



Solaris Common Desktop Environment: Handbok för systemadministratörer

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94043
USA

Artikelnummer: 805-5139-10
Oktober, 1998

Copyright 1998 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, CA 94303-4900 USA. Med ensamrätt.

Den här produkten och tillhörande dokumentation är skyddad genom copyright och distribueras under licens, vilket begränsar hur det kan användas, kopieras, distribueras och dekompileras. Ingen del av denna produkt eller tillhörande dokument får återges i någon form utan uttryckligt skriftligt tillstånd från Sun eller från eventuella licensgivare. Tredjepartsprogram, inklusive teckensnittsteknik i produkten är skyddade genom copyright och licensierade från leverantörer till Sun.

Delar av denna produkt kan komma från Berkeley BSD system, licensierad från University of California. UNIX är ett registrerat varumärke i USA och i andra länder och är exklusivt licensierat genom X/Open Company, Ltd.

Sun, Sun Microsystems, Sun-logotypen, SunSoft, SunDocs, SunExpress och Solaris är varumärken, registrerade varumärken eller servicemärken från Sun Microsystems, Inc. i USA eller i andra länder. Alla SPARC-varumärken används under licens och är varumärken eller registrerade varumärken från SPARC International, Inc. i USA eller i andra länder. Produkter som bär SPARC-varumärken baseras på arkitektur utvecklad av Sun Microsystems, Inc. Koden och dokumentationen till bildmodulerna DtComboBox och DtSpinBox tillhandahålls av Interleaf, Inc. Copyright 1993, Interleaf, Inc..

De grafiska användargränssnitten OPEN LOOK och Sun™ är utvecklade av Sun Microsystems, Inc. för dess användare och licenstagare. Sun vill tacka Xerox för de banbrytande insatser i forskning och utveckling av visuellt och grafiskt användargränssnitt som de gjort för datorindustrin. Sun innehar en licens utan ensamrätt från Xerox till Xerox grafiska användargränssnitt. Licensen täcker också Suns licensinnehavare som implementerar OPEN LOOK GUI:s och på annat sätt samtycker till Suns skriftliga licensavtal.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DF AR 227.7202-3(a).

DENNA PUBLIKATION TILLHANDAHÅLLS "I BEFINTLIGT SKICK" UTAN NÅGRA SOM HELST GARANTIER, VARKEN UNDERFÖRSTÅDDA ELLER UTTRYCKLIGA, INKLUSIVE, MEN INTE BEGRÄNSAD TILL, DE UNDERFÖRSTÅDDA FÖRSÄLJNINGSGARANTIERNA, LÄMPLIGHET FÖR ETT SÄRSKILT ÄNDAMÅL, ELLER ATT DE INTE INKRÄKTAR PÅ ANDRA FÖRETAGS RÄTTIGHETER.

Copyright 1998 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, Californie 94303-4900 Etats-Unis. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées du système Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, SunSoft, SunDocs, SunExpress, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc. Le code et la documentation pour les produits DtComboBox et DtSpinBox ont été fournis par Interleaf, Inc. Copyright 1993, Interleaf, Inc

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REPENDRE A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.



Papperet kan återvinnas

Innehåll

Inledning	xix
1. Konfigurera inloggningshanteraren	1
Starta inloggningsservern	2
Hantera lokala skärmar och nätverksskärmar	2
Söka efter inloggningsserverns process-ID	3
Visa en inloggningsbild på en lokal skärm	4
Köra inloggningsservern utan lokal skärm	4
Kommandoradsinloggning på lokala skärmar	5
Anpassa textskärmen	5
Visa en inloggningsbild på en nätverksskärm	6
Kontrollera åtkomst till inloggningsservern	7
Felsökning	8
Inaktivera inloggningsservern	9
Inloggningsbilden	9
Ändra utseendet på inloggningsbilden	10
▼ Ändra logotyp	11
▼ Ändra välkomstmeddelandet	11
▼ Ändra teckensnitt	12
▼ Ange olika texter för olika språk	12

Ändra hur inloggningsbilden fungerar skärmvis	13
Ändra X-serverbehörighet	13
Starta kommandon innan inloggningsbilden visas	15
Starta en Felsäker session	15
När användarsessionen har avslutats	16
Inloggningsserverns omgivning	16
Ändra användar- eller systemsökvägen	17
▼ Ändra systemskalet	17
▼ Ändra tidszon	18
Administrera inloggningshanteraren	18
Inloggningshanterarens filer	19
2. Konfigurera sessionshanteraren	21
Vad är en session?	21
Startsession	22
Aktiv session	22
Hemsession	22
Skärmspecifika sessioner	22
Starta en session	23
När sessionen startar	23
Läsa in och köra skriptet .dtprofile	24
Läsa in och köra skriptet Xsession.d	24
Visa välkomstmeddelandet	25
Ställa in skrivbordets sökvägar	25
Samla ihop tillgängliga program	26
Läsa in och köra skripten .profile och .login (valfritt)	26
Starta ToolTalk	27
Starta sessionshanterarklienten	27
Ladda sessionsresurser	27

Starta färghanteraren	28
Starta arbetsytehanteraren	29
Starta sessionsprogram	29
Ytterligare anpassningar av sessionsstarten	29
▼ Ange omgivningsvariabler	30
▼ Ange resurser	30
▼ Ange skärmspecifika resurser	31
▼ Ändra program för startsessionen	32
▼ Ange en skärmspecifik session	32
Köra ytterligare kommandon vid sessionsstart och utloggning	32
▼ Köra ytterligare kommandon vid sessionsstart	33
▼ Köra ytterligare kommandon vid utloggning	33
▼ Återskapa en session från säkerhetskopia	33
▼ Undersöka problem vid sessionsstart	34
Filer och kataloger för sessionshanteraren	34
3. Felsökning för inloggning och problem vid sessionstart	35
Startfiler för inloggning	35
Fellogningskataloger	36
Användarstartfil	37
Solaris CDE, startexempel	37
4. Lägga till och hantera program	39
Programhanterarens struktur	39
Programhanterarens plats i filsystemet	40
Hur programhanteraren hittar och samlar ihop program	40
Prioritetsregler för insamling av program	41
Programgrupper på standardskrivbordet	41
Exempel på hur programgrupper samlas in	42
Lägga till program i programhanteraren	43

	Lägga till program i programhanteraren	43
▼	Lägga till skrivbordsanpassade program i programhanteraren	44
▼	Registrera befintliga program eller program som inte är skrivbordsanpassade	45
▼	Lägga till en programikon i befintlig programgrupp	45
	Skapa och hantera allmänna programgrupper	46
▼	Skapa en systemtäckande allmän programgrupp	46
▼	Skapa en egen allmän programgrupp	46
▼	Anpassa en inbyggd programgrupp	46
	Ändra sökvägen som används för att söka efter program	47
	Standardsökvägen	47
	Lägga till en programsver i programsökvägen	48
	Allmän administration av programhanteraren	49
▼	Ta bort ett program	49
▼	Uppdatera programhanteraren under en session	49
	Ändra textredigeraren och terminalemulatorn	50
▼	Ändra standardtextredigerare och standardterminalemulator	50
5.	Registrera ett program	53
	Programregistrering – översikt	54
	Funktioner som medföljer programregistrering	54
	Syftet med programregistrering	56
	Generella steg för registrering av program	57
	Steg 1: Ändra teckensnitts- och färgresurser	58
	Steg 2: Skapa skrivbordsprogrammets rot	59
	Steg 3: Skapa registreringspaketets kataloger	59
	Steg 4: Skapa åtgärderna och datatyperna för programmet	62
	Steg 5: Placera hjälpfilerna i registreringspaketet	65
	Steg 6: Skapa ikoner för programmet	65
	Steg 7: Skapa programgruppen	67

Steg 8: Registrera programmet med hjälp av dtappintegrate	73
Exempel på hur du kan skapa ett registreringspaket	76
Information om "BraTextredigerare"	76
Steg för att registrera "BraTextredigerare"	77
6. Diverse konfigurationer	83
Solaris CDE – katalogstruktur	83
/usr/dt	83
/etc/dt	84
/var/dt	84
\$Hemkatalog	85
Huvudkonfigurationsfiler	85
Xconfig	85
Xservers	85
Starta inloggningsservern	86
Montera en installerad CDE från en annan installationsadress för en arbetsstation eller nätverksserver	88
▼ Montera en installerad CDE	88
▼ Avmontera en monterad CDE-katalog	89
Konfigurera skrivbordet för att använda flera skärmar	89
▼ Starta skrivbordet på flera skärmar	90
Nätverksanslutna skrivbord	91
Använda X-terminaler	92
Inloggningsspråkområde och teckensnittssökväg	93
Använda arbetsstationer som X-terminaler	94
▼ Använda Väljaren för att markera en värd-CDE-inloggning	94
▼ Använda en specifik värd-CDE-inloggning	95
▼ Använda den första tillgängliga värdinloggningen	95
Särskilda CDE-konfigurationer	95

Anpassa utskrift av brev	96
Konvertera kalendrar till nya dataformat	97
Lägga till AnswerBook-paketet från nätverket	97
Installera CDE-omgivning utanför CDE-skrivbordet	98
Skrivbordsomgivningsfil	98
Använda diskett och CD-skiva med MAE (Macintosh Application Environment)	99
7. Konfigurera skrivbordet i nätverk	101
Nätverk på skrivbord – översikt	101
Typer av nätverkstjänster för skrivbord	102
Typiska nätverkssituationer	102
Andra nätverkssituationer	104
Sammanfattning – servertyper	105
Generella steg för att konfigurera nätverket	106
Konfigurera operativsystemets basnätverkstjänster för skrivbordet	106
Inloggningskonto för användare	107
Konfigurera distribuerad filsystemåtkomst	107
Konfigurera åtkomst av fjärrskrivare	108
Konfigurera e-post	109
Konfigurera som X-verifiering	109
Konfigurera skrivbordsklienter och servrar	109
Konfigurera inloggnings- och sessionstjänster	109
Konfigurera andra programrelaterade tjänster	110
Administrera programtjänster	114
Omgivningsvariabler för sökvägar	115
Konfigurera en programserver och dess klienter	115
Konfigurera databas-, ikon- och hjälptjänster	117
Speciella konfigurationer för nätverksprogram	118

8.	Konfigurera och administrera utskrifter från skrivbordet	123
	Lägga till och ta bort skrivare	123
	▼ Lägga till en skrivare till skrivbordet	124
	▼ Ta bort en skrivare från skrivbordet	124
	Ändra intervall för jobbuppdatering	125
	Ikonbilder till skrivaren	125
	Ikonfilnamn och ikonfilstorlekar	125
	▼ Ändra ikon, skrivarnamn och beskrivning globalt	125
	Konfigurera standardskrivare	126
	▼ Ändra destination för standardutskrift	126
	Vad som sker vid utskrift	127
9.	Skrivbordets sökvägar	129
	Skrivbordets sökvägar och deras omgivningsvariabler	130
	Ange värde för en sökväg	131
	▼ Visa aktuellt värde för en sökväg (utdatavariabel)	131
	▼ Göra egna ändringar av sökvägen	131
	▼ Göra systemtäckande ändringar av sökvägen	132
	Programsökväg	132
	Standardsökväg för program	132
	Omgivningsvariabler för programsökvägar	133
	Syntax för indatavariabler för programsökväg	133
	Hur programsökvägens värde sätts samman	133
	Ändra prioritetsordning på den lokala systemtäckande adressen	134
	Hur programsökvägen påverkar sökvägarna för databas, ikon och hjälp	134
	Databassökväg (åtgärd/datatyper)	135
	Databasens standardsökväg	136
	Hur programsökvägen påverkar databassökvägen	136

	Databassökvägens omgivningsvariabler	136
	Syntax för databassökvägens indatavariabler	136
	Hur databassökvägen sätts samman	137
	Ikonsökväg	137
	Standardikonsökväg	137
	Hur programsökvägen påverkar ikonsökvägen	138
	Ikonsökvägens omgivningsvariabler	138
	Syntax för ikonsökvägens indatavariabler	138
	Hur ikonsökvägen sätts samman	138
	Hjälpsökväg	139
	Standardhjälpsökväg	139
	Hur programsökvägen påverkar hjälpsökvägen	139
	Hjälpsökvägens omgivningsvariabler	140
	Syntax för hjälpsökvägens indatavariabler	140
	Hur ikonsökvägen sätts samman	140
	Lokaliserade sökvägar	141
10.	Introduktion till åtgärder och datatyper	143
	Introduktion till åtgärder	144
	Hur åtgärder skapar programikoner	146
	Hur åtgärder använder datafiler som argument	148
	Ytterligare användningsområden för åtgärder	149
	Introduktion till datatyper	149
	Vad är en datatyp?	149
	Hur datatyper ansluter datafiler till åtgärder	150
	Skriva ut datatyper från skrivbordet	153
11.	Skapa åtgärder och datatyper med Skapa åtgärd	155
	Vad Skapa åtgärd gör	155
	Begränsningar i Skapa åtgärd	156

Åtgärdsbegränsningar	156
Datatypsbegränsningar	157
Skapa en åtgärd och datatyp för ett program med Skapa åtgärd	157
▼ Skapa en åtgärd för ett program	157
▼ Skapa en eller flera datatyper för programmet	160
Använd dialogrutan Sök ikonuppsättning för att ange en ikon	165
12. Skapa åtgärder manuellt	169
När åtgärder måste skapas manuellt	170
COMMAND-åtgärder	170
MAP-åtgärder	170
TT_MSG-åtgärder (ToolTalk-meddelanden)	171
Skapa en åtgärd manuellt: Allmänna steg	171
Konfigurationsfiler för åtgärder	171
▼ Skapa en åtgärd manuellt	171
Exempel på hur du skapar en COMMAND-åtgärd	172
Exempel på hur du skapar en MAP-åtgärd	173
▼ Starta om databasen för åtgärder och datatyper	174
Skapa åtgärdsfil (ikon) för åtgärd	174
Ange åtgärdens ikonbild	176
▼ Ändra en befintlig åtgärdsdefinition	177
Prioritet i åtgärdsdefinitioner	178
Skapa en exekveringssträng för COMMAND-åtgärder	179
Generella egenskaper för exekveringssträngar	179
Skapa åtgärder som inte använder argument	180
Skapa åtgärder som accepterar släppta filer	181
Skapa en åtgärd som frågar efter ett filargument	181
Skapa åtgärder som accepterar eller frågar efter en släppt fil	182
Skapa en åtgärd som frågar efter icke-filargument	182

Tolka filargument som en sträng	183
Skalmöjligheter för åtgärder	183
Skapa COMMAND-åtgärder för flera filargument	183
Fönsterstöd och terminalemulatorer för COMMAND-åtgärder	185
Ange fönsterstöd för åtgärden	186
Ange kommandoradsalternativ för terminalemulatorn	186
Ange en annan standardterminalemulator	187
Begränsa åtgärder för vissa argument	187
Begränsa åtgärder till en viss datatyp	188
Begränsa åtgärder med antalet argument	188
▼ Ange metod för dubbelklickning och släppfunktionen	188
Begränsa åtgärder baserat på argumentläge	189
Skapa åtgärder som kör program i fjärrsystem	190
Skapa en åtgärd som kör ett fjärrprogram	190
Använda variabler i åtgärds- och datatypsdefinitioner	191
Använda strängvariabler i en åtgärd	191
Använda omgivningsvariabler i åtgärder och datatyper	192
Starta åtgärder från en kommandorad	192
Syntax för <code>dtaction</code>	192
Skapa en åtgärd som kör en annan åtgärd	193
Skapa åtgärder som kör som andra användare	193
Skapa språkanpassade åtgärder	194
Adresser för språkanpassade åtgärder	194
▼ Språkanpassa en befintlig åtgärd	194
Skapa åtgärder för ToolTalk-program	195
Fälten <code>addressing</code> och <code>disposition</code>	195
Meddelanden som inte stöds	195
13. Skapa datatyper manuellt	197

Varför du måste skapa datatyper manuellt	197
Komponenter i datatypsdefinitionen: Villkor och attribut	198
Skapa datatyper manuellt: Allmänna steg	198
Konfigurationsfiler för datatyper	199
▼ Skapa en datatypsdefinition	199
Exempel på hur du skapar en egen åtgärd och datatyp	200
Definiera datatypens dataattribut	201
Ange ikonbild för datatypen	202
Associera datatyper med åtgärder	202
Dölja filer baserat på datatyp	203
Ange metoder när en fil manipuleras	203
Definiera dataattribut för en datatyp	204
Namnbaserade datatyper	205
Sökvägsbaserade datatyper	206
Datatyper baserade på namn och sökväg	206
Använda fillägen som typvillkor	207
Innehållsbaserad datatypsbestämning	209
▼ Skapa en datatyp med flera oberoende villkor	209
Skapa språkanpassade datatyper	210
Adresser för språkanpassade datatyper	210
▼ Språkanpassa en datatyp	210
14. Skapa skrivbordsikoner	213
Ikonbildsfilen	213
Ikonfilformat	214
Ikonfilnamn	214
Storlekskonventioner	214
Ikonsökväg	216
Komma åt ikoner via nätverket	216

Associera ikoner	216
Ange ikonfiler	216
▼ Associera en ikon med en funktion eller datatyp	216
▼ Visa en ikon i huvudgruppen	217
▼ Associera en ikon med ett programfönster	218
▼ Använda filhanteraren som ett ikonfönster	218
Rekommendationer för hur du skapar ikoner	219
Hur färger används	219
15. Särskild anpassning av huvudgruppen	221
Konfigurationsfiler för huvudgruppen	221
Standardhuvudgruppens konfigurationsfil	222
Sökvägen till huvudgruppens konfigurationsfiler	222
Hur huvudgruppen sätts samman: prioritetsregler	223
Skapa huvudgruppsfiler dynamiskt	223
Administrera anpassning av användargränssnitt	224
▼ Förhindra egna anpassningar	224
▼ Återställa en borttagen kontroll eller panelundermeny	224
Hur huvudgruppsdefinitionen är organiserad	225
Komponenter i huvudgruppen	225
Allmän syntax för huvudgruppsdefinitionen	226
Anpassa huvudpanelen	228
▼ Lägga till en kontroll på huvudpanelen	229
▼ Ta bort en kontroll	229
▼ Ändra en kontroll	230
▼ Byta position för kontrollen	231
▼ Ersätta en huvudgruppskontroll	231
Ange vilken ikonbild som ska användas av kontrollen	232
Skapa och ändra panelundermenyer	233

- ▼ Skapa en ny systemtäckande panelundermeny 234
 - Anpassa inbyggda panelundermenyer 234
- ▼ Ta bort den automatiska stängningen (Auto-Close) av panelundermenyer 236
- Definiera kontroller i huvudgruppen 237
 - Definitioner till kontroller i huvudgruppen 237
 - Kontrolltyper 237
- ▼ Skapa en ny kontroll 238
- Anpassa området för växling av arbetsyta 243
- ▼ Ändra standardantalet arbetsytor 244
- ▼ Ändra antalet knapprader i växlingsområdet 244
- ▼ Ändra eller lägga till kontroller i växlingsområdet 244
- Allmän konfiguration av huvudgruppen 245
 - Allmänna steg 245
- ▼ Ändra huvudgruppens standardposition 245
- ▼ Namnge kontroller på huvudpanelen 246
- ▼ Ändra klickmetod för kontroller 246
- ▼ Skapa en helt ny huvudgrupp 246
 - Exempel på hur du skapar en egen huvudgrupp med tre rader 247
- 16. Anpassa arbetsythanteraren 249**
 - Arbetsythanterarens konfigurationsfiler 250
 - ▼ Skapa och ändra en egen konfigurationsfil 250
 - ▼ Skapa en systemtäckande konfigurationsfil 251
 - ▼ Inkludera andra filer 251
 - ▼ Starta om arbetsythanteraren 252
 - Anpassa arbetsytor 252
 - ▼ Ändra antalet systemtäckande arbetsytor 252
 - ▼ Namnge systemtäckande arbetsytor 253
 - ▼ Skapa ytterligare bakgrunder 253

- ▼ Ersätta bakgrunden med en grafisk bild 254
- Arbetsytahanterarens menyer 254
 - Arbetsytahanterarens menysyntax 255
- ▼ Så här lägger du till ett nytt menyalternativ på menyn Arbetsyta 255
- ▼ Så här anpassar du menyn Arbetsyta 256
- ▼ Skapa en ny Arbetsyta-meny 258
- ▼ Skapa en ny Fönster-meny 258
- Anpassa knappkopplingar 259
 - Syntax för knappkoppling 259
- ▼ Lägga till knappkopplingar 260
- ▼ Skapa nya knappkopplingsuppsättningar 261
- Anpassa tangentkopplingar 261
 - Förinställda tangentkopplingar för skrivbordet 261
 - Syntax för tangentkopplingar 262
 - ▼ Skapa en anpassad tangentkopplingsuppsättning 262
- Växla mellan förinställda och egna uppsättningar 263
- 17. Hantera programresurser, teckensnitt och färger 265**
 - Ange programresurser 265
 - ▼ Ange systemtäckande resurser 266
 - ▼ Anger egna resurser 266
 - Hur skrivbordet startar resurser 266
 - Processhanteraresurser 266
 - Definiera UNIX-kopplingar 267
 - ▼ Ange EMACS-stildefinitioner 267
 - ▼ Ändra EMACS-stildefinitioner 267
 - UNIX-kopplingar i filen UNIXbindings 267
 - Hantera teckensnitt 271
 - Ange teckensnittsresurser för skrivbordet 272

▼	Visa tillgängliga teckensnitt	273
▼	Ange teckensnitt på kommandoraden	273
	XLFD (X Logical Font Description)	274
	Lagring av användares filsystem för teckensnittsgrupper	276
	Skapa teckensnittsgrupp för systemadministratör	276
	Hantera färger	277
	Färgpaletter	277
	Färguppsättningar	277
	Kontrollera färg med Inställningar	281
	Antal färger som används i Inställningar	281
	Ange skuggtjocklek för programfönster	285
18.	Konfigurera språkpassade skrivbordsessioner	287
	Hantera omgivningsvariabeln LANG	287
	Ange språk för flera användare	288
	Ange språk för en session	289
	Ange språk för en användare	289
	Omgivningsvariabeln LANG och sessionskonfiguration	289
	Ange andra NLS-omgivningsvariabler	290
	Söka efter teckensnitt	290
	Språkpassa resursfilerna app-defaults	290
	Språkpassa åtgärder och datatyper	291
	Språkpassa ikoner och bitmappar	291
	Språkpassa bakgrundsnamn	292
	Språkpassa palettnamn	292
	Språkpassa hjälpvolym	293
	Språkpassa meddelandekataloger	293
	Köra språkpassade fjärranslutna skrivbordsprogram	293
	Återställa tangentbordstabellen	294

A. Direkthjälp (man page) för dtconfig(1) 295

Index 297

Inledning

I den här handboken beskrivs hur Solaris Common Desktop Environment (CDE) ser ut och fungerar för erfarna användare. Boken tar bland annat upp:

- hur du anpassar systemstart, inloggning och startsession
- hur du lägger till program och förser program och deras data med gränssnitt
- hur du konfigurerar skrivbordsprocesser, program och data via nätverk
- hur du anpassar skrivbordet, t ex fönsterhantering, utskrift, färger och teckensnitt

Obs! Termen "x86" refererar till Intels 8086-familj av mikroprocessorer, inklusive processorerna Pentium och Pentium Pro samt kompatibla mikroprocessorer ifrån AMD och Cyrix. I det här dokumentet refererar termen "x86" till den allmänna plattformens arkitekturen, medan "*Intel-version*" ingår i produktnamnet.

Den här boken riktar sig till

Bokens målgrupp:

- Systemadministratörer – många av uppgifterna i boken kräver rotbehörighet.
- Vana användare som vill anpassa systemet på ett sätt som inte går att göra med skrivbordets verktyg. Skrivbordet ger användarspecifika sökvägar till de flesta konfigurationsfilerna.

Innan du läser denna bok

Användare bör känna till innehållet i följande böcker:

- *Solaris CDE: Användarhandbok*
- *Solaris Common Desktop Environment: Användarhandbok för byte av miljö*

Hur boken är upplagd

Handboken innehåller följande kapitel:

Kapitel 1 handlar om hur du konfigurerar utseendet och funktionen hos inloggningshanteraren.

Kapitel 2 handlar om hur skrivbordet sparar och hämtar sessioner och hur du anpassar startsessionen.

Kapitel 3 beskriver startfiler i Solaris CDE, möjliga startproblem med Solaris CDE och förslag till lösningar för att undvika startproblem.

Kapitel 4 handlar om hur program ställs samman i programhanteraren och hur nya program läggs till.

Kapitel 5 beskriver hur du skapar ett registreringspaket för ett program.

Kapitel 6 tar upp olika delmoment inom mer avancerad konfiguration, t ex egna inloggningskonfigurationer, inställningar för flera fönster, skrivbord och X-terminaler som ingår i ett nätverk, ändring av punktfiler för användaren, utskriftsinställningar för e-post, inställningar av skrivbordsmiljö och olika typer av felloggfiler.

Kapitel 7 beskriver hur du distribuerar tjänster, program och data från skrivbordet över ett nätverk.

Kapitel 8 handlar om hur du lägger till och tar bort skrivare och hur du anger en standardskrivare.

Kapitel 9 beskriver hur skrivbordet hittar program, hjälpfiler, ikoner och andra skrivbordsrelaterade data över nätverket.

Kapitel 10 introducerar begreppen "åtgärder" och "datatyper" och förklarar hur dessa förser användaren med gränssnitt gentemot program.

Kapitel 11 beskriver hur du använder programmet Skapa åtgärd för att skapa åtgärder och datatyper.

Kapitel 12 beskriver hur du skapar åtgärdsdefinitioner genom att redigera en databaskonfigurationsfil.

Kapitel 13 beskriver hur du skapar datatypsdefinitioner genom att redigera en databaskonfigurationsfil.

Kapitel 14 beskriver hur du använder ikonredigeraren och hur du namnger, ändrar storlek på och anger sökväg för ikoner för skrivbordet.

Kapitel 15 beskriver hur du skapar nya heltäckande kontroller och panelundermenyer samt andra panelanpassningar.

Kapitel 16 beskriver hur du anpassar fönster, musknappskopplingar, tangentkopplingar och menyer i arbetsytahanteraren.

Kapitel 17 beskriver hur du ställer in programresurser, färger och teckensnitt.

Kapitel 18 beskriver systemadministrationen i system som kör internationella sessioner.

Bilaga A är en kopia av direkthjälpen (man pages) till dtconfig(1).

Beställa dokument från Sun

I SunDocssm-programmet ingår fler än 250 handböcker från Sun Microsystems. Du som bor i USA, Kanada, Europa eller Japan kan köpa dokumentationspaket eller enskilda handböcker med hjälp av SunDocs-programmet.

En lista över de dokument som kan beställas och anvisningar om hur du gör finns i katalogavsnittet i SunExpressTM på Internet-adressen <http://www.sun.com/sunexpress>.

Typografiska markeringar

Följande tabell beskriver de typografiska markeringar som används i den här boken.

TABELL P-1 Typografiska konventioner

Teckensnitt eller symbol	Innebörd	Exempel
AaBbCc123	Namn på kommandon, filer och kataloger, utdata på skärmen	Redigera din <code>.login</code> -fil. Använd <code>ls -a</code> för att lista alla filer. <code>datornamn% You have mail.</code>
AaBbCc123	Vad du skriver i jämförelse med vad som matas ut på skärmen	<code>datornamn% su</code> <code>Password:</code>
<i>AaBbCc123</i>	Kommandoradens platshållare: ersätt med ett riktigt namn eller värde	Om du vill ta bort en fil, skriver du <code>rm filnamn</code> .
<i>AaBbCc123</i>	Boktitlar, nya ord eller termer eller ord som ska framhävas	Läs kapitel 6 i <i>Användarhandboken</i> . Detta kallas <i>klassalternativ</i> . Du <i>måste</i> vara superanvändare för att kunna göra detta.

Ledtext i skalet för kommandoexempel

Följande tabell visar standardledtexten och superanvändarledtexten för C-skal, Bourne-skal och Korn-skal.

TABELL P-2 Ledtext i skalet

Skal	Ledtext
Ledtext för C-skal	<code>datornamn%</code>
Ledtext för superanvändare med C-skal	<code>datornamn#</code>

TABELL P-2 Ledtext i skalet (forts.)

Skal	Ledtext
Ledtext för Bourne-skal och Korn-skal	\$
Ledtext för superanvändare med Bourne-skal och Korn-skal	#

Konfigurera inloggningshanteraren

Inloggningshanteraren är en server som visar en inloggningsbild, verifierar användarna och startar användarens session. Den här grafiska inloggningen är enklare än den traditionella teckenbaserade inloggningen. Skärmarna som hanteras av inloggningsservern kan direkt anslutas till inloggningsservern, till en X-terminal eller till en arbetsstation i nätverket.

Obs! Du måste vara rotanvändare för att kunna starta, stoppa eller anpassa inloggningsservern.

- "Starta inloggningsservern" på sidan 2
- "Hantera lokala skärmar och nätverksskärmar" på sidan 2
- "Felsökning" på sidan 8
- "Inaktivera inloggningsservern" på sidan 9
- "Ändra utseendet på inloggningsbilden" på sidan 10
- "Administrera inloggningshanteraren" på sidan 18
- "Inloggningshanterarens filer" på sidan 19

Inloggningsservern:

- kan visa en inloggningsbild på punktbaserade skärmar utan begäran eller på begäran av punktbaserade skärmar lokalt eller i nätverk
- förser systemfönster med direktanslutna tecken
- kan visa en valskärm där användaren kan visa inloggningsbilder från andra inloggningsservrar i nätverket
- tillåter kontrollerad åtkomst till inloggningsservern
- omfattar även traditionell teckenbaserad inloggning

Skärmar som hanteras av inloggningshanteraren kan anslutas direkt till inloggningshanterarservern eller till en X-terminal eller arbetsstation i nätverket. För

lokala skärmar startar inloggningsservern automatiskt en X-server och visar en inloggningssbild. För nätverksskärmar (t ex X-terminaler) stöds XDMCP 1.0, (X Display Manager Protocol), med vilket skärmar kan begära att inloggningsservern visar en inloggningssbild.

Starta inloggningsservern

Inloggningsservern startas normalt när systemet startas. Du kan också starta inloggningsservern från en kommandorad.

- Om du vill att inloggningsservern ska starta när systemet startas, skriver du `/usr/dt/bin/dtconfig -e`.

Inloggningsservern kommer sedan att startas automatiskt när du startar om.

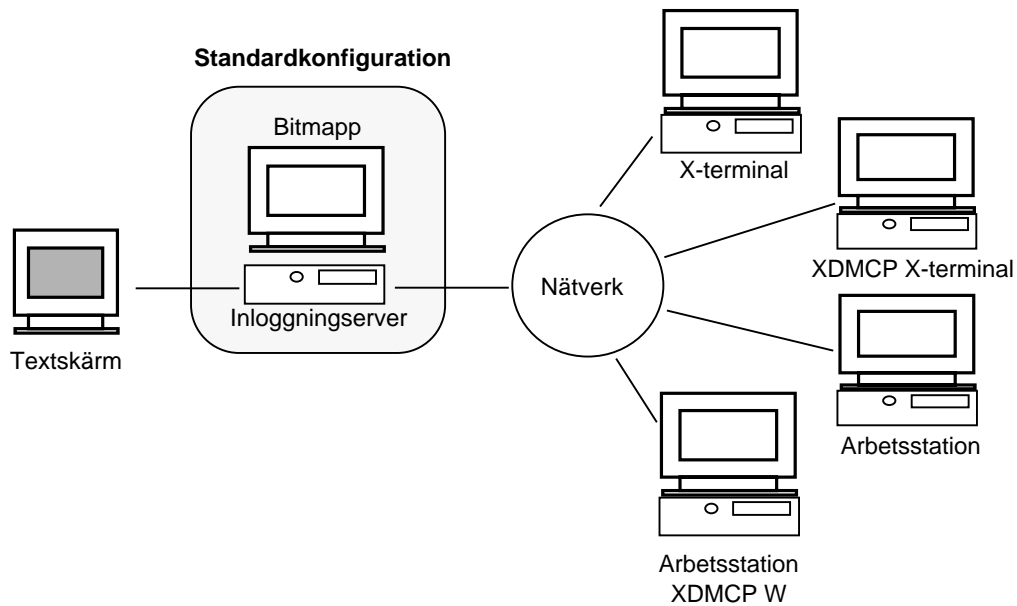
Mer information om skrivbordskonfigurationsverktyget `dtconfig` finns i Bilaga A. Där finns en kopia av direkthjälpavsnittet (man page) till `dtconfig.1`.

- Om du vill starta inloggningsservern från kommandoraden, skriver du `/usr/dt/bin/dtlogin -daemon; exit`.

Obs! Även om det är möjligt att starta inloggningsservern från kommandoraden när du ska testa konfigurationen, bör du normalt starta inloggningsservern i samband med att systemet startas.

Hantera lokala skärmar och nätverksskärmar

Figur 1-1 visar en möjlig konfiguration av inloggningsservern.



Figur 1-1 Exempel på konfiguration av inloggningsserver

Söka efter inloggningsserverns process-ID

Som standard lagras inloggningsserverns process-ID i `/var/dt/Xpid`.

Om du vill ändra detta anger du resursen `Dtlogin.pidFile` i filen `Xconfig`. Om filen ändras måste katalogen finnas när inloggningsservern startar.

Om du vill ändra `Xconfig`, kopierar du `Xconfig` från `/usr/dt/config` till `/etc/dt/config`. När du har ändrat `/etc/dt/config/Xconfig` ser du till att inloggningsservern läser `Xconfig` igen genom att skriva:

```
/usr/dt/bin/dtconfig -reset
```

Detta kör kommandot `kill -HUP inloggningsserverprocess-ID`.

Om du t ex vill lagra inloggningsserverns process-ID i `/var/minserver/Dtpid`, anger du följande i filen `Xconfig`:

```
Dtlogin.pidFile: /var/minserver/Dtpid
```

När inloggningsservern startas om lagras dess process-ID i `/var/minserver/Dtpid`. Katalogen `/var/minserver` måste finnas när inloggningsservern startas.

Visa en inloggningsbild på en lokal skärm

När inloggningsservern startar kontrolleras filen `Xservers` för att avgöra om en X-server måste startas, samt om och i så fall hur inloggningsskärmar ska visas på lokala skärmar och nätverksskärmar.

Om du vill ändra `Xservers`, kopierar du `Xservers` från `/usr/dt/config` till `/etc/dt/config`. När du har ändrat `/etc/dt/config/Xservers` ser du till att inloggningsservern läser `Xservers` igen genom att skriva:

```
/usr/dt/bin/dtconfig -reset
```

Detta kör kommandot `kill -HUP inloggningsserverprocess-ID`.

Formatet på en `Xservers`-rad är:

skärnmamn skärmklass skärmtyp X-serverkommando

där

skärnmamn – meddelar inloggningsservern vilket anslutningsnamn som ska användas vid anslutning till X-servern (:0 i följande exempel). Värdet * (asterisk) utökas till *värddatornamn:0*. Den angivna siffran måste matcha den siffra som anges i anslutningsnumret för *X-serverkommando*.

skärmklass – identifierar resurser som är anknutna till skärmen (Local i följande exempel).

skärmtyp – meddelar inloggningsservern om skärmen är en lokal skärm eller en nätverksskärm och hur alternativen för Kommandoradsinloggning i inloggningssfönstret (local@console i följande exempel) ska hanteras.

X-serverkommando – identifierar kommandoraden, anslutningsnumret och andra alternativ som inloggningsservern kommer använda för att starta X-servern (`/usr/bin/X11/X: 0` i följande exempel). Det angivna anslutningsnumret måste matcha den siffra som anges i *skärnmamn*.

Standardraden för `Xservers` ser ut ungefär så här:

```
:0 Local local@console /usr/bin/X11/X :0
```

Köra inloggningsservern utan lokal skärm

Om inloggningsserversystemet inte har någon punktbaserad skärm, kör du inloggningsservern utan lokal skärm genom att kommentera bort raden `Xservers` för den lokala skärmen med hjälp av tecknet # (nummertecken). Exempel:

```
# :0 Local local@console /usr/bin/X11/X :0
```

När inloggningsservern startar körs den i bakgrunden och väntar på en förfrågan från nätverksskärmar.

Kommandoradsinloggning på lokala skärmar

När användaren väljer Kommandoradsinloggning på inloggningsbilden avbryts X-servern tillfälligt, vilket gör att den traditionella kommandoradsinloggningen på den punktbaseade skärmen kan användas. När användaren har loggat in och sedan ut, eller efter angiven time-out, startas X-servern om.

Obs! Alternativet Kommandoradsinloggning kan inte användas på nätverksskärmar.

Skärmtypen kontrollerar hur kommandoradsinloggningen fungerar. Formatet för *skärmtyp* är:

- `local@skärmenhet`
- `local`
- `foreign`

När `local@skärmenhet` anges antar inloggningsservern att X-servern och `/dev/skärmenhet` finns på samma fysiska enhet och att en kommandoradsinloggning (normalt `getty`) körs på enheten. När användaren väljer kommandoradsinloggning kommer X-servern att avslutas och ge åtkomst till kommandoradsinloggning (`getty`) på `/dev/skärmenhet`.

Om du vill inaktivera alternativet Kommandoradsinloggning på en skärm anger du `none` som *skärmenhet*. *Standardsskärmenhet* är `console`. När `local` anges, används `console` som standard för *skärmenhet*. När `foreign` anges kan Kommandoradsinloggning inte användas.

Obs! Alternativet Kommandoradsinloggning inaktiveras på den lokala skärmen när inloggningsservern startar från kommandoraden.

Anpassa textskärmen

Om inloggningssystemet har en direktansluten textskärm som fungerar som ett systemfönster, kan du ange `none` som *skärmenhet* för att inaktivera Kommandoradsinloggning på den punktbaseade skärmen.

Om kommandoradsinloggning (`getty`) körs på både textskärmen och den punktbaseade skärmen kan du ändra *skärmenhet* till enheten för kommandoradsinloggning (`getty`) på den punktbaseade skärmen.

Om t ex den punktbaseade skärmens kommandoradsinloggning (`getty`) finns på enheten `/dev/tty01` ändrar du *skärmtyp* till `local@tty01`.

Visa en inloggningsbild på en nätverksskärm

Inloggningsservern kan acceptera förfrågningar från nätverksskärmar om att visa en inloggningsbild på en viss skärm. Nätverksskärmen är normalt en X-terminal men den kan också vara en arbetsstation.

Om du ska hantera förfrågningar från nätverksskärmen måste inloggningsservern stödja XDMCP (X Display Manager Protocol) 1.0. Detta protokoll gör att inloggningsservern kan förhandla med nätverksskärmen genom att acceptera eller avslå förfrågningar. XDMCP finns inbyggt i de flesta X-terminaler.

XDMCP, direktförfrågningar från nätverksskärmar

När du konfigurerar X-terminalen för att använda XDMCP direkt (frågeläge), anger du namnet på inloggningsservaren. När X-terminalen startas kontaktas inloggningsservern automatiskt, och en inloggningsbild visas på X-terminalen. Mer information om hur du konfigurerar X-terminalen för XDMCP-direktläge finns i dokumentationen till X-terminalen.

De flesta X-servrar stöder också alternativet `-query`. I det här läget fungerar X-servern som om den var en X-terminal, och kontaktar inloggningsservarens värddator direkt med en begäran om att inloggningsbilden ska visas på X-servern. Om du t ex startar X-servern på en punktbaserad skärm på arbetsstationen `bridget`, visar inloggningsservern `anita` en inloggningsbild på X-servern:

```
X -query anita
```

XDMCP, indirekta förfrågningar från nätverksskärmar

När du konfigurerar X-terminalen att använda indirekt XDMCP, anger du namnet på inloggningsservaren. När X-terminaler startas kontaktas inloggningsservern. Denna visar, via en skärm, en lista med andra inloggningsservrar i nätverket. I denna lista kan användaren välja en värddator, och värddatorn visar en inloggningsbild på användarens X-terminal. Mer information om hur du konfigurerar X-terminalen för XDMCP i indirekt läge finns i dokumentationen till X-terminalen.

I detta fall stöder X-servrar alternativet `-indirect`, med vilket X-servern kontaktar inloggningsservern i XDMCP indirekt läge.

Hantera nätverksskärmar utan XDMCP

Det kan hända att äldre X-terminaler inte stöder XDMCP. Om du vill att inloggningsservern ska kunna visa en inloggningsbild på den här typen av X-terminal, tar du med X-terminalnamnet i listan i filen `xservers`.

Eftersom skärmen finns i nätverket ingår värddatornamnet som en del i *skärmmamn*. *Skärmklass* används för att ange resurser som är specifika för en viss klass av X-terminaler. (I dokumentationen för X-terminalen finns information om hur du

anger X-terminalens skärmklass.) *Skärmtyp* för `foreign` anger för inloggningsservern att den ska ansluta till en befintlig X-server i stället för att starta en egen. I detta fall anges inte ett *X-serverkommando*.

Exempel

Följande rader i filen `ruby` gör att inloggningsservern visar två inloggningsbilder på två X-terminaler som inte stöder XDMCP (`ruby` och `wolfie`):

```
ruby.blackdog.com:0 AcmeXsta foreign
wolfie:0 PandaCo foreign
```

Kontrollera åtkomst till inloggningsservern

Normalt kan vilken värddator som helst i nätverket som har tillgång till inloggningsservern begära att en inloggningsbild ska visas. Du kan begränsa åtkomsten till inloggningsservern genom att ändra filen `Xaccess`.

Om du vill ändra `Xaccess`, kopierar du `Xaccess` från `/usr/dt/config` till `/etc/dt/config`. När du har ändrat `/etc/dt/config/Xaccess` ser du till att inloggningsservern läser `Xaccess` igen genom att skriva:

```
/usr/dt/bin/dtconfig -reset
```

Detta kör kommandot `kill -HUP inloggningsserverprocess-ID`.

Direkt XDMCP

När en värddator försöker ansluta till en inloggningsserver via direkt XDMCP, jämförs värddatornamnet med posten `Xaccess` för att avgöra om värddatorn har åtkomstbehörighet till inloggningsservern. Alla `Xaccess`-poster är värddatornamn, som kan innehålla jokertecknen `*` (asterisk) och `?` (frågetecken). En `*` (asterisk) motsvarar noll eller flera tecken och ett `?` (frågetecken) motsvarar ett enskilt valfritt tecken. Ett `!` (utropstecken) före posten innebär att värddatorn inte har åtkomstbehörighet till servern, men om det inte finns något utropstecken har värddatorn behörighet till servern.

`Text` innehåller `Xaccess` följande tre poster:

```
amazon.waterloo.com
*.dept5.waterloo.com
!*
```

Den första posten ger åtkomst till inloggningsservern från värddatorn `amazon.waterloo.com`, och den andra posten innebär att alla värddatorer vars fullständiga domännamn slutar med `dept5.waterloo.com` har åtkomstbehörighet. Den sista posten hindrar åtkomst från alla andra värddatorer.

Indirekt XDMCP

När en värddator försöker ansluta till en inloggningsserver via indirekt XDMCP, jämförs värddatornamnet med posten `Xaccess` för att avgöra om värddatorn har behörighet till inloggningsservern. Alla `Xaccess`-poster överensstämmer med posterna för direkt XDMCP, inklusive jokertecken, förutom att alla poster markeras med strängen `CHOOSE`. Till exempel:

```
amazon.waterloo.com  CHOOSE BROADCAST
*.dept5.waterloo.com  CHOOSE BROADCAST
!*  CHOOSE BROADCAST
```

Den första posten ger även här åtkomst till inloggningsservern från värddatorn `amazon.waterloo.com`, och den andra posten innebär att alla värddatorer vars fullständiga domännamn slutar med `dept5.waterloo.com` har åtkomstbehörighet. Den sista posten hindrar åtkomst från alla andra värddatorer.

Något av följande steg visas efter `CHOOSE`:

- `BROADCAST`
- *lista över värddatornamn*

`BROADCAST` gör att inloggningsservern meddelar (broadcast) inloggningsserverns undernätverk att en lista över tillgängliga inloggningsvärdar ska skapas. En lista över värddatornamn talar om för inloggningsservern att den ska använda denna lista som listan över tillgängliga inloggningsvärdar. Till exempel:

```
amazon.waterloo.com  CHOOSE shoal.waterloo.com alum.waterloo.com
*.dept5.waterloo.com  CHOOSE BROADCAST
!*  CHOOSE BROADCAST
```

Om `amazon.waterloo.com` ansluter via indirekt XDMCP, visas en lista som innehåller `shoal` och `alum`. Om `alice.dept5.waterloo.com` ansluter visas en lista över alla tillgängliga inloggningsvärdar i inloggningsserverns undernätverk. Andra förfrågningar för XDMCP indirekt avvisas.

Ett annat sätt är att definiera ett eller fler makron som innehåller en lista över värddatornamn. Till exempel:

```
%list1  shoal.waterloo.com alum.waterloo.com
amazon.waterloo.com  CHOOSE %list1
```

Felsökning

Som standard loggar inloggningsservern fel i filen `/var/dt/Xerrors`. Om du vill ändra detta anger du resursen `Dtlogin.errorLogFile` i filen `Xconfig`. Angiven katalog måste finnas när inloggningsservern startar.

Om du t ex vill att inloggningsservern ska logga fel i filen
`/var/mylogs/Derrors`, anger du följande i filen `Xconfig`:

```
Dtlogin.errorLogFile: /var/mylogs/Derrors
```

När du startar om inloggningsservern loggas fel i filen `/var/mylogs/Derrors`.
Katalogen `/var/mylogs` måste finnas när inloggningsservern startas.

Inaktivera inloggningsservern

- Om du vill inaktivera startprocessen för inloggningsservern när du startar om datorn, skriver du:

```
/usr/dt/bin/dtconfig -d
```

Detta uppmanar systemet att inte starta inloggningsservern när du startar om.

- Om du vill inaktivera inloggningsservern genom att ta bort dess process-ID, skriver du:

```
/usr/dt/bin/dtconfig -kill
```

Detta kör kommandot `kill inloggningsserverprocess-ID`.

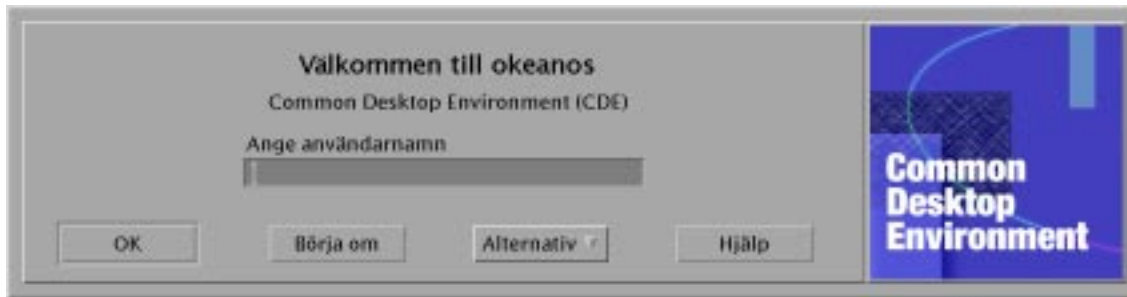
Obs! Om du tar bort inloggningsserverns process avbryts alla användarsessioner som hanteras av inloggningsservern.

Du kan också avbryta inloggningsservern genom att terminera process-ID:t.
Inloggningsserverns process-ID lagras i `/var/dt/Xpid` eller i den fil som anges i `Xconfig` av resursen `Dtlogin.pidFile`.

Om du har loggat in på skrivbordet samtidigt som du har använt programmet `kill` för att avbryta inloggningsservern, kommer skrivbordssessionen att avslutas omedelbart.

Inloggningsskärmen

Inloggningsskärmen som inloggningsservern visar är ett attraktivt alternativ till den traditionella teckenbaserade inloggningsskärmen, och den omfattar dessutom fler funktioner.



Figur 1-2 Inloggningsbilden

På samma sätt som vid en teckenbaserad inloggning skriver användaren ett användarnamn följt av ett lösenord. Om användaren verifieras startas en skrivbordsession för användaren. När användaren avslutar sessionen visas en ny inloggningsbild, och processen börjar om igen.

Om du vill anpassa inloggningsbilden kan du:

- Ändra hur inloggningsbilden ser ut
- Konfigurera X-serverbehörighet
- Ändra standardspråk
- Starta kommandon innan inloggningsbilden visas
- Ändra innehållet på inloggningsbildens Språk-meny
- Ange vilket kommando som ska starta användarens session
- Exekvera kommandon innan användarens skrivbordsession startar
- Exekvera kommandon när användarens session har avslutats

Allt detta kan göras för alla skärmar eller för en skärm i taget.

Ändra utseendet på inloggningsbilden

Du kan anpassa inloggningsbilden genom att ändra logotypen, grafiken, välkomstmeddelandet och teckensnitten.

Om du vill ändra `Xresources`, kopierar du `Xresources` från `/usr/dt/config/språk` till `/etc/dt/config/språk`. Inloggningsskärmen kommer visa alla ändringar nästa gång som inloggningsskärmen visas. Om du vill tvinga inloggningsbilden att visas en gång till, markerar du Återställ inloggningsskärm på inloggningsbildens Alternativ-meny.

Följande attribut till inloggningsbilden kan bestämmas av resursspecifikationer i filen `Xresources`:

Dtlogin*logo*bitmapFile - en bitmapsfil eller bildpunktsbeskrivningsfil kan visas som logotypbild

Dtlogin*greeting*persLabelString - eget välkomstmeddelande

Dtlogin*greeting*labelString - välkomstmeddelande

Dtlogin*greeting*fontList - Teckensnitt för välkomstmeddelande

Dtlogin*labelFont - Teckensnitt för knappar och etiketter

Dtlogin*textFont - Teckensnitt för hjälp- och felmeddelanden

Dtlogin**språk**languageName - Alternativ text för språkområdesnamn *språk*

▼ Ändra logotyp

◆ **Ange resursen** Dtlogin*logo*bitmapFile **i** Xresources.

Logotypen kan vara en färgad bildpunktsmönsterfil eller en bitmapsfil.

I följande exempel används bitmappen Minlogo som logotyp:

```
Dtlogin*logo*bitmapFile: /usr/local/lib/X11/dt/bitmaps/Minlogo.bm
```

▼ Ändra välkomstmeddelandet

Som standard visas meddelandet Välkommen till *värddatornamn* på inloggningsskärmen. Så här kan du ändra meddelandet:

◆ **Ange resursen** Dtlogin*greeting*labelString **i** Xresources.

Värdet på resursen labelString innehåller *%Lokal_värd%*, som ersätts med inloggningsserverns värddatornamn, och *%skärmmamn%*, som ersätts med X-serverns skärmmamn.

I följande exempel ändras välkomstmeddelandet till Här är *värddatornamn!*:

```
Dtlogin*greeting*labelString: Här är %Lokal_värd%!
```

När användarnamnet har skrivits in visas meddelandet Välkommen *användarnamn* som standard. Du kan byta meddelande genom att ändra resursen

Dtlogin*greeting*persLabelString i Xresources. Värdet på resursen persLabelString kan innehålla *%s*, vilket ersätts med *användarnamn*.

I följande exempel ändras det personliga välkomstmeddelandet till Hej *användarnamn*.

```
Dtlogin*greeting*persLabelString: Hej %s
```

▼ Ändra teckensnitt

Du ändrar teckensnittet som används i inloggningsbilden genom att ange följande resurs i `Xresources`:

```
Dtlogin*greeting*fontList - teckensnitt för välkomstmeddelande
```

```
Dtlogin*labelFont - teckensnitt för knappar och etiketter
```

```
Dtlogin*textFont - teckensnitt för hjälp- och felmeddelanden
```

Om du vill se en lista över tillgängliga teckensnitt skriver du:

```
xlsfonts [-alternativ] [-fn mönster]
```

I följande exempel används ett stort teckensnitt till välkomstmeddelandet (värdet du anger måste rymmas på en rad):

```
Dtlogin*greeting*fontList: -dt-interface \  
system-medium-r-normal-xxl*-*-**-*-*-*-*-*:*
```

▼ Ange olika texter för olika språk

Om du vill att texten på inloggningsbildens Språk-meny ska vara på ett visst språk i stället för standardspråket, ändrar du resursens namnresurs

```
Dtlogin*språk*languageName i Xresources:
```

```
Dtlogin*En_US*languageName: Amerikansk engelska
```

Texten `Amerikansk engelska` visas i stället för språkområdesnamnet `En_US`.

Ändra inloggningsbildens funktion

Om du vill anpassa inloggningsbildens funktion ändrar du resurserna som anges i filen `Xconfig`.

Om du vill ändra `Xconfig`, kopierar du `Xconfig` från `/usr/dt/config` till `/etc/dt/config`. När du har ändrat `/etc/dt/config/Xconfig` ser du till att inloggningsservern läser `Xconfig` igen genom att skriva:

```
/usr/dt/bin/dtconfig -reset
```

Detta kör kommandot `kill -HUP inloggningsserverprocess-ID`.

Följande resurser anges i filen `Xconfig`:

```
Dtlogin*authorize - Xaccess filspecifikation
```

```
Dtlogin*environment - X-serveromgivning
```

```
Dtlogin*language - standardspråk
```

Dtlogin*languageList - lista över språk i Språk-menyn på inloggningsskärmen
Dtlogin*resources - Xresources specifikation
Dtlogin*setup - Xsetup filspecifikation
Dtlogin*startup - Xstartup filspecifikation
Dtlogin*session - Xsession filspecifikation
Dtlogin*failsafeClient - Xfailsafe kommandofilsspecifikation
Dtlogin*reset - Xreset kommandofilsspecifikation
Dtlogin*userPath - PATH för Xsession och Xfailsafe
Dtlogin*userPath - PATH för Xsetup, Xstartup och Xfailsafe
Dtlogin*systemShell - SHELL för Xsetup, Xstartup och Xfailsafe
Dtlogin.timeZone - TZ för alla kommandofiler

Ändra hur inloggningsskärmen fungerar skärmvis

I exemplet nedan påverkar ändringar av resursen Xconfig inloggningsskärmen funktion. Resurser som visas med en asterisk (*) kan anges per skärm. Detta gör att du kan ange anpassade inloggningsskärmar för vissa skärmar. Om du vill ange en resurs för en viss skärm, anges resursen som Dtlogin**skärnmamn**resurs. Om du t ex vill stänga av användarbaserad behörighetskontroll för skärm expo:0 men vill att det ska vara aktiverat för andra skärmar, skriver du:

```
Dtlogin*expo_0*authorize: False
```

Obs! Alla specialtecken i skärnmamnet som : (kolon) eller . (punkt), byts ut mot ett _ (understreck).

Ändra X-serverbehörighet

Som standard kan X-serveråtkomstkontroll användas för inloggningsserven för varje användare. Åtkomstkontrollen bygger på den information som sparas i filen *Hemkatalog*/.Xauthority. Endast användare som har tillgång till den här filen har också tillgång till X-server. Detta är den vanligaste metoden för att kontrollera X-serverbehörighet.

Ett alternativ till användarbaserad behörighetskontroll är värddatorbaserad behörighetskontroll. Om en värddator är garanterad behörighet till X-servern, kan alla användare på den värddatorn ansluta till X-servern. Värddatorbaserad kontroll används när:

- Äldre R2- och R3-X-klienter inte kan ansluta till en X-server med användarbaserad behörighetskontroll.
- Obehöriga personer kan snappa upp behörighetsdata som skickas mellan X-klienten och X-servern i ett oskyddat nätverk.

Resursen `Xconfig Dtlogin*authorize` gör att inloggningsservern använder användarbaserad X-serverätkomstkontroll. Om du vill använda värddatorbaserad behörighetskontroll ändrar du värdet på verifieringsresursen till `False`, till exempel:

```
Dtlogin*authorize: False
```

Ändra X-serveromgivning

Om du vill ge X-servern en eller flera omgivningsvariabler och värden när den startas av inloggningsservern, anger du dem med resursen `Dtlogin*environment` i `Xconfig`. Till exempel:

```
Dtlogin*environment: VAR1=foo VAR2=bar
```

Detta gör variablerna `VAR1` och `VAR2` tillgängliga för den lokala X-serverprocessen. Dessa variabler exporteras också till kommandofilerna `Xsession` och `Xfailsafe`.

Ändra standardspråk

När användaren loggar in på skrivbordet från inloggningsskärmen körs användarsessionen med det språk som markeras på undermenyn `Språk` till `Alternativ`-menyn. Om användaren inte väljer något språk används inloggningsserverns standardspråk. Du styr värdet på standardspråket med resursen `Dtlogin*language` i `Xconfig`. Till exempel:

```
Dtlogin*language: Ja_JP
```

Information om hur du kontrollerar vilka språk som är installerade i systemet finns i systemdokumentationen.

Ändra innehållet på inloggningsskärmens Språk-meny

Normalt skapas inloggningsskärmens `Språk`-meny av en lista över de språk som finns installerade i systemet. När användaren väljer ett språk från menyn, visas inloggningsskärmen på nytt med det valda språket. När användaren loggar in nästa gång startas en skrivbordssession för användaren med det språket.

Du kan ange en egen lista över språk genom att ändra resursen `Dtlogin*languageList` i `Xconfig`:

```
Dtlogin*languageList: En_US De_DE
```

Nu visas bara `En_US` och `De_DE` på inloggningsskärmens `Språk`-meny.

Starta kommandon innan inloggningsbilden visas

När X-servern har startat men innan inloggningsbilden visas, körs kommandofilen `Xsetup`. `Xsetup` körs med rotbehörighet och startar kommandon som måste köras innan inloggningsbilden kan visas.

Om du vill ändra `Xsetup`, kopierar du `Xsetup` från `/usr/dt/config` till `/etc/dt/config`. Nästa gång inloggningsbilden visas körs den ändrade `Xsetup` när användarsessionen avslutas.

Starta kommandon innan användarsessionen

När användaren har skrivit användarnamnet och lösenordet och de har verifierats men innan användarsessionen börjar, körs kommandofilen `Xstartup`. `Xstartup` körs med rotbehörighet och startar kommandon som måste köras som rot innan användarsessionen kan börja.

Om du vill ändra `Xstartup`, kopierar du `Xstartup` från `/usr/dt/config` till `/etc/dt/config`. Nästa gång användaren loggar in körs den ändrade `Xstartup`.

Starta en skrivbordssession

Som standard startas användarsessionen genom att kommandofilen `Xsession` körs. `Xsession` körs med användarens behörighet och startar de kommandon som behövs för att starta skrivbordssystemet.

Obs! Uppdatera inte kommandofilen `Xsession` direkt.

Se Kapitel 2 om du vill ha mer information om hur du anpassar användarens startsession på skrivbordet.

Starta en Felsäker session

Om användaren väljer Felsäker session på undermenyn Sessioner på inloggningsbildens Alternativ-meny, körs skriptet `Xfailsafe`. `Xfailsafe` körs med användarbehörighet och startar de kommandon som behövs för att starta en minimal fönsteromgivning, oftast ett terminalfönster och ytterligare en fönsterhanterare.

Om du vill ändra `Xfailsafe`, kopierar du `Xfailsafe` från `/usr/dt/config` till `/etc/dt/config`. Nästa gång användaren loggar in körs den ändrade `Xfailsafe`.

När användarsessionen har avslutats

När användaren avslutar skrivbordssystemet eller en felsäker session, körs kommandofilen `Xreset`. `Xreset` körs med rotbehörighet och startar kommandon som måste köras som rot efter att användarsessionen avslutas.

Om du vill ändra `Xreset`, kopierar du `Xreset` från `/usr/dt/config` till `/etc/dt/config`. Nästa gång användaren loggar in körs den ändrade `Xreset`.

Inloggningsserverns omgivning

Inloggningsserverns omgivning exporteras till kommandofilerna `Xsetup`, `Xstartup`, `Xsession`, `Xfailsafe` och `Xreset`. Mer information om de här omgivningarna finns i Tabell 1-1. Ytterligare variabler kan också exporteras av inloggningsservern.

TABELL 1-1 Inloggningsserverns omgivningar

Omgivningsvariabel	Xsetup	Xstartup	Xsession	Xreset	Beskrivning
LANG	X	X	X	X	Standard eller valt språk
XAUTHORITY	X	X	X	X	Alternativ X-behörighetsfil
PATH	X	X	X	X	Värde för resursen <code>Dtlogin*userPath</code> (<code>Xsession</code> , <code>Xfailsafe</code>) eller resursen <code>Dtlogin*systemPath</code> (<code>Xsetup</code> , <code>Xstartup</code> , <code>Xreset</code>)
DISPLAY	X	X	X	X	X-server-anslutningsnummer
SHELL	X	X	X	X	Skal anges i <code>/etc/passwd</code> (<code>Xsession</code> , <code>Xfailsafe</code>) eller resursen <code>Dtlogin*systemShell</code> (<code>Xsetup</code> , <code>Xstartup</code> , <code>Xreset</code>)
TZ	X	X	X	X	Värdet för resursen eller tidszonen <code>Dtlogin.timeZone</code> bestäms av systemet
USER		X	X	X	Användarnamn

TABELL 1-1 Inloggningsserverns omgivningar (forts.)

Omgivnings- variabel	Xsetup	Xstartup	Xsession	Xreset	Beskrivning
HOME		X	X	X	Hemkatalogen anges i /etc/passwd
LOGNAME		X	X	X	Användarnamn

Ändra användar- eller systemsökvägen

Inloggningsservern anger omgivningsvariabeln PATH när kommandofilerna Xsession och Xfailsafe körs. Du kan bifoga en alternativ sökväg för dessa skript.

Ändra användarsökväg

- ◆ **Ange resursen** Dtlogin*userPath **i** Xconfig. **Till exempel:**

```
Dtlogin*userPath: /usr/bin:/etc:/usr/sbin:/usr/ucb:/usr/bin/X11
```

Ändra systemsökväg

- ◆ **Ange resursen** Dtlogin*systemPath **i** Xconfig. **Till exempel:**

```
Dtlogin*systemPath: /usr/bin/X11:/etc:/bin:/usr/bin:/usr/ucb
```

▼ Ändra systemskalet

Inloggningsservern anger omgivningsvariabeln SHELL när kommandofilerna Xsetup, Xstartup och Xfailsafe körs. Standard är /bin/sh. Om du vill ange ett annat skal för dessa kommandofiler, anger du resursen Dtlogin*systemShell i Xconfig. Till exempel:

```
Dtlogin*systemShell: /bin/ksh
```

▼ Ändra tidszon

Inloggningsservern anger omgivningsvariabeln TZ när kommandofilerna Xsetup, Xstartup, Xsession, Xfailsafe och Xreset körs. Standardvärdet kommer från systemet så du behöver oftast inte ändra dess upplägg. Om du vill ange en annan tidszon för dessa kommandofiler anger du resursen Dtlogin.timeZone i Xconfig. Till exempel:

```
Dtlogin.timeZone: CST6CDT
```

Administrera inloggningshanteraren

När inloggningsservern startar, startar också processen dtlogin. Under processen dtlogin avläses filen Xconfig, så att den inledande serverkonfigurationen kan bestämmas och andra konfigurationsfiler för inloggningsserven kan hittas. Inloggningsservern läser sedan filen Xservers för att kontrollera om den ska hantera några skärmar, och den läser också filen Xaccess för att kontrollera behörighet till inloggningsservern.

Om inloggningsservern får information från filen Xservers om att den måste hantera en lokal skärm, startas en X-server på det sätt som anges i filen Xservers, och sedan visas inloggningsskärmen på den skärmen.

Om inloggningsservern får information från filen Xservers om att den måste hantera en nätverksskärm, antas det att X-servern redan körs med det angivna skärmmamnet, och inloggningsskärmen visas på den skärmen.

Inloggningsservern väntar sedan på en XDMCP-förfrågan från nätverket.

För varje skärm som hanteras skapar inloggningsservern först en ny dtlogin-process för den skärmen. Detta innebär att om inloggningsservern hanterar n skärmar, finns det $n+1$ dtlogin-processer. Inloggningsservern kör kommandofilen Xsetup, öppnar filen Xresources och kör sedan dtgreet för att visa inloggningsskärmen. När användaren har skrivit användarnamnet och lösenordet och de har verifierats, körs kommandofilen Xstartup och sedan kommandofilen Xsession eller Xfailsafe. När användaren har avslutat sessionen kör inloggningsservern kommandofilen Xreset.

Om inloggningsservern får en indirekt XDMCP-förfrågan, körs dtchooser för att visa en lista över inloggningsskärmar på den skärmen. När användaren väljer en värddator från listan, kommer inloggningsservern på den värddatorn att hantera skärmen.

Inloggningsservern letar som standard efter konfigurationsfilerna Xaccess, Xconfig, Xfailsafe, Xreset, *språk*/Xresources, Xservers, Xsetup och

Xstartup, först i /etc/dt/config och därefter i /usr/dt/config. Den fil som hittas först används.

Inloggningshanterarens filer

Standardsökvägen till inloggningshanterarens filer är:

- /usr/dt/bin/dtlogin - inloggningsservern och skärmhanteraren
- /usr/dt/bin/dtgreet - visar en inloggningsbild för en skärm
- /usr/dt/bin/dtchooser - visar en valbild för en skärm
- /usr/dt/bin/Xsession - startar en skrivbordssession
- /usr/dt/config/Xfailsafe - startar en felsäker session
- /usr/dt/config/Xconfig - konfigurationsfil för inloggningsserven
- /usr/dt/config/Xservers - inloggningsserverns skärmbeskrivningsfil
- /usr/dt/config/Xaccess - inloggningsserverns åtkomstbeskrivningsfil
- /usr/dt/config/*språk*/Xresources - skärmlayoutresurser
- /usr/dt/config/Xsetup - skärminställningsfil
- /usr/dt/config/Xstartup - startfil före sessionen
- /usr/dt/config/Xreset - återställningsfil efter sessionen
- /var/dt/Xpid - process-ID för inloggningsservern
- /var/dt/Xerrors - felloggningsfil för inloggningsservern

Konfigurera sessionshanteraren

Sessionshanterarens uppgift är att starta skrivbordet och automatiskt spara och återställa färger, teckensnitt, musfunktioner, ljudvolym och tangentbordsklickningar samt program som körs.

- "Vad är en session?" på sidan 21
- "Starta en session" på sidan 23
- "När sessionen startar" på sidan 23
- "Ytterligare anpassningar av sessionsstarten " på sidan 29
- "Filer och kataloger för sessionshanteraren" på sidan 34

Med sessionshanteraren kan du:

- Anpassa startsessionen för alla skrivbordsanvändare
- Anpassa omgivning och resurser för alla skrivbordsanvändare
- Ändra sessionens startmeddelande
- Ändra parametrar för sessionens startverktyg och bakgrundsprogram
- Anpassa användningen av skrivbordsfärger för alla användare

Vad är en session?

En session är en mängd program, inställningar och resurser som visas på användarens skrivbord. Sessionshanteraren är en uppsättning konventioner och protokoll som gör att en användarsession kan sparas och återställas. En användare kan logga in i systemet och ta del av samma program, inställningar och resurser som fanns när användaren loggade ut. När användaren loggar in på skrivbordet för första

gängen kommer en standardsession att laddas. Därefter används begreppen aktiv session och hemsession.

Startsession

När en användare loggar in för första gången kommer sessionshanteraren att skapa användarens startsession med standardvärden. Normalt startas filhanteraren och hjälpvolymen Introduktion till skrivbordet.

Aktiv session

Den session som användaren kör betraktas alltid som aktiv session, oavsett om den har återställts vid inloggning från en sparad hemsession, en sparad aktiv session eller från systemets standardstartsession. Sessionshanteraren sparar automatiskt den aktiva sessionen, baserat på användarens Startinställningar i Inställningar, när användaren avslutar sessionen. När användaren loggar in nästa gång startas den tidigare sparade aktiva sessionen, vilket innebär att skrivbordet återställs till samma status som när användaren loggade ut.

Hemsession

Du kan också låta skrivbordet återställas till samma status varje gång användaren loggar in, oavsett vilken status det hade när användaren loggade ut. Användaren kan spara status i aktiv session och sedan, med Startinställningar i Inställningar, låta sessionshanteraren starta den sessionen varje gång användaren loggar in.

Skärmspecifika sessioner

Om en användare vill köra en viss session för en viss skärm, kan han eller hon skapa en skärmspecifik session. Användaren kan göra det genom att kopiera katalogen *Hemkatalog/.dt/sessions* till *Hemkatalog/.dt/skärm*, där *skärm* är det verkliga, ogiltiga värddatornamnet (`pablo:0` är t ex giltigt, medan `pablo.gato.com:0` eller `unix:0` inte är giltiga). När användaren loggar in på skärmen `pablo:0`, kommer sessionshanteraren att starta de skärmspecifika sessionerna.

Starta en session

Sessionshanteraren startas med `/usr/dt/bin/Xsession`. När användaren loggar in med inloggningshanteraren, startas `Xsession` som standard.

Användaren kan också logga in med den traditionella teckenbaserade inloggningen (`getty`) och starta sessionshanteraren manuellt med verktygen som startar en X-server, t ex `xinit`. Till exempel: `xinit /usr/dt/bin/Xsession`.

När sessionen startar

När sessionshanteraren startar går den igenom följande steg för att starta användarens session:

1. Läser in och kör kommandofilen *Hemkatalog*/`.dtprofile`
2. Läser in och kör kommandofilerna `Xsession.d`
3. Visar ett välkomstmeddelande
4. Ställer in skrivbordets sökvägar
5. Samlar ihop tillgängliga program
6. Läser eventuellt in och kör *Hemkatalog*/`.profile` eller *Hemkatalog*/`.login`
7. Startar ToolTalk[®], bakgrundsprogrammet för meddelanden
8. Laddar sessionsresurser
9. Startar färghanteraren
10. Startar arbetsytahanteraren
11. Startar sessionens program

Följande avsnitt beskriver stegen ovanför.

Läsa in och köra skriptet .dtprofile

När sessionen startar, körs kommandofilen `Xsession`, som läser in och kör användarens *Hemkatalog*/`.dtprofile`-kommandofil.

Hemkatalog/`.dtprofile`-kommandofilen är en `/bin/sh` eller `/bin/ksh`-kommandofil som gör att användaren kan ange omgivningsvariabler för sessionen. Mer information om hur du anger omgivningsvariabler finns i "Ytterligare anpassningar av sessionsstarten " på sidan 29.

Om *Hemkatalog*/`.dtprofile`-kommandofilen inte finns första gången som användaren loggar in på skrivbordssystemet kommer `Xsession` att kopiera skrivbordsstandardens `sys.dtprofile` till *Hemkatalog*/`.dtprofile`.

Skrivbordets standardvärde är `/usr/dt/config/sys.dtprofile`. Om du vill ändra `sys.dtprofile`-kommandofilen, kopierar du `sys.dtprofile` från `/usr/dt/config` till `/etc/dt/config` och redigerar den nya filen.

Läsa in och köra skriptet Xsession.d

När *Hemkatalog*/`.dtprofile`-kommandofilen har lästs in och körts, läser kommandofilen `Xsession` in kommandofilen `Xsession.d` och kör den. Detta skript används för att allokeras omgivningsvariabler och starta eventuella bakgrundsprogram för användarens session. Standardbeskrivningen för `Xsession.d`-kommandofiler är:

0010.`dtpaths` – anpassningsbara dokument sökvägar för skrivbordet
0020.`dtims` – startar valfri indata metoder server
0030.`dttmpdir` – skapar en tillfällig katalog för varje användare och session
0040.`xmbind` – anger `$XMBINDDIR` som skrivbordsstandard

Det kan finnas ytterligare kommandofiler från olika leverantörer i `Xsession.d`.

`Xsession` läser först in och kör alla filer i `/etc/dt/config/Xsession.d` och därefter filerna i katalogen `/usr/dt/config/Xsession.d`.

Skrivbordets `Xsession.d`-standardkommandofiler finns i katalogen `/usr/dt/config/Xsession.d`. Om du vill ändra `Xsession.d`-kommandofilen, kopierar du kommandofilen från `/usr/dt/config/Xsession.d` till `/etc/dt/config/Xsession.d` och redigerar den nya filen. Du måste ha exekveringsbehörighet för att kunna göra detta.

Om du vill att `Xsession` automatiskt ska läsa in och köra en egen kommandofil, kopierar du den till `/etc/dt/config/Xsession.d`.

Obs! När du ändrar eller skapar en `Xsession.d`-kommandofil, måste du kontrollera att alla förgrundskommandon du kör är korta, eftersom den tid som kommandot tar direkt påverkar starttiden. Om ett förgrundskommando inte avslutas hänger sig sessionsstarten. Kommandon som körs i en `Xsession.d`-kommandofil och som du vill fortsätta köra under hela sessionen ska köras i bakgrunden.

Visa välkomstmeddelandet

När *Hemkatalog*/`.dtprofile`- och `Xsession.d`-kommandofilerna har lästs in och körts visar `Xsession` ett välkomstmeddelande på skärmen. Du kan ändra välkomstmeddelandet eller stänga av det helt och hållet. Klienten `dthello` används för att visa meddelandet.

Om du vill byta meddelandetext ändrar du `dthello`-alternativen genom att redigera variabeln `dtstart_hello[0]`.

Om du vill ändra `dtstart_hello[0]`, skapar du en `/etc/dt/config/Xsession.d`-kommandofil som anger det nya värdet. Om du vill visa dagens meddelande för alla användare, skapar du en körbar `sh`- eller `ksh`-kommandofil, t ex `/etc/dt/config/Xsession.d/minavar` och ställer in `dtstart_hello[0]` så här:

```
dtstart_hello[0]="/usr/dt/bin/dthello -file /etc/motd &"
```

På samma sätt kan användarna ändra välkomstmeddelandet för sina sessioner genom att ange `dtstart_hello[0]` i *Hemkatalog*/`.dtprofile`.

Om du vill stänga av välkomstmeddelandet anger du `dtstart_hello[0]=" "`.

Mer information om `dthello` finns i avsnittet om `dthello` i direkthjälpen (man pages).

Ställa in skrivbordets sökvägar

Skrivbordssökvägar skapas vid inloggning av `dtsearchpath`. Det finns två kategorier av omgivningsvariabler som används av `dtsearchpath`:

Indatavariabler – Systemtäckande och egna omgivningsvariabler vars värden anges av systemadministratören eller av slutanvändaren.

Utdatavariabler – Variabler som har skapats och tilldelats värden av `dtsearchpath`. Värdet på varje variabel är sökvägen till skrivbordssessionen.

Om du vill ändra kommandoradsalternativ för `dtsearchpath` ändrar du variabeln `dtstart_searchpath`. Om du vill ändra variabeln `dtstart_searchpath` för alla användare, skapar du en körbar `sh`- eller `ksh`-kommandofil (t ex

/etc/dt/config/Xsession.d/minavar), och anger `dtstart_searchpath` så här:

```
dtstart_searchpath="/usr/dt/bin/dtsearchpath"
```

Användarna kan bara ändra `dtsearchpath`-alternativ för sina egna sessioner genom att ange `dtstart_searchpath` i *Hemkatalog*/.dtprofile.

Mer information om `dtsearchpath` finns i Kapitel 9. Mer information om `dtsearchpath`-alternativ finns i direkthjälpavsnittet (man page) om `dtsearchpath`.

Samla ihop tillgängliga program

När skrivbordssökvägar har angetts är nästa steg att samla ihop tillgängliga program med hjälp av `dtappgather`. Om du vill ändra kommandoradsalternativ för `dtappgather` ändrar du variabeln `dtstart_appgather`. Om du vill ändra variabeln `dtstart_appgather` för alla användare, skapar du en körbar sh- eller ksh-kommandofil (t ex /etc/dt/config/Xsession.d/minavar), och anger `dtstart_appgather` så här:

```
dtstart_appgather="/usr/dt/bin/dtappgather &"
```

Användarna kan bara ändra `dtappgather`-alternativ för sina egna sessioner genom att ange `dtstart_appgather` i *Hemkatalog*/.dtprofile.

Mer information om `dtappgather` -alternativ finns i direkthjälpavsnittet (man page) om `dtappgather(4)`.

Läsa in och köra skripten .profile och .login (valfritt)

`Xsession` kan läsa in och köra en användares vanliga *Hemkatalog*/.profile- eller *Hemkatalog*/.login-kommandofiler. Som standard gäller inte de här skripten. Du anger för `Xsession` att .profile- eller .login-kommandofilen ska läsas in genom att ange `true` för `DTSOURCEPROFILE`.

Om du vill ändra `DTSOURCEPROFILE` för alla användare skapar du en /etc/dt/config/Xsession.d-kommandofil som anger det nya värdet. Om du vill ange `true` för `DTSOURCEPROFILE` för alla användare, skapar du en körbar sh- eller ksh-kommandofil (t ex /etc/dt/config/Xsession.d/minavar), och anger `DTSOURCEPROFILE` så här:

```
DTSOURCEPROFILE=true
```

Användarna kan på samma sätt ändra `DTSOURCEPROFILE` för sina egna sessioner genom att ange `true` för `DTSOURCEPROFILE` i *Hemkatalog*/.dtprofile.

Starta ToolTalk

Med hjälp av ToolTalk, bakgrundsprogrammet för meddelanden, `ttsession`, kan oberoende program kommunicera utan att ha direkt kännedom om varandra. Programmen skapar och skickar ToolTalk-meddelanden för att kommunicera med varandra. `ttsession` kommunicerar i nätverket för att leverera meddelanden.

Om du vill ändra kommandoradsalternativ för `ttsession` ändrar du variabeln `dtstart_ttsession`. Om du vill ändra variabeln `dtstart_ttsession` för alla användare, skapar du en körbar `sh`- eller `ksh`-kommandofil (t ex `/etc/dt/config/Xsession.d/minavar`) och anger `dtstart_ttsession` så här:

```
dtstart_ttsession="/usr/dt/bin/ttsession -s"
```

Användarna kan på samma sätt ändra `ttsession`-alternativ för sina egna sessioner genom att ange `dtstart_ttsession` i *Hemkatalog*/`.dtprofile`.

Mer information om `ttsession`-alternativ finns i direkthjälpavsnittet (man page) om `ttsession`. Mer information om `ttsession` finns i *Common Desktop Environment: ToolTalk Messaging Overview*.

Starta sessionshanterarklienten

Nu startar `Xsession /usr/dt/bin/dtsession`, som innehåller sessionens startprocess.

Ladda sessionsresurser

Sessionshanteraren använder X-serveregenskapen `RESOURCE_MANAGER` för att göra skrivbordsresurser tillgängliga för alla program. Sessionshanteraren öppnar `RESOURCE_MANAGER` genom att:

- Ladda systemets standardresurser
- Slå samman alla systemtäckande resurser som anges av systemadministratören.
- Slå samman användarspecificerade resurser.

Skrivbordets standardresurser finns i `/usr/dt/config/språk/sys.resources`. Dessa resurser görs tillgängliga för alla användarsessioner via egenskapen `RESOURCE_MANAGER`. Den här filen bör inte redigeras eftersom den skrivs över vid varje skrivbordsinstallation.

Du kan utvidga systemets standardresurser genom att skapa `/etc/dt/config/språk/sys.resources`. I den här filen kan du åsidosätta standardresurserna eller ange ytterligare resurser för alla skrivbordsanvändare. Eftersom den här filen kopplas samman med skrivbordets standardresurser när sessionen startar, ska endast nya eller uppdaterade resursspecifikationer placeras i den här filen. Resurser som specificeras i filen blir tillgängliga för alla

användarsessioner via egenskapen `RESOURCE_MANAGER`. Resurser som anges i den här filen har företräde framför de som anges i resursfilen för standardskrivbordet.

Användare kan utvidga skrivbordets standardresurser och systemtäckande resurser med filen *Hemkatalog/.Xdefaults*. Resurser som anges i den här filen blir tillgängliga för alla användarsessioner via egenskapen `RESOURCE_MANAGER`. Resurser som anges i den här filen har företräde framför de som anges i resursfiler för standardskrivbordet eller resursfiler från systemadministratörer.

Obs! Hjälpprogrammet X Toolkit Intrinsic anger att det öppnar resurser för ett program antingen från `RESOURCE_MANAGER` eller från *Hemkatalog/.Xdefaults*, men inte från båda. Vanligtvis innebär detta att användarens *Hemkatalog/.Xdefault*-fil ignoreras. Sessionshanteraren kan dock hantera *Hemkatalog/.Xdefaults* genom att den slås samman med `RESOURCE_MANAGER` när sessionen startar, vilket beskrivs ovan. Om en användare ändrar *Hemkatalog/.Xdefaults*, blir ändringarna inte synliga för nya program förrän användaren anropar åtgärden Ladda om resurser. Åtgärden Ladda om resurser instruerar sessionshanteraren om att `RESOURCE_MANAGER` ska startas om med standardresurser, systemtäckande resurser och användarspecificerade resurser. På detta sätt görs ändringar av systemtäckande och egna resursfiler tillgängliga för alla program.

Mer information finns i

- "Ange programresurser" på sidan 265
- Direkthjälpen (man) för `dtresourcesfile(4)`

Starta färghanteraren

Sessionshanteraren fungerar som en färghanterare för skrivbordssystemet och tillhandahåller följande uppsättning av `dtsession`-resurser som kan användas för att konfigurera den.

`foregroundColor` – kontrollerar om en bildpunkt är tillgänglig för förgrundsfasen

`dynamicColor` – anger om färger som inte kan redigeras är tillgängliga

`shadowPixmap` – anger om färger ska ha en övre eller undre skugga

`colorUse` – begränsar färgtilldelning

`writeXrdbColors` – anger om `*background-` eller `*foreground-`resurserna placeras i resursdatabasen

Du kan ange färghanteraresurser för alla användare genom att skapa `/etc/dt/config/språk/sys.resources` och ange färghanteraresurser i den filen.

Användarna kan på samma sätt ange färghanteraresurser för sina egna sessioner genom att ange resurserna i *Hemkatalog/.Xdefaults*.

Mer information om hur du anger färghanteraresurser finns i "Hantera färger" på sidan 277.

Starta arbetsytahanteraren

Sessionshanteraren startar arbetsytahanteraren. Normalt startas `/usr/dt/bin/dtwm`. En alternativ fönsterhanterare kan anges med resursen `wmStartupCommand`.

Du kan ange en annan fönsterhanterare för alla användare genom att skapa `/etc/dt/config/språk/sys.resources` och ange den fullständiga sökvägen till fönsterhanteraren och alternativ för den med resursen `Dtsession*wmStartupCommand` i den filen.

Användarna kan bara ange en annan fönsterhanterare för egna sessioner genom att ange resursen `Dtsession*wmStartupCommand` i *Hemkatalog*/`.Xdefaults`.

Mer information om fönsterhanteraren finns i Kapitel 16.

Starta sessionsprogram

När sessionen startar, startar sessionshanteraren om alla program som sparades som en del av sessionen. Systemets standardprogramuppsättning, som ska återställas som en del av användarens startsession, finns i `/usr/dt/config/språk/sys.session`. Den här filen ska inte redigeras eftersom den skrivs över vid kommande skrivbordsinstallationer.

Mer information finns i direkthjälpen (`man`) för `dtactionfile(4)`.

En systemadministratör kan ersätta den programuppsättning som startas som en del av användarens startsession genom att kopiera `/usr/dt/config/språk/sys.session` till `/etc/dt/config/språk/sys.session` och ändra den senare filen. Till skillnad från resursfilen används den här filen som en fullständig ersättning till skrivbordets standardfil, så du kan skapa en kopia av systemets standardfiler och göra nödvändiga ändringar.

Ytterligare anpassningar av sessionsstarten

I det här avsnittet beskrivs hur du:

- Anger omgivningsvariabler

- Anger resurser
- Använder skärmberoende sessioner
- Kör skript vid inloggning
- Återskapar en säkerhetskopierad session

▼ Ange omgivningsvariabler

- ◆ Om du vill ange systemtäckande omgivningsvariabler skapar du en fil i katalogen `/etc/dt/config/Xsession.d` som anger och exporterar variabeln.

Om du t ex skapar en körbar ksh-kommandofil, `/etc/dt/config/Xsession.d/minavar`, som innehåller:

```
export MINVARIABEL="värde"
```

anges variabeln `MINVARIABEL` i alla användares omgivning vid nästa inloggning.

- ◆ Om du vill ange egna omgivningsvariabler anger du variabeln i *Hemkatalog*/`.dtprofile`.

Till exempel:

```
export MINVARIABEL="värde"
```

anger variabeln `MINVARIABEL` i alla användares omgivning vid nästa inloggning.

Obs! Sessionshanteraren läser inte automatiskt filerna `.profile` eller `.login`. Du kan dock konfigurera den så att dessa filer används. Se "Läsa in och köra skripten `.profile` och `.login` (valfritt)" på sidan 26.

▼ Ange resurser

- ◆ Om du vill ange systemtäckande resurser lägger du till resurserna i filen `/etc/dt/config/språk/sys.resources`. (Först måste du skapa filen.)

Obs! `.dtprofile` stöder bara `/bin/sh-` eller `/bin/ksh-syntax`.

Om du t ex specificerar följande i `/etc/dt/config/C/sys.resources`:

```
AnApplication*resource: värde
```

anges resursen `AnApplication*resource` i varje användares `RESOURCE_MANAGER`-egenskap vid nästa inloggning.

- ◆ Om du vill ange egna resurser lägger du till resurserna i filen *Hemkatalog/.Xdefaults*.

▼ Ange skärmspecifika resurser

Du kan ange skärmspecifika resurser för alla användare i systemet. Användarna kan också ange skärmspecifika resurser för sina egna sessioner. Du kan därför ange resurser beroende på vilken skärm som användaren använder för att logga in till skrivbordet.

- ◆ Om du vill ange skärmspecifika resurser för alla skrivbordsanvändare i systemet skapar du filen `/etc/dt/config/språk/sys.resources` som anger den skärmspecifika resursen.

- ◆ Om du vill ange egna skärmspecifika resurser anger du resursen i *Hemkatalog/.Xdefaults*.

Du begränsar resurser genom att innesluta dem i `cpp`-villkorsprogramsatser. Makrot `DISPLAY_skärnmamn` definieras beroende av värdet i variabeln `DISPLAY_`. Detta görs genom att alla punkt- och kolontecken omvandlas till understrykning (`_`) och alla skärmspecifikationer tas bort, och slutligen läggs `DISPLAY_` till som prefix till resultatet.

`$DISPLAY` för `:0` blir t ex `DISPLAY_0`, och `$DISPLAY` för `blanco.gato.com:0.0` blir `DISPLAY_blanco_gato_com_0`. Det resulterande värdet kan användas som en del i ett `cpp`-test i en sessionsresursfil. Om du t ex specificerar följande i `/etc/dt/config/C/sys.resources`:

```
MittPrgm*resource: värde
```

```
#ifdef DISPLAY_blanco_gato_com_0
MittPrgm*resource: specialvärde1
#endif
```

```
#ifdef DISPLAY_pablo_gato_com_0
MittPrgm*resource: specialvärde2
#endif
```

anges resursen `MittPrgm*resource` i `RESOURCE_MANAGER` till `specialvärde1` när användaren loggar in på skärmen `blanco.gato.com:0`, `specialvärde2` när användaren loggar in på `pablo.gato.com:0`; och `värde` när användaren loggar in på en annan skärm.

▼ Ändra program för startsessionen

Du kan ange att andra program ska starta som en del av användarens startsession.

1. **Kopiera** `/usr/dt/config/språk/sys.session` till `/etc/dt/config/språk/sys.session`.

2. **Ändra den nya filen** `sys.session`.

Alla poster i `sys.session` ser ut så här:

```
dtsmcmd -cmd kommandon_och_alternativ
```

Om du vill starta ytterligare ett program som en del av användarens startsession, anger du en ny `sys.session`-post med fullständig sökväg. Om du t ex vill starta `/usr/bin/X11/xclock` som en del av användarens startsession, lägger du till posten `xclock` till `/etc/dt/config/C/sys.session`:

```
#  
# Starta xclock...  
#  
dtsmcmd -cmd "/usr/bin/X11/xclock -digital"
```

▼ Ange en skärmspecifik session

En användare kan ange en skärmspecifik session för att justera sessionen för en viss skärm.

- ◆ **Kopiera katalogen *Hemkatalog*** `/.dt/sessions` till ***Hemkatalog*** `/.dt/skärm`, där *skärm* är det verkliga, ogiltiga värddatornamnet (`pablo:0` är t ex giltigt, medan `pablo.gato.com:0` eller `unix:0` inte är giltiga).

Du kan t ex skapa en skärmspecifik session för skärmen `pablo.gato.com:0`:

```
cp -r Hemkatalog/.dt/sessions Hemkatalog/.dt/pablo:0
```

När användaren loggar in på skärmen `pablo.gato.com:0`, startar sessionshanteraren den skärmspecifika sessionen.

Köra ytterligare kommandon vid sessionsstart och utloggning

Användarna kan ange att ytterligare kommandon ska startas när de loggar in till skrivbordsessionen. Detta är användbart vid X-inställningar som inte sparas av

sessionshanteraren. Användaren kan t ex använda `xsetroot` för att anpassa rotpekaren (arbetsyta). Ett annat användningsområde är att starta program som inte kan sparas och återställas av sessionshanteraren. Om ett program inte vill starta när sessionen startar, kan användaren starta klienten med den här metoden.

▼ Köra ytterligare kommandon vid sessionsstart

- ◆ **Skapa filen *Hemkatalog*/.dt/sessions/sessionetc som innehåller kommandona.**

Normalt är den här filen ett skript som måste ha exekveringsbehörighet. Processer som startas i `sessionetc` ska köras i bakgrunden.

Obs! Använd inte `sessionetc` för att starta klienter som automatiskt återställs av sessionshanteraren. Om du gör det kan flera kopior av programmet startas. Du kanske inte ser alla kopior omedelbart eftersom fönstren kan staplas på varandra.

▼ Köra ytterligare kommandon vid utloggning

En kompletteringsfil till `sessionetc` är `sessionexit`. Använd `sessionexit` för att utföra vissa bearbetningar vid sessionsavslutningar som inte hanteras av sessionshanteraren.

- ◆ **Skapa filen *Hemkatalog*/.dt/sessions/sessionexit som innehåller kommandona.**

Liksom `sessionetc` är den här filen normalt en kommandofil med exekveringsbehörighet.

▼ Återskapa en session från säkerhetskopia

När sessionshanteraren sparar en session lagras sessionsinformationen katalogen *Hemkatalog*/.dt/sessions, eller i katalogen *Hemkatalog*/.dt/skärm om en skärmspecifik session används. I de här katalogerna skapas en underkatalog med namnet `current` eller `home` som lagrar information för den aktiva sessionen eller hemsessionen. Innan sessionsinformationen lagras, skapar sessionshanteraren en säkerhetskopia av den tidigare sessionen med det namnet och lagrar den i `current.old` eller `home.old`.

1. **Logga in med Felsäker session eller Kommandorad från inloggningsfönstret.**
2. **Kopiera katalogen för den säkerhetskopierade sessionen till aktivt namn, t ex till säkerhetskopierad hemsession:**

```
cp -r Hemkatalog/.dt/sessions/home.old \  
    Hemkatalog/.dt/sessions/home
```

En skärmspecifik session kan återskapas på samma sätt.

▼ Undersöka problem vid sessionsstart

- ◆ **Kontrollera filen *Hemkatalog*/.dt/startlog.**

Sessionshanteraren loggar varje användares startsession i den här filen.

Filer och kataloger för sessionshanteraren

- /usr/dt/bin/Xsession
- /usr/dt/config/Xsession.d/*
- /usr/dt/bin/dtsession
- /usr/dt/bin/dtsession_res
- *Hemkatalog*/.dt/sessions/current
- *Hemkatalog*/.dt/sessions/home
- *Hemkatalog*/.dt/*skärm*/current
- *Hemkatalog*/.dt/*skärm*/home

Felsökning för inloggning och problem vid sessionstart

I det här kapitlet finns flera avsnitt med information om Solaris CDEs startfiler, möjliga problem vid start av Solaris CDE och förslag på lösningar av startproblem.

- "Startfiler för inloggning" på sidan 35
- "Felloppningskataloger" på sidan 36
- "Användarstartfil" på sidan 37
- "Solaris CDE, startexempel" på sidan 37

Startfiler för inloggning

Efter att inloggningshanteraren för Solaris CDE har verifierat användaren, anropar den följande startskript för att starta skrivbordet:

```
/usr/dt/bin/Xsession
```

Den första användarspecifika fil som `Xsession` anropar är *Hemkatalog*/`.dtprofile`.

Första gången en ny användare loggar in i Solaris CDE, kopieras filen `.dtprofile` till användarens hemkatalog. Filen är förinställd att inte utföra någonting. Den innehåller däremot ett antal kommentarer om på vilka sätt den kan redigeras. Användaren kan redigera den här filen och lägga till användarspecifika omgivningsvariabler.

Obs! Precis som `Xsession`-kommandofilen som anropar `.dtprofile`, använder den här filen `ksh`-syntax.

En användbar redigeringsåtgärd är att kommentera bort sista raden i filen `.dtprofile`:

```
DTSOURCEPROFILE=true
```

Den här raden anger att användarens *Hemkatalog*/`.login` (för `csch`-användare) eller *Hemkatalog*/`.profile` (för användare av andra skal) ska läsas in vid startprocessen.

Felloggningskataloger

Välj Felsäker session på Alternativ-menyn i inloggningsbilden i Solaris CDE om du vill kringgå den normala Xsession-starten och undersöka felloggningskataloger samt åtgärda problem med användarpunktfiler. Tabell 3-1 visar felloggar och var de är placerade.

TABELL 3-1 Felloggningskataloger

Placering	Felloggningsfiler
<code>/var/dt/Xerrors</code>	Systemfel i inloggningsfönstret för Solaris CDE med tidigare ursprung än användarens inloggning
<i>Hemkatalog</i> / <code>.dt/startlog</code>	Startfel i Solaris CDE under Xsession, <code>.dtprofile</code> , <code>.login</code> eller <code>.profile</code>
<i>Hemkatalog</i> / <code>.dt/errorlog</code>	Solaris CDE-fel efter Xsession-start
<i>Hemkatalog</i> / <code>.dt/sessionlogs</code>	Katalog för sessionsloggar för fel i sessionshanteraren och fönsterhanteraren

Vissa fel visas även i Systemfönstret. Om inte Systemfönstret är aktiverat är namnet på systemets reservloggningsfil `wscn` med tid- och visningskoderna sammanlänkade. Till exempel:

```
/usr/tmp/wscnAAAA004EE:0.0
```

Användarstartfil

Om du vill att filen `.login` (för `ssh`-användare) eller filen `.profile` (för `sh`- eller `ksh`-användare) ska läsas in och köras, kommenterar du bort sista raden i `.dtprofile`-filen som ska läsas:

```
DTSOURCEPROFILE=true
```

I normala fall är detta den enda åtgärd som behövs. I vissa fall behöver du ändra i filerna `.login` eller `.profile` eftersom de kan innehålla kommandon som inte är kompatibla med inloggningshanteraren i Solaris CDE. Om det finns ett fel i något kommando kan det bero på att filen förväntar sig en indatakälla från terminalens tangentbord, t ex `stty`, `tset` eller ett "vänta på indata"-kommando.



Varning! Om `.dtprofile` är inställt för att läsa in en `.login`- eller `.profile`-fil och den innehåller felaktiga kommandon som kraschar skalet, kommer försöket att starta skrivbordet att misslyckas. Skrivbordet kommer inte att visas. Istället kommer Solaris CDE:s inloggningsbild att visas på nytt. Startfel från `.login` eller `.profile` visas i *Hemkatalog*/`.dt/startlog`. Använd en felsäker inloggnings-session eller kommandoradsinloggning för att söka efter felaktiga kommandon i `.login` eller `.profile`.

Mer information om olika problem och lösningar finns i filen `.dtprofile`. Normalt sett är felaktiga kommandon kopplade till terminalinformation och styrning.

Solaris CDE, startexempel

Det här avsnittet innehåller förslag på hur du kan redigera följande användarstartfiler:

- `.login` (för `ssh`-användare)
- `.profile` (för `sh`- eller `ksh`-användare)
- `.Xdefaults`

Startprocessen för Solaris CDE anger en skalvariabel som heter `DT` i kommandofilen `.login` eller `.profile` så att du kan genomföra en kontroll under Solaris CDE-startprocessen. Detta förhindrar att terminalrelaterade kommandon, t ex `stty` och `stty`, körs. Hänvisa till följande skriptexempel:

```
.login (C-skal)
if ( ! ${?DT} ) then
```

```

    stty erase '^h'

endif

.profile (sh eller ksh)
if [ ! ``$DT'' ]; then

    stty erase '^h'

fi

```

Denna extra syntax runt DT stoppar varningsmeddelanden från C-skalet när DT inte är definierad. DT är inte definierad när användaren loggar in från en textbaserad inloggningsledning i systemfönstret. Mer information om hur du ställer in punktfiler finns i filen *Hemkatalog/.dtprofile*.

Obs! Du anger tty-inställningar för Solaris CDE i *.Xdefaults*.

Användarens *.Xdefaults*-fil används som källa för användarspecifika resurser vid starten av Solaris CDE. *ttyModes* anger t ex att användarens favoritinställningar för tty används i terminalemuleringsfönster, t ex *dtterm* och *xterm*. Följande rad visar en normal *ttyModes*-inställning i filen *.Xdefaults*:

```
*ttyModes: erase ^H intr ^C kill ^U start ^Q stop ^S susp ^Z`
```

Obs! Favoritresurser för Solaris CDE skiljer sig från standardinställningarna.

Fönsterhanteraren i skrivbordets placering av ikoner är ett exempel på en standardinställning. Följande rad i filen *.Xdefaults* visar standardinställningen för ikonplacering:

```
Dtwm*iconPlacement: right top
```

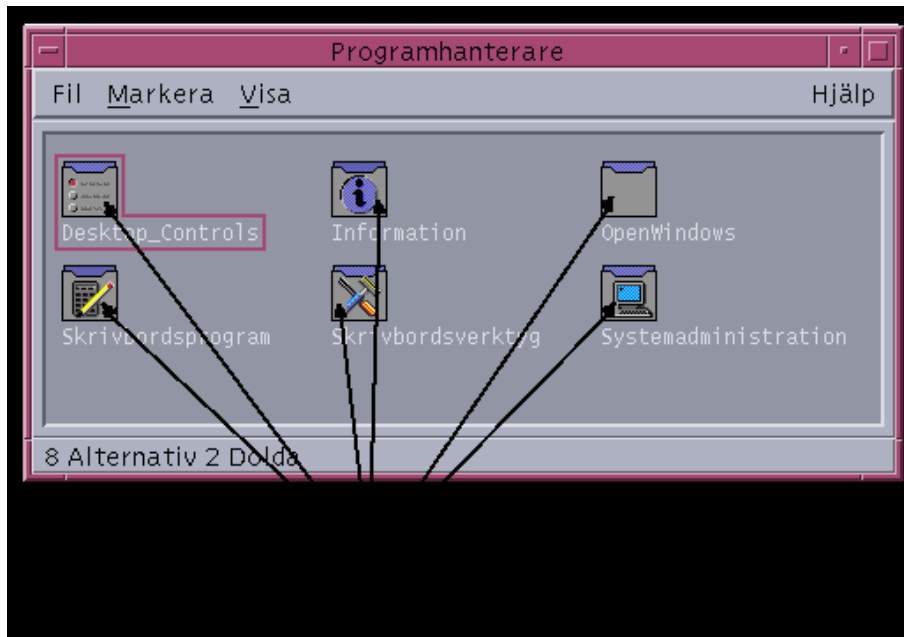
Lägga till och hantera program

Programhanteraren är den funktion på skrivbordet som innehåller program som är tillgängliga för användaren.

- "Programhanterarens struktur" på sidan 39
- "Lägga till program i programhanteraren" på sidan 43
- "Skapa och hantera allmänna programgrupper" på sidan 46
- "Ändra sökvägen som används för att söka efter program" på sidan 47
- "Allmän administration av programhanteraren" på sidan 49
- "Ändra textredigeraren och terminalemulatorn" på sidan 50

Programhanterarens struktur

Programhanterarens översta nivå innehåller normalt kataloger. Var och en av dessa kataloger, med innehåll, kallas för en *programgrupp*.



Figur 4-1 Programgrupper i programhanteraren

Programgrupperna och deras innehåll samlas in från olika platser lokalt och från nätverket.

Programhanterarens plats i filsystemet

I filsystemet ligger programhanteraren i katalogen

`/var/dt/appconfig/appmanager/inloggningsnamn-värddatornamn-skärm.`

Katalogen skapas dynamiskt varje gång användaren loggar in.

Om t ex användaren karinj loggar in från skärmen wxyz:0, skapas katalogen

`/var/dt/appconfig/appmanager/karinj-wxyz-0.`

Hur programhanteraren hittar och samlar ihop program

Programhanteraren utgörs av insamlade lokala programgrupper och programgrupper i nätverket. Programgrupperna samlas in från kataloger i programsökvägen.

Standardsökvägen för program består av platserna i Tabell 4-1.

TABELL 4-1 Standardsökväg för program

Definitionsområde	Placering
Inbyggd	<code>/usr/dt/appconfig/appmanager/språk</code>
Systemtäckande	<code>/etc/dt/appconfig/appmanager/språk</code>
Egen	<i>Hemkatalog</i> /.dt/appmanager

Vid inloggningen skapas programhanterarens översta nivå genom att länkar skapas från programgrupperna (kataloger) i programsökvägens kataloger till programhanterarkatalogen

`/var/dt/appconfig/appmanager/inloggningsnamn-värddatornamn-skärm.`

Insamlingen utförs av skrivbordsverktyget `dtappgather`, som körs automatiskt av inloggningshanteraren efter att användaren har loggat in.

På skrivbordet finns t ex den inbyggda programgruppen:

`/usr/dt/appconfig/appmanager/språk/Desktop_Tools`

Vid inloggningen skapas en symbolisk länk till:

`/var/dt/appconfig/appmanager/inloggning-värddatornamn-skärm/Desktop_Tools`

Programsökvägen kan omfatta kataloger i nätverket. På detta sätt kan programgrupper samlas in från olika system i nätverket. Mer information finns i "Lägga till en programserver i programsökvägen" på sidan 48.

Prioritetsregler för insamling av program

Vid dubbleringar i sökvägen har egna programgrupper prioritet över systemtäckande grupper, och systemtäckande grupper har prioritet över inbyggda grupper. Om t ex både `/usr/dt/appconfig/appmanager/C/Desktop_Tools` och `/etc/dt/appconfig/appmanager/C/Desktop_Tools` finns, används programgruppen under `/etc`.

Programgrupper på standardskrivbordet

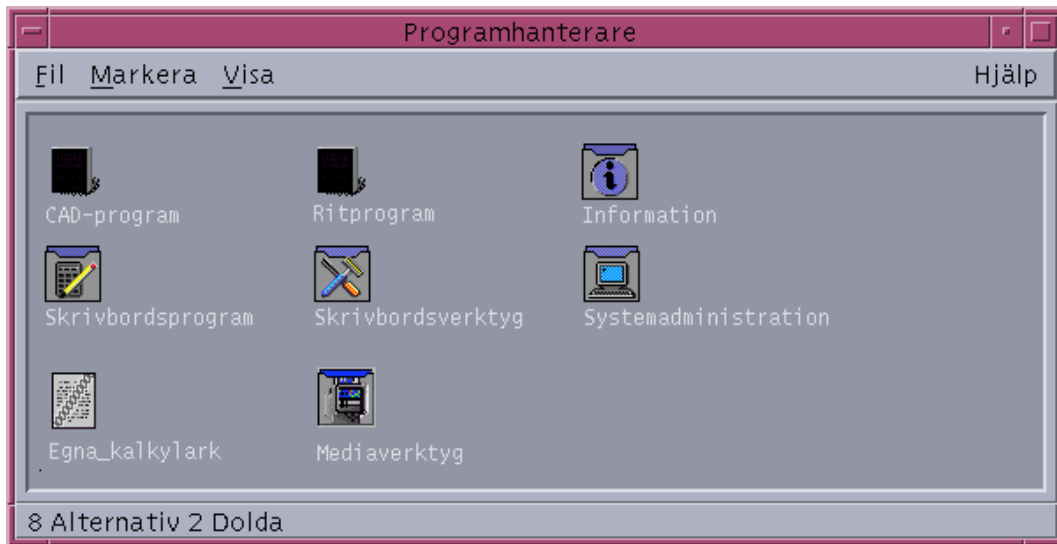
På ett skrivbord som inte är anpassat finns fyra programgrupper.

- Skrivbordsprogram
- Skrivbordsverktyg

- Desktop_Controls
- Information
- Systemadministration

Exempel på hur programgrupper samlas in

I Figur 4-2 visas ett fönster i programhanteraren som innehåller en rad olika programgrupper. I Tabell 4-2 visas vilka kataloger som programgrupperna har samlats från.



Figur 4-2 Ett typiskt programhanterarfönster

TABELL 4-2 Källor till programgrupperna i Figur 4-2

Namn	Insamlad katalog
CAD-program	<code>/net/ApServA/etc/dt/appconfig/appmanager/C/CAD_App</code>
Ritprogram	<code>/etc/dt/appconfig/appmanager/C/DrawingApp</code>
Skrivbordsprogram	<code>/usr/dt/appconfig/appmanager/C/Desktop_Apps</code>
Skrivbordsverktyg	<code>/usr/dt/appconfig/appmanager/C/Desktop_Tools</code>

TABELL P-2 Källor till programgrupperna i Figur 4-2 (forts.)

Namn	Insamlad katalog
Information	/usr/dt/appconfig/appmanager/C/Information
OpenWindows	/usr/dt/appconfig/appmanager/C/Information
Systemadministration	/etc/dt/appconfig/appmanager/C/System_Admin
Egna_kalkylblad	/users/anna/.dt/appmanager/MySpreadSheet
Mediaverktyg	/etc/dt/appconfig/appmanager/C/Media_Tools

Om programgrupperna Information eller Systemadministration har anpassats, kommer de i stället att samlas in från /etc/dt/appconfig/appmanager/C.

Gruppen CAD-program samlas in eftersom systemet ApServA har lagts till programsökvägen (se "Lägga till en programserver i programsökvägen" på sidan 48). Egna_kalkylblad är en personlig programgrupp som bara är tillgänglig för användaren anna.

Lägga till program i programhanteraren

När ett program har lagts till i programhanteraren kan det startas med en ikon i en programgrupp.

Många program har egna programgrupper. Programgruppen är en katalog på programhanterarens översta nivå som innehåller programikon och andra filer som hör till programmet.

Vissa program har inga egna programgrupper. I stället finns dess startikon i en allmän programgrupp. Du kan t ex skapa en tom programgrupp med namnet "Spel" som behållare för alla spel som du installerar i systemet.

Lägga till program i programhanteraren

Det finns två sätt att lägga till program i programhanteraren:

- Registrera programmet

- Lägga till en programikon utan att registrera programmet

Registrera program

Programregistrering ger en fullständig programintegration.

Ett registrerat program:

- Har en egen programgrupp.
- Har sina konfigurationsfiler för skrivbordet samlade på ett ställe. Dessa konfigurationsfiler kallas för *registreringspaket*.
- Kan ha en registrerad hjälpvolym.

Program kan registreras på två sätt:

- När du installerar ett skrivbordsanpassat program sker registreringen automatiskt. Se "Lägga till skrivbordsanpassade program i programhanteraren" på sidan 44.
- Genom att skapa ett registreringspaket kan befintliga program registreras. Se "Registrera befintliga program eller program som inte är skrivbordsanpassade" på sidan 45.

Med registreringspaket går det lättare att administrera program på skrivbordet. Registreringspaket skapas inte på samma plats i filsystemet som används för skrivbordets konfigurationsfiler.

Lägga till program utan registreringspaket

Det här är det bästa sättet att lägga till program när du vill att programhanteraren bara ska innehålla en ikon som startar programmet.

Program som har lagts till utan registreringspaket:

- Kan ha sin egen programgrupp, men ikonerna placeras normalt i en befintlig grupp.
- Har sina skrivbordskonfigurationsfiler placerade i skrivbordets sökväg.

Se "Lägga till en programikon i befintlig programgrupp" på sidan 45.

▼ Lägga till skrivbordsanpassade program i programhanteraren

Ett skrivbordsanpassat program är ett program som registreras automatiskt i programhanteraren när det installeras. Bland programmets filuppsättningar finns ett registreringspaket för skrivbordet.

1. **Installera programmet genom att följa instruktionerna till programmet.**
2. **När installationen är klar dubbelklickar du på Ladda om program i programgruppen Skrivbordsverktyg.**

3. **Kontrollera att installationen är klar genom att:**
 - a. **Öppna programhanteraren och kontrollera att den nya programgruppen finns där.**
 - b. **Öppna programmet genom att öppna programgruppen och dubbelklicka på dess programikon.**

▼ Registrera befintliga program eller program som inte är skrivbordsanpassade

Det här är det bästa sättet att fullständigt integrera program med skrivbordet.

På skrivbordet finns verktyget `dtappintegrate`, som skapar länkar mellan registreringspaketets filer och katalogerna i skrivbordets sökväg.

Mer information om skrivbordsregistrering finns i Kapitel 5.

▼ Lägga till en programikon i befintlig programgrupp

I den här proceduren beskrivs hur du lägger till en programikon i en befintlig programgrupp.

På skrivbordet finns bl a programgruppen Systemadministration som reserverats för olika program och skript som hjälper dig att administrera system. Om du har ett skript som användarna ofta använder kanske du vill att de ska kunna köra skriptet genom att dubbelklicka på en ikon i programgruppen Systemadministration.

1. **Använd Skapa åtgärd för att skapa en åtgärdsdefinition för programmet.**
Mer information om Skapa åtgärd finns i Kapitel 11.

2. **Skapa en körbar fil med samma namn som åtgärden i programgruppens katalog. Innehållet i filen spelar ingen roll.**

Om du t ex har skapat en åtgärd med namnet "Ordna" som startar ett verktyg för systemadministration, skapar du den körbara filen

```
/etc/dt/appconfig/appmanager/språk/System_Admin/Cleanup
```

Skapa och hantera allmänna programgrupper

En allmän programgrupp är en programgrupp (katalog) som inte hör till någon viss programprodukt. Den inbyggda programgruppen Skrivbordsverktyg är en allmän programgrupp som innehåller ikoner för ett stort antal likartade program som inte hör till någon enskild produkt.

Du kan skapa flera allmänna programgrupper. Du kan till exempel skapa en grupp som heter Spel som innehåller alla spel i systemet.

En allmän programgrupp kan vara systemtäckande eller egen.

▼ Skapa en systemtäckande allmän programgrupp

1. **Logga in som root.**
2. **Skapa en katalog i `/etc/dt/appconfig/appmanager/språk`.**
Programgruppen får samma namn som katalogen.
3. **Dubbelklicka på Ladda om program i programgruppen Skrivbordsverktyg.**

▼ Skapa en egen allmän programgrupp

1. **Skapa en katalog i `Hemkatalog/.dt/appmanager`.**
Programgruppen får samma namn som katalogen.
2. **Dubbelklicka på Ladda om program i programgruppen Skrivbordsverktyg.**

▼ Anpassa en inbyggd programgrupp

1. **Logga in som root.**
2. **Om programgruppen finns i `/usr/dt/appconfig/appmanager/språk`, kopierar du programgruppen till `/etc/dt/appconfig/appmanager/språk`.**
Följande kommando kopierar till exempel programgruppen Skrivbordsverktyg:

```
cp -r /usr/dt/appconfig/appmanager/C/Desktop_Tools \  
/etc/dt/appconfig/appmanager/C
```

Den nya programgruppen kommer att prioriteras framför den inbyggda.

3. Ändra kopian av programgruppen. Du kan t ex lägga till nya åtgärdsfiler (körbara filer med samma namn som åtgärden).
4. Kontrollera ändringarna genom att logga ut och sedan in igen.

Ändra sökvägen som används för att söka efter program

Det viktigaste skälet att ändra programsökvägen är för att lägga till en programserver. När du lägger till en programserver i sökvägen samlas serverns alla systemtäckande programgrupper in.

Mer information om programsökvägen finns i "Programsökväg" på sidan 132.

Standardsökvägen

Standardsökvägen innehåller katalogerna i Tabell 4-3.

TABELL 4-3 Kataloger i standardsökvägen

Definitionsområde	Katalog i sökvägen
Egen	<i>Hemkatalog</i> /.dt/appmanager
Systemtäckande	/etc/dt/appconfig/appmanager/ <i>språk</i>
Inbyggd	/usr/dt/appconfig/appmanager/ <i>språk</i>

Lägga till en programserver i programsökvägen

Det kan hända att du, utöver ändringen av programsökvägen, också måste utföra andra konfigurationer för att kontakten med programservern ska fungera. Se "Administrera programtjänster" på sidan 114.

Ange en systemtäckande programsökväg

1. Logga in som root.

2. Om inte filen `/etc/dt/config/Xsession.d/0010.dtpaths` redan finns skapar du den genom att kopiera

```
/usr/dt/config/Xsession.d/0010.dtpaths.
```

3. Öppna `/etc/dt/Xsession.d/0010.paths` för redigering. Lägg till eller redigera en rad som anger och exporterar variabeln `DTSPSYSAPPHOSTS`:

```
export DTSPSYSAPPHOSTS=värdsnamm:[,hostname]
```

I följande rad läggs systemet `ApServA` till i programsökvägen:

```
export DTSPSYSAPPHOSTS=ApServA:
```

4. Informera alla användare i systemet om att de måste logga ut och in igen för att ändringen ska träda i kraft.

Ange en egen programsökväg

1. Öppna *Hemkatalog*/`.dtprofile` för redigering.

2. Lägg till eller redigera en rad som anger och exporterar variabeln

```
DTSPUSERAPPHOSTS:
```

```
export DTSPUSERAPPHOSTS=värdsnamm:[,hostname]
```

I följande rad läggs systemen `ApServB` och `ApServC` till i programsökvägen:

```
export DTSPUSERAPPHOSTS=ApServB:,ApServC:
```

3. Logga ut och sedan in igen.

Allmän administration av programhanteraren

Allmän administration av programhanteraren omfattar:

- Ta bort program
- Ladda om programdatabasen under en session

▼ Ta bort ett program

Om ett program har registrerats med verktyget `dtappintegrate`, kan du också använda `dtappintegrate` för att göra det motsatta. När ett program avregistreras tas dess programgrupp bort från programhanteraren, och dess åtgärder, datatyper och ikoner är inte längre tillgängliga.

1. **Logga in som root.**
2. **Kör kommandot:**

```
dtappintegrate -s prgm_rot -u
```

▼ Uppdatera programhanteraren under en session

Du måste återskapa programhanteraren om du lägger till program och vill att dessa ändringar ska träda i kraft omedelbart.

- ◆ **Öppna programgruppen Skrivbordsverktyg och dubbelklicka på Ladda om åtgärder.**

Ladda om program är användbart för att uppdatera programhanteraren när program har lagts till på en programserver. Ladda om program upptäcker dock inte program som har tagits bort från en programserver eller program som har flyttats från en plats till en annan. Dessa ändringar träder i kraft när användaren loggar ut och sedan in igen.

Ändra textredigeraren och terminalemulatorn

Både textredigeraren och terminalemulatorn kan startas med en kontroll i huvudgruppen eller genom att du dubbelklickar på en ikon i programhanteraren.

Dessa program startas också av andra skrivbordsaktiviteter.

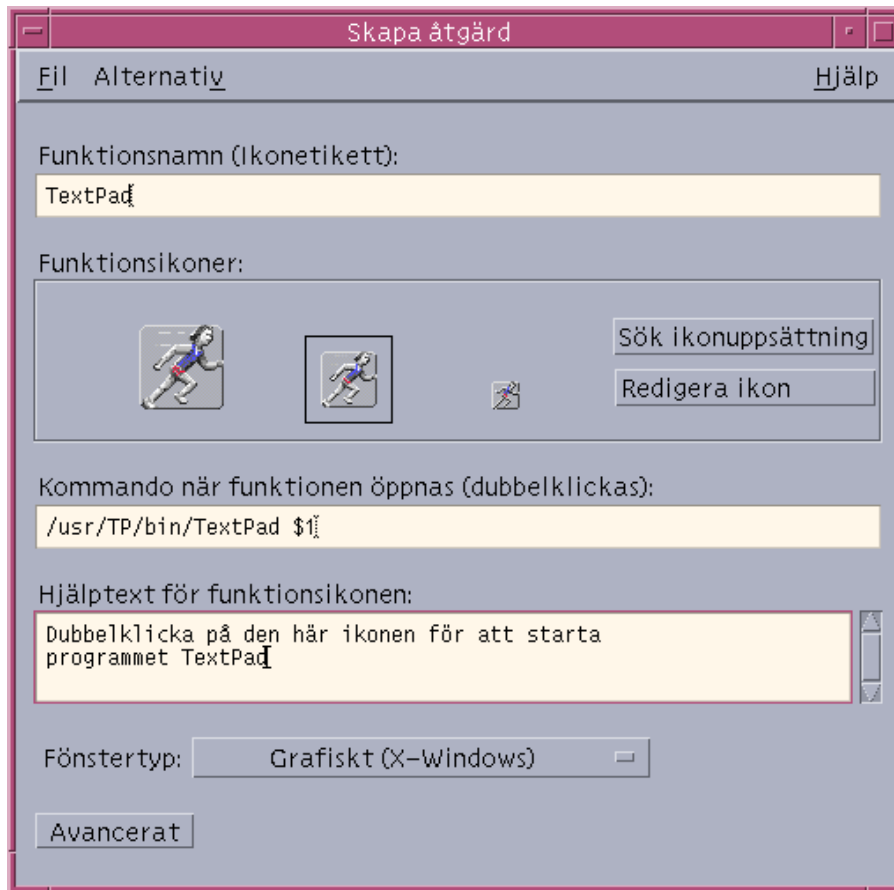
- Textredigeraren öppnas när användaren markerar textfilen i filhanteraren och väljer Öppna på menyn Markera. Standardredigeraren är `dtpad`.
- En terminalemulator körs när användaren väljer Öppna terminal på filhanterarens Fil-menü eller när en åtgärd öppnar terminalemulatorfönstret. Standardterminalen är `dtterm`.

Du kan konfigurera skrivbordet så att andra textredigerare eller terminalemulatorer används i dessa situationer.

▼ Ändra standardtextredigerare och standardterminalemulator

1. Logga in som root om ändringen ska vara systemtäckande.
2. Skapa en åtgärd för den nya textredigeraren eller terminalemulatorn.

- Du kan använda programmet Skapa åtgärd. I Figur 4-3 visas ett fönster i Skapa åtgärd som fyllts i för programmet TextPad. Mer information om Skapa åtgärd finns i Kapitel 11.



Figur 4-3 Fönstret Skapa åtgärd

- Du kan *också* skapa åtgärden manuellt:

```

ACTION TextPad
{
    LABEL      TextPad
    TYPE       COMMAND
    WINDOW_TYPE NO_STDIO
    EXEC_STRING /usr/TP/bin/TextPad %(File)Arg_1%
    DESCRIPTION Dubbelklicka på den här ikonen för att starta \
                programmet TextPad.
}

```

Mer information om hur du skapar åtgärdsdefinitioner manuellt finns i Kapitel 12.

3. Placera konfigurationsfilen med den nya åtgärden i rätt katalog:

- Systemtäckande: `/etc/dt/appconfig/types/språk`
- Egen: *Hemkatalog*/`.dt/types`

4. Om den inte redan finns, skapar du en lämplig `user-prefs.dt`-fil genom att kopiera `/usr/dt/appconfig/types/språk/user-prefs.dt` till:

- Systemtäckande: katalogen `/etc/dt/appconfig/types/språk`
- Egen: katalogen *Hemkatalog*/`.dt/types`

5. Redigera åtgärden `TextEditor` eller `Terminal` i den systemtäckande eller egna filen `user-prefs.dt`. Ändra raden `MAP_ACTION` så att den tilldelar åtgärden till den nya åtgärden.

Ändra t ex raden:

```
MAP_ACTION Dtpad
```

till:

```
MAP_ACTION TxtPd
```

6. Spara filen `user-prefs.dt`.

7. Uppdatera åtgärdsdatabasen genom att dubbelklicka på Ladda om åtgärder i programgruppen `Skrivbordsverktyg`.

Registrera ett program

Det här kapitlet beskriver hur du skapar ett registreringspaket för ett program och hur du registrerar programmet.

- "Programregistrering – översikt" på sidan 54
- "Generella steg för registrering av program" på sidan 57
- "Steg 1: Ändra teckensnitts- och färgresurser" på sidan 58
- "Steg 2: Skapa skrivbordsprogrammets rot" på sidan 59
- "Steg 3: Skapa registreringspaketets kataloger" på sidan 59
- "Steg 4: Skapa åtgärderna och datatyperna för programmet" på sidan 62
- "Steg 5: Placera hjälpfilerna i registreringspaketet" på sidan 65
- "Steg 6: Skapa ikoner för programmet" på sidan 65
- "Steg 7: Skapa programgruppen" på sidan 67
- "Steg 8: Registrera programmet med hjälp av dtappintegrate" på sidan 73
- "Exempel på hur du kan skapa ett registreringspaket" på sidan 76

När ett program är helt registrerat på skrivbordet har det:

- En egen programgrupp på översta nivån i programhanteraren
- En åtgärd som startar programmet. Åtgärden representeras av en ikon i programgruppen
- Datatyper för sina datafiler (valfritt)

Programregistrering är en icke inträngande åtgärd i programmet:

- Den ändrar inte programmets körbara filer. Du kan därför registrera befintliga program i ett system.
- Ingen av programmets inbyggda filer (t ex körbara filer och `app-defaults`) behöver flyttas.

- Registrering är enkel att ångra. Verktöget `dtappintegrate`, som används för att registrera program, innehåller ett kommandoradsalternativ som upphäver processen.

Du kan behöva skapa ett registreringspaket om du är:

- En systemadministratör som vill registrera ett befintligt program på skrivbordet
- En programmerare som vill skapa ett installationspaket till ett skrivbordsprogram

Programregistrering – översikt

I det här avsnittet förklaras:

- Syftet med programregistrering
- Funktioner som tillförs programmet genom programregistrering

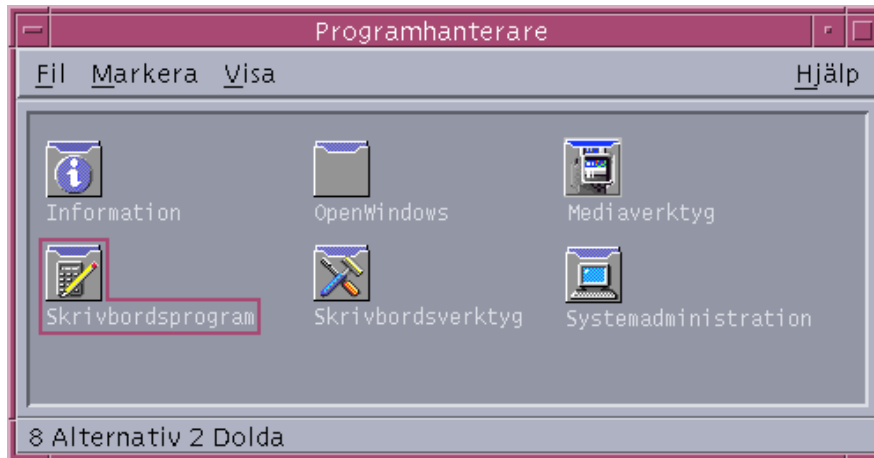
Obs! Ett utförligt exempel som visar hur du registrerar ett befintligt program finns i "Exempel på hur du kan skapa ett registreringspaket" på sidan 76.

Funktioner som medföljer programregistrering

Programregistrering erbjuder ett grafiskt sätt för användare att:

- Leta rätt på programmet.

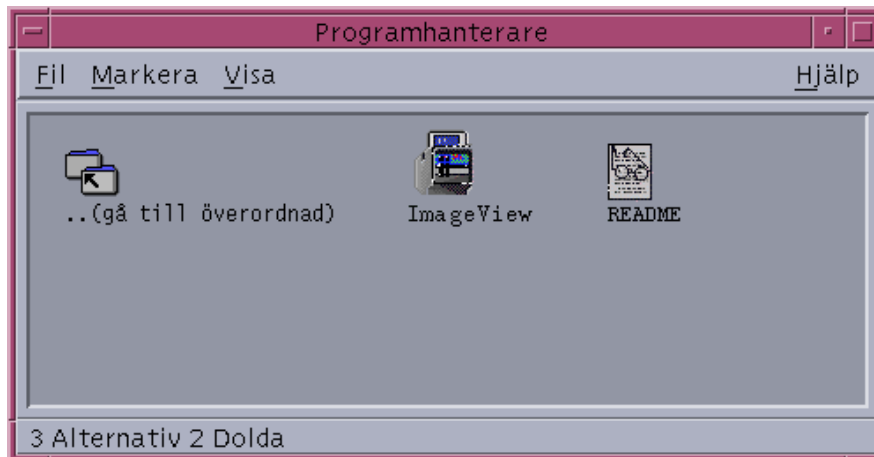
Vid installationen "registreras" programmet i programhanteraren och får en egen programgrupp.



Figur 5-1 Programgrupper på översta nivån i programhanteraren

- Starta programmet.

Programgruppen med ditt program innehåller en ikon som användaren kan dubbelklicka på för att starta programmet.

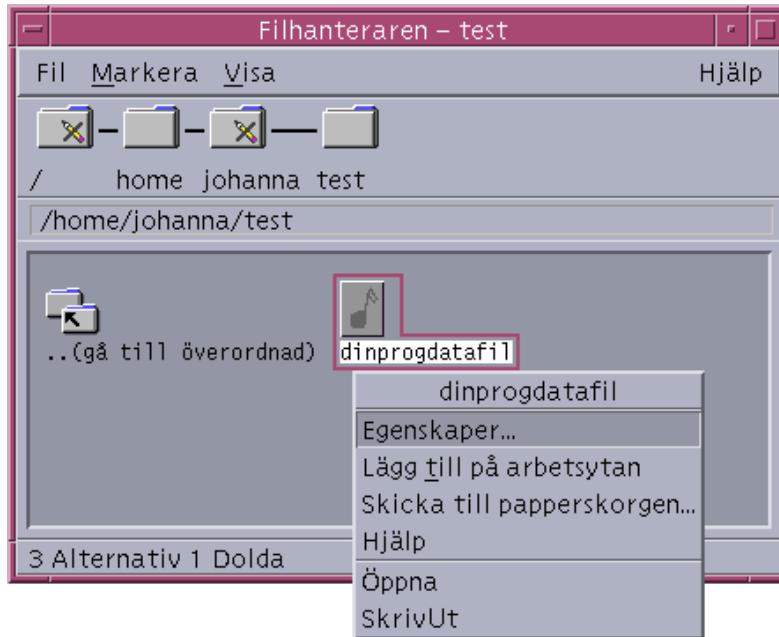


Figur 5-2 En programgrupp som innehåller en ikon som startar programmet

- Identifiera och ändra datafiler. Programmets datafiler kommer att ha en unik ikon i filhanteraren.

Användaren kan använda datafilikoner för att:

- Starta (Öppna) programmet
- Skriva ut datafiler



Figur 5-3 En datafiles snabbmeny med "Öppna" och "Skriv ut"

- Utföra andra åtgärder, t ex posthantering, komprimering, visning eller uppspelning av data (ljud).

Syftet med programregistrering

Ett registrerat skrivbordsprogram innehåller vissa konfigurationsfiler som används av skrivbordet för att visa programmets användargränssnitt:

- Filer med åtgärds- och datatypsdefinitioner
- Ikonbildfiler (bildpunktsbeskrivning eller bitmapp)
- En katalog och filer som skapar programgruppen
- Skrivbordets hjälpfiler och huvudgruppens definitionsfiler (valfritt)

För att dessa filer ska kännas igen och användas av skrivbordet måste de finnas i vissa kataloger som anges i skrivbordets sökvägar.

Det kan vara svårt att administrera ett program om dess konfigurationsfiler är utspridda i flera kataloger. Därför kan ett program lagra alla skrivbordskonfigurationsfiler i en och samma katalog. Denna filgruppering kallas för *registreringspaket*.

Om programmet är skrivbordsanpassat innehåller det ett registreringspaket som en del av installationspaketet. Om du är systemadministratör och skapar konfigurationsfilerna själv kan du skapa registreringspaketet själv.

Konfigurationsfilerna i registreringspaketet är inte tillgängliga på skrivbordet eftersom de inte finns i lämpliga sökvägs kataloger. Processen att placera de här filerna på lämpliga platser kallas *registrering* eller *integrering* av programmet.

I skrivbordssystemet finns ett verktyg, `dtappintegrate`, som utför registreringen genom att skapa symboliskt länkade representationer av filerna i de korrekta sökvägs katalogerna.

Många skrivbordsanpassade program kör `dtappintegrate` automatiskt under installationen. Om du är systemadministratör och integrerar ett befintligt program kan du köra det själv när du har skapat registreringspaketet.

När ett program är registrerat på ett systems skrivbord är programmet tillgängligt för alla användare i systemet. Om systemet är konfigurerat som en skrivbordsprogramserver blir programmet också tillgängligt i andra system i nätverket.

Verktyget `dtappintegrate` har ett kommandoradsalternativ som upphäver processen genom att bryta länkarna. Detta gör det enkelt att ta bort programmet från programhanteraren så att det kan flyttas till en annan programserver eller uppdateras.

Generella steg för registrering av program

Obs! Ett utförligt exempel på hur du kan använda de här stegen för att skapa ett programpaket finns i "Exempel på hur du kan skapa ett registreringspaket" på sidan 76.

- 1. Ändra alla programresurser som anger teckensnitt och färger. Annars kommer inte skrivbordets dynamiska teckensnitt och färger att fungera korrekt.**
Se "Steg 1: Ändra teckensnitts- och färgresurser" på sidan 58.
- 2. Skapa en rotadress för programmet.**
Se "Steg 2: Skapa skrivbordsprogrammets rot" på sidan 59.
- 3. Skapa katalogstrukturen under programroten.**
Se "Steg 3: Skapa registreringspaketets kataloger" på sidan 59
- 4. Skapa åtgärderna och datatyperna för programmet.**
Se "Steg 4: Skapa åtgärderna och datatyperna för programmet" på sidan 62.
- 5. Placera hjälpfilerna i lämplig katalog.**

Se "Steg 5: Placera hjälpfilerna i registreringspaketet" på sidan 65.

6. Skapa ikonerna för programmet

Se "Steg 6: Skapa ikoner för programmet" på sidan 65.

7. Skapa programgruppen för programmen.

Se "Steg 7: Skapa programgruppen" på sidan 67.

8. Registrera programmet med hjälp av dtappintegrate.

Se "Steg 8: Registrera programmet med hjälp av dtappintegrate" på sidan 73.

Steg 1: Ändra teckensnitts- och färgresurser

Obs! Ett exempel på hur du kan ändra resurser för ett program finns i Steg 1 i "Exempel på hur du kan skapa ett registreringspaket" på sidan 76.

Skrivbordet tillhandahåller mekanismer för att ange och ändra gränssnittsteckensnitt och fönsterfärger. För att ett program ska kunna använda de här mekanismerna korrekt kan du behöva ändra programmets `app-defaults`-fil.

Ändra teckensnittsresurser

Obs! Det här avsnittet gäller för program som är skapade med OSF/Motif 1.2TM (eller senare versioner). Inställningar kan inte ange gränssnittsteckensnitt för program som är skrivna med tidigare versioner av OSF/Motif.

Inställningar på skrivbordet anger gränssnittsteckensnitt för program som är skrivna med OSF/Motif 1.2 (eller senare versioner) om programmet inte anger programspecifika gränssnittsteckensnitt.

Inställningar tillhandahåller två teckensnitt:

systemteckensnitt – används i systemområden som etiketter, menyer och knappar

användarteckensnitt – används i redigerbara områden, t ex textfält

Varje teckensnitt finns i sju storlekar, med etiketten 1 t o m 7 i dialogrutan Teckensnitt. Teckensnitten i Inställningar är kopplade till faktiska teckensnitt i systemet via resurser för Inställningar som anges i `/usr/dt/app-defaults/språk/Dtstyle`.

Om du vill att programmet ska använda teckensnitten i Inställningar, bör du ta bort alla programresurser som anger teckensnitt för gränssnittet. Skrivbordet anger automatiskt korrekta programresurser:

`FontList` – ange som systemteckensnitt

XmText*FontList - ange som användarteckensnitt

XmTextField*FontList - ange som användarteckensnitt

Ändra färgresurser

Inställningar erbjuder möjlighet att ändra programfärger dynamiskt. Programmet måste vara en OSF/Motif 1.1- eller OSF/Motif 1.2-klient. Klienter som skrivs med andra verktygslådor kan inte ändra färg dynamiskt. Färgändringarna träder i kraft när klienten startas om.

Det enklaste sättet att använda de dynamiska färger som tillhandahålls av skrivbordet är att ta bort alla programfärgresurser för bakgrunds- och förgrundsfärg.

Steg 2: Skapa skrivbordsprogrammets rot

Obs! Ett exempel på hur du kan skapa en rotkatalog för ett skrivbordsprogram finns i Steg 2 i "Exempel på hur du kan skapa ett registreringspaket" på sidan 76.

Registreringspaketets filer för programmet är grupperade under en katalog som kallas programrot eller *prgm_rot*. Katalogen *prgm_rot* som används för skrivbordskonfigurationsfilerna kan vara samma katalog som programmets installationskatalog (*prgm_rot*) eller någon annan katalog.

Anta exempelvis att ett program är installerat under katalogen `/usr/BTE`. Samma katalog kan användas som *prgm_rot* för skrivbordskonfigurationsfilerna. Om du däremot integrerar ett befintligt program som inte är skrivbordsanpassat, skapar du en annan *prgm_rot*-katalog för skrivbordssystemet. Det här förhindrar att konfigurationsfilerna som du skapar blir överskrivna när du uppdaterar programmet.

En systemadministratör kan t ex vilja skapa en katalog som heter `/etc/desktop_approots/BTE` som skrivbordssystemets *prgm_rot*-katalog.

Steg 3: Skapa registreringspaketets kataloger

Obs! Ett exempel på hur du kan skapa registreringspaketkataloger för ett program finns i Steg 3 i "Exempel på hur du kan skapa ett registreringspaket" på sidan 76.

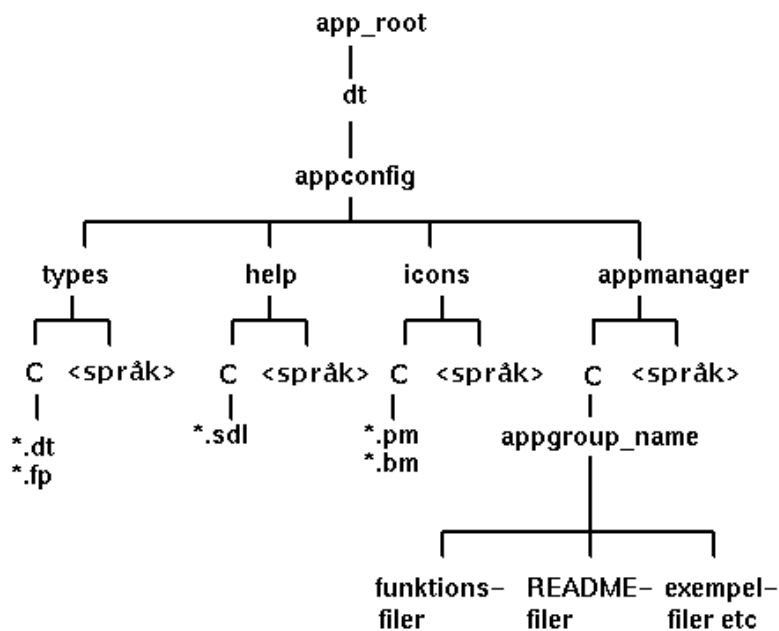
Registreringspaketet är den grupp med skrivbordskonfigurationsfiler som används av skrivbordet för att tillhandahålla ett grafiskt gränssnitt för programmet.

Registreringspaketet – innehåll

Skrivbordskonfigurationsfilerna innehåller:

- Filer med åtgärds- och datatypsdefinitioner
- Filer med ikonbilder
- En programgruppskatalog och dess innehåll
- Hjälpdatafiler och en konfigurationsfil för huvudgruppen (valfritt)

Registreringspaketet samlas under en katalog på högsta nivån som kallas programroten eller *prgm_rot*.



Figur 5-4 Ett registreringspaket under en programrotskatalog

Huvudkategorierna av konfigurationsfält under katalogen *prgm_rot/dt/appconfig* visas i Tabell 5-1.

TABELL 5-1 Konfigurationsfält – huvudkategorier

Underkatalog	Innehåll
types	Filer med åtgärds- och datatypsdefinitioner
help	Skrivbordshjälpfiler
icons	Bitmapp- och bildpunktsbeskrivningsfiler som används av programmets åtgärder och datatyper
appmanager	Katalogen och innehållet som skapar programgruppen

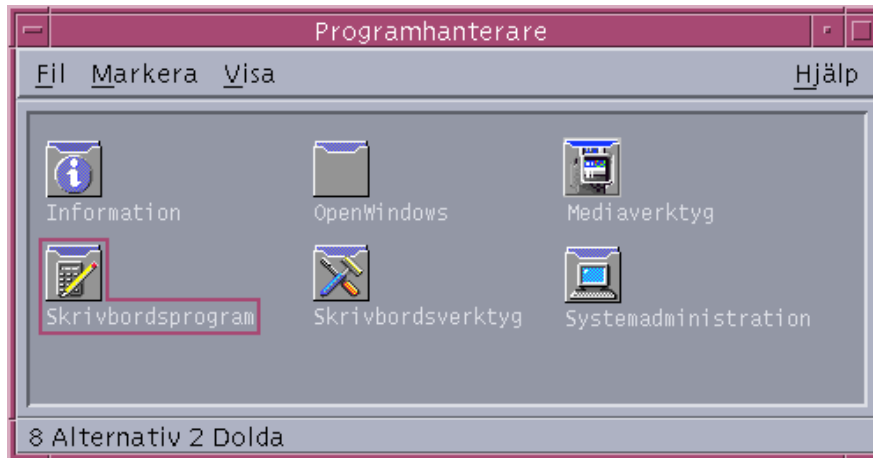
Var och en av huvudkategorierna har underkataloger för språkberoende filer. Standardspraksfiler placeras i katalogen C.

Skapa registreringspaketet

- ◆ **Skapa de här katalogerna. Om du tillhandahåller språkberoende konfigurationsfiler skapar du en separat katalog för varje språk. Om du bara använder ett språk placerar du filerna i katalogen C.**

- *prgm_rot/dt/appconfig/types/språk*
- *prgm_rot/dt/appconfig/help/språk*
- *prgm_rot/dt/appconfig/icons/språk*
- *prgm_rot/dt/appconfig/appmanager/språk/programgruppsnamn*, där *programgruppsnamn* är namnet på programgruppen.

I Figur 5-5 visas programhanteraren med en grupp vars *programgruppsnamn* är "Mediaverktyg".



Figur 5-5 Programgrupp på högsta nivån i programhanteraren

Verktyget dtappintegrate fungerar bara för skrivbordskonfigurationsfilerna i katalogerna types, help, icons och appmanager. Programmets binära körbara app-defaults och meddelandekatalogfiler administreras separat.

Steg 4: Skapa åtgärderna och datatyperna för programmet

Obs! Ett exempel på hur du kan skapa åtgärderna och datatyperna för ett program finns i Steg 4 i "Exempel på hur du kan skapa ett registreringspaket" på sidan 76.

Åtgärder och datatyper tillhandahåller ett användargränssnitt för programmet.

- Åtgärder tillhandahåller ett användargränssnitt där kommandot kan starta programmet.
- Datatyper tillhandahåller anpassade utseenden och beteenden för programmets datafiler.

Nödvändiga åtgärder och datatyper för ett program

Ett typiskt program kan kräva följande åtgärds- och datatypsdefinitioner:

- En åtgärd som öppnar programmet.
- En datatyp för datafilerna i programmet. Om du skapar en datatyp behöver du också skapa:
 - En Öppna-åtgärd för datafilerna i programmet
 - En SkrivUt-åtgärd för datafilerna i programmet

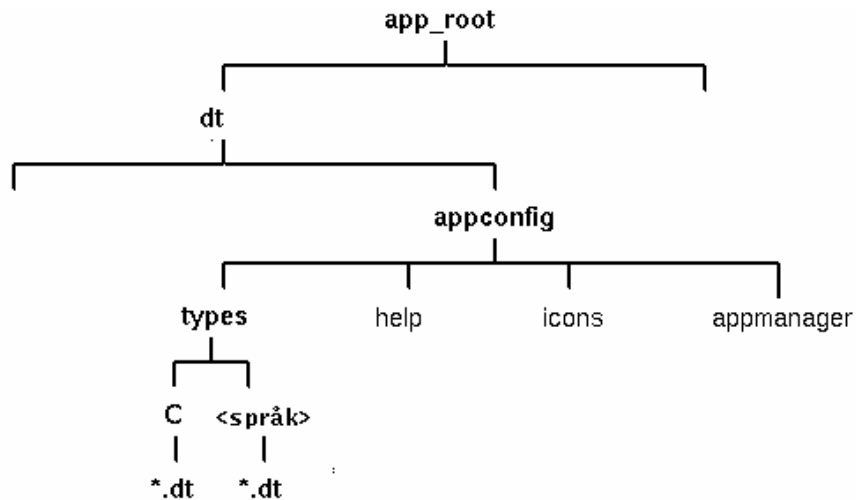
- En datatyp för programgruppen (se "Konfigurera programgruppen så att den använder en unik ikon" på sidan 69).

En beskrivning av hur åtgärder och datatyper används på skrivbordet finns i Kapitel 10.

Adress för konfigurationsfiler för åtgärds- och datatypsdefinitioner

Åtgärder och datatyper definieras i konfigurationsfiler. Det enda namnkravet för filer som innehåller åtgärds- och datatypsdefinitioner är att de måste ha filnamnstillägget `.dt`. Du kanske vill att filen ska heta *åtgärdsnamn*.dt eller *programnamn*.dt.

Placera filer som innehåller åtgärder och datatyper under programroten i katalogen `prgm_rot/dt/appconfig/types/språk`. Standardspråk är C.



Figur 5-6 Filer med åtgärds- och datatypsdefinitioner

Olika sätt att skapa åtgärder och datatyper

Du kan skapa åtgärds- och datatypsdefinitioner för ett program på två sätt:

- Använd verktyget Skapa åtgärd.
 - Skapa åtgärd innehåller ett lättanvänt gränssnitt med textfält som du fyller i. Verktyget har emellertid vissa begränsningar.
- Skapa definitionerna manuellt.
 - Det här kräver att du lär dig syntax för att skapa definitionerna, men ger dig samtidigt tillgång till hela uppsättningen funktioner.

Skapa åtgärder och datatyper med hjälp av Skapa åtgärd

Den här proceduren använder hjälpmedlet Skapa åtgärd för att skapa åtgärder och datatyper för programmet.

Mer information om Skapa åtgärd finns i direkthjälpen och i Kapitel 11.

1. **Öppna programgruppen `Skrivbords_prog` och dubbelklicka på Skapa åtgärd.**
2. **Använd Skapa åtgärd för att skapa åtgärds- och datatypsdefinitioner för programmet och dess datatyp.**
Konfigurationsfilen som skapas av Skapa åtgärd sparas i *Hemkatalog/.dt/type/åtgärdsnamn.dt*. Åtgärdsfilen (den körbara filen med samma namn som åtgärden) placeras i din hemkatalog.
3. **Testa åtgärden med hjälp av åtgärdsfilen som har skapats i din hemkatalog.**
4. **Kopiera åtgärdsdefinitionsfilen *Hemkatalog/.dt/type/åtgärdsnamn.dt* till katalogen *prgm_rot/dt/appconfig/types/språk*.**
5. **När programgruppskatalogen har skapats (se "Steg 7: Skapa programgruppen" på sidan 67), kopierar du åtgärdsfilen *Hemkatalog/åtgärdsnamn* till katalogen *prgm_rot/dt/appconfig/appmanager/språk/programgruppsnamn*.**

Skapa åtgärder och datatyper manuellt

- ♦ **Skapa en konfigurationsfil som innehåller åtgärds- och datatypsdefinitioner för programmet.**

Åtgärds- och datatypsdefinitionsfilerna måste namnges enligt mönstret *namn.dt*.

Du kan placera alla åtgärds- och datatypsdefinitioner i en fil eller distribuera dem bland flera filer. För varje fil använder du ett filnamn som systemadministratörer enkelt kopplar till ditt program.

Åtgärds- och datatypsnamn måste bestå av ett ord (inga inbäddade blanksteg). Du kan använda ett understrykningstecken. Enligt konventionen är den första bokstaven i åtgärds- eller datatypsnamnet versal. Använd inte ett befintligt åtgärdsnamn eller filnamn. Använd ett namn som avancerade användare och systemadministratörer enkelt kopplar till ditt program.

Om du vill att programmets ikon ska ha ett annat namn än åtgärdsnamnet tar du med fältet `LABEL` i åtgärdsdefinitionen.

Mer information om hur du skapar åtgärder och datatyper finns i:

- Kapitel 10
- Kapitel 11

- Kapitel 12
- Kapitel 13

Steg 5: Placera hjälpfilerna i registreringspaketet

Obs! Ett exempel på hur du kan lägga till hjälpfiler i ett registreringspaket finns i Steg 5 i "Exempel på hur du kan skapa ett registreringspaket" på sidan 76.

Om programmet innehåller en skrivbordshjälpvolym (en hjälpvolym som är skapad med skrivbordssystemets hjälputvecklingsystem), bör hjälpvolymens huvudfil (* .sdl) placeras i katalogen *prgm_rot/appconfig/help/språk*.

Grafik som används av hjälpfilerna placeras vanligen i underkatalogen *graphics*. Grafiken måste finnas i samma katalog i förhållande till huvudhjälpvolymfilen (* .sdl) som när hjälpvolymen skapades.

Om programmet inte innehåller en hjälpvolym kan du skapa en sådan, om du har tillgång till hjälputvecklingsystemet.

Det finns två integrationsnivåer för en hjälpvolym:

- Full integration.

När skrivbordshjälpen är fullständigt integrerad kan hjälpvolymen öppnas från programmet, t ex via objekthjälp och Hjälp-menyn. Full integrering innebär att programmets körbara filer ändras.

- Delvis integration.

När skrivbordshjälpen är delvis integrerad är den tillgänglig från högsta nivån i hjälphanteraren. Du kan emellertid inte komma åt hjälpvolymen från programmets fönster. Du kan också låta en åtgärd komma åt hjälpen från programgruppen. I följande åtgärdsexempel visas hjälpvolymen som finns i hjälpens huvudfil *MittPrgm.sdl*:

```
ACTION VisaMittPrgmHjälp
{
  LABEL           MittPrgmHjälp
  ARG_COUNT       0
  TYPE            COMMAND
  WINDOW_TYPE     NO_STDIO
  EXEC_STRING     /usr/dt/bin/dthelpview -helpVolume MittPrgm
  DESCRIPTION     Visar hjälp för programmet MittPrgm.
}
```

Steg 6: Skapa ikoner för programmet

Obs! Ett exempel på hur du kan skapa ikonfiler för ett program finns i Steg 6 i "Exempel på hur du kan skapa ett registreringspaket" på sidan 76.

Skrivbordet tillhandahåller standardikoner för åtgärder, datatyper och programgrupper. Troligen vill du emellertid skapa unika ikoner för programmet.

Ikonerna placeras i katalogen `prgm_rot/dt/appconfig/icons/språk`.

Nödvändiga ikoner för skrivbordet

Programmet använder de här ikonbilderna på skrivbordet:

- **Åtgärdsikon.** Det här är den ikon som användaren dubbelklickar på för att starta programmet (åtgärderna). `ICON`-fältet för den åtgärd som startar programmet innehåller en referens till den här ikonen.

Den finns i tre storlekar: mycket liten, medelstor och stor.

- **Datatypsikon.** Den här ikonen används för att representera programmets datafiler i filhanteraren. Datatypsdefinitionens `ICON`-fält innehåller en referens till den här ikonen.

Om programmet stöder flera datatyper bör du ha en särskild ikon för varje datatyp.

Den finns i två storlekar: mycket liten och medelstor.

- **Programgruppsikon.** Det här är den ikon som representerar katalogen på högsta nivån i programhanteraren. `ICON`-fältet för programgruppens datatypsdefinition innehåller en referens till den här ikonen. (Se "Steg 7: Skapa programgruppen" på sidan 67.)

Den finns i två storlekar: mycket liten och medelstor.

Du kan behöva stödja både bildpunktsbeskrivnings- och bitmappsversioner av varje ikon för att stödja både färgskärmar (åtta bitar och större) och monokroma skärmar (färre än åtta bitar).

TABELL 5-2 Namnkonventioner för ikonfiler

Storlek, Sidformat, Grad	Bildpunktsstorlek	Bitmappsnamn	Bildpunktsmönster
mycket liten	16 gånger 16	<i>grundnamn.t.bitmap</i>	<i>grundnamn.t.pbm</i>
medelstor	32 gånger 32	<i>grundnamn.m.bitmap</i>	<i>grundnamn.m.pbm</i>
stor	48 gånger 48	<i>grundnamn.l.bitmap</i>	<i>grundnamn.l.pbm</i>

Om du inte tillhandahåller bitmappsfiler anger skrivbordet färgspecifikationerna för bildpunktsbeskrivningsfilerna till svartvitt. Det är dock möjligt att den här tilldelningen inte ger det utseende du vill ha.

Mer information om ikoner finns i "Ikonbildsfiler" på sidan 213.

Steg 7: Skapa programgruppen

Obs! Ett exempel på hur du skapar en programgrupp finns i Steg 7 i "Exempel på hur du kan skapa ett registreringspaket" på sidan 76.

När du har skapat åtgärds- och datatypsdefinitionerna för programmet måste du skapa konfigurationsfilerna som ska skapa det som användaren ser på skärmen: programgruppen och dess innehåll.

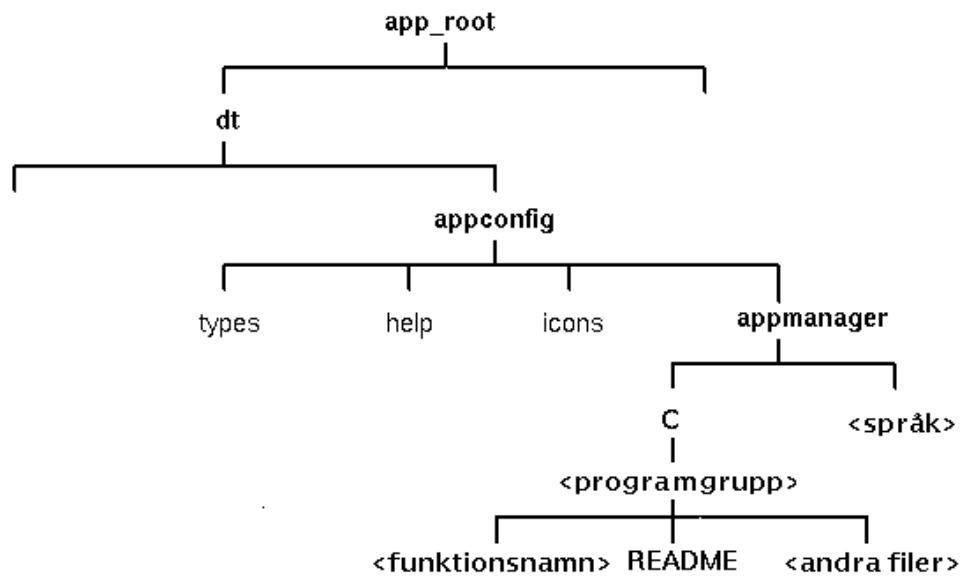
Programgruppen är en katalog på högsta nivån i programhanteraren (se Figur 5-1).

Du skapar programgruppen i tre steg:

- Skapa programgruppskatalogen i registreringspaketet.
- Valfritt: konfigurera programgruppen så att den använder en unik ikon. Det innebär att du skapar datatypsdefinitionen för programgruppskatalogen.
- Skapa innehållet i programgruppen.

Skapa programgruppskatalogen

Om du vill skapa en programgrupp skapar du katalogerna i registreringspaketet under `appmanager`, enligt anvisningarna i Figur 5-7.



Figur 5-7 Katalogen appmanager

Programgruppnamn

<Programgruppsnamn> i Figur 5-7 är namnet på programgruppen.



Figur 5-8 Programgruppsnamnet (<programgruppsnamn>)

Namnet kan vara ett valfritt tillåtet fil- eller katalognamn. Använd ett namn som beskriver programmet.

Konfigurera programgruppen så att den använder en unik ikon

Skrivbordet tillhandahåller en standardikon för programgrupper. Troligen vill du emellertid använda en anpassad ikon.

Om du vill använda en unik ikon för programgruppen måste du skapa:

- En datatyp för katalogen som visas på översta nivån i programhanteraren.
- Öppna- och SkrivUt-åtgärder för datatypen.

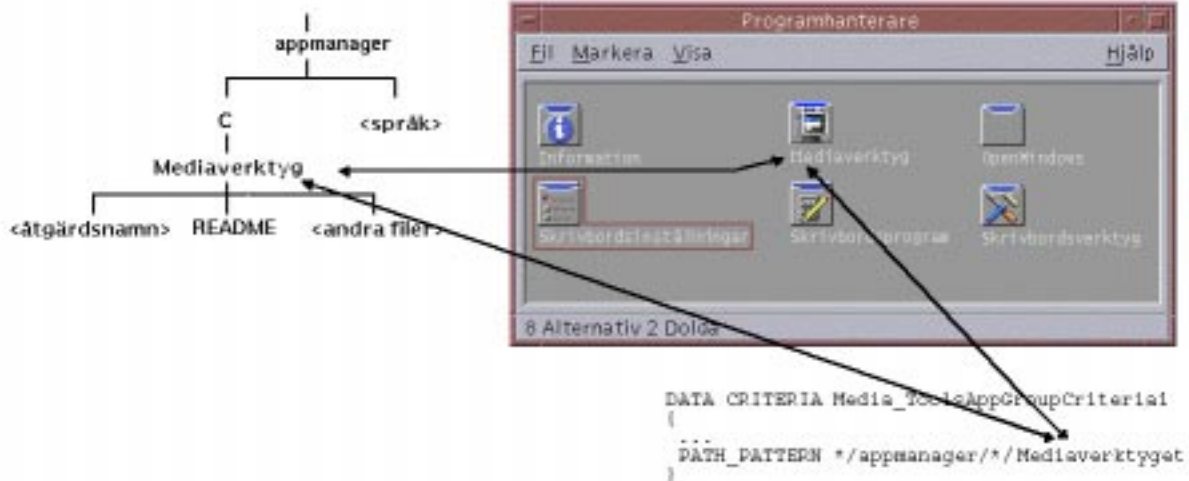
Anta exempelvis att du vill skapa en programgrupp som heter Mediaverktyg. Följande datatypsdefinition, som finns i filen *prgm_rot/dt/appconfig/types/språk/namn.dt*, anger en unik ikon som programgruppsikon.

```
DATA_ATTRIBUTES Media_ToolsAppgroup
{
    ACTIONS          OpenInPlace,OpenNewView
    ICON             MediaTools
    DESCRIPTION      Dubbelklicka för att öppna Mediaverktygets \
                    programgrupp
}

DATA_CRITERIA Media_ToolsAppgroupCriterial
{
    DATA_ATTRIBUTES_NAME Media_ToolsAppgroup
    MODE                 d
    PATH_PATTERN         */appmanager/*/Mediaverktyg
}
```

Attributavsnittet i definitionen anger vilken ikon som ska användas. Villkorsavsnittet i definitionen anger att datatypen ska definieras till en katalog som heter *Media_Tools*, som är en underkatalog till en katalog som heter *appmanager*.

Figur 5–9 visar förhållandet mellan programgruppsnamnet och datatypsdefinitionen. Fältet *PATH_PATTERN* i datatypsdefinitionen tilldelar en unik ikon till programgruppen.



Figur 5-9 Hur en programgrupp får en unik ikon

Du bör också skapa en Öppna- och en SkrivUt-åtgärd för programgruppens datatyp:

```
ACTION Open
{
  ARG_TYPE  Media_ToolsAppGroup
  TYPE      MAP
  MAP_ACTION OpenAppGroup
}
```

```
ACTION Print
{
  ARG_TYPE  Media_ToolsAppGroup
  TYPE      MAP
  MAP_ACTION PrintAppGroup
}
```

Åtgärderna `OpenAppGroup` och `PrintAppGroup` är inbyggda åtgärder som är definierade i `/usr/dt/appconfig/types/språk/dtappman.dt`.

Skapa innehållet i programgruppen

Det viktigaste elementet i programgruppen är den ikon som startar programmet (en åtgärdsikon). Om programgruppen innehåller en uppsättning program finns det vanligen en ikon för varje program.

Förutom en eller flera åtgärdsikoner kan programgruppen innehålla:

- En eller flera README-filer
- Ett eller flera exempel på datafiler
- Mallar

- En ikon som användaren kan dubbelklicka på för att visa hjälpinformation
- En direkthjälp
- En särskild huvudgruppskontroll

Programgruppen kan innehålla underkataloger.

Skapa åtgärdsfilen (programikon)

Programgruppen bör innehålla en ikon som startar programmet. Om gruppen tillhandahåller en uppsättning program bör det finnas en ikon för varje program. Ikonerna kallas *programikoner* eller *åtgärdsikoner*, eftersom de representerar en underliggande åtgärd.

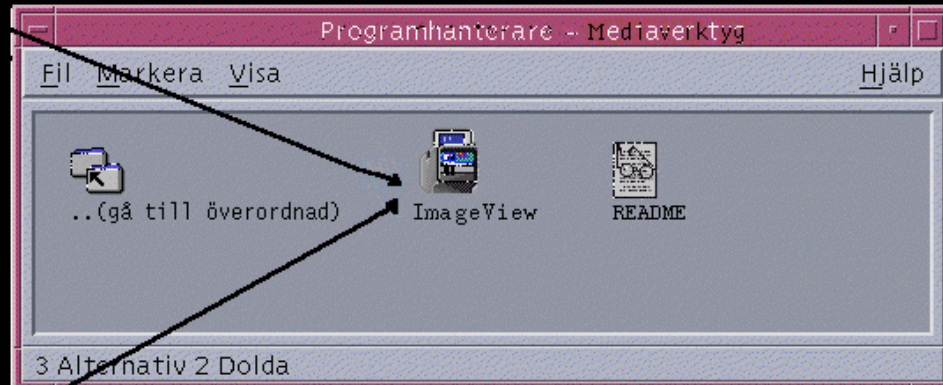
Du skapar en åtgärdsikon genom att skapa en körbar fil med samma namn som åtgärden den kommer att köra:

```
prgm_rot/dt/appconfig/appmanager/programgruppsnamn/åtgärdsnamn
```

Filen kallas en *åtgärdsfil*, eftersom den används för att skapa en visuell representation av den underliggande åtgärden.

Om du t ex har skapat en åtgärd som heter BraTextredigerare som kör programmet BraTextredigerare, skapar du en körbar fil som heter BraTextredigerare. I filhanteraren och programhanteraren använder åtgärdsfilen ikonbilden som anges i åtgärdsdefinitionen.

Figur 5–10 åskådliggör förhållandet mellan åtgärdsdefinitionen, åtgärdsfilen och den faktiska posten i programhanterarfönstret.



Figur 5-10 Programikonen är en fil i programgruppen

Readme-filer

Till skrivbordet finns datatypen README som du använder till programmets README-filer. Uttrycket Readme förknippas oftast med något som är viktigt, dvs en fil eller något som ska läsas så snart som möjligt. Använd en av följande namngivningskonventioner:

- README
- readme
- README.*
- Read.*.Me
- read.*.me
- READ.*.ME

Skapa en särskild kontroll för huvudgruppen

För det mesta behöver du inte ange en kontrolldefinition för huvudgruppen. Användaren kan lägga till programmet i huvudgruppen genom att släppa åtgärdsikonen på installationsikonen på en panelundermeny.

Du kanske vill skapa en konfigurationsfil för Huvudgruppen som innehåller en kontrolldefinition för programmet om du vill att användare ska kunna installera en kontroll som fungerar annorlunda än åtgärdsikonen, om t ex kontrollen övervakar en fil och ändrar utseende när den övervakade filen ändras.

Huvudgruppens konfigurationsfiler placeras i katalogen `prgm_rot/dt/appconfig/types/språk`. Filerna måste använda namngivningskonventionen `namn.fp`.

Om du anger en konfigurationsfil med en kontroll, kan användaren lägga till kontrollen i en panelundermeny genom att släppa filen `*.fp` på installationsikonen på panelundermenyn.

Följande definition kan exempelvis placeras i en konfigurationsfil på huvudgruppen i programgruppen. Om användaren släpper den här filen på en installationsikon på en panelundermeny skapas en kontroll i panelundermenyn, som kör en upplaga av programmet BraTextredigerare. Om BraTextredigerare redan körs flyttas fönstret högst upp i fönsterstacken på den aktuella arbetsytan.

```
CONTROL BestTextEditorControl
{
  TYPE      icon
  ICON      BTEFPanel
  PUSH_RECALL  True
  CLIENT_NAME  BTEd
  PUSH_ACTION  BTEditor
  DROP_ACTION  BTEditor
  HELP_STRING  Startar programmet BraTextredigerare.
}
```

Mer information om hur du skapar konfigurationsfiler för huvudgruppen finns i:

- Kapitel 15
- direkthjälpen (man) till `dtfpfil(4)`

Steg 8: Registrera programmet med hjälp av dtappintegrate

Obs! Ett exempel på hur du kan registrera ett program finns i Steg 8 i "Exempel på hur du kan skapa ett registreringspaket" på sidan 76.

När du har skapat ett registreringspaket under en programrot kan du utföra själva programregistreringen.

Med programregistrering skapar du länkar mellan registreringspaketet och katalogerna i skrivbordssystemets sökvägar (se "Hur dtappintegrate integrerar program " på sidan 75).

Registrera ett program med dtappintegrate

Om programmet är skrivbordsanpassat körs vanligen dtappintegrate automatiskt som sista steg i installationsprocessen. Om det inte körs automatiskt, eller om du har skapat konfigurationsfiler för att integrera ett icke skrivbordsanpassat program, kan du köra dtappintegrate manuellt.

1. Logga in som root.

2. Kör kommandot:

```
/usr/dt/bin/dtappintegrate -s prgm_rot
```

där *prgm_rot* är skrivbordssystemets programrotskatalog. Mer information finns i direkthjälpen (man) till dtaction(1).

3. Öppna programgruppen Skrivbordsverktyg och dubbelklicka på Ladda om åtgärder.

4. Kontrollera att programmet är korrekt registrerat:

- a. Visa översta nivån i programhanteraren. Den nya programgruppen ska visas i programhanteraren.
- b. Öppna programgruppen och dubbelklicka på åtgärdsikonen.

Syntax och alternativ för dtappintegrate

```
dtappintegrate -s prgm_rot [-t målsökväg ] [-l språk ] [-u]
```

<code>-s prgm_rot</code>	Nödvändig parameter som anger vilken programrot som programmet är installerat under.
<code>-t målsökväg</code>	Valfri parameter. Som standard används placeringen /etc/dt/appconfig. Anger adressen dit skrivbordets konfigurationsfiler är länkade. Du måste använda en adress i programmets sökväg.
<code>-l språk</code>	Valfri parameter. Som standard används alla språk. Anger vilka språkberoende skrivbordskonfigurationsfiler som ska integreras.
<code>-u</code>	Valfri parameter, avintegrerar programmet och tar bort alla länkar som skapades under integreringen.

Hur dtappintegrate integrerar program

Funktionen hos dtappintegrate är att skapa länkar mellan de installerade filerna och de platser där skrivbordet letar efter konfigurationsfiler.

Åtgärder och datatyper

dtappintegrate skapar symboliska länkar från åtgärds- och datatypsdefinitionsfilerna i registreringspaketet till katalogen för hela systemet enligt åtgärdsdatabasens hjälpsökväg. Detta åstadkoms genom att länkar skapas från

```
prgm_rot/dt/appconfig/types/språk/*.dt
```

till

```
/etc/dt/appconfig/types/språk/*.dt
```

Hjälpinformationsfiler

dtappintegrate skapar symboliska länkar från hjälppilerna i registreringspaketet till katalogen för hela systemet enligt åtgärdsdatabasens hjälpsökväg. Detta åstadkoms genom att länkar skapas från

```
prgm_rot/dt/appconfig/help/språk/hjälpfil.sdl
```

till

```
/etc/dt/appconfig/help/språk/hjälpfil.sdl
```

Ikonfiler

dtappintegrate skapar symboliska länkar från ikonfilerna i registreringspaketet till katalogen för hela systemet enligt åtgärdsdatabasens hjälpsökväg. Detta åstadkoms genom att länkar skapas från

prgm_rot/dt/appconfig/icons/språk/ikonfiler

till

/etc/dt/appconfig/icons/språk/ikonfiler

Programgrupp

För att placera programgruppen för programmet på översta nivån i programhanteraren skapar `dtappintegrate` en länk mellan programgruppskatalogen i registreringspaketet och den systemtäckande adressen enligt programmets sökväg. Detta åstadkoms genom att länkar skapas från katalogen

prgm_rot/dt/appconfig/appmanager/språk/programgruppsnamn

till

/etc/dt/appconfig/appmanager/språk/programgruppsnamn

Exempel på hur du kan skapa ett registreringspaket

Följande steg skapar ett registreringspaket för ett befintligt, icke skrivbordsanpassat program som heter BraTextredigerare.

Information om ”BraTextredigerare”

I exemplet görs följande antaganden om programmet BraTextredigerare:

- Det har installerats i katalogen `/usr/BTE`.
- Användarens sessionsspråk är standardspråket, `C`.
- Kommandoraden som startar BraTextredigerare är:

`BTEd {filnamn}`

där *filnamn* är namnet på datafilen som ska öppnas i det nya fönstret. BraTextredigerare skapar ett eget fönster, d v s det körs inte i ett terminalemuleringsfönster.

- BraTextredigerare skapar och använder två typer av datafiler:

- Dokumentationsfiler. Namnkonventionen *.bte används. BraTextredigerare tillhandahåller en kommandorad för utskrift av .bte-datafiler. Syntax för det här kommandot är:

```
BTEPrint [-d mål] [-s] filnamn
```

där:

-d *mål* – anger målskrivaren.

-s – anger tyst utskrift. Programmets dialogruta för utskrift visas inte.

filnamn – anger den fil som ska skrivas ut.

- Mallfiler. Namnkonventionen *.tpl används. Mallfiler kan inte skrivas ut.
- De befintliga standardfilerna som inte är desktopprogram för BraTextredigerare innehåller resurser för gränssnittsteckensnitt samt förgrunds- och bakgrundsfärger.
- En hjälpvolym för BraTextredigerare skapades med skrivbordets Help Developer's Kit. När hjälpvolymen skapades användes följande källfiler:

```
.../BTEHelp.htg
.../graphics/BTE1.xwd
.../graphics/BTE2.xwd
```

och skapade den filen .../BTEHelp.sdl.

Steg för att registrera "BraTextredigerare"

Följande stegvisa procedur registrerar BraTextredigerare.

1. Ändra teckensnitt- och färgresurser.

I BraTextredigerares app-defaults-fil tar du bort resurser som anger:

- Teckensnitt för text
- Färger för förgrund och bakgrund

2. Skapa programroten.

Skapa katalogen:

```
/desktop_approots/BTE
```

Om du integrerar ett befintligt program bör du skapa programrotskatalogen någon annanstans än på installationsplatsen för programmet. Annars kan konfigurationsfilerna som du skapar tas bort när du uppdaterar programmet.

3. Skapa registreringspaketets kataloger.

Skapa följande kataloger:

```
/desktop_approots/BTE/dt/appconfig/types/C
/desktop_approots/BTE/dt/appconfig/help/C
/desktop_approots/BTE/dt/appconfig/icons/C
/desktop_approots/BTE/dt/appconfig/appmanager/C/BraTextredigerare
```

4. Skapa åtgärderna och datatyperna för programmet.

a. Skapa konfigurationsfilen för åtgärds- och datatypsdefinitionerna:

```
/desktop_approots/BTE/dt/appconfig/types/C/BTE.dt
```

b. Skapa åtgärdsdefinitionen för att köra BraTextredigerare:

```
ACTION BTEditor
{
    WINDOW_TYPE    NO_STUDIO
    ICON           BTERun
    DESCRIPTION    Dubbelklicka på ikonen eller släpp \
                  en BTE-datafil på den för att starta \
                  BraTextredigeraren.
    EXEC_STRING    /usr/BTE/BTEd %Arg_1%
}
```

c. Skapa datatypen för *.bte-filer:

```
DATA_ATTRIBUTES BTEDataFile
{
    DESCRIPTION    Datafil för BraTextredigeraren.
    ICON           BTEData
    ACTIONS        Open,Print
}

DATA_CRITERIA BTEDataFileCriterial
{
    DATA_ATTRIBUTES_NAME    BTEDataFile
    NAME_PATTERN             *.bte
    MODE                     f
}
```

d. Skapa datatypen för *.tpl-filer:

```
DATA_ATTRIBUTES BTETemplateFile
{
    DESCRIPTION    Mallfil för BraTextredigeraren.
    ICON           BTETempl
    ACTIONS        Open
}

DATA_CRITERIAL BTETemplateFileCriterial
{
    DATA_ATTRIBUTES_NAME    BTETemplateFile
    NAME_PATTERN             *.tpl
    MODE                     f
}
```

e. Skapa Öppna-åtgärden för *.bte-filer.

```
ACTION Open
{
  ARG_TYPE   BTEDataFile
  TYPE       MAP
  MAP_ACTION BTEditor
}
```

f. Skapa SkrivUt-åtgärden för *.bte-filer.

Här följer enkla SkrivUt-åtgärder som skriver ut datafiler. De här åtgärderna kräver ett värde för omgivningsvariabeln LPDEST och ignorerar utskriftsalternativet `-s`. (Om LPDEST inte anges kan åtgärden misslyckas.)

```
ACTION Print
{
  ARG_TYPE   BTEDataFile
  TYPE       MAP
  MAP_ACTION BTEPrintData
}

ACTION BTEPrintData
{
  WINDOW_TYPE NO_STDIO
  EXEC_STRING  BTEPrint -d $LPDEST %Arg_1%
}
```

Här följer en annan version av åtgärden BTEPrintData och ett medföljande skript. Tillsammans hanterar de situationer där LPDEST inte anges eller tyst utskrift begärs.

```
ACTION BTEPrintData
{
  WINDOW_TYPE NO_STDIO
  EXEC_STRING  /usr/BTE/bin/BTEenvprint \
               %(File)Arg_1%
}
```

Innehållet i kommandofilen `/usr/BTE/bin/BTEenvprint` är:

```
# BTEenvprint
#!/bin/sh
DEST='''
SILENT='''
if [ $LPDEST ] ; then
  DEST='-d $LPDEST'
fi
BTEPrint $DEST SILENT $1
```

g. Skapa Öppna-åtgärden för *.tpl-filer.

```
ACTION Open
{
  ARG_TYPE   BTETemplateFile
  TYPE       MAP
}
```

```
MAP_ACTION BTEditor
}
```

h. Skapa SkrivUt-åtgärden för *.tpl-filer.

```
ACTION Print
{
  ARG_TYPES BTETemplateFile
  TYPE      MAP
  MAP_ACTION NoPrint
}
```

NoPrint är en inbyggd åtgärd som visar en dialogruta där det står att filen inte går att skriva ut.

5. Placera hjälpfilerna i registreringspaketet.

a. Placera hjälpfilerna på följande adresser:

```
/desktop_approots/BTE/dt/appconfig/help/C/BTEHelp.sdl
  /desktop_approots/BTE/dt/appconfig/help/C/graphics/BTE1.xwd
  /desktop_approots/BTE/dt/appconfig/help/C/graphics/BTE2.xwd
```

b. Skapa filen

```
/desktop_approots/BTE/dt/appconfig/types/C/BTEhelp.dt.
```

Placera följande åtgärdsdefinition i filen:

```
ACTION BTEHelp
{
  WINDOW_TYPE      NO_STDIO
  EXEC_STRING      /usr/dt/bin/dthelpview -helpVolume \
                  BTEHelp.sdl
  DESCRIPTION      Öppnar hjälpen för BraTextredigerare.
}
```

6. Skapa ikoner för programmet.

Använd Ikonredigeraren för att skapa ikonerna. Tabell 5-3 visar vilken storlek på rutnät som ska användas.

TABELL 5-3 Rutnät för olika storlekar på ikoner

Namn	Storlek, Sidformat, Grad
<i>grundnamn.t.pm</i>	16 gånger 16
<i>grundnamn.m.pm</i>	32 gånger 32
<i>grundnamn.l.pm</i>	48 gånger 48

Skapa de här ikonfilerna i katalogen

/desktop_approots/BTE/dt/appconfig/icons/C:

- Ikoner som representerar den åtgärd som kör programmet: BTERun.t.pm, BTERun.m.pm, BTERun.l.pm
- Ikoner som representerar *.bte-filer: BTEData.t.pm, BTEData.m.pm,
- Ikoner som representerar *.tpl-filer: BTETempl.t.pm, BTETempl.m.pm
- Ikoner som representerar programgruppen (används i steg 7): BTEApp.t.pm, BTEApp.m.pm

7. Skapa programgruppen.

a. Skapa katalogen om du inte redan har gjort det.

/desktop_approots/BTE/dt/appconfig/appmanager/C/BraTextredigerare

- b. Det här steget är valfritt. Det skapar en unik ikon för programgruppen genom att skapa en datatyp och associerade åtgärder för programgruppen. Om du utelämnar det här steget använder programgruppen standardikonen.**

Lägg till följande datatyps- och åtgärdsdefinitioner i filen

/desktop_approots/BTE/dt/appconfig/types/C/BTE.dt. Datatypen anger vilken ikon som ska användas av programgruppen BraTextredigerare. Åtgärderna ger samma Öppna- och SkrivUt-beteende som hos de inbyggda programgrupperna.

```
DATA_ATTRIBUTES BestTextEditorAppGroup
{
    ACTIONS  OpenInPlace,OpenNewView
    ICON     BTEApp
}

DATA_CRITERIA BestTextEditorAppGroupCriterial
{
    DATA_ATTRIBUTES_NAME BestTextEditorAppGroup
    MODE                  d
}
```

```

        PATH_PATTERN    */appmanager*/BraTextredigerare
    }

ACTION Open
{
    ARG_TYPE    BestTextEditorAppGroup
    TYPE        MAP
    MAP_ACTION  OpenAppGroup
}

ACTION Print
{
    ARG_TYPE    BestTextEditorAppGroup
    TYPE        MAP
    MAP_ACTION  PrintAppGroup
}

```

- c. Skapa en ikon i programgruppen som ska starta programmet. För att göra det skapar du filen:**

```

/desktop_approots/BTE/dt/appconfig/appmanager/C \
/BraTextredigerare/BTEditor

```

och gör filen körbar.

- d. Skapa åtgärdsfilen i programgruppen som ska öppna hjälpvolymen. För att göra det skapar du filen:**

```

/desktop_approots/BTE/dt/appconfig/appmanager/C \
/BraTextredigerare/BTEHelp

```

och gör filen körbar.

- e. Placera andra filer i programgruppen, t ex "Readme"-filer, exempeldata och mallfiler.**

8. Registrera programmet.

I ett terminalemuleringsfönster:

- a. Logga in som root.**

- b. Kör kommandot:**

```

/usr/dt/bin/dtappintegrate -s /desktop_approots/BTE

```

- c. Öppna programgruppen Skrivbordsverktyg och dubbelklicka på Ladda om åtgärder.**

Diverse konfigurationer

Det här kapitlet handlar om installation och systemadministration.

- "Solaris CDE – katalogstruktur" på sidan 83
- "Huvudkonfigurationsfiler" på sidan 85
- "Starta inloggningsservern" på sidan 86
- "Montera en installerad CDE från en annan installationsadress för en arbetsstation eller nätverksserver" på sidan 88
- "Konfigurera skrivbordet för att använda flera skärmar" på sidan 89
- "Nätverksanslutna skrivbord" på sidan 91
- "Använda X-terminaler" på sidan 92
- "Inloggningsspråkområde och teckensnittssökväg" på sidan 93
- "Använda arbetsstationer som X-terminaler" på sidan 94
- "Särskilda CDE-konfigurationer" på sidan 95

Solaris CDE – katalogstruktur

Det här avsnittet beskriver huvudkatalogerna i skrivbordsomgivningen.

`/usr/dt`

Den här katalogen är Solaris CDE:s installationsadress. Den kan också vara en monteringspunkt för en fjärrhanterare. I Tabell 6-1 beskrivs underkatalogerna `/usr/dt`.

TABELL 6-1 /usr/dt Underkataloger

Underkatalog	Beskrivning
/bin	SCDE-program och SCDE-hjälpmedel
/lib	SCDE körtidsdelade bibliotek
/config	Standardfiler för systemkonfiguration
/man	[Valfritt] direkthjälp (man)
/app-defaults	Standardprogramresurser
/appconfig	Standardprogramikoner, typer och åtgärder
/examples	[Valfritt] CDE-kod/CDE-program, exempel
/include	[Valfritt] Tilläggsfiler (include files) för utvecklare
/palettes	Färger
/share	CDE AnswerBook, dokumentation och standardbakgrund

/etc/dt

Den här katalogen innehåller anpassade arbetsstationsspecifika konfigurationsfiler. Filerna gör att du kan anpassa din omgivning på följande sätt:

- Ange konfigurationsalternativ för X-server
- Använda flera skärmar
- Anpassa arbetsstationens åtgärdsfiler, datatyper, ikoner och teckensnitt

/var/dt

Den här katalogen används för att lagra temporära filer för Solaris CDE-program t ex inloggningshanteraren och programhanteraren.

\$Hemkatalog

Den här katalogen innehåller användarspecifika filer som gäller användarens skrivbordsinställningar. Där ingår ändringar av program, färginställningar, Arbetsyta-menyn, huvudgruppen och felloggar.

Huvudkonfigurationsfiler

De flesta anpassningar av din skrivbordsomgivning kan inkludera många filer. Följande filer är två exempel:

- `/usr/dt/config/Xconfig`
- `/usr/dt/config/Xservers`

Xconfig

`Xconfig` är huvudkonfigurationsfilen som används av `dtlogin`. Den anger inloggningsresurser och adresser för ytterligare filer som krävs av `dtlogin`. Följande exempel visar några rader som finns i standard-`Xconfig`-filen:

```
Dtlogin.errorLogFile:      /var/dt/Xerrors

Dtlogin.servers:          /usr/dt/config/Xservers

Dtlogin*session:         /usr/dt/bin/Xsession
```

Mer information om den här filen finns i Bilaga A. Du kan också granska de förklarande kommentarerna i själva filen.

Xservers

`dtlogin`, liksom XDM (X Display Manager) som den bygger på, använder filen `Xservers` för att ange hur den lokala Xservern startas. Som standard ser filens sista rad ut så här:

```
:0 Local local@console /usr/openwin/bin/Xsun :0
```

- `:0` innebär att Xservers skärm är `<localhost:0>`.
- `local` anger att den nya X11-servern ska startas lokalt.

Obs! Ersätt `local` med `foreign` om du vill ansluta till en aktiv X11-server.

- `console` innebär att kommandoradsinloggningen använder `/dev/console`.
 - `/usr/openwin/bin/Xsun` är sökvägen till X11-servern.
-

Obs! Omgivningarna Solaris CDE och OpenWindowsTM™ kör samma X11-server.

Mer information om den här filen finns i direkthjälpen (man pages) till `dtlogin(1)`. Du kan också granska de förklarande kommentarerna i själva filen.

Starta inloggningsservern

Inloggningsservern startas för det mesta automatiskt när systemet startas. Du kan också starta inloggningsservern från en kommandorad, men du måste logga in som rotanvändare först.

Så här anger du att inloggningsservern ska starta när systemet startas:

- ◆ **Skriv följande kommando och tryck på Retur:**

```
# /usr/dt/bin/dtconfig -e
```

Det här gör att en `S99dtlogin`-fil läggs till i katalogen `/etc/rc2.d`. Inloggningsservern startar automatiskt när du startar om.

Så här hindrar du att inloggningsservern startar automatiskt när systemet startas:

- ◆ **Skriv följande kommando och tryck på Retur:**

```
# /usr/dt/bin/dtconfig -d
```

Så här startar du inloggningsservern från en kommandorad:

- ◆ **Skriv följande kommando och tryck på Retur:**

```
# /usr/dt/bin/dtlogin -daemon; exit
```

Obs! Även om det går att starta inloggningsservern från kommandoraden för tillfällig konfigurationstestning, bör inloggningsservern normalt startas när systemet startas.

Så här avbryter du inloggningsservern, Xserver, och hela Solaris CDE-skrivbordet:

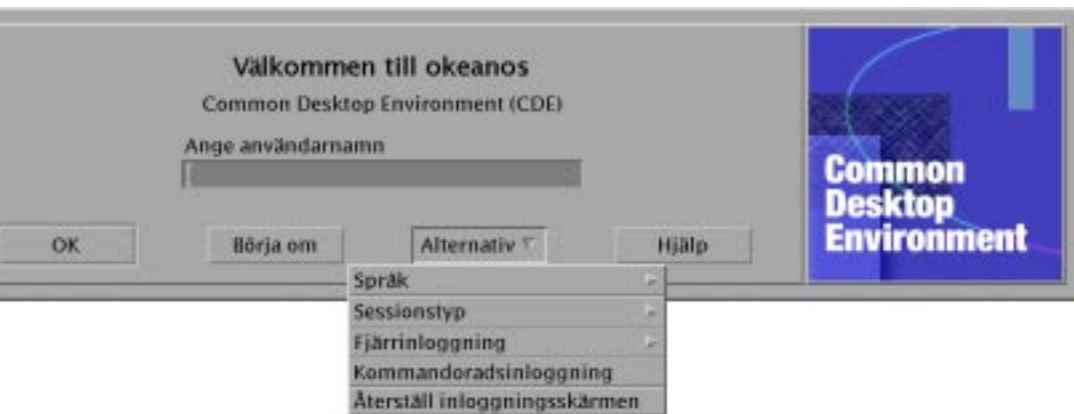
- ♦ **Skriv följande kommando och tryck på Retur:**

```
# /usr/dt/bin/dtconfig -kill
```

Mer information om skrivbordskonfigurationsverktyget `dtconfig` finns i Bilaga A. Där finns en kopia av direkthjälpssnittet (man page) till `dtconfig.1`.

Så här avslutar du Solaris CDE:s inloggningsskärm med hjälp av alternativet Kommandoradsinloggning:

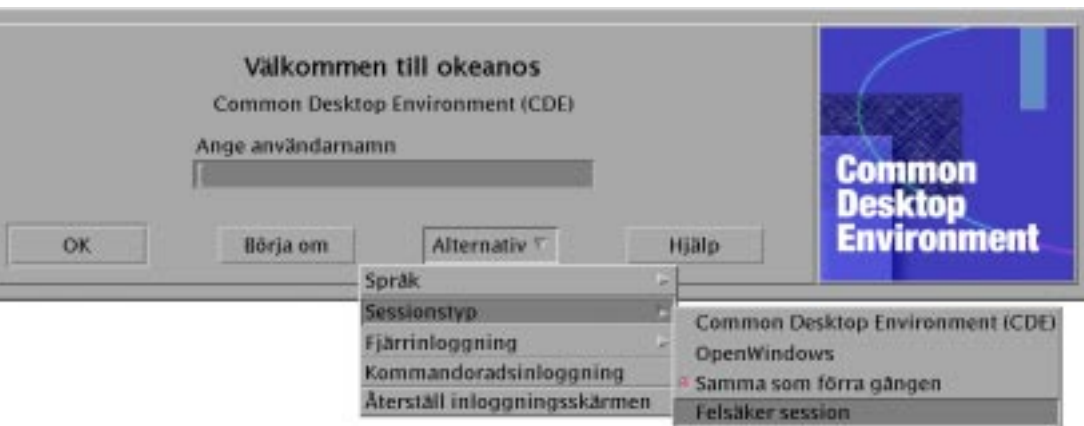
- ♦ **Välj Kommandoradsinloggning på Alternativ-menyn på Solaris CDE:s inloggningsskärm. När skärmen är tom trycker du på Retur för att visa en ledtext för inloggning.**



Obs! Skrivbordets bakgrundsprogram för inloggning kommer att starta skrivbordets inloggningsskärm automatiskt när du är färdig med kommandoradsinloggningen.

Så här startar du ett enstaka terminalemuleringsfönster:

- ♦ **Välj Felsäker session på undermenyn Session på Alternativ-menyn på Solaris CDE:s inloggningsskärm.**



Om du vill låta Xservern stå på väljer du Felsäker session på undermenyn Session. Då startas ett `xterm`-fönster. Alternativet Felsäker session är alltid tillgängligt, även när alternativet Kommandoradsinloggning inte är tillgängligt.

Montera en installerad CDE från en annan installationsadress för en arbetsstation eller nätverksserver

Du kan montera en installerad CDE från en annan installationsadress för arbetsstation eller nätverksserver om du inte har det nödvändiga diskutrymme på din arbetsstation, eftersom den proceduren inte använder något lokalt diskutrymme.

Obs! Eftersom katalogstrukturen `/usr/dt` skiljer sig åt mellan olika versioner av Solaris och olika klientarbetsstationer (SPARC-katalogstrukturen skiljer sig från Intel-katalogstrukturen), måste klientarbetsstationen montera en lämplig `/usr/dt`-bild för NFS-server. Ett SPARC-system med Solaris 2.5 bör exempelvis montera `/usr/dt` från ett annat Solaris 2.5-system med `/usr/dt`-installerad CDE.

▼ Montera en installerad CDE

1. **Montera en `/usr/dt`-katalog från en tidigare installerad arbetsstation eller nätverksserver till katalogen `/usr/dt` på din arbetsstation.**
2. **Skriv `/usr/dt/bin/dtconfig -inetd`**

3. Aktivera Solaris skrivbordsinloggning genom att skriva:

```
/usr/dt/bin/dtconfig -e
```

4. Starta om arbetsstationen.

Mer information om skrivbordskonfigurationsverktyget `dtconfig` finns i Bilaga A. Där finns en kopia av direkthjälpssnittet (man page) till `dtconfig(1)`.

▼ Avmontera en monterad CDE-katalog

1. Koppla ur Solaris skrivbordsinloggning genom att skriva:

```
/usr/dt/bin/dtconfig -d
```

2. Skriv `/usr/dt/bin/dtconfig -inetd.ow`

3. Avmontera `/usr/dt`.

4. Starta om arbetsstationen.

Mer information om skrivbordskonfigurationsverktyget `dtconfig` finns i Bilaga A. Där finns en kopia av direkthjälpssnittet (man page) till `dtconfig(1)`.

Konfigurera skrivbordet för att använda flera skärmar

Standardinloggningen startar skrivbordet på en skärm. Du kan redigera filen `Xconfig` så att skrivbordet startas på flera skärmar. Du måste vara inloggad som rotanvändare för att ändra den här filen.

Obs! Om du vill att redigeringen av information om inloggningskonfiguration ska gå snabbare, använder du alternativet `Felsäker session` för att köra ett enda `xterm`-fönster i stället för hela skrivbordet.

▼ Starta skrivbordet på flera skärmar

1. Skapa en kopia av filen `Xservers` genom att skriva följande kommando och trycka på Retur:

```
# cp /usr/dt/config/Xservers /etc/dt/config/Xservers
```

Obs! `/etc/dt/config/Xservers` åsidosätter `/usr/dt/config/Xservers`.

2. Redigera filen `/etc/dt/config/Xservers` och ange två bildskärmsminnen (skärmar):

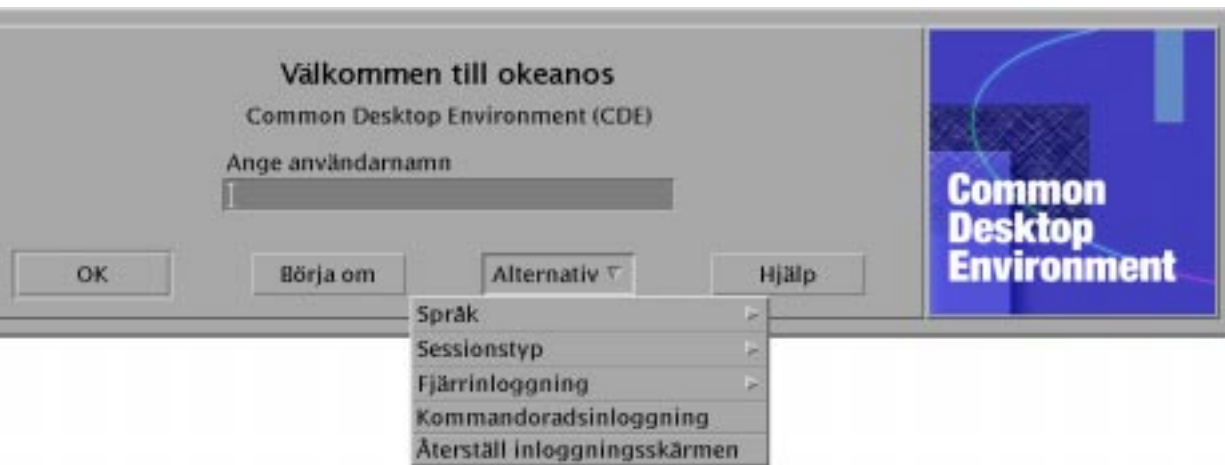
- a. Leta rätt på raden:

```
:0 Local local_uid@console root /usr/openwin/bin/Xsun :0
```

- b. Lägg till två `-dev`-alternativ i slutet av raden. I detta exempel antas att deras namn är `/dev/fb0` och `/dev/fb1`. Lägg till följande, föregånget av ett blanksteg:

```
-dev /dev/fb0 -dev /dev/fb1
```

3. Välj Återställ inloggningsskärmen på Alternativ-menyn.



Obs! När du kör Solaris CDE på flera skärmar visas en huvudgrupp på varje skärm. Dessa huvudgrupper är oberoende av varandra och har olika konfigurationsfiler.

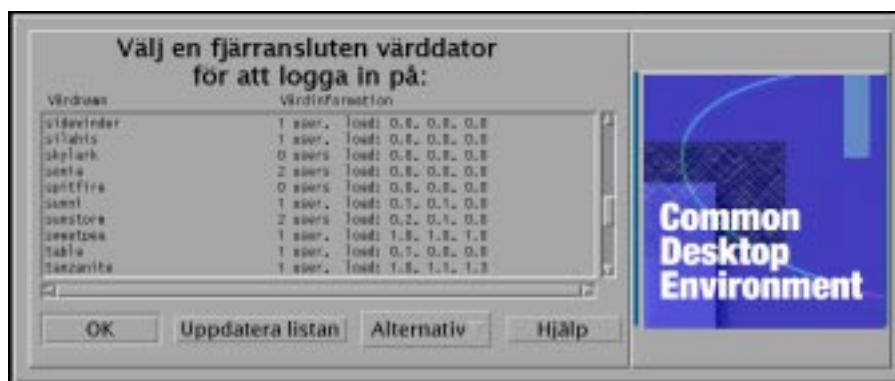
Nätverksanslutna skrivbord

Inloggningshanteraren för Solaris CDE är nätverksmedveten. Inloggningsskärmar svarar som standard på frågor från Väljaren i Solaris CDE.

Så här visar du en lista med fjärranslutna värddatorer:

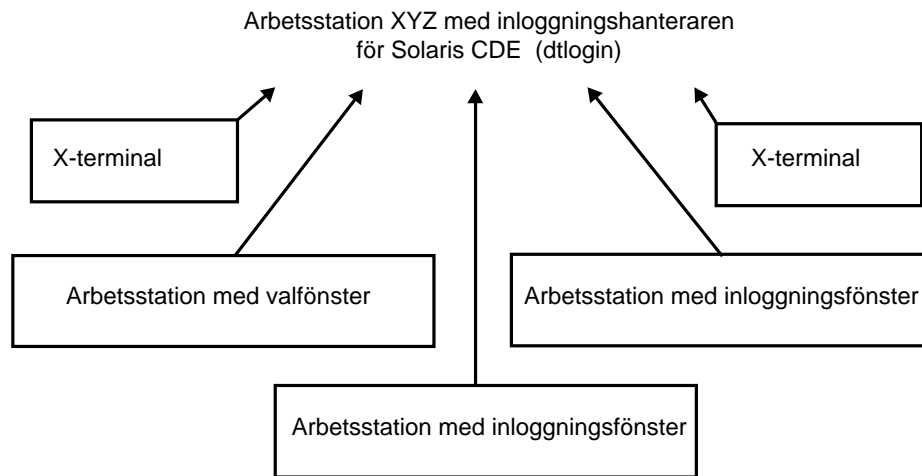
- ♦ **Välj Välj server från en lista på Fjärrinloggning-menyn på Solaris CDE inloggningsskärm.**

Mer information om hur du kan köra Véljaren oberoende av inloggningsskärmen finns i "Använda arbetsstationer som X-terminaler" på sidan 94. Följande skärm visar ett exempel på en lista över tillgängliga servrar i Véljaren.



Du kan markera en ledig server i listan och använda dess resurser via Véljaren för att logga in på Solaris CDE.

En nätverksanslutna arbetsstation som kör inloggningshanteraren för Solaris CDE kan både stödja en lokal skrivbordsanvändare och flera fjärranvändare inklusive Xterminal-användare genom att arbetsstationen används som Solaris CDE-skrivbordsserver. (Se Figur 6-1.)



Figur 6-1 Nätverksanslutna skrivbord

Obs! Figur 6.1 kan beskriva en blandad maskinvaruomgivning.

Använda X-terminaler

Alla X-terminaler som stöder XDM-protokollet kan använda Väljaren för att logga in på Solaris CDE. Mer information om hur du kan köra Väljaren finns i "Använda arbetsstationer som X-terminaler" på sidan 94. Programvaran till SPARC Xterminal™ 2.0 (eller senare) fungerar bra med Solaris CDE. Klicka på Egenskaper på Xterminalens kontrollfönster och välj Session som kategori för att visa installationsskärmen som visas i Figur 6-2.



Figur 6-2 Session, installationskärm

Välj None som lokal fönsterhanterare (Local Window Manager), och XDM Indirect som XDM-anslutning till fjärrsystemet som kör Solaris CDE.

Inloggningspråkområde och teckensnittssökväg

Markera det språk du vill ha när du loggar in på Solaris CDE. Följande Solaris CDE-inloggningskärm visar alternativet Språk på Alternativ-menyn.



På arbetsstationer innehåller teckensnittssökvägar automatiskt teckensnitt (och aliasnamn) som är associerade med både det språk du vill använda och systemets basspråkområde, "C". För X-terminaler tillhandahålls de här teckensnitten automatiskt av en X11-teckensnittsserver.

Använda arbetsstationer som X-terminaler

Om du har äldre arbetsstationer eller arbetsstationer med 16 MB minne eller mindre, kan du använda dem som X-terminaler när de laddas med Solaris 2.4-programvara eller senare (eller Solaris 2.3-programvara med Xservers stora korrigeringsfil).

Obs! Om teckensnitten på arbetsstationens skärm inte visas normalt när du utför nedanstående procedur kanske inte Solaris CDE är installerat på värdarbetsstationen. Använd kommandoradsalternativet `pkgadd` för att lägga till Solaris CDE:s teckensnittspaket `SUNWdtft()` separat från en utgåva av Solaris CDE 1.0.x.

▼ Använda Väljaren för att markera en värd-CDE-inloggning

1. Avsluta det fönstersystem som körs.

2. Starta Väljaren. Skriv kommandon (med hjälp av `ssh`) från systemfönstrets kommandorad:

```
setenv OPENWINHOME /usr/openwin
```

```
/usr/openwin/bin/X -indirect CDE-loginvärd
```

Xserver startar och visar Väljarskärmen från värddatorn.

3. Välj den inloggningsvärd du vill ha från Väljarskärmen.

▼ Använda en specifik värd-CDE-inloggning

- ◆ Skriv följande kommando och tryck på Retur:

```
/usr/openwin/bin/X -once -query CDE-loginvärd
```

Xserver startar och visar inloggningsskärmen från värddatorn. Alternativet `-once` avslutar servern efter en inloggnings- eller utloggningssession. Om alternativet `-once` inte läggs till visas inloggningsskärmen igen efter utloggningen från Solaris CDE.

▼ Använda den första tillgängliga värdinloggningen

- ◆ Skriv följande kommando och tryck på Retur:

```
/usr/openwin/bin/X -broadcast
```

Det här startar Xserver. Kommandot skickar en fråga på det lokala delnätet efter en XDM-inloggningstjänst (X Display Manager). Om några system på delnätet kör inloggningshanteraren för Solaris CDE (eller något annat XDM-baserat inloggningsfönster), placerar den första värddator som svarar sitt inloggningsfönster på skrivbordet.

Särskilda CDE-konfigurationer

Det här avsnittet handlar om särskilda konfigurationer.

Anpassa utskrift av brev

Postverktyget i CDE skriver ut brev med hjälp av åtgärden Skriv ut som är definierad för datatypen `DTMAIL_FILE`. Den använder inte utskriftskommandot som anges av OpenWindows Postverktyg. Om du vill ändra inställningarna för utskrift måste du ändra åtgärden Print.

Ändra åtgärden Print

1. Skapa följande fil i en textredigerare:

```
Hemkatalog/.dt/types/dtmail.dt
```

2. Skriv följande rader i den här filen:

```
#  
  
# Ändra standardinställning för åtgärden Print i postverktyget  
  
#  
  
ACTION Print  
  
{  
  
LABEL          Print  
  
ARG_TYPE       DTMAIL_FILE  
  
TYPE           COMMAND  
  
WINDOW_TYPE    NO_STDIO  
  
EXEC_STRING    sh -c ' \  
  
                dtmailpr -p -f %(File)Arg_1% | mp -m -l | \  
  
                dtlp -u %(File)Arg_1%;'  
  
}
```

Obs! `dtmailpr` är ett utskriftsfilter som skiljer av bilagor så att man får en ström av ren oformaterad text som resultat. `dtlp` är CDE:s standardgränssnitt för `lp`. `%(File)Arg_1%` är filen som skrivs ut.

3. Ändra fältet `EXEC_STRING` så att det innehåller de utskriftskommandon du vill använda.

4. Starta om Postverktyget.

Konvertera kalendrar till nya dataformat

Dataformatet i version 4 är ett nytt format som stöds av CDE:s Kalender. OpenWindows Kalender kan inte läsa det här formatet. Om du behöver växla fram och tillbaka mellan plattformarna OpenWindows och CDE, bör du inte konvertera kalendern till dataformatet version 4. Du kan emellertid använda kommandofilen `sdtcm_convert` för att rensa i kalendern.

Obs! Du kan ta reda på kalenderns dataversion genom att välja Om Kalender på Kalenderns Hjälp-meny.

Använd annars kommandofilen `sdtcm_convert` för att konvertera kalendern till dataformatet version 4.

Du får mer information om den här filen i direkthjälpen (man) till `sdtcm_convert(1)`.

Obs! Om du bara installerade CDE-paketet för slutanvändare installerades inte direkthjälpen. För att kunna visa direkthjälpen måste du installera den med verktyget `pkgadd`.

Lägga till AnswerBook-paketet från nätverket

När du installerar Solaris CDE ger installationsmenyn möjlighet att installera AnswerBook:s CDE-paket. Om du väljer Ja installeras AnswerBook-paketet i katalogen `/usr/dt/share/answerbooks/språk` (*språk* är det lokala namnet på den specifika AnswerBook-översättningen).

Eftersom AnswerBook-paketet kräver 109 MB diskutrymme kanske du inte vill installera det. Följande två alternativ är emellertid tillgängliga:

- Du kan montera AnswerBook-paketet till `/usr/dt/share/answerbooks/språk` (eller länka till det från `/net/. . .`).

- Du kan lägga till omgivningsvariabeln `AB_CARDCATALOG` i *Hemkatalog*/`.dtprofile`. Om du t ex har en exporterad version i ditt nätverk lägger du till följande rad i *Hemkatalog*/`.dtprofile`:

```
export AB_CARDCATALOG=/net/världnamn/usr/dt/share/answerbooks \
/språk/ab_cardcatalog
```

Installera CDE-omgivning utanför CDE-skrivbordet

Omgivningsvariabler som används av dina Solaris CDE-program kan installeras utanför CDE-skrivbordet. Du kan t ex logga in på en fjärrarbetsstation och visa ett CDE-program tillbaka till din arbetsstation. Använd CDE-verktyget `dtsearchpath` för att installera olika CDE-skalomgivningsvariabler.

För Bourne- och Korn-skal skriver du följande kommando:

```
eval `/usr/dt/bin/dtsearchpath`
```

För C-skal skriver du följande kommando:

```
eval `/usr/dt/bin/dtsearchpath -c`
```

Du kan följa det här kommandot genom att ange `DISPLAY` tillbaka till den lokala arbetsstationen och sedan fjärrköra CDE-programmet. Resultaten visas på den lokala arbetsstationen.

Obs! I det här exemplet ingår Bourne- och Korn-skalets `dtsearchpath`-syntax i CDE:s exempelimplementering, som är tillgänglig på alla plattformar som kör CDE. C-skalets alternativ (`-c`) är däremot bara tillgängligt på Sun-plattformar.

Skrivbordsomgivningsfil

CDE-skrivbordet tillhandahåller en körtidsomgivning för OpenWindows DeskSet™-programmen. Det krävs inga särskilda CDE-installationer för den här funktionen. Vissa av omgivningens inställningar kommer från följande fil:

```
/usr/dt/config/Xsession.d/0015.sun.env
```

Om du vill lägga till eller ta bort omgivningsinställningar för en viss arbetsstation, kan den här filen antingen redigeras på plats eller först kopieras till följande filadress:

```
/etc/dt/config/Xsession.d/0015.sun.env
```

Ett exempel på ett tillägg kan vara att starta OpenWindows gamla hjälpprogram Tangentbord för användning med OpenWindows DeskSet-program. Eftersom de flesta program på ett CDE-skrivbord (och de flesta Sun-användare) inte använder

hjälpprogrammet Tangentbord, har detta utelämnats från standardstartsekvensen för att förbättra skrivbordets startprestanda.

Se även kommentarerna i filen `0015.sun.env` om du vill ha ytterligare information om valfri start av OpenWindows Tangentbord (`vkdb`).

Använda diskett och CD-skiva med MAE (Macintosh Application Environment)

Om du har installerat MAE (Macintosh Application Environment) 1.0 och tänker använda flyttbara media (disketter eller CD-skivor) via programmet filhanteraren i OpenWindows, måste du redigera filen `/etc/rmmount.conf`. Det säkerställer att MAE fungerar korrekt med flyttbara media i filhanteraren.

Ändra filen `/etc/rmmount.conf`

1. **Bli rotanvändare genom att skriva `su` och skriv därefter ditt lösenord.**
2. **Ändra till katalogen `/etc`. Skriv följande kommando och tryck på Retur:**

```
# cd /etc
```

3. **Öppna filen `/etc/rmmount.conf` i en textredigerare.**

4. **Flytta följande rad under `#Åtgärder` till slutet av listan:**

```
action floppy action_macfs.so
```

Till exempel:

```
# more rmmount.conf
```

```
# @(#)rmmount.conf 1.2      92/09/23 SMI
```

```
#
```

```
# Konfigurationsfil för utbytbara media.
```

```
#
```

```
# Identifierare för filsystem
```

```
ident hsfs ident_hsfs.so cdrom

ident ufs ident_ufs.so cdrom floppy

ident pcfs ident_pcfs.so floppy

ident macfs ident_macfs.so floppy

# Åtgärder

action cdrom action_filemgr.so

action floppy action_filemgr.so

action floppy action_macfs.so
```

5. Spara och avsluta.

När MAE är installerat infogas raden `macfs action` i filen `rmmount.conf`. Om MAE avinstalleras tas raden bort.

Den här ändringen gör att du kan använda flyttbara media i OpenWindows eller filhanteraren i CDE och fortfarande använda Macintosh-disketter i MAE. Den hindrar dock MAE från att ta emot oformaterade disketter, oläsbara disketter eller DOS-disketter. Alla dessa disketter går till Filhanteraren i OpenWindows med den här installationsproceduren.

Om du vill att MAE ska ha åtkomst till oläsbara disketter, oformaterade disketter eller DOS-disketter, måste raden `action floppy action action_macfs.so` placeras före raden `action floppy action_filemgr.so`.

Konfigurera skrivbordet i nätverk

Skrivbordet är utformat för att fungera i nätverksintensiva omgivningar.

- "Nätverk på skrivbord – översikt" på sidan 101
- "Generella steg för att konfigurera nätverket" på sidan 106
- "Konfigurera operativsystemets basnätverkstjänster för skrivbordet" på sidan 106
- "Konfigurera skrivbordsklienter och servrar" på sidan 109
- "Administrera programtjänster" på sidan 114

Skrivbordets arkitektur gör att systemadministratörer kan distribuera dataresurser över hela nätverket, inklusive:

- Program
- Datafiler för program.
- Skrivbordssessionstjänster (skrivbordsprogram som inloggningshanteraren och filhanteraren)
- Hjälptjänster (hjälpfiler kan placeras på en central hjälpserver)

Nätverk på skrivbord – översikt

Operativsystemet erbjuder flera nätverkstjänster, inklusive distribuerade filsystem och fjärrkörning. X-servern erbjuder ytterligare nätverksfunktioner, inklusive tillgång till fjärrskärmar och tjänster för säkerhetshantering.

Skrivbordet placerar ett användargränssnitt ovanpå de här nätverksfunktionerna. Avsikten med gränssnittet och dess underliggande arkitektur är att göra nätverkssystem:

- Enklare att använda. Användare kan köra program och komma åt datafiler utan att veta exakt var i nätverket programmen och filerna finns.
- Enklare att administrera. Skrivbordet innehåller verktyg för programintegrering och nätverkssökvägar som gör det enklare för systemet att hitta fjärrdata och fjärrinstallerade program. Dessutom gör skrivbordets filnamnsavbildningsprocess det lättare att administrera komplexa nätverk som innehåller flera servrar.
- Flexibelt. Skrivbordets administrativa funktioner är utformade för vissa vanliga nätverkssituationer, men kan även användas för många andra anpassade nätverkskonfigurationer.

Typer av nätverkstjänster för skrivbord

Via ett nätverk kan en användare komma åt olika datatjänster som distribueras på andra system, t ex:

- Skrivbordssessionen och dess program, t ex arbetsytahanteraren och filhanteraren
- Andra program
- Datafiler

I nätverkssammanhang används uttrycket *server* för att beskriva ett system som tillhandahåller datatjänster för ett eller flera system. När ett system tar emot tjänster från en server, kallas det *klient* till den servern.

I ett komplext nätverk kan ett system använda tjänster som finns på flera olika system i nätverket. Ett system kan också fungera som en viss typ av server (t ex en sessionsserver) och samtidigt vara en klient (t ex till en programserver).

Typiska nätverkssituationer

Från ett skrivbordsperspektiv kan en typisk nätverkskonfiguration innehålla kombinationer av dessa huvudkomponenter:

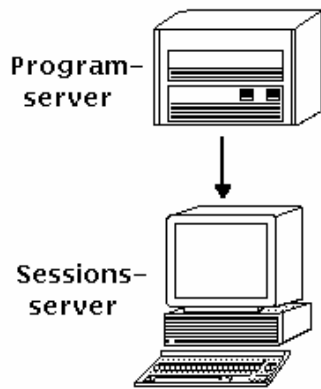
Skärmar – där X-servern körs

Inloggnings- eller sessionsservrar – där skrivbordsprogrammen (inloggningshanteraren, arbetsytahanteraren och liknande) körs

Programservrar – där andra program körs

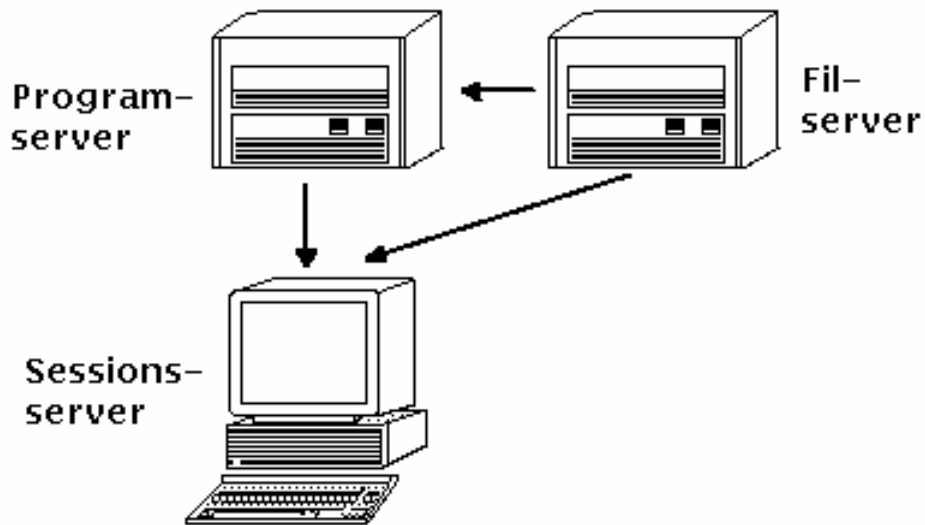
Filservrar – där data som används av program finns

En av de vanligaste nätverkskonfigurationerna innebär att system ansluter till en programserver. Figur 7-1 föreställer en arbetsstation som använder en programserver. X-servern och skrivbordssessionen körs på arbetsstationen.



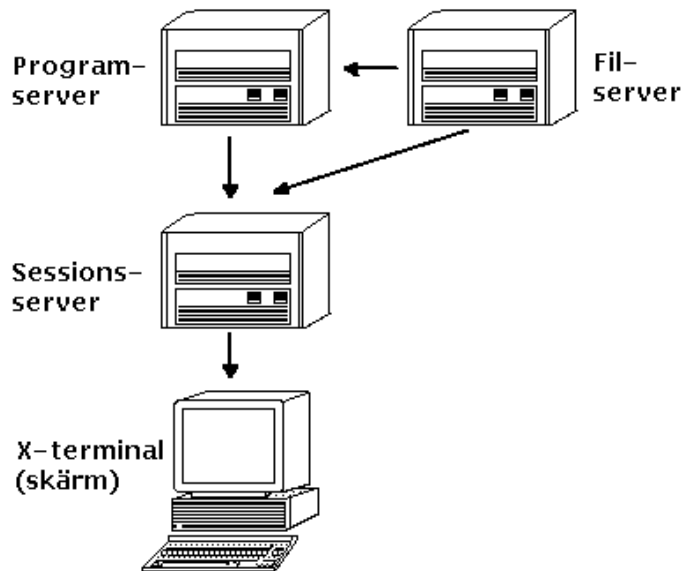
Figur 7-1 Programserver ger en skrivbordssession tillgång till olika tjänster.

Nätverk brukar också använda filserverar till att lagra stora mängder data. Ett program som körs på en programserver eller ett skrivbordsprogram (filhanteraren behöver t ex tillgång till datafiler för att kunna visa dem i ett filhanterarfönster) kan använda sig av dessa data.



Figur 7-2 Filserver ger program- och sessionsservrar tillgång till data.

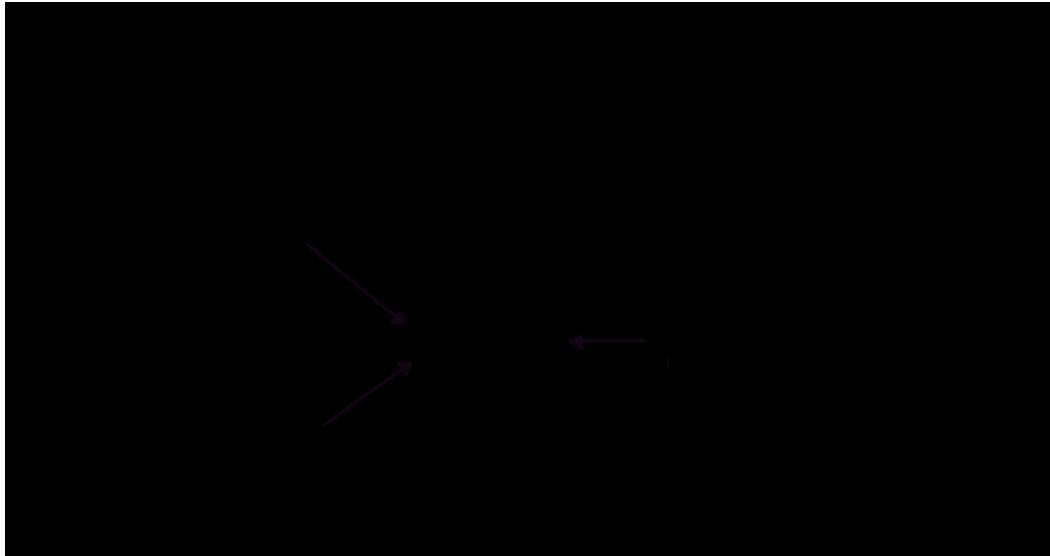
X-terminaler kör X-servern och kommer åt skrivbordssessionstjänster från ett annat system.



Figur 7-3 X-terminaler får sessionstjänster från en sessionsserver

Andra nätverkssituationer

Skrivbordet är flexibelt och kan stödja mer komplexa nätverkskonfigurationer. Dessa omfattar normalt olika tjänster som, tillsammans med filserverar, görs tillgängliga för programserverar.



Figur 7-4 De tjänster som behövs för en skrivbordsprogramserver kan distribueras

Sammanfattning – servertyper

Skärmar – där X-servern körs

Inloggnings- och sessionsserver – systemet som kör skrivbordssessionen (inloggningshanteraren, sessionshanteraren, fönsterhanteraren, filhanteraren och liknande).

Programserver – ett system där ett program körs. Kallas även *exekveringsdator*.

Filserver – ett system där datafiler för program lagras.

Hjälpsserver – ett system där hjälpfiler lagras.

Åtgärdsdatabasserver – ett system där filer som innehåller åtgärds- och datatypsdefinitioner lagras.

Ikonserver – ett system där ikonfiler lagras.

Nätverket kan innehålla ytterligare servrar, exempelvis en lösenordsserver, postserver, videoservert osv.

Generella steg för att konfigurera nätverket

Det finns tre allmänna steg för att konfigurera skrivbord i nätverket:

1. Konfigurera operativsystemets basnätverkstjänster.

Dessa tjänster tillhandahålls av operativsystemet och är nödvändiga för skrivbordet. Se "Konfigurera operativsystemets basnätverkstjänster för skrivbordet" på sidan 106.

2. Installera och konfigurera skrivbordets program och tjänster för nätverk.

Detta är tjänsterna som behövs för skrivbordet, oavsett vilken typ av klient och server som installeras. Se "Konfigurera skrivbordsklienter och servrar" på sidan 109.

3. Konfigurera en viss typ av server eller klient.

När du t ex konfigurerar en programserver krävs andra metoder än när du konfigurerar en filserver. Se "Administrera programtjänster" på sidan 114.

Konfigurera operativsystemets basnätverkstjänster för skrivbordet

Skrivbordet kräver följande basnätverkskonfiguration:

- Användaren måste ha ett inloggningskonto på sessionsservern och på varje system som tillhandahåller skrivbordstjänster för sessionsservern. Användaren måste ha samma användar-ID och grupp-ID på alla klienter och serversystem.
- Systemen måste ha tillgång till fjärrfilsystem som innehåller data som används av sessionen och andra program.
- Skrivarkön för kommandot `lp` måste vara konfigurerad för åtkomst av fjärrskrivare.
- `sendmail` måste vara konfigurerat för e-posttjänster.
- X-verifiering måste vara angivet.

Inloggningskonto för användare

I det här avsnittet beskrivs vad som krävs för inloggningskonton på skrivbord i nätverket.

Inloggningskonton

Användaren måste ha inloggningskonto på:

- Alla system som tillhandahåller tjänster för skrivbordet, inklusive programserverar, filserverar och system som innehåller nätverksskrivare.
- Alla sessionsserverar som användaren använder. Sessionsserverar används normalt med X-terminaler.

Konsekvent användar- och grupp-ID

UNIX-användare identifieras av ett inloggningsnamn och ett numeriskt användar-ID (UID). Användaren måste ha samma inloggningsnamn och användar-ID på alla klienter och serversystem i nätverket.

UNIX-användare kopplas också till en eller flera inloggningsgrupper. Varje grupp har ett gruppnamn och ett numeriskt grupp-ID (GID). I ett nätverk ska alla system använda samma gruppnamn och grupp-ID.

Mer information finns i direkthjälpen (man) för `id(1)` och `id(1M)`.

Konfigurera distribuerad filsystemåtkomst

Skrivbordet använder NFS för att dela filer mellan system. Du måste identifiera alla filsystem i nätverket som innehåller delade filer, och kontrollera att de är korrekt monterade i alla lämpliga system.

Normalt måste du tillhandahålla följande fjärrfilåtkomst:

- Användarens hemkatalog måste delas med alla skrivbordsklienter och serversystem. Detta är nödvändigt eftersom:
 - Hemkatalogen innehåller datafiler som program i fjärrsystem ska ha tillgång till. Program som använder datafiler har ofta hemkatalogen som standardadress för datafilerna.
 - Hemkatalogen är standardverifieringskatalogen för `dtspcd`. Mer information om `dtspcd` finns i "Konfigurera styrbakgrundsprogram för underprocesser" på sidan 113.

- Om användaren kräver åtkomst till datafiler som inte finns i hemkatalogen, måste dessa datafiler delas av alla skrivbordsklienter och serversystem som arbetar med datafilerna.
- Skrivbordsinstallations- och skrivbordskonfigurationskatalogerna (`/usr/dt` och `/etc/dt`) måste delas av alla skrivbordsklients- och serversystem, så att alla användarens program har tillgång till samma skrivbordskonfigurationsfiler.

Hemkatalog för nätverk

Arbetsstationer i nätverk fungerar mest effektivt när användaren har en enstaka hemkatalog som delas mellan alla klienter och serversystem i nätverket.

Med en nätverksansluten hemkatalog kan användaren använda olika system i nätverket utan att egna anpassningar och konfigurationer går förlorade. Detta beror på att egna anpassningar, och informationen som krävs för att återställa föregående session, sparas i underkataloger till hemkatalogen.

En hemkatalog behövs också för:

- Standard-X-behörighetsmekanismen (se "Konfigurera som X-verifiering" på sidan 109).
- Styrbakgrundsprogram för underprocesser. Detta bakgrundsprogram startar fjärrprogram och måste kunna skriva i användarens hemkatalog.

Konsekventa filnamn

Du bör konfigurera nätverket så att användarna kan komma åt sina datafiler från alla system med samma namn. Detta kallas för att använda *konsekventa filnamn*, och uppnås normalt genom att du skapar lämpliga symboliska länkar. Du kan t ex konfigurera alla system så att varje användares hemkatalog är tillgänglig som `/users/inloggningsnamn` genom att skapa en symbolisk länk till katalogens verkliga plats.

Konfigurera åtkomst av fjärrskrivare

Skrivbordssystemet använder skrivarkön för kommandot `lp` för att komma åt lokala skrivare och fjärrskrivare. Mer information om hur du konfigurerar kommandot `lp` finns i direkthjälpen (`man page`) för `lpadmin` (1M).

Innan du försöker skriva ut med skrivbordssystemets grafiska gränssnitt, bör du kontrollera om du kan skriva ut på alla skrivare med hjälp av kommandot `lp`.

Se till att använda konsekventa namn på utskriftsenheter. Om t ex en viss skrivare kallas `Postscript1` i systemet som den är direktansluten till, bör namnet `Postscript1` användas av alla system som har fjärråtkomst till skrivaren.

Konfigurera e-post

Skrivbordssystemets postverktyg använder `sendmail` för att leverera post mellan system. Mer information om hur du konfigurerar anslutningsbarhet för e-post finns i direkt hjälpen (man) till `sendmail(1M)`.

Innan du försöker skicka och ta emot post från skrivbordet, ska du kontrollera att du kan skicka och ta emot post med hjälp av kommandot `mailx`.

Konfigurera som X-verifiering

Skrivbordet använder standard X-mekanism för att verifiera om fjärrprogram (X-klienter) har åtkomst till en lokal skärm. Det enklaste sättet att göra detta är ge alla användare en nätverkshemkatalog. Detta garanterar att följande krav uppfylls:

- Användaren måste ha läs- och skrivbehörighet till filen `Hemkatalog/.Xauthority`.
- Filen `.Xauthority` på en programserver måste innehålla en "magic cookie" för skärmen där programmet kommer att köras.

Mer information finns i direkt hjälpen (man) till `X(1)` och `xauth(1)`.

Konfigurera skrivbordsklienter och servrar

Det här avsnittet beskriver krav för nätverkskonfiguration som är specifika för skrivbordssystemet, dvs de här funktionerna tillhandahålls av skrivbordssystemet i stället för det grundläggande operativsystemet.

Avsnittet består av två delar:

- Konfigurera inloggnings- och sessionstjänster.
- Konfigurera tjänster som är nödvändiga för program och dess data. Detta omfattar program-, databas-, ikon-, fil- och hjälpservrar och deras klienter.

Konfigurera inloggnings- och sessionstjänster

En inloggnings- och sessionsserver är ett system som tillhandahåller skrivbordstjänster (inloggningshanteraren, sessionshanteraren, filhanteraren, fönsterhanteraren osv) på en skärm med X-servern.

En sessionsserver innehåller normalt tjänster för X-terminaler. En nätverkskonfiguration kan dock ställas in så att sessionstjänsterna koncentreras till en eller flera servrar som är tillgängliga för både X-terminaler och arbetsstationer.

Inloggningshanteraren är en skrivbordskomponent som svarar för inloggnings-tjänster på andra skärmar. När användaren har loggat in startas sessionshanteraren åt användaren.

Mer information om hur du konfigurerar inloggnings- och sessionsserverar samt X-terminaler finns i "Visa en inloggningsbild på en nätverksskärm" på sidan 6.

Konfigurera andra programrelaterade tjänster

I det här avsnittet beskrivs de gemensamma nätverkskraven för skrivbordssystemet:

- Programservrar
- Databasservrar
- Ikonservrar
- Hjälpservrar

Konfigurera skrivbordsklienter och -servrar

1. Gör den nätverkskonfiguration för operativsystemet som krävs av skrivbordet tillgänglig.

Se "Konfigurera operativsystemets basnätverkstjänster för skrivbordet" på sidan 106.

2. Installera skrivbordet eller minimiuppsättningen med filer:

Du måste installera:

- Hela uppsättningen körtidsfiler i Common Desktop Environment
- *eller* de här filuppsättningarna: CDE-MIN och CDE-TT

Obs! Installation och filuppsättningar kan variera mellan olika leverantörer.

3. Konfigurera systemet för bakgrundsprogrammet för filnamsservern ToolTalk_{rpc.ttdbserver}.

Detta ska ske automatiskt när skrivbordet installeras. Mer information finns i "Konfigurera databasservern ToolTalk " på sidan 114.

4. Installera och konfigurera styrbakgrundsprogram för underprocesser (*dtspcd*).

Detta ska ske automatiskt när skrivbordet installeras. Mer information finns i "Konfigurera styrbakgrundsprogram för underprocesser" på sidan 113.

5. Montera alla nödvändiga fjärrdata.

Data betraktas som "fjärrdata" om de finns i ett annat system än programmet som använder dessa data.

Till exempel:

- Om ett program använder data som finns på en filserver, måste programmet montera dessa filer.
- Om ikoner till filhanteraren finns på en ikonservr måste sessionsservern montera de filerna.
- Om ett nätverk använder en hjälpservr för hjälpfiler, måste sessionsservern och alla programservrar montera hjälpdata.

Mer information om monteringspunkter finns i nästa avsnitt, "Konfigurera monteringspunkter för fjärrfilsystem" på sidan 111.

Konfigurera monteringspunkter för fjärrfilsystem

När skrivbordet skickar filnamn från ett system till ett annat måste det överföra, eller *avbilda*, dessa filnamn till namn som målsystemet känner igen. Tilldelningen är nödvändig eftersom en fil kan monteras på olika platser i olika system, och därför måste gå att komma åt med olika namn. Filen `/projects/big` på `sysA` kanske kallas `/net/sysA/projects/big` på `sysB`.

Krav för tilldelning av filnamn

För att tilldelningen ska göras korrekt måste något av följande vara uppfyllt:

- Kommandot `mount` används för att montera filsystem statiskt. Dessa typer av statisk montering är normalt konfigurerade i filer som `/etc/checklist`, `/etc/mnttab` eller `/etc/filesystems`.

För att tilldelningen av filnamn ska fungera korrekt mellan olika system, måste monterade filsystem använda konsekventa värddatornamn. Om en värddator har flera namn (t ex `alias`, eller om värddatorn har fler än en LAN-adress som är känd under olika namn) måste du använda samma namn och format på namnet för alla monteringar.

- *Eller också* använder du funktionen för automatisk montering för att montera filsystem vid standardmonteringspunkten för `/net`.
- *Eller också* kan du använda funktionen för automatisk montering för att montera filsystem på andra platser än `/net`. Omgivningsvariabeln `DTMOUNTPOINT` anges för att ange monteringspunkten. Se nästa avsnitt, "Ange värde på `DTMOUNTPOINT`" på sidan 112.

Mer information om automontering finns i direkthjälpen (man) till `automount(1M)`.

Ange värde på DTMOUNTPOINT

Du måste ange omgivningsvariabeln DTMOUNTPOINT om båda nedanstående villkor är sanna:

- Automonteringen används för att montera filsystem.
- Och fjärrfilssystem monteras på andra platser än /net.

DTMOUNTPOINT måste anges för processer, inklusive:

- Användarens skrivbordsprocesser som startas automatiskt när användaren loggar in, t ex skrivbordshanteraren (dtwm) och filhanteraren (dtfile)
- Systemprocesser, t ex `rpc.ttdbserver` och `dtspcd`, som startas av mekanismer som `inetd`
- Program som startas av skrivbordet på lokala system eller fjärrsystem
- Program som startas av användaren från skalets kommandorad

Så här anger du DTMOUNTPOINT för de här processerna:

1. Redigera filen `/etc/inetd.conf`:

a. Leta reda på posten `dtspcd` och lägg till:

```
-mount_point monteringspunkt
```

b. Leta reda på posten `rpc.ttdbserver` och lägg till:

```
-m monteringspunkt
```

Om t ex funktionen för automatisk montering används med monteringspunkten `/nfs`, är posterna i `/etc/inetd.conf`:

```
dtspcd stream tcp nowait root /usr/dt/bin/dtspcd \  
/usr/dt/bin/dtspcd -mount_point /nfs  
rpc stream tcp wait root /usr/dt/bin/rpc.ttdbserver \  
100083 1 rpc.ttdbserver -m /nfs
```

2. Utför proceduren på det system som läser om `/etc/inetd.conf`. Mer information finns i direkthjälpen (`man`) till `inetd(1M)`.

3. Ange DTMOUNTPOINT på ett sådant sätt att dess värde ärvs av en användare som loggar in.

Detta kan utföras genom att man anger variabeln i `/etc/dt/config/Xsession.d`. Mer information om att ange omgivningsvariabler finns i "Ange omgivningsvariabler" på sidan 30.

Konfigurera styrbakgrundsprogram för underprocesser

Skrivbordets tjänst för underprocesskontroll (subprocess control, SPC) omfattar kommandoexekvering för klient.

Styrbakgrundsprogrammet för underprocesser (`dtspcd`) används av skrivbordssystemet för att starta fjärrprogram. Det är ett `inet`-bakgrundsprogram som tar emot förfrågningar om att exekvera kommandon från fjärrklienter. Mer information om hur du konfigurerar `inet`-bakgrundsprogram finns i direkthjälpen (man page) till `inetd.conf(1M)`.

Skrivbordets åtgärdsanropsbibliotek använder tjänsten SPC för att anropa fjärråtgärder.

Konfigurera `dtspcd`

- ♦ **Kontrollera att `dtspc` är korrekt registrerad i både `/etc/services` och `/etc/inetd.conf`.**

Se direkthjälpen (man) till `dtspcd(1M)`.

SPC-säkerhet

Verifieringen för tjänsten underprocesskontroll baseras på filsystemverifiering. Kommandot `dtspcd` måste ha tillgång till en *verifieringskatalog* som också är monterad till alla SPC-klientsystem.

Som standard är verifieringskatalogen för `dtspcd` användarens hemkatalog. Du kan dock konfigurera `dtspcd` så att en annan plats används genom att ange alternativet `-auth_dir` i katalogen `/etc/inetd.conf`. Mer information finns i direkthjälpen (man) till `dtspcd(1M)`.

Eftersom SPC-verifieringen baseras på filsystemverifiering, är tjänsten SPC bara lika säker som det distribuerade filsystemet. Om du använder skrivbordssystemet i ett nätverk där du inte litar på det distribuerade filsystemet, kan du inaktivera `dtspcd`. Du inaktiverar `dtspcd` genom att kommentera bort posten `dtspc` i `/etc/services`.

Konfigurera omgivningsvariabler för fjärrkörning

När skrivbordet använder en åtgärd för att starta ett program i ett fjärrsystem, kopieras användarens omgivningsvariabler till fjärrsystemet och placeras i programmets omgivning.

Normalt ändras några av omgivningsvariablerna innan de kopieras till fjärrsystemet. Du kan konfigurera både komponenten för åtgärdsanropet och

underprocesstyrningen för att bearbeta ytterligare omgivningsvariabler innan variablerna placeras i programmets omgivning.

Mer information om standardkonfigurationen och hur du ändrar den finns i direkthjälpen (man) för `dtactionfile(4)` och `dtspcdenv(4)`.

Konfigurera databasservern ToolTalk

En komponent i ToolTalk är ToolTalk-databasservern,
`/usr/dt/bin/rpc.ttdbserver`.

Databasservern ToolTalk används av meddelandetjänsten ToolTalk och till filnamnstilddelning. Den registreras normalt i `/etc/inetd.conf` när skrivbordssystemet installeras och behöver ingen ytterligare konfiguration.

Mer information om ToolTalk-databasservern och dess konfigurationsalternativ finns i direkthjälpen (man) för `rpc.ttdbserver(1M)`.

Konfigurera meddelandeservern ToolTalk

ToolTalk-meddelandeservern är `ttsession`. Som standard krävs ingen konfiguration. Den startas av kommandofilen `Xsession` vid inloggningen.

Mer information om ToolTalk-meddelandeservern och dess konfigurationsalternativ finns i direkthjälpen (man) för `ttsession`.

Konfigurera bakgrundsprogrammet för Kalender

En komponent i programmet Kalender är Kalender-bakgrundsprogrammet `rpc.cmsd`. Den registreras normalt i `/etc/inetd.conf` när skrivbordssystemet installeras och behöver ingen ytterligare konfiguration.

Mer information om Kalenderbakgrundsprogram och dess konfigurationsalternativ finns i direkthjälpen (man page) för `rpc.cmsd(1)`.

Administrera programtjänster

I det här avsnittet beskrivs vissa konfigurationskrav för:

- Programserverar och deras klienter
- Skrivbordsserverar som tillhandahåller särskilda tjänster – databasserverar, ikonservrar och hjälpservrar

Här beskrivs också nätverkskrav för två speciella konfigurationer av nätverksprogram:

- Värddatorer för fjärrkörning
- Program som kör över olika monterade filsystem

Omgivningsvariabler för sökvägar

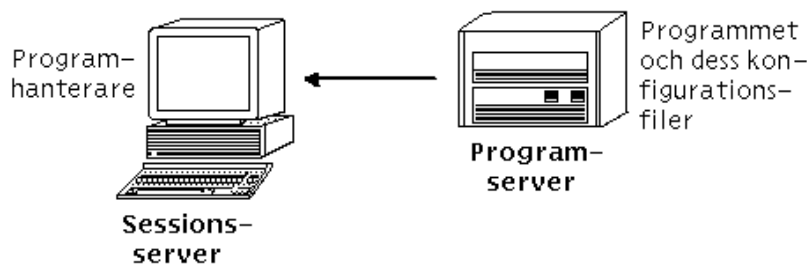
Skrivbordet använder en uppsättning omgivningsvariabler för att ange vilken sökväg som olika hitta skrivbordskonfigurationsfiler, t.ex åtgärdsdatabaser, datatyps-databaser, hjälpfiler och ikonfiler.

Mer information om hur omgivningsvariabler för sökvägar används finns i Kapitel 9 eller i direkthjälpavsnittet (man page) för dtenvvar(5).

Konfigurera en programserver och dess klienter

I standardkonfigurationen för programservern innehåller programservern alla binära filer och konfigurationsfiler som är associerade med programmet, inklusive:

- Programmets körbara filer
- Standardkonfigurationsfilerna för program, t ex programstandardfiler, meddelandekatalog och delade bibliotek.
- Skrivbordets konfigurationsfiler:
 - Filer med åtgärds- och datatypsdefinitioner
 - Filer med ikonbilder
 - Hjälpdatafiler för skrivbordet.



Figur 7-5 Standardkonfiguration för programservern

Konfigurera en programserver

1. **Gör den nätverkskonfiguration för operativsystemet som krävs av skrivbordet tillgänglig.**
Se "Konfigurera operativsystemets basnätverkstjänster för skrivbordet" på sidan 106.
2. **Gör den allmänna skrivbordskonfiguration som krävs för servrar tillgänglig.**
Se "Konfigurera skrivbordsklienter och -servrar" på sidan 110.
3. **Installera programmet eller programmen.**
4. **Om ett program inte registreras automatiskt måste du registrera det.**
Se Kapitel 5.

Konfigurera klienten till en programserver

1. **Gör den nätverkskonfiguration för operativsystemet som krävs av skrivbordet tillgänglig.**
Se "Konfigurera operativsystemets basnätverkstjänster för skrivbordet" på sidan 106.
2. **Gör den allmänna skrivbordskonfiguration som krävs för klienter tillgänglig.**
Se "Konfigurera skrivbordsklienter och -servrar" på sidan 110.

3. **Gör den allmänna skrivbordskonfiguration som krävs för klienter tillgänglig.**

För hela systemet – ange och exportera variabeln `DTSPSYSAPPHOSTS` i `/etc/dt/config/Xsession.d/0010.dtpaths`

Personlig – ange och exportera variabeln `DTSPUSERAPPHOSTS` i `Hemkatalog/.dtprofile`

Följande rad i `/etc/dt/config/Xsession.d/0010.dtpaths` lägger t ex till ett system med värddatornamnen `SysAAA` och `SysBBB` till programsökvägen:

```
export DTSPSYSAPPHOSTS=SysAAA: ,SysBBB:
```

Mer information om hur du anger programsökväg finns i:

- "Programsökväg" på sidan 132

- "Ange värde för en sökväg" på sidan 131

Konfigurera databas-, ikon- och hjälptjänster

Normalt installeras ett programs åtgärds- och datatypsdefinitioner, ikon- och hjälpdatafiler till samma system som programmet själv installeras till.

Detta kan vara en typisk konfiguration av hjälpdatafiler:

- Hjälpdatafilerna för filhanteraren finns normalt på sessionsservern. Skrivbordet kan hitta dem eftersom hjälpsökvägen automatiskt söker på rätt plats på sessionsservern.
- Hjälpfilerna till andra program finns normalt på samma programserver som programmet. Sessionsservern kan hitta dem eftersom en ändring av programsökvägen automatiskt innebär en ändring av hjälpsökvägen.

Ibland kanske du vill placera databas- (åtgärder och datatyper), hjälp- eller ikondata någon annanstans i nätverket. Om nätverket t ex använder flera sessionsservrar kan du skapa en hjälpserver där alla hjälpfiler för skrivbordsprogram (filhanteraren, inställningar och liknande) lagras. Detta sparar diskutrymme eftersom hjälpfilerna då inte behöver dupliceras på varje sessionsserver.

Skapa en databas-, hjälp- eller ikonserver

1. **Gör den nätverkskonfiguration för operativsystemet som krävs av skrivbordet tillgänglig.**

Se "Konfigurera operativsystemets basnätverkstjänster för skrivbordet" på sidan 106.

2. **Gör den allmänna skrivbordskonfiguration som krävs för klienter tillgänglig.**

Se "Konfigurera skrivbordsklienter och -servrar" på sidan 110.

3. **Installera databas-, hjälp- eller ikonfilerna.**

Filerna kan finnas var som helst i systemet. Det kan dock underlätta att använda följande adresser, eftersom detta är de kataloger som automatiskt söks när systemet har utnämnts till programserver.

- Databasfiler: `/etc/dt/appconfig/types/språk`
- Hjälpfiler: `/etc/dt/appconfig/help/språk`
- Ikonfiler: `/etc/dt/appconfig/icons/språk`

Om du installerar en databasserver måste åtgärderna skrivas så att de anger var kommandona (EXEC_STRING) ska köras. Se "Ange en värddator för fjärrkörning" på sidan 119.

Konfigurera sessionsservern att hitta en databas-, ikon- eller hjälpserver

1. Gör den nätverkskonfiguration för operativsystemet som krävs av skrivbordet tillgänglig.

Se "Konfigurera operativsystemets basnätverkstjänster för skrivbordet" på sidan 106.

2. Gör den allmänna skrivbordskonfiguration som krävs för klienter tillgänglig.

Se "Konfigurera skrivbordsklienter och -servrar" på sidan 110.

3. Lägg till databas-, ikon- eller hjälpdatabasen i lämplig sökväg.

- Om du har placerat datafilerna på de platser som anges i Steg 3 på sidan 117 i "Skapa en databas-, hjälp- eller ikonserver" på sidan 117, kan du ändra programsökvägen.
- Om du har placerat datafilerna någon annanstans måste du ändra den specifika sökvägen.

Om du t ex har placerat hjälpfilerna i katalogen `/etc/dt/help` i systemet SysCCC lägger du till följande rad i /

```
etc/dt/config/Xsession.d/0010.dtpaths:
```

```
export DTSPSYSHELP=/net/SysCCC/etc/dt/help
```

Mer information om hur du anger sökvägar finns i:

- "Databassökväg (åtgärd/datatyper)" på sidan 135
- "Ikonsökväg" på sidan 137
- "Hjälpsökväg" på sidan 139
- "Ange värde för en sökväg" på sidan 131

Speciella konfigurationer för nätverksprogram

I det här avsnittet beskrivs hur du konfigurerar system för att köra program:

- På en annan plats i systemet som innehåller åtgärden – på en värddator för fjärrkörning
- Lokalt via filsystemmonteringar

Ange en värddator för fjärrkörning

I den typiska programserverkonfigurationen finns åtgärdsdefinitionen i samma system som programmets körbara filer. Åtgärder kan dock skrivas för att köra kommandon i andra system. I den här konfigurationen kallas systemet som innehåller programmet för *exekveringsdator*.

Åtgärdsdefinitionen kan finnas på sessionsservern eller i ett system som tillhandahåller åtgärds- och datatypstjänster åt sessionsservern (kallas även en *databasserver* eller *databasvärd*).

Åtgärdsdefinitioner använder fältet EXEC_HOST för att ange var deras kommandon (EXEC_STRING) ska köras. Följande åtgärdsdefinition anger t ex att en xload-klient ska köras i systemet med värddatornamnet SysDDD:

```
ACTION XloadSysDDD
{
  TYPE      COMMAND
  EXEC_HOST SysDDD
  EXEC_STRING /usr/bin/X11/xload -label SysDDD
}
```

Om fältet EXEC_HOST anger mer än ett värddatornamn, försöker skrivbordssystemet köra EXEC_STRING på alla värddatorer i tur och ordning tills det hittar en sträng som kan köra åtgärden. Följande EXEC_HOST-fält anger t ex att åtgärden först ska försöka köra EXEC_STRING på SysDDD, och om detta inte går ska den försöka i SysEEE.

```
EXEC_HOST SysDDD,SYSEEE
```

Om fältet EXEC_HOST inte anges för en åtgärd får det standardvärdet %DatabaseHost%. Värdet på %DatabaseHost% hämtas från databassökvägen.

Anta t ex att databasens sökväg har ändrats genom att följande rad har lagts till i /etc/dt/config/Xsession.d/0010.dtpaths:

```
DTSPSYSDATABASEHOSTS=SysAAA: /net/SysBBB/etc/dt/appconfig/types/C
```

SysAAA anges med hjälp av den värddatorkvalificerade syntaxen - SysAAA:. En åtgärdsdefinition som hittas med det här elementet i sökvägen anger databasvärden till SysAAA. En åtgärd som hittas med delen /net/SysBBB... av sökvägen anger databasvärden till det lokala systemet eftersom syntaxen inte innehåller värddatorkvalificeraren.

Konfigurera värddatorn för fjärrkörning

1. Gör den nätverkskonfiguration för operativsystemet som krävs av skrivbordet tillgänglig.

Se "Konfigurera operativsystemets basnätverkstjänster för skrivbordet" på sidan 106.

2. **Gör den allmänna skrivbordskonfiguration som krävs för servrar tillgänglig.**
Se "Konfigurera skrivbordsklienter och -servrar" på sidan 110.
3. **Kontrollera att programmen är rätt installerade och konfigurerade för lokal exekvering.**

Konfigurera systemet som innehåller åtgärdsdefinitionen

1. **Gör den nätverkskonfiguration för operativsystemet som krävs av skrivbordet tillgänglig.**
Se "Konfigurera operativsystemets basnätverkstjänster för skrivbordet" på sidan 106.
2. **Gör den allmänna skrivbordskonfiguration som krävs för servrar tillgänglig.**
Se "Konfigurera skrivbordsklienter och -servrar" på sidan 110.
3. **Skapa och installera åtgärdsdefinitioner och programgrupper.**
Se "Skapa åtgärder som kör program i fjärrsystem" på sidan 190 och "Skapa och hantera allmänna programgrupper" på sidan 46.

Konfigurera sessionsservern

1. **Gör den nätverkskonfiguration för operativsystemet som krävs av skrivbordet tillgänglig.**
Se "Konfigurera operativsystemets basnätverkstjänster för skrivbordet" på sidan 106.
2. **Gör den allmänna skrivbordskonfiguration som krävs för klienter tillgänglig.**
Se "Konfigurera skrivbordsklienter och -servrar" på sidan 110.
3. **Ändra åtgärdssökvägen till att omfatta databasvärden.**
Se "Databassökväg (åtgärd/datatyper)" på sidan 135.
4. **Ändra programsökvägen till att omfatta värddatorm för exekvering.**
Se "Programsökväg" på sidan 132.

Köra program lokalt

Standardkonfigurationen för programservrar kör program på programservern. Ibland kan det dock vara bättre att programmet är installerat i ett fjärrsystem men körs lokalt på sessionsservern.

Figur 7-6 Exekvering via monteringspunkter

Konfigurera programservern

Ingen speciell konfiguration behövs.

Konfigurera sessionsservern

- ♦ **Ändra programsökvägen. Använd den lokala absoluta sökvägen till programmet.**

Du kan t ex använda följande variabeldefinition för att söka efter ett program som har registrerats i `sysAAA`:

```
DTSPSYSAPPHOSTS=/net/SysAAA/etc/dt/appconfig/appmanager/C
```

Sessionsservern måste ha tillgång till programmets konfigurationsfiler, t ex programstandardfiler, meddelandekataloger och delade bibliotek.

Konfigurera och administrera utskrifter från skrivbordet

En skrivbordsanvändare kan skriva ut filer på flera olika sätt. De olika förfaringssätten kan delas in i två huvudkategorier: utskrifter från skrivbordet och från ett program.

Du kan skriva ut från skrivbordet genom att:

- Markera en fil i filhanteraren och välja Skriv ut från Markera-menyn eller ikonens meny
 - Dra en fil från filhanteraren till huvudgruppens skrivarikon eller panelundermenyn Egen skrivare
 - Dra en fil från filhanteraren till en skrivare i utskriftshanterarens huvudfönster
- Om du vill skriva ut från ett program använder du kommandot Skriv ut, som normalt finns på en meny eller nås via en ikon i programfönstret.
- "Lägga till och ta bort skrivare" på sidan 123
 - "Ikonbilder till skrivaren" på sidan 125
 - "Konfigurera standardskrivare" på sidan 126
 - "Vad som sker vid utskrift" på sidan 127

Lägga till och ta bort skrivare

I det här avsnittet beskrivs hur du lägger till och tar bort skrivare från skrivbordet.

▼ Lägga till en skrivare till skrivbordet

1. Lagg till en skrivare till systemets konfiguration.

Följ instruktionerna i dokumentationen för systemadministratörer för operativsystemet.

2. Kör kommandot:

```
env LANG=språk /usr/dt/bin/dtprintinfo -populate
```

3. Starta om utskriftshanteraren genom att dubbelklicka på Ladda om åtgärder i programgruppen Skrivbordsverktyg i programhanteraren. Kontrollera att skrivaren visas.

4. Skicka ett meddelande till användarna att även de ska starta om utskriftshanteraren och köra Ladda om åtgärder.

Varje gång utskriftshanteraren anropas läser den systemets skrivarkonfigurationslista. Om en ny skrivare upptäcks skapas automatiskt en ny skrivaråtgärd och ikon för skrivaren till skrivbordet. Du behöver inte göra något mer för att skrivaren ska visas på skrivbordet.

▼ Ta bort en skrivare från skrivbordet

1. Ta bort skrivaren från systemkonfigurationen.

Följ instruktionerna i dokumentationen för systemadministratörer för operativsystemet.

2. Starta om utskriftshanteraren genom att dubbelklicka på Ladda om åtgärder i programgruppen Skrivbordsverktyg i programhanteraren. Kontrollera att skrivaren är borta.

3. Skicka ett meddelande till användarna att även de ska starta om utskriftshanteraren och köra Ladda om åtgärder.

Varje gång utskriftshanteraren anropas läser den systemets skrivarkonfigurationslista. Om den upptäcker att en skrivare har tagits bort från listan, tas skrivarens åtgärd och ikon automatiskt bort från utskriftshanteraren och filhanteraren. Du behöver inte göra något mer för att ta bort skrivaren från skrivbordet.

Obs! Utskriftshanteraren kan inte ta bort skrivare från huvudgruppen. Därför ska du skicka ett meddelande till alla användare när du tar bort en skrivare, så att de tar bort alla ikoner för den skrivaren på huvudgruppen.

Ändra intervall för jobbuppdatering

Om du vill ändra hur ofta informationen i utskriftshanteraren ska uppdateras, ändrar du intervallet för jobbuppdatering. Normalt begär utskriftshanteraren information om utskriftsjobb från skrivaren var 30:e sekund. Du ändrar detta med skjutreglaget Uppdatera intervall i dialogrutan Välj alternativ (visas när du väljer Välj alternativ på Visa-menyn).

Ikonbilder till skrivaren

När du lägger till en skrivare tilldelas den automatiskt standardskrivarikonen. Om du vill göra en annan ikon tillgänglig placerar du ikonfilerna i `/etc/dt/appconfig/icons/språk` eller i någon annan katalog i ikonsökvägen. Användarna kan sedan markera den här ikonen för att ersätta standardikonen.

Du måste skapa en komplett uppsättning (mycket liten, normal och stor) ikoner, annars visas de inte i ikonväljaren i utskriftshanteraren.

Mer information om ikonsökvägen finns i "Ikonsökväg" på sidan 137.

Ikonfilnamn och ikonfilstorlekar

Ikonernas filnamn ska se ut så här:

basnamn.storlek.typ

där:

storlek – l (stor), m (normal), t (pytteliten). Mer information om ikonstorlekar finns i "Storlekskonventioner" på sidan 214.

typ – pm (färgbildpunktsbeskrivning), bm (bitmapp).

Ikonfilnamnet på en medelstor respektive en mycket liten ikon för en färgskrivare blir då `ColorPrinter.m.pm` och `ColorPrinter.t.pm`.

Mer information om hur du skapar ikoner finns i Kapitel 14.

▼ Ändra ikon, skrivarnamn och beskrivning globalt

Du ska ändra de globala skrivaregenskaperna så fort du har lagt till skrivaren, innan användarna har ändrat dem med utskrifthanteraren. När användarna väl har ändrat skrivaregenskaperna med utskriftshanteraren visas inte de ändringar som du gör.

Redigera filen `/etc/dt/appconfig/types/språk/utskriftskö.dt` med önskad information om ikon, skrivarnamn och beskrivning:

1. I fältet `ICON` uppdaterar du *basnamn* till den nya ikonens basnamn.
2. I fältet `LABEL` uppdaterar du *etikettnamn* till det nya skrivarnamnet.
3. Uppdatera texten i fältet `DESCRIPTION`.

Här kan du skriva var skrivaren finns, typen av skrivare och skrivarkontakter.
Om du vill skriva mer än en rad, placerar du ett \ i slutet av raden. Till exempel:

```
DESCRIPTION    Det här är en PostScript-skrivare i byggnad 1, \
                rum 123. Ring 555-5555 om du har problem.
```

Konfigurera standardskrivare

Standardskrivaren används när användaren:

- Släpper ett objekt på skrivarikonen på huvudgruppen
- Markerar ett objekt i filhanteraren sedan väljer Skriv ut från vald meny eller från ikonens objektmeny
- Skrifer ut från program som använder standardskrivare

▼ Ändra destination för standardutskrift

Så här ändrar du standardskrivare för alla användare:

1. **Öppna filen** `/etc/dt/config/Xsession.d/0010.dtpaths`.
Om `/etc/dt/config/Xsession.d/0010.dtpaths` inte finns kopierar du den från `/usr/dt/config/Xsession.d/0010.dtpaths`.
2. I fältet `LPDEST=skrivare` uppdaterar du *skrivare* till den nya destinationen för standardutskrift.
Om raden inte finns lägger du till raden `LPDEST=skrivare`, där *skrivare* är namnet på skrivaren som ska vara standardskrivare.
3. **Användarna måste logga ut och in igen.**

Så här ändrar en enstaka användare standardskrivare:

- ◆ **Kopiera en annan skrivare till huvudgruppen från panelundermenyn Egen skrivare.**

Om du vill ändra standardskrivaren för en enskild användare måste du:

1. Gå till hemmappen och öppna filen `.dtprofile`.
2. Lägg till eller redigera en rad som anger värden för omgivningsvariabeln LPDEST:

```
LPDEST=skrivarenhet; export LPDEST
```

Om du använder `bash` är syntaxen:

```
setenv LPDEST skrivarenhet
```

Följande rad ändrar t ex standardskrivaren till den skrivare vars enhetsnamn är `laser3d`.

```
LPDEST=laser3d; export LPDEST
```

Om du använder `csh` är syntaxen:

```
setenv LPDEST laser3d
```

Vad som sker vid utskrift

När en utskriftsbegäran initieras av att en fil släpps på skrivarikonen, utför systemet följande:

1. Systemet söker i datatyps-databasen efter det släppta objektets definition.
2. Om det finns en unik skrivare-åtgärd för datatypen (anges med fältet `ARG_TYPE` i utskrifts-åtgärden) används den, annars används standardutskrifts-åtgärden (`dtlp`). Om t ex filen är en PostScript[®]-fil, använder systemet Print-åtgärden för PostScript-filer. Den här åtgärden definieras i `/usr/dt/appconfig/types/språk/dt.dt`. Om du använde verktyget Skapa åtgärd för den här datatypen, används det utskriftskommando du angav som utskrifts-åtgärd för att skriva ut filer av den här typen.
3. Filen levereras till skrivaren med det vanliga UNIX-undersystemet `lp` för utskrift.

Skrivbordets sökvägar

Skrivbordet använder sökvägar för att hitta program och tillhörande skrivbordsfiler.

- "Skrivbordets sökvägar och deras omgivningsvariabler" på sidan 130
- "Ange värde för en sökväg" på sidan 131
- "Programsökväg" på sidan 132
- "Databassökväg (åtgärd/datatyper)" på sidan 135
- "Ikonsökväg" på sidan 137
- "Hjälpökväg" på sidan 139
- "Lokaliserade sökvägar" på sidan 141

Skrivbordet tillhandahåller fyra sökvägar, som beskrivs i Tabell 9-1.

TABELL 9-1 Skrivbordets sökvägar

Sökvägar	Beskrivning
Program	Används för att söka efter program. Programhanteraren använder programsökvägen för att dynamiskt fylla den översta nivån när användaren loggar in.
Databas	Används för att ange ytterligare adresser för åtgärds- och datatypsdefinitionsfiler (*.fd) samt huvudgruppsfiler (*.dt-filer).
ikoner	Används för att ange ytterligare adresser för ikoner.
Hjälpdata	Används för att ange ytterligare adresser för hjälpdata.

Sökvägen kan innehålla både lokala kataloger och fjärrkataloger. När en sökväg omfattar fjärrkataloger måste du konfigurera fjärrfilsåtkomst för dessa adresser. Ett system hittar t ex program på en programserver eftersom programservern är listad i programmets sökväg.

När en sökväg innehåller en fjärradress måste du konfigurera fjärrfilsåtkomst för adressen. Mer information finns i "Konfigurera distribuerad filsystemåtkomst" på sidan 107.

Skrivbordets sökvägar och deras omgivningsvariabler

Skrivbordets sökvägar skapas vid inloggningen av skrivbordsverktyget `dtsearchpath`. Verktyget `dtsearchpath` använder en kombination av omgivningsvariabler och inbyggda adresser för att skapa sökvägarna.

Omgivningsvariabler som `dtsearchpath` kallas för *indatavariabler*. Dessa anges av systemadministratören eller slutanvändaren. Indatavariabler använder namngivningskonventionen `DTSP*`.

När `dtsearchpath` körs vid inloggningen sätts värdena som tilldelas till dessa variabler samman, inbyggda adresser läggs till och värden för *utdatavariabler* skapas. Det finns en utdatavariabel för varje sökväg.

TABELL 9-2 Omgivningsvariabler för skrivbordets sökvägar

Sökväg för:	Omgivningsvariabel för utdata	Systemtäckande indatavariabel	Egen indatavariabel
Program	<code>DTAPPSEARCHPATH</code>	<code>DTSPSYSAPPHOSTS</code>	<code>DTSPUSERAPPHOSTS</code>
Databas ¹	<code>DTDATABASESEARCHPATH</code>	<code>DTSPSYSDATABASEHOSTS</code>	<code>DTSPUSERDATABASEHOSTS</code>
ikoner	<code>XMICONSEARCHPATH</code> , <code>XMICONBMSEARCHPATH</code>	<code>DTSPSYSICON</code>	<code>DTSPUSERICON</code>
Hjälpdata	<code>DTHELPSEARCHPATH</code>	<code>DTSPSYSHELP</code>	<code>DTSPUSERHELP</code>

1. Åtgärder, datatyper, och huvudgruppsdefinitioner

Komponenter använder utdatavariablernas värden. Programhanteraren använder t ex värdet för programmets sökväg (DTAPPSEARCHPATH) för att hitta programgrupper.

Ange värde för en sökväg

Du kan ändra systemtäckande och egna sökvägar. Ändringarna görs genom att värden anges för systemtäckande eller egna indatavariabler. Alla ändringar du gör läggs till de inbyggda sökvägsadresserna.

▼ Visa aktuellt värde för en sökväg (utdatavariabel)

- ◆ Använd kommandot `dtsearchpath` för att visa aktuella värden för sökvägarna:

- Om du vill ta reda på värdet för den aktuella (inloggade) användaren skriver du kommandot:

```
dtsearchpath -v
```

- Om du vill ta reda på värdet för en annan användare skriver du kommandot:

```
dtsearchpath -u användare
```

Sökvägens värden omfattar dessa variabler:

%H – används i DTHELPSEARCHPATH. Hjälpvolymens namn.

%B – används i XMICONSEARCHPATH. Basnamnet på en ikonfil.

%M – ikonfilens storlek (.l, .m, .s, .t) används i XMICONSEARCHPATH.

%L – värdet på omgivningsvariabeln LANG.

▼ Göra egna ändringar av sökvägen

1. Öppna *Hemkatalog*/`.dtprofile` för redigering.
2. Lägga till eller redigera en rad som definierar och exporterar den egna indatavariabeln.

Följande rad lägger t ex till en adress i användarens egen programsökväg:

```
export DTSPUSERAPPHOSTS=/projects1/editors
```

3. Logga ut och in igen för att aktivera ändringen.

▼ Göra systemtäckande ändringar av sökvägen

1. Logga in som root.
2. Om inte filen `etc/dt/config/Xsession.d/0010.dtpaths` redan finns skapar du den genom att kopiera `/usr/dt/config/Xsession.d/0010.dtpaths`.
3. Öppna `etc/dt/config/Xsession.d/0010.paths` för redigering. Lägg till eller redigera en rad som definierar och exporterar den systemtäckande indatavariabeln.
Följande rad lägger t ex till en adress i den systemtäckande hjälpsökvägen:

```
export DTSPYSHELP=/applications/helpdata
```
4. Informera alla användare i systemet att de måste logga ut och in igen för att ändringarna ska träda i kraft.

Programsökväg

Programmets sökväg är den primära sökväg som används av skrivbordet för att hitta program i det lokala systemet och på programserverar i hela nätverket.

När adresser läggs till i programmets sökväg uppdateras andra sökvägar (databas, ikon och hjälp) automatiskt så att de visar adresserna för dessa data. Programmets sökväg erbjuder alltså relativt enkel administration för program och deras skrivbordskonfigurationsfiler. Se "Hur programsökvägen påverkar sökvägarna för databas, ikon och hjälp" på sidan 134.

Standardsökväg för program

Standardsökvägen för program innehåller egna, systemtäckande och inbyggda adresser. *Standardspråk* är C.

Egen adress – *Hemkatalog*/.dt/appmanager

Systemtäckande adress – /etc/dt/appconfig/appmanager/*språk*

Inbyggd adress – /usr/dt/appconfig/appmanager/*språk*

Omgivningsvariabler för programsökvägar

Programsökvägen sätts samman utifrån inbyggda adresser och följande indatavariabler:

DTSPSYSAPPHOSTS – indatavariabel för systemtäckande programsökväg

DTSPUSERAPPHOSTS – indatavariabel för egen programsökväg

Den sammanlagda sökvägen anges av utdatavariabeln DTAPPSEARCHPATH.

Syntax för indatavariabler för programsökväg

Syntaxen för variablerna DTSPSYSAPPHOSTS och DTSPUSERAPPHOSTS är

VARIABLE=*adress* [, *adress* . . .]

där *adress* kan ha syntaxen:

/sökväg – anger en katalog i det lokala (sessionsserver-) systemet. Använd den här syntaxen för att lägga till en lokal katalog.

värdsnamm: – anger den systemtäckande katalogen

/etc/dt/appconfig/appmanager/språk i systemet *värdsnamm*. Använd den här syntaxen för att lägga till en programserver.

värdsnamm/*sökväg* – anger en katalog i fjärrsystemet *värdsnamm*.

lokalvärd: – den lokala systemtäckande adressen. Det här nyckelordet används för att ändra prioritetsordning på den lokala systemtäckande adressen. Se "Ändra prioritetsordning på den lokala systemtäckande adressen" på sidan 134.

Hur programsökvägens värde sätts samman

Värdet på programsökvägen (DTAPPSEARCHPATH) skapas genom att följande adresser sätts samman i prioritetsordning:

- Adresser som anges med variabeln DTSPUSERAPPHOSTS
- Egen standardadress: *Hemkatalog*/.dt/appmanager
- Standardadress: /etc/dt/appconfig/appmanager/ *språk*
- Platser som anges med variabeln DTSPSYSAPPHOSTS

■ /usr/dt/appconfig/appmanager/*språk*

Syntaxen:

värdsnamm:

utökas så att den anger katalogen /etc/dt/appconfig/appmanager i systemet *värdsnamm*.

Ändra prioritetsordning på den lokala systemtäckande adressen

Vanligtvis har den lokala systemtäckande adressen /etc/dt/appconfig/appmanager/*språk*) företräde framför fjärradresser. Lokala programgrupper har därför prioritet över fjärrgrupper med samma namn. Om både det lokala systemet och fjärrsystemet innehåller programgruppen Skrivare (/etc/dt/appconfig/appmanager/*språk*/Printers), används den lokala gruppen.

Programsökvägens indatavariabler tillhandahåller syntax för att ange prioritetsordning för de lokala systemtäckande programgrupperna:

localhost:

Anta att systemet måste ha tillgång till programserverna SysA, SysB och SysC och du vill att de systemtäckande programgrupperna i SysB ska ha företräde framför alla lokala grupper med samma namn.

Följande värde för DTSPSYSAPPHOSTS ger det här beteendet:

```
DTSPSYSAPPHOSTS=SysB:,localhost:,SysA:,SysC:
```

Hur programsökvägen påverkar sökvägarna för databas, ikon och hjälp

När ett tillägg görs i programsökvägen läggs motsvarande adresser automatiskt till även för databas-, ikon- och hjälpsökvägen. På det här sättet kan du lägga till en programserver till en sökväg genom att bara ange indatavariabeln i programsökvägen.

Om du t ex anger DTSPSYSAPPHOSTS enligt följande:

```
export DTSPSYSAPPHOSTS=servera:
```

så påverkas sökvägarna enligt beskrivningen i Tabell 9-3.

TABELL 9-3 Påverkade sökvägar

Sökvägar	Katalog som lagts till sökväg
Program	servera:/etc/dt/appconfig/appmanager/ <i>språk</i>
Databas	servera:/etc/dt/appconfig/types/ <i>språk</i>
Ikon	servera:/etc/dt/appconfig/icons/ <i>språk</i>
Hjälp	servera:/etc/dt/appconfig/help/ <i>språk</i>

Och om du anger DTSPSYSAPPHOSTS enligt följande:

```
export DTSPSYSAPPHOSTS=/projects1/apps
```

så påverkas sökvägarna enligt beskrivningen i Tabell 9-4 .

TABELL 9-4 Påverkade sökvägar

Sökvägar	Katalog som lagts till sökväg
Program	/projects1/apps/appmanager/ <i>språk</i>
Databas	/projects1/apps/types/ <i>språk</i>
Ikon	/projects1/apps/icons/ <i>språk</i>
Hjälp	/projects1/apps/help/ <i>språk</i>

Databassökväg (åtgärd/datatyper)

Databassökvägen hjälper skrivbordet att söka i vissa adresser efter filer som innehåller:

- Åtgärds- och datatypsdefinitioner (*.dt-filer)
- Huvudgruppsdefinitioner (*.fp-filer).

Du kanske måste ändra databassökvägen när du skapar en databasserver, eller när du lägger till en lokal adress för databasfiler.

Databasens standardsökväg

Standardsökvägen för databasen omfattar egna, systemtäckande och inbyggda adresser. *Standardspråk* är C.

Egen plats – *Hemkatalog*/.dt/types

Systemtäckande adress – /etc/dt/appconfig/types/*språk*

Inbyggd adress – /usr/dt/appconfig/types/*språk*

Hur programsökvägen påverkar databassökvägen

När en adress läggs till i programsökvägen läggs lämplig underkatalog för databasen till i databassökvägen (se "Hur programsökvägen påverkar sökvägarna för databas, ikon och hjälp" på sidan 134).

Om t ex programservern *hosta*: läggs till i programsökvägen, läggs katalogen *hosta*: /etc/dt/appconfig/types/*språk* automatiskt till i databassökvägen.

Databassökvägens omgivningsvariabler

Ikonsökvägen sätts samman utifrån inbyggda adresser och följande indatavariabler:

DTSPSYSDATABASEHOSTS – indatavariabel för sökväg till systemtäckande databas

DTSPUSERDATABASEHOSTS – indatavariabel för sökväg till egen databas

Följande indatavariabler anger adresser utanför programsökvägen.

Den sammansatta databassökvägen anges av utdatavariabeln
DTDATABASESEARCHPATH.

Syntax för databassökvägens indatavariabler

Syntaxen för variablerna DTSPSYSDATABASEHOSTS och DTSPUSERDATABASEHOSTS är

VARIABLE=*adress* [, *adress* . . .]

där *adress* kan ha syntaxen:

/sökväg – anger en katalog i det lokala (sessionsserver-) systemet. Använd den här syntaxen för att lägga till en lokal katalog.

värdsnamm: – anger den systemtäckande katalogen
/etc/dt/appconfig/types/språk i systemet *värdsnamm*.

värdsnamm: */sökväg* – anger en katalog i det fjärrsystemet *värdsnamm*.

Hur databassökvägen sätts samman

Värdet för databassökvägen (DTDATABASESEARCHPATH) skapas genom att följande adresser sätts samman i prioritetsordning:

- Platser som anges med variabeln DTSPUSERDATABASEHOSTS
- Platser som fås från variabeln DTSPUSERAPPHOSTS
- Egen standardadress: *Hemkatalog/.dt/types*
- Standardadress: */etc/dt/appconfig/types/språk*
- Platser som anges med variabeln DTSPSYSDATABASEHOSTS
- Platser som fås från variabeln DTSPSYSAPPHOSTS
- */usr/dt/appconfig/types/språk*

Syntaxen:

värdsnamm:

utökas så att den anger katalogen */etc/dt/appconfig/types* i systemet *värdsnamm*.

Ikonsökväg

Ikonsökvägen hjälper skrivbordet att söka i angivna adresser efter filer som innehåller bitmapp- och bildpunktsmönsterfiler.

Standardikonsökväg

Standardikonsökvägen innehåller egna, systemtäckande och inbyggda adresser. *Standardspråk* är C.

Egen plats – *Hemkatalog/.dt/icons*

Systemtäckande adress – */etc/dt/appconfig/icons/språk*

Inbyggd adress - /usr/dt/appconfig/icons/*språk*

Hur programsökvägen påverkar ikonsökvägen

När en adress läggs till i programsökvägen läggs lämplig underkatalog för ikonerna till i ikonsökvägen (se "Hur programsökvägen påverkar sökvägarna för databas, ikon och hjälp" på sidan 134).

Om t ex programservern *hosta*: läggs till i programsökvägen, läggs katalogen *hosta:/etc/dt/appconfig/icons/språk* automatiskt till i ikonsökvägen.

Ikonsökvägens omgivningsvariabler

Ikonsökvägen sätts samman utifrån inbyggda adresser och följande indatavariabler:

DTSPSYSICON - indatavariabel för sökväg till systemtäckande ikon

DTSPUSERICON - indatavariabel för egen ikonsökväg

Följande indatavariabler anger adresser utanför programsökvägen.

Den sammansatta ikonsökvägen anges av två utdatavariabler:

XMICONSEARCHPATH - används av färgbildskärmar

XMICONBMSEARCHPATH - används av monokroma bildskärmar

Syntax för ikonsökvägens indatavariabler

Syntaxen för variablerna DTSPSYSICON och DTSPUSERICON är

VARIABEL=adress [,adress . . .]

där *adress* kan ha syntaxen:

/sökväg - anger en katalog i det lokala (sessionsserver-) systemet. Använd den här syntaxen för att lägga till en lokal katalog.

Om du vill ange en adress i ett annat system använder du nätverksfilnamnet, exempelvis */nfs/servera/projects/icons*.

Hur ikonsökvägen sätts samman

Värdet för ikonsökvägen (XMICONSEARCHPATH och XMICONBMSEARCHPATH) skapas genom sammansättning av följande adresser, listade i prioritetsordning:

- Platser som anges med variabeln DTSPUSERICON
- Platser som fås från variabeln DTSPUSERAPPHOSTS
- Egen standardadress: *Hemkatalog*/.dt/icons
- Standardadress: /etc/dt/appconfig/icons/*språk*
- Platser som anges med variabeln DTSPSYSICON
- Platser som fås från variabeln DTSPSYSAPPHOSTS
- /usr/dt/appconfig/icons/*språk*

Sökvägarna för färgskärm och monokrom skärm skiljer sig bara åt genom prioriteten som tilldelas bildpunktsbeskrivningar och bitmappar. Variabeln XMICONSEARCHPATH listar bildpunktsbeskrivningar före bitmappar. XMICONBMSEARCHPATH listar bitmappar före bildpunktsbeskrivningar.

Hjälpsökväg

Hjälpsökvägen hjälper skrivbordet att söka i angivna adresser efter filer som innehåller hjälpinformation som ska registreras i systemet.

Standardhjälpsökväg

Standardhjälpsökvägen innehåller egna, systemtäckande och inbyggda adresser. *Standardspråk* är C.

Egen plats - *Hemkatalog*/.dt/help

Systemtäckande adress - /etc/dt/appconfig/help/*språk*

Inbyggd adress - /usr/dt/appconfig/help/*språk*

Hur programsökvägen påverkar hjälpsökvägen

När en adress läggs till i programsökvägen läggs lämplig underkatalog för hjälpen till i hjälpsökvägen (se "Hur programsökvägen påverkar sökvägarna för databas, ikon och hjälp" på sidan 134).

Om t ex programservern *hosta*: läggs till i programsökvägen, läggs katalogen *hosta*:/etc/dt/appconfig/help/*språk* automatiskt till i hjälpsökvägen.

Hjälpsökvägens omgivningsvariabler

Hjälpsökvägen sätts samman utifrån inbyggda adresser och följande indatavariabler:

DTSPSYSHELP – indatavariabel för sökväg till systemtäckande hjälp

DTSPUSERHELP – indatavariabel för sökväg till egen hjälp

Följande indatavariabler anger adresser utanför programsökvägen.

Den sammansatta databassökvägen anges av utdatavariabeln DTHELPPSEARCHPATH.

Syntax för hjälpsökvägens indatavariabler

Syntaxen för variablerna DTSPSYSHELP och DTSPUSERHELP är

VARIABLE=adress [, adress . . .]

där *adress* kan ha syntaxen:

/sökväg – anger en katalog i det lokala (sessionsserver-) systemet. Använd den här syntaxen för att lägga till en lokal katalog.

Om du vill ange en adress i ett annat system använder du nätverksfilnamnet, exempelvis */nfs/servera/projects/help*.

Hur ikonsökvägen sätts samman

Värdet för hjälpsökvägen (DTHELPPSEARCHPATH) skapas genom sammansättning av följande adresser, listade i prioritetsordning:

- Platser som anges med variabeln DTSPUSERHELP
- Platser som fås från variabeln DTSPUSERAPPHOSTS
- Egen standardadress: *Hemkatalog/.dt/help*
- Standardadress: */etc/dt/appconfig/help/språk*
- Platser som anges med variabeln DTSPSYSHELP
- Platser som fås från variabeln DTSPSYSAPPHOSTS
- */usr/dt/appconfig/help/språk*

Lokaliserade sökvägar

Utdatavariablerna innehåller poster för både lokaliserade adresser och standardadresser (C).

Standardprogramsökvägen är t ex:

```
Hemkatalog/ .dt/appmanager  
/etc/dt/appconfig/appmanager/språk  
/etc/dt/appconfig/appmanager/C  
/usr/dt/appconfig/appmanager/språk  
/usr/dt/appconfig/appmanager/C
```

där *språk* ersätts med värdet i omgivningsvariabeln LANG.

För varje definitionsområde (systemtäckande och inbyggda) har den språkspecifika adressen prioritet över standardadressen.

Introduktion till åtgärder och datatyper

Åtgärder och *datatyper* är kraftfulla komponenter som används för att integrera program i skrivbordssystemet. Med hjälp av dessa kan du skapa ett användargränssnitt för att starta program och ändra deras datafiler.

- "Introduktion till åtgärder" på sidan 144
- "Introduktion till datatyper" på sidan 149

Det här kapitlet introducerar begreppen åtgärder och datatyper. Det handlar om:

- Varför du kan behöva skapa åtgärder och datatyper för program.
- Hur åtgärder och datatyper förhåller sig till varandra.
- Hur åtgärder och datatyper förhåller sig till utskrifter från skrivbordet.

Procedurerna och reglerna för att skapa åtgärder och datatyper beskrivs i tre kapitel i handboken.

- Kapitel 11 beskriver hur du kan skapa åtgärder och datatyper genom att använda skrivbordsverktyget Skapa åtgärd.

Du använder Skapa åtgärd för att skapa åtgärder och datatyper till de flesta program, utan att du behöver lära dig syntaxreglerna för definitionerna.

- Kapitel 12 och Kapitel 13 beskriver hur du skapar åtgärder och datatyper manuellt genom att skapa och redigera konfigurationsfiler.

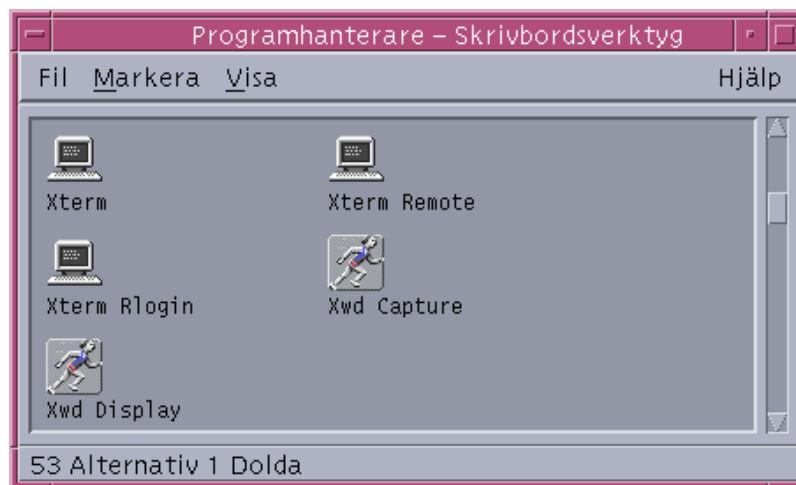
När du vill använda andra åtgärder som inte stöds av Skapa åtgärd måste du skapa dessa åtgärder och datatyper manuellt.

Introduktion till åtgärder

Åtgärder är instruktioner som automatiserar skrivbordsuppgifter, t ex att köra program och öppna datafiler. Åtgärder fungerar på ungefär samma sätt som makron och programmeringsfunktioner. Varje åtgärd har ett namn som används för att köra åtgärden.

När du definierar en åtgärd kan den användas för att anpassa användargränssnittet så att olika aktiviteter blir lättare att utföra. Skrivbordet kan ansluta nya komponenter för användargränssnittet, t ex ikoner, kontroller för huvudgruppen och menyalternativ för åtgärder.

Programgruppen Skrivbordsverktyg i programhanteraren innehåller t ex ikoner som startar olika hjälpmedel.



Figur 10-1 Åtgärdsikoner i programgruppen Skrivbordsverktyg

När du dubbelklickar på en ikon kommer en åtgärd att köras. Här är t ex en del av definitionen till åtgärden som körs när användaren dubbelklickar på ikonen för Xwd Display. Åtgärden definieras i konfigurationsfilen

`/usr/dt/appconfig/types/språk/xclients.dt`.

```
ACTION Xwd
{
  LABEL      Xwd Display
  TYPE       COMMAND
  EXEC_STRING /usr/bin/X11/xwd -noclick -in \
              %(File)Arg_1"Xwd-fil som ska visas:%"
  ...
}
```

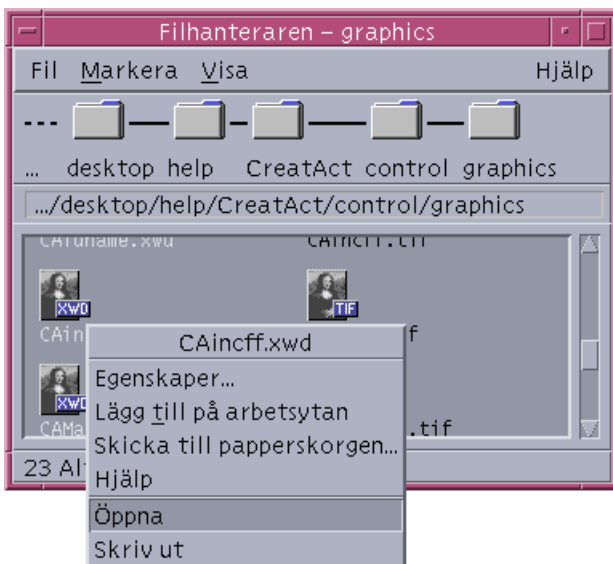

Kommandot i åtgärdens EXEC_STRING körs när användaren dubbelklickar på ikonen.

Huvudgruppen använder också åtgärder. Här är t ex en del av definitionen till ikonen Terminal på panelundermenyn Egna program. Kontrollen definieras i konfigurationsfilen /usr/dt/appconfig/types/språk/dtwwm.fp.

```
CONTROL Term
{
  ICON      Fpterm
  LABEL     Terminal
  PUSH_ACTION Dtterm
  ...
}
```

I fältet PUSH_ACTION anges vilken åtgärd som ska köras när användaren klickar på kontrollen. I det här fallet är det en åtgärd som heter Dtterm.

Ett annat vanligt användningsområde för åtgärder är menyer. Datafiler har normalt åtgärder på Markera-menyn i filhanteraren. XWD-filer (filer med namn som slutar på .xwd eller .wd) har t ex åtgärden Öppna som visar skärmbilden genom att köra åtgärden Xwud.



Figur 10-2 Åtgärden Öppna för filer med datatypen XWD

Åtgärderna på Markera-menyn anges i datatypsdefinitionen för XWD-filen.

Definitionen finns i konfigurationsfilen

/usr/dt/appconfig/types/språk/xclients.dt.

```
DATA_ATTRIBUTES XWD
{
  ACTIONS      Open,Print
  ICON         Dtxwd
}
```

```
...  
}
```

Datatypen XWD och motsvarande Open- och Print-åtgärder förklaras i "Hur datatyper ansluter datafiler till åtgärder" på sidan 150.

Hur åtgärder skapar programikoner

Titta på ikonen Visa xwd i programgruppen Skrivbordsverktyg. Om du dubbelklickar på den här ikonen körs X-klienten `xwud`. Ikonen representerar dock inte direkt det körbara programmet `xwud` i `/usr/bin/X11/xwud`.

Ikonen med namnet Visa xwd-fil visas i programgruppen eftersom det finns en fil i den katalogen som heter `Xwud` (se Figur 10-3). Den filen motsvarar en underliggande åtgärd med samma namn: `Xwud`. I åtgärdsdefinitionen är åtgärdsnamnet det namn som följer efter nyckelordet `ACTION`:

```
ACTION Xwud  
{  
  LABEL      Xwd Display  
  TYPE       COMMAND  
  WINDOW_TYPE NO_STUDIO  
  EXEC_STRING /usr/bin/X11/xwud -noclick -in \  
              %(File)Arg_1"Xwd-fil som ska visas:"%  
  DESCRIPTION Xwd Display (Xwud) XwdDisplay-åtgärden \  
              visar en xwd-fil som skapats med åtgärden \  
              Xwd Capture (Xwd). Den använder \  
              kommandot xwud.  
}
```

Filen kallas för en *åtgärdsfil* eftersom den är knuten till en åtgärd. En fil är en åtgärdsfil när den är körbar och har samma namn som åtgärden. Ikonen i programhanteraren (eller filhanteraren) kallas för en *åtgärdsikon* eller *programikon*, eftersom programmet startar då du dubbelklickar på den.



Figur 10-3 Programikonen (åtgärdsikonen) representerar en åtgärdsfil

När programhanteraren hittar en körbar fil letar den igenom åtgärdens databas för att se om det finns några åtgärder vars namn matchar filnamnet. Om så är fallet vet programhanteraren att filen är en åtgärdsfil.

Innehållet i en åtgärdsfil är ointressant eftersom den innehåller kommentarer till skrivbordsfunktionen.

Obs! En *åtgärdsfil* är inte detsamma som *åtgärdsdefinitionsfil*. *Åtgärdsfilen* är en fil med samma namn som åtgärden. Den används för att skapa *programikonen* i filhanteraren eller programhanteraren. En *åtgärdsdefinitionsfil* är den fil som heter *namn.dt* och innehåller definitionen av åtgärden.

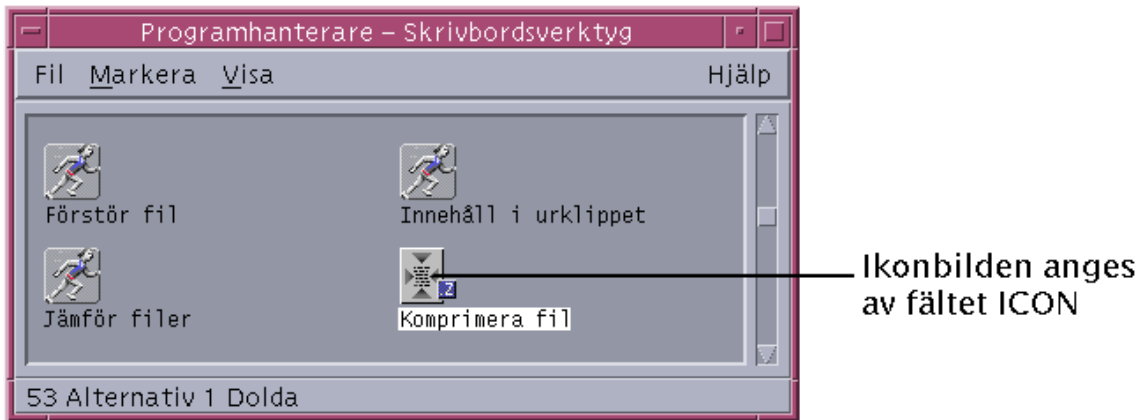
När skrivbordet upptäcker att en fil är en åtgärdsfil, används den underliggande åtgärden för att definiera åtgärdsfilens utseende och metod.

- I fältet EXEC_STRING anges hur programikonen ska fungera. För ikonen Visa xwd-fil anger EXEC_STRING att åtgärdsikonen ska köra X-klienten xwud med vissa kommandoradsargument.
- I fältet LABEL anges programikonens etikett.
- I fältet DESCRIPTION beskrivs texten som visas när användaren begär objekthjälp.
- Programikonen Xwud använder standardikonbilden för åtgärder, eftersom dess åtgärdsdefinition inte innehåller fältet ICON, som anger en annan bild.

Ikonen Komprimera fil använder däremot en annan ikonbild eftersom dess underliggande åtgärdsdefinition innehåller ett ICON-fält.

Till exempel:

```
ACTION Compress
{
  LABEL   Compress File
  ICON    Dtcmps
  ...
}
```



Figur 10-4 Ikonbild som anges med fältet ICON i åtgärdsdefinitionen

Åtgärden Xwud kallas en *kommandoåtgärd* eftersom dess definition innehåller kommandot som ska köras (EXEC_STRING). Fältet TYPE i åtgärdsdefinitionen definierar åtgärdstypen.

Inledningsvis visas ikonen Visa xwd i programgruppen Skrivbordsverktyg. Du kan dock skapa ytterligare kopior av åtgärdsikonen i valfri katalog där du har skrivbehörighet. Om åtgärdsdefinitionen för Xwud ingår i databasen, blir alla körbara filer du skapar med namnet Xwud åtgärdsfiler som representerar den åtgärden. Ikonen i programhanteraren eller filhanteraren används för att köra åtgärden.

Hur åtgärder använder datafiler som argument

Ett *argument* till ett kommando är normalt en fil som kommandot arbetar med. Åtgärder kan skrivas för att acceptera filargument.

EXEC_STRING för åtgärden Xwud anger t ex att ett filargument krävs:

```
EXEC_STRING    /usr/bin/X11/xwud -noclick -in \
               %(File)Arg_1"Xwd-fil som ska visas:"%
```

Termen Arg motsvarar ordet *argument*. Syntaxen Arg_1 betecknar det första argumentet, och (File) innebär att åtgärden behandlar argumentet som en fil.

Det enklaste sättet för användaren att tillhandahålla ett filargument är att släppa en datafil på programikonen. Skrivbordssystemet avgör sökvägen till den släppta filen och placerar den i kommandoraden i stället för texten mellan symbolerna %(File)Arg_1"Xwd-fil som ska visas: "%). Kommandot som körs blir då:

```
/usr/bin/X11/xwud -noclick -in sökväg
```

När användaren dubbelklickar på programikonen avgör skrivbordssystemet med hjälp av EXEC_STRING att ett filargument krävs, och en dialogruta som uppmanar

användaren att skriva ett filnamn eller en sökväg visas. I fallet med åtgärden Xwd blir ledtexten:

Xwd-fil som ska visas:

Filnamnet eller sökvägen som användaren anger används som filargument.

Ytterligare användningsområden för åtgärder

Förutom för att starta program används åtgärder på skrivbordet för att skapa funktionalitet i:

- Huvudgruppen.

Definitionen av en ikon i huvudgruppen omfattar fält som anger vilken åtgärd som körs när användaren klickar på ikonen eller släpper en fil på den. Mer information finns i "Definiera kontroller i huvudgruppen" på sidan 237.

- Meny

Med syntaxen till definitionerna för Fönster- och Arbetsyta-menyn kan du ange vilken åtgärd som ska köras av ett menyalternativ. Mer information finns i "Arbetsytahanterarens menyer" på sidan 254 och under dtwmrc(4) i direkthjälpen (man page).

- Kommunikation mellan program.

Ett program kan vara utformat för att skicka och ta emot information med en speciell typ av åtgärd som kallas ToolTalk-meddelande (TT_MSG). TT_MSG-åtgärder beskrivs i utvecklingsmiljödokumentationen för skrivbordssystemet.

Introduktion till datatyper

När användaren skapar en ny datafil varierar filikonens utseende och metod i filhanteraren, beroende på den typ av datafil som användaren har skapat. Med skrivbordets datatypmekanism går det att skapa anpassat utseende och en anpassad metod för filer och kataloger.

Vad är en datatyp?

En datatyp är en konstruktion som definieras i skrivbordets databas. Nedan finns t ex definitionen till datatypen XWD. Definitionen finns i konfigurationsfilen `/usr/dt/appconfig/types/språk/xclients.dt`.

```

DATA_ATTRIBUTES XWD
{
  ACTIONS      Open,Print
  ICON         Dtxwd
  NAME_TEMPLATE %s.xwd
  MIME_TYPE    application/octet-stream
  SUNV3_TYPE   xwd-file
  DESCRIPTION  Den här filen innehåller en bild i XWD-\
               format. Dessa filer skapas vanligtvis genom\
               att en bild tas av fönster med hjälp av åtgärden XwdCapture. \
               Dess datatyp heter XWD. XWD-filer \
               har namn som slutar på '.xwd' eller '.wd'.
}

DATA_CRITERIA XWD1
{
  DATA_ATTRIBUTES_NAME  XWD
  MODE                   f
  NAME_PATTERN           *.xwd
}

DATA_CRITERIA XWD2
{
  DATA_ATTRIBUTES_NAME  XWD
  MODE                   f
  NAME_PATTERN           *.wd
}

```

Alla datatypsdefinitioner består av två delar:

`DATA_ATTRIBUTES` – beskriver datatypens utseende och egenskaper.

`DATA_CRITERIA` – avgör regler (namn och innehåll) för hur en fil ska kategoriseras så att den tillhör en datatyp.

Fältet `DATA_ATTRIBUTES_NAME` kopplar villkoren till attributen.

Det kan finnas flera `DATA_CRITERIA` till ett `DATA_ATTRIBUTE`. Datatypen XWD har t ex två villkor för att ange två olika namnvillkor (`NAME_PATTERN`): namn som slutar med `.xwd` respektive `.wd`.

Hur datatyper ansluter datafiler till åtgärder

Ta datatypen XWD som exempel. Användaren skapar en fil av XWD-typ genom att ge filen något av filnamnställagen `.xwd` eller `.wd`. Skrivbordssystemet använder filnamnet som *villkor* vid bestämningen av filens datatyp.

Varje fil med datatypen XWD får:

- En unik ikonbild som hjälper användaren att känna igen datafilerna.
- Objekthjälp som informerar om datatypen.
- En egen Markera-meny i filhanteraren som innehåller åtgärderna Öppna och Skriv ut. Åtgärden Open (Öppna) för XWD-filer kör åtgärden Xwud.

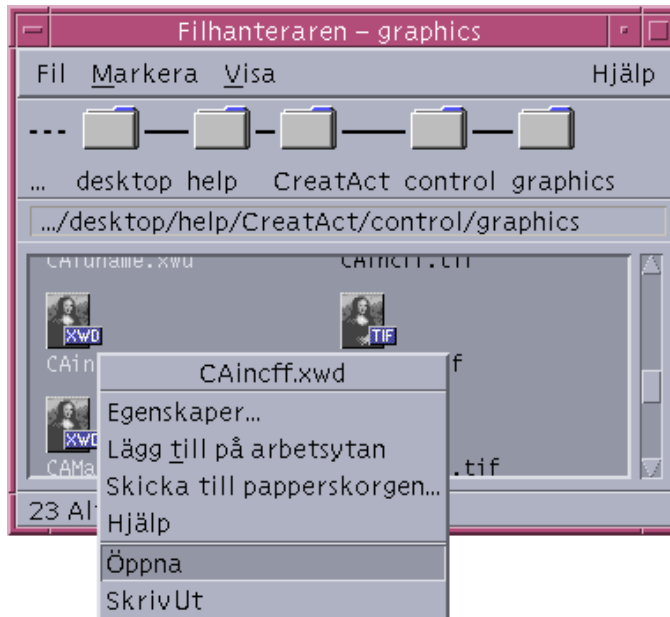
Köra åtgärder från Markera-menyn

Markera-menyn i filhanteraren är bara aktiv när en fil eller en katalog är markerad. Vilket kommando som finns längst ned på Markera-menyn beror på datatypen. Om en XWD-fil är markerad finns alternativen Öppna och Skriv ut på Markera-menyn.

Fältet `ACTIONS` i datatypsdefinitionen anger vilka kommandon som läggs till längst ned på datatypens Markera-menyn.

```
DATA_ATTRIBUTES XWD { ACTIONS Öppna,Skriv ut ... }
```

Innehållet på Markera-menyn beror på datatypen. Det finns många datatyper som innehåller åtgärden Open (Öppna). Det innebär att när du markerar en fil av en sådan datatyp i filhanteraren och visar Markera-menyn, finns kommandot Öppna på menyn.



Figur 10-5 Markera-menyn för en XWD-fil

Åtgärden Open (Öppna) kör normalt programmet som datafilen hör till. Om du t ex öppnar en XWD-fil körs åtgärden Xwud, vilken i sin tur kör X-klienten xwud för att visa skärmbilden. Med andra ord är åtgärden Open (Öppna) synonym med åtgärden Xwud, för datatypen XWD. Om du öppnar en fil av typen TEXTFILE öppnas Textredigeraren. När du öppnar en BM- (bitmapp) eller PM-fil (bildpunktsmönster) körs ikonredigeraren.

Du kan skapa olika Öppna-åtgärder som utför olika saker med två möjligheter till åtgärdsdefinitioner:

- Åtgärdsindelning.

Med åtgärdsstilldelning skapar du en åtgärd som kör en annan åtgärd i stället för att köra ett kommando direkt. Du kan t ex skapa en Öppna-åtgärd som tilldelas (kör) åtgärden Xwud.

- Datatypsrestriktioner för en åtgärd.

Åtgärdsdefinitioner kan innehålla fältet ARG_TYPE som begränsar åtgärden till vissa datatyper. Du kan t ex ange att Öppna-åtgärden som kör åtgärden Xwud bara gäller för fält med datatypen XWD.

Nedan finns definitionen av den åtgärd som tilldelar Öppna-åtgärden åtgärden Xwud för datatypen XWD. Den finns i databaskonfigurationsfilen /usr/dt/appconfig/types/C/xclients.dt.

```
ACTION Open
{
  LABEL      Open
  ARG_TYPE   XWD
  TYPE       MAP
  MAP_ACTION Xwud
}
```

Fältet TYPE anger att detta är en avbildningsåtgärd. MAP_ACTION anger att åtgärden kör åtgärden Xwud. Fältet ARG_TYPE anger att åtgärden bara gäller för filer med datatypen XWD.

Jämför föregående definition av åtgärden Öppna med nästa definition, som finns i databasfilen /usr/dt/appconfig/types/C/dt.dt.

```
ACTION Open
{
  LABEL      Open
  ARG_TYPE   BM
  TYPE       MAP
  MAP_ACTION Dticon
}
```

Den här definitionen gäller för filer med datatypen (ARG_TYPE) BM (bitmappsfiler). Definitionen tilldelar åtgärden Öppna åtgärden Dticon, som kör Ikonredigeraren.

Definiera hur datatypens dubbelklickningar fungerar

Vad som händer när du dubbelklickar för en viss datatyp definieras av den första posten i fältet ACTIONS. För datatypen XWD innebär ett dubbelklick att åtgärden Öppna körs, vilken i sin tur kör åtgärden Xwud.

Släppa en datafil på en åtgärdsikon

När användaren släpper en datafil på en åtgärdsikon körs åtgärden med datafilen som argument (se "Hur åtgärder använder datafiler som argument" på sidan 148).

När t ex datafilen XWD släpps på ikonerna Visa xwd körs åtgärden Xwud med den datafilen som argument. Detta kör X-klienten xwud med datafilen.

Skriva ut datatyper från skrivbordet

En datafil kan skrivas ut på följande sätt från skrivbordet:

- Använd om möjligt kommandot Skriv ut på filhanterarens Markera-meny.
- Släpp en datafil på skrivbordets målområde för utskrift (skrivarikonerna på huvudgruppen eller en skrivarikon i utskriftshanteraren).

Dessutom går det att skriva ut från många program.

När du skriver ut från skrivbordet används åtgärden Skriv ut. Skriv ut är, på samma sätt som Öppna, ett åtgärdsnamn som används till flera olika typer av data. Därför används åtgärdsavbildning och fältet ARG_TYPE för Skriv ut-åtgärder för att anpassa utskriften efter varje datatyp.

Nedan finns t ex åtgärden Skriv ut för datatypen XWD. Definitionen finns i /usr/dt/appconfig/types/*språk*/xclients.dt.

```
ACTION Print
{
  LABEL      Print
  ARG_TYPE   XWD
  TYPE       MAP
  MAP_ACTION NoPrint
}
```

Den här Print (Skriv ut)-åtgärden, som är specifik för XWD-filer, tilldelas en NoPrint-åtgärd. NoPrint är en särskild åtgärd som definieras i /usr/dt/appconfig/types/*språk*/dt.dt. Åtgärden NoPrint visar en dialogruta som anger att den här datatypen inte kan skrivas ut.

Jämför Print-åtgärden för XWD med följande Print-åtgärd för PCL-filer:

```
ACTION Print
{
  LABEL      Print
  ARG_TYPE   PCL
  TYPE       MAP
  MAP_ACTION PrintRaw
}
```

I filen /usr/dt/appconfig/types/*språk*/print.dt definieras åtgärden PrintRaw, som innehåller följande kommandorad för utskrift av PCL-filerna:

```
ACTION PrintRaw
{
  TYPE       COMMAND
  WINDOW_TYPE NO_STDIO
  EXEC_STRING /usr/dt/bin/dt1p -w %(File)Arg_1%
}
```


Skapa åtgärder och datatyper med Skapa åtgärd

Skapa åtgärd är ett verktyg för att skapa:

- En åtgärd som startar ett program
- En eller flera datatyper för ett programs datafiler
- Åtgärder för att öppna och skriva ut programmets datafiler

Skapa åtgärd är också användbart för att skapa enkla åtgärder som kör operativsystemkommandon och skalskript.

- "Vad Skapa åtgärd gör" på sidan 155
- "Begränsningar i Skapa åtgärd" på sidan 156
- "Skapa en åtgärd och datatyp för ett program med Skapa åtgärd" på sidan 157

Referensinformation finns i direkthjälpen (man) för dtcreate(1X).

Vad Skapa åtgärd gör

Skapa åtgärd omfattar ett huvudfönster och en uppsättning dialogrutor som du använder för att skapa en åtgärd och tillhörande datatyper.

Skapa åtgärd gör följande:

- Skapar en åtgärdsdefinition som kör ett kommando.
- Skapar filen *Hemkatalog/.dt/types/åtgärdsnamn.dt*. Den här filen lagrar åtgärden och de datatypsdefinitioner som skapas för programmet.
- Skapar en *åtgärdsfil* i användarens Hemkatalog. Åtgärdsfilen är en körbar fil med samma namn som åtgärden.

Åtgärdsfilen representeras i filhanteraren av en *programikon* som startar programmet när du dubbelklickar på den.

Du kan också göra åtgärdsikonen till ett målområde genom att ange en släppbar datatyp när du skapar åtgärden.

- Skapar en eller flera datatyper för programmets datafiler (valfritt).
- Skapar en Öppna-åtgärd för varje datatyp.
- Skapar en Skriva ut-åtgärd för varje datatyp (valfritt).
- Startar om databasen för åtgärder och datatyper. Detta gör att åtgärderna och datatyperna träder i kraft omedelbart.

Begränsningar i Skapa åtgärd

Skapa åtgärd är utformat för att skapa åtgärder och datatyper för program som ska köras. Åtgärder och datatyper kan dock vara mycket flexibla och omfatta ytterligare funktionalitet som bara kan nås om du skapar definitionerna manuellt.

Mer information finns i:

- Kapitel 12
- Kapitel 13

Åtgärdsbegränsningar

Du kan inte använda Skapa åtgärd för att skapa åtgärder för ett program om något av följande villkor är sanna:

- Kommandoraden kräver ett argument som inte är ett filargument (parameter).

Du kan t ex inte använda Skapa åtgärd för att skriva en åtgärd för kommandot:

```
lp -denhet filnamn
```

där användaren måste ange *enhet* varje gång kommandot ska köras.

- Programikonen måste ha ett annat namn än åtgärden.

Du kan t ex inte använda Skapa åtgärd för att tillhandahålla en lokal språkversion av en befintlig åtgärd.

- Åtgärden kräver någon av specialåtgärderna i åtgärdsdatabasen.

Exempel på sådana specialåtgärder är åtgärder som:

- Startar kommandon i fjärrsystem

- Anropar andra åtgärder
- Måste köras som en annan användare (t ex en superanvändare)
- Använder tilldelningsåtgärden
- Fungerar olika beroende på antalet filargument som anges för åtgärden

Datatypsbegränsningar

Du kan inte använda Skapa åtgärd för att skapa datatypen till ett program om något av följande villkor är sanna:

- Datatypen har andra åtgärder än Öppna och Skriv ut kopplade till sig.
- Åtgärden Öppna för datatypen är inte åtgärdens kommando.

Du kan t ex inte använda Skapa åtgärd för att skapa datatypen som tillhandahåller en unik ikon för den katalog som representerar programmets programgrupp.

Skapa en åtgärd och datatyp för ett program med Skapa åtgärd

Det finns några saker som du måste känna till om programmet innan du kör Skapa åtgärd.

- Kommandoraden för att starta programmet.

Du måste veta om kommandoraden innehåller ett obligatoriskt filargument, ett valfritt filargument eller inget filargument.

Om programmet kräver argument som inte består av en fil, kan du inte använda Skapa åtgärd.

- De typer av datafiler som ett program accepterar.

Vissa program accepterar bara en typ av data. Andra (t ex en ASCII-redigerare eller grafikredigerare) accepterar flera datatyper.

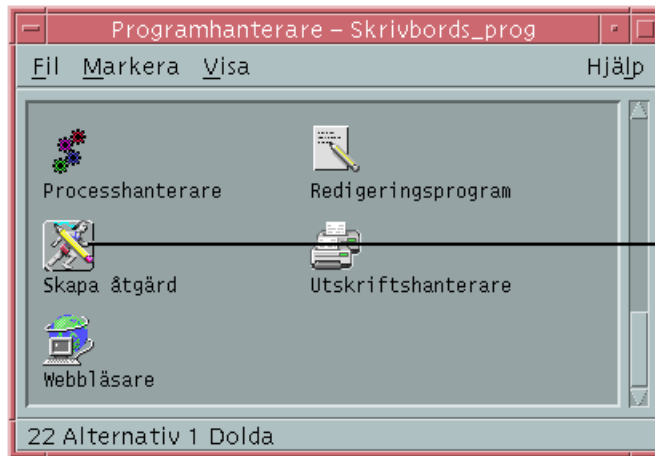
- På vilket sätt programmet identifierar sina datafiler.

Detta kan vara en namngivningskonvention (t ex filnamn som slutar med `.doc`). Det kan också bero på innehållet i filen. Även om programmet inte använder någon namngivningskonvention, kan du ange en sådan för åtgärdsikonen.

- Valfritt: Kommandoraden för att skriva ut filer.

▼ Skapa en åtgärd för ett program

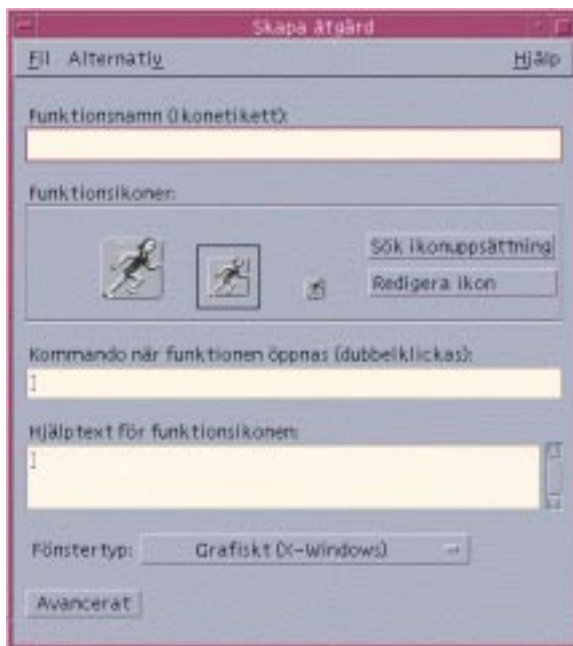
1. Dubbelklicka på Skapa åtgärd i programgruppen Skrivbordsprogram.



Dubbelklicka på ikonen om du vill köra Skapa åtgärd

Figur 11-1 Ikonen för Skapa åtgärd i programhanteraren

Detta öppnar huvudfönstret i Skapa åtgärd.



Figur 11-2 Huvudfönstret i Skapa åtgärd

2. **Skriv namnet på åtgärdsikonen i textfältet Åtgärdsnamn.**

3. **Använd någon av åtgärdsikonerna för att ange ikon för programmet. Inledningsvis visas standardikonen.**

- Om du vill välja en annan tillgänglig ikon kan du klicka på Sök ikonuppsättning för att visa dialogrutan Sök ikonuppsättning. Se "Använd dialogrutan Sök ikonuppsättning för att ange en ikon" på sidan 165.
- Om du vill skapa nya ikoner väljer du Redigera ikon för att köra Ikonredigeraren.

4. **Skriv vilket kommando som ska starta programmet i textfältet Kommando när åtgärden öppnas.**

Använd syntaxen $\$n$ som filargument, t ex:

```
emacs
  bitmap $1
  diff $1 $2
  lp -oraw $1
```

Om kommandoraden innehåller ett filargument ($\$n$), blir åtgärdsikonen ett område där du kan släppa filer.

Kommandoraden skickas inte till ett skal såvida du inte särskilt anger att skal ska användas. Följande rader använder skalbearbetning:

```
/bin/sh -c 'ps | lp'
/bin/sh -c 'spell $1 | more'
```

5. **Skriv texten till objekthjälpen för åtgärdsikonen i textfältet Hjälptext för åtgärdsikonen.**

Texten radbryts automatiskt i textfältet. Dessa radbrytningar bevaras dock inte när objekthjälpen visas på skärmen. Om du vill infoga en hård radbrytning använder du $\backslash n$.

6. **Välj det fönsterstöd som åtgärden kräver i menyfältet Fönstertyp.**

Grafiskt (X-Windows) – programmet skapar ett eget fönster

Terminal (automatisk stängning) – programmet körs i ett kommandofönster som stängs automatiskt när programmet avslutas

Terminal (manuell stängning) – programmet körs i ett kommandofönster som är öppet ända tills användaren stänger det

Ingen visning – programmet visar inga utdata på skärmen

7. **Utför följande:**

- Information om hur du skapar en eller flera datatyper för datafiler om programmet har sådana finns i "Skapa en eller flera datatyper för programmet" på sidan 160."
- Om du inte vill skapa en datatyp kan du spara åtgärden genom att välja Spara från Fil-menyn. Testa sedan den nya åtgärden genom att dubbelklicka på ikonen i Hemkatalogen.

▼ Skapa en eller flera datatyper för programmet

1. Definiera en åtgärd för ett program enligt anvisningarna i föregående avsnitt, "Skapa en åtgärd för ett program" på sidan 157.
2. Klicka på Avancerat i fönstret Skapa åtgärd för att visa fler alternativ.



Figur 11-3 Specialåtgärder i huvudfönstret till Skapa åtgärd

3. Om du vill att programikonen ska fråga efter ett filargument när någon dubbelklickar på den, skriver du texten till ledtexten i textfältet "När åtgärden öppnas, fråga efter".

Använd följande riktlinjer för textfältet:

- Du måste använda det här fältet om programmets kommandorad har ett *obligatoriskt* filargument.
- Lämna detta fält tomt om kommandoraden inte innehåller något filargument.
- Om filargumentet är valfritt i programmets kommandorad kan du välja om du vill använda det. Om du bifogar ledtext kommer åtgärdsikonen att fråga efter

filnamnet när du dubbelklickar på ikonen. Om du inte bifogar ledtext kommer programmet att köras med en nollvariabel som filargument.

4. Ange de filtyper som åtgärden kommer att acceptera som argument:

- Om åtgärden ska acceptera alla datatyper markerar du Alla datatyper.
- Om åtgärden bara ska acceptera de datatyper som du har skapat för programmet markerar du Endast från listan ovan.

Inledningsvis kommer listan Datatyper som använder den här åtgärden att vara tom. Efterhand som du skapar datatyper för program kommer de att läggas till i listan.

5. Klicka på Lägg till (bredvid listan Datatyper som använder den här åtgärden) för att visa dialogrutan Lägg till datatyp.



Figur 11-4 Dialogrutan Lägg till datatyp i Skapa åtgärd

6. **Valfritt:** Om du inte vill använda datatypens standardnamn skriver du ett nytt namn på datatypen i textfältet **Datatypfamiljens namn**.

Namnet får inte innehålla blanksteg. Datatypsnamnet är inte synligt för programmets användare, utan används i åtgärds- och datatypsdata-baser för att identifiera datatypsdefinitionen.

7. **Klicka på Redigera (bredvid rutan Särskilda egenskaper) för att visa dialogrutan Egenskaper.**



Figur 11-5 Dialogrutan Egenskaper i Skapa åtgärd

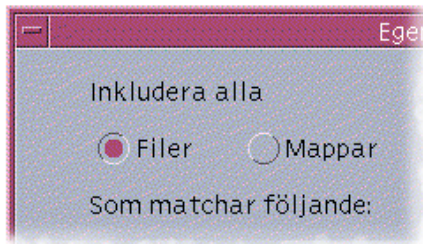
Datatypens egenskaper är de villkor som används för att skilja en datatyp från en annan. Du kan välja ett eller flera av följande villkor:

Filer eller mapper – datatypen används endast för filer eller på mapper

Namnmonster – datatypen bygger på filnamnet

Behörighetsmönster – rättighet att läsa, skriva och använda
Innehåll – innehåll i en viss del av en fil

8. Ange om datatypen ska representera en fil eller en mapp.



Figur 11-6 Ange en fil- eller katalogegenskap för datatypen.

9. Om datatypen är beroende av namnet markerar du kryssrutan Namnmönster och fyller i textfältet.



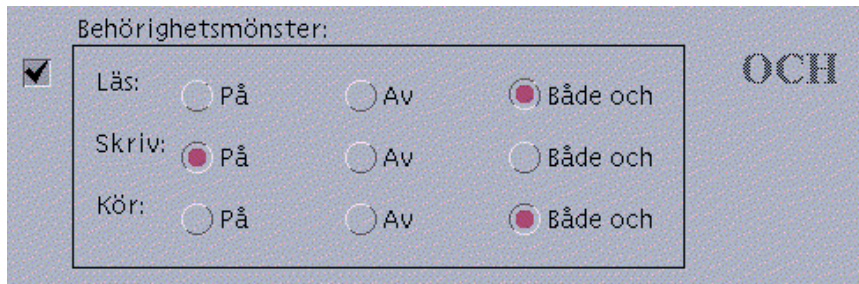
Figur 11-7 Ange filnamnsegenskap för en datatyp

Du kan använda * och ? som jokertecken:

* – matchar en serie tecken

? – matchar ett enskilt tecken

10. Om datatypen är beroende av behörigheter markerar du kryssrutan Behörighetsmönster och markerar behörighet för datatypen.



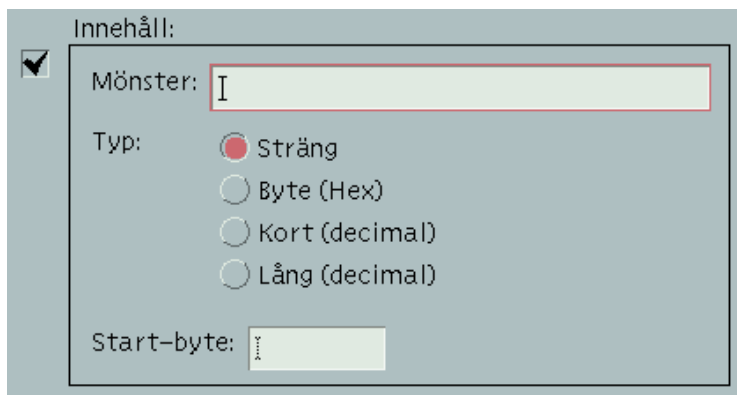
Figur 11-8 Ange behörighetsegenskaper för datatypen

På – filen måste ha den angivna behörigheten

Av – filen måste sakna den angivna behörigheten

Både och – den angivna behörigheten spelar ingen roll

11. Om datatypen beror på dess innehåll kan du markera kryssrutan **Innehåll och bifoga den information du vill ge** – klicka på **Mönster och Typ**. Du kan också ange i vilken position som sökningen ska starta.



Figur 11-9 Ange innehållsegenskaper för en datatyp

Obs! Om du använder en innehållsbaserad datatyp kan detta påverka systemets prestanda.

12. Klicka på **OK** för att stänga dialogrutan **Egenskaper**.

Egenskaperna visas i fältet **Särskilda egenskaper** med dessa koder:

- d – en katalog
- r – fil med läsrättigheter
- w – fil med skrivrättigheter
- x – fil med användningsrättigheter
- ! – NOT, logisk operator
- & – AND, logisk operator

13. Skriv hjälptexten för datafilen i fältet Hjälptext för datatypikonen.

14. Använd någon av åtgärdsikonerna för att ange ikon för programmet. Inledningsvis visas standardikonen.

- Om du vill välja en annan tillgänglig ikon kan du klicka på Sök ikonuppsättning för att visa dialogrutan Sök ikonuppsättning. Se "Använd dialogrutan Sök ikonuppsättning för att ange en ikon" på sidan 165.
- Om du vill skapa nya ikoner klickar du på Redigera ikon för att köra Ikonredigeraren.

15. Bekräfta kommandot i textfältet Kommando för att öppna den här datatypen. Detta är kommandot som kommer att användas när användaren dubbelklickar på datafilen.

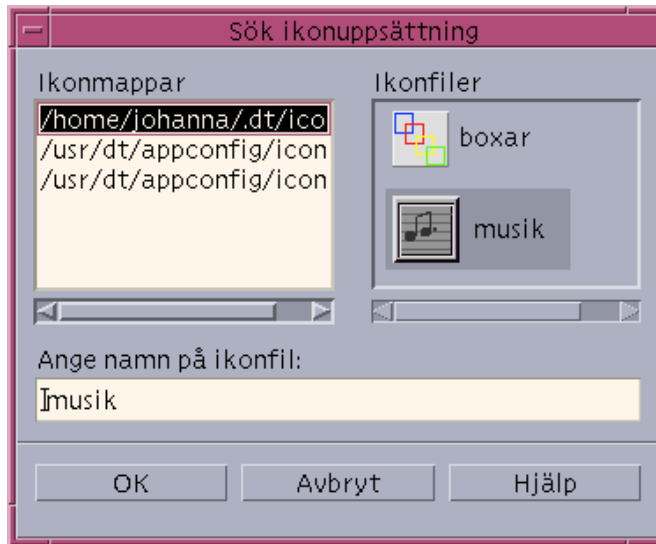
16. Valfritt: Om programmet innehåller ett kommando för utskrift av datafiler från kommandoraden skriver du kommandot i textfältet Kommando för att skriva ut den här datatypen med syntaxen \$n för ett filargument.

17. Utför något av följande steg för att spara datatypsdefinitionen:

- Klicka på OK för att spara datatypen och stäng dialogrutan Lägg till datatyp.
- Klicka på Använd för att spara datatypen utan att stänga dialogrutan Lägg till datatyp. Detta gör att du omedelbart kan definiera ytterligare en ny datatyp för en åtgärd.

Använd dialogrutan Sök ikonuppsättning för att ange en ikon

Dialogrutan Sök ikonuppsättning visas när du klickar på Sök ikonuppsättning i huvudfönstret till Skapa åtgärd, eller i fönstret Lägg till datatyp. Använd den här dialogrutan för att ange vilken ikon som ska användas till åtgärden eller datatypen.



Figur 11-10 Dialogrutan Sök ikonuppsättning

I dialogrutan Sök ikonuppsättning kan du ange en uppsättning ikonbildsfilerna som finns i:

- En katalog i ikonsökvägen. I listan Ikonmappar finns alla kataloger i ikonsökvägen.
- Ett registreringspaket som ska integreras med skrivbordssystemet med hjälp av dtappintegrate. Dessa ikoner finns ännu inte i någon katalog i ikonsökvägen, men placeras där av dtappintegrate.

Obs! Åtgärds- och datatypsdefinitioner som skapas med Skapa åtgärd använder basnamnet på ikonfilen (namnet minus suffixet för typ och storlek). Ikoner för åtgärder och datatyper som skapas med Skapa åtgärd måste placeras i kataloger i ikonsökvägen.

Ange en uppsättning ikoner i ikonsökvägen

1. **I listan Ikonmappar i dialogrutan Sök ikonuppsättning dubbelklickar du på mappsökvägen som innehåller ikonerna.**

Listan Ikonfiler visar alla ikoner som finns i mappen.

2. **Klicka på ikonerna du vill använda i listan Ikonfiler.**

Basnamnet på ikonfilen placeras i textfältet Ange namn på ikonfil.

3. **Välj OK.**

Ange en ikon i ett registreringspaket

För systemadministratörer eller programmerare som skapar ett registreringspaket finns ikonfilerna inledningsvis i en katalog i registreringspaketet:

`prgm_rot/dt/appconfig/icons/språk`

När du har registrerat ikonerna med `dtappintegrate`, kopieras ikonfilerna till `/etc/dt/appconfig/icons/språk` i ikonsökvägen.

Använd proceduren nedan för att ange ikoner som är en del av registreringspaketet:

- 1. I textfältet Ange namn på ikonfil i dialogrutan Sök ikonuppsättning skriver du ett basnamn på ikonfilen.**
- 2. Välj OK.**
Skapa åtgärd visar en dialogruta som meddelar att ikonerna inte fanns i någon katalog längs ikonsökvägen.
- 3. Välj Ingen ändring i dialogrutan som visas.**

Skapa åtgärder manuellt

Det finns två sätt att skapa åtgärder:

- Använda skrivbordsprogrammet Skapa åtgärd
- Skapa en åtgärdsdefinition manuellt

När du skapar en åtgärd manuellt måste du redigera en databasfil. Det här kapitlet beskriver hur du skapar åtgärdsdefinitioner manuellt.

- "När åtgärder måste skapas manuellt" på sidan 170
- "Skapa en åtgärd manuellt: Allmänna steg " på sidan 171
- "Skapa en exekveringssträng för COMMAND-åtgärder" på sidan 179
- "Fönsterstöd och terminalemulatorer för COMMAND-åtgärder" på sidan 185
- "Begränsa åtgärder för vissa argument" på sidan 187
- "Skapa åtgärder som kör program i fjärrsystem" på sidan 190
- "Använda variabler i åtgärds- och datatypsdefinitioner" på sidan 191
- "Starta åtgärder från en kommandorad" på sidan 192
- "Skapa språkanpassade åtgärder" på sidan 194
- "Skapa åtgärder för ToolTalk-program" på sidan 195
- En introduktion till åtgärder finns i Kapitel 10.
- Information om hur du använder Skapa åtgärd finns i Kapitel 11.
- Referensinformation om åtgärdsdefinitioner finns i direkthjälpen (man) för dtactionfile(4).

När åtgärder måste skapas manuellt

Det finns tre grundtyper av åtgärder:

- COMMAND
- MAP
- TT_MSG

Verktyget Skapa åtgärd är utformat för att skapa vissa typer av COMMAND- och MAP-åtgärder. Alla TT_MSG-åtgärder måste skapas manuellt.

Mer information finns i "Begränsningar i Skapa åtgärd" på sidan 156.

COMMAND-åtgärder

En *kommandoåtgärd* utför ett kommando som startar ett program eller ett verktyg, kör en kommandofil i skalet eller exekverar ett operativsystemkommando. Definitionen av åtgärden innehåller kommandot som ska köras (EXEC_STRING).

Verktyget Skapa åtgärd kan användas till att skapa de vanligaste typerna av kommandoåtgärder. Det kan dock finnas situationer där du måste skapa åtgärden manuellt. Du måste exempelvis skapa en COMMAND-åtgärd manuellt om åtgärden anger:

- Flerfilsargument med olika ledtexter för varje argument.
- Åtgärdsanrop – åtgärder som startar andra åtgärder.
- Argumentantalsberoende åtgärd – en åtgärd som fungerar olika för olika antal filargument skapas.
- Värddator för fjärrkörning – ett program kan köras på ett annat system än det som innehåller åtgärdsdefinitionen.
- Användarändring – åtgärden kan köras som en annan användare (t ex kan superanvändarens lösenord efterfrågas, och därefter körs åtgärden som root).

MAP-åtgärder

En *avbildningsåtgärd* är en åtgärd som avbildas på en annan åtgärd i stället för att den direkt anger ett kommando eller ToolTalk-meddelande.

Med tilldelning kan du ange alternativa namn på åtgärder. En inbyggd kommandoåtgärd med namnet IconEditor startar Ikonredigeraren. Databasen innehåller också en Öppna-åtgärd, som i definitionen är begränsad till bitmapps- och bildpunktsfiler (av fältet ARG_TYPE). Denna åtgärd har avbildats på åtgärden

IconEditor. På så sätt kan användaren starta Ikonredigeraren genom att markera en bitmapps- eller bildpunktsfil i filhanteraren och därefter välja Öppna på Markera-menyn.

Skapa åtgärd erbjuder begränsad tilldelning för Öppna- och Skriv ut-åtgärder. Alla andra tilldelningsåtgärder måste skapas manuellt.

TT_MSG-åtgärder (ToolTalk-meddelanden)

TT_MSG åtgärder skickar ToolTalk-meddelanden. Alla TT_MSG-åtgärder måste skapas manuellt.

Skapa en åtgärd manuellt: Allmänna steg

Det här avsnittet beskriver hur du skapar en konfigurationsfil för en åtgärdsdefinition.

Konfigurationsfiler för åtgärder

Konfigurationsfiler som innehåller åtgärdsdefinitioner måste uppfylla följande krav:

- Filerna måste använda namngivningskonventionen *namn*.dt
- Filerna måste finnas i databasens (åtgärder och datatyper) sökväg.
Standardsökväg:

Egna åtgärder – *Hemkatalog*/.dt/types

Åtgärder för hela systemet – /etc/dt/appconfig/types/*språk*

Inbyggda åtgärder – /usr/dt/appconfig/types/*språk* Den här katalogen ska du inte använda.

Mer information om hur du ändrar sökväg för åtgärder och datatyper finns i "Ange värde för en sökväg" på sidan 131.

▼ Skapa en åtgärd manuellt

1. **Öppna en befintlig databasfil eller skapa en ny.**
Se föregående avsnitt, "Konfigurationsfiler för åtgärder" på sidan 171.
2. **Skapa en åtgärdsdefinition med syntaxen:**

```

ACTION åtgärdsnamn
{
    TYPE åtgärdstyp
    åtgärdsfält
    ...
}

```

där:

åtgärdsnamn – anger namnet som används för att köra åtgärden

åtgärdstyp – COMMAND (standard), MAP eller TT_MSG.

åtgärdsfält – ett av de nödvändiga eller valfria fälten för den här typen av åtgärd.

Alla fält består av ett nyckelord och ett värde.

Många av åtgärdsfälten beskrivs i det här kapitlet. Mer information finns i direkthjälpen (man page) för dtactionfile(4).

3. Spara filen.

4. Om du vill att åtgärdsikonen ska ha en unik bild skapar du en ikon till åtgärden. Standardadressen för ikoner är:

- Egna ikoner: *Hemkatalog*/.dt/icons
- Ikoner för hela systemet: /etc/dt/appconfig/icons/språk. *Standardspråk* är C.

Mer information finns i "Ange åtgärdens ikonbild" på sidan 176.

5. Dubbelklicka på Ladda om åtgärder i programgruppen Skrivbordsverktyg.

6. Skapa åtgärdsfilen för åtgärden. Åtgärdsfilen skapar en ikon i filhanteraren eller programhanteraren som representerar åtgärden. (Om åtgärden är skriven för att starta ett program kallas ikonen för *programikon*.)

Om du vill skapa en åtgärdsfil skapar du en körbar fil med samma namn som ett *åtgärdsnamn*. Du kan placera den här filen i en valfri katalog, som du har skrivbehörighet till. Du kan skapa så många åtgärdsfiler du vill.

Exempel på hur du skapar en COMMAND-åtgärd

I följande steg skapas en egen åtgärd som startar ett faxprogram i fjärrsystemet ProgServerA. Kommandot som startar faxprogrammet är:

```
/usr/fax/bin/faxcompose [filnamn]
```

1. Skapa filen *Hemkatalog*/.dt/types/Fax.dt.

2. Placera följande åtgärdsdefinition i filen:

```

ACTION FaxComposer
{

```

```

TYPE      COMMAND
ICON      fax
WINDOW_TYPE  NO_STDIO
EXEC_STRING  /usr/fax/bin/faxcompose -c %Arg_1%
EXEC_HOST   AppServerA
DESCRIPTION  Kör faxprogrammet

```

```

}
```

Fälten `WINDOW_TYPE` och `EXEC_STRING` beskriver åtgärdens uppförande.

`WINDOW_TYPE` – nyckelordet `NO_STDIO` anger att åtgärden inte behöver köras i ett terminalemulatorfönster.

Se "Ange fönsterstöd för åtgärden" på sidan 186.

`EXEC_STRING` – syntaxen `%Arg_1%` accepterar en släppt fil. Om någon dubbelklickar på åtgärdsikonen, öppnar åtgärden ett tomt faxassistentfönster.

Se "Skapa en exekveringssträng för `COMMAND`-åtgärder" på sidan 179.

3. Spara filen.

4. Skapa följande ikonbildsfil i *Hemkatalog*/`.dt/icons` med hjälp av Ikonredigeraren:

- fax.m.pm, 32 gånger 32 bildpunkter
- fax.t.pm, 16 gånger 16 bildpunkter

5. Dubbelklicka på Ladda om åtgärder i programgruppen Skrivbordsverktyg.

6. Skapa en körbar fil med namnet `FaxComposer` i en katalog som du har skrivbehörighet till (t ex hemkatalogen).

Exempel på hur du skapar en MAP-åtgärd

Anta att de flesta filer du faxar skapas med Textredigeraren och är av datatypen `TEXTFILE` (filer med namnet `*.txt`).

Stegen nedan lägger till menyalternativet "Fax" på datatypens Markera-menyn.

1. Öppna filen *Hemkatalog*/`.dt/types/Fax.dt`, som skapades i föregående exempel.

2. Lägg till den här tilldelningsdefinitionen i filen:

```

ACTION Fax
{
  ARG_TYPE  TEXTFILE
  TYPE      MAP
  MAP_ACTION  FaxComposer
}

```

3. Spara filen.

4. **Kopiera en dataattributsdefinition för TEXTFILE från /usr/dt/appconfig/types/*språk*/dtpad.dt till den nya filen *Hemkatalog*/.dt/types/textfile.dt. Lägg till åtgärden Fax i fältet ACTIONS.**

```
DATA_ATTRIBUTES TEXTFILE
{
  ACTIONS    Open,Print,Fax
  ICON       Dtpenpd
  ...
}
```

5. Spara filen.

6. **Öppna programhanteraren och dubbelklicka på Ladda om åtgärder i programgruppen Skrivbordsverktyg.**

▼ Starta om databasen för åtgärder och datatyper

För att nya och redigerade åtgärdsdefinitioner ska träda i kraft måste skrivbordet läsa om databasen.

- ◆ **Öppna programgruppen Skrivbordsverktyg och dubbelklicka på Ladda om åtgärder.**

- ◆ **Eller kör kommandot:**

```
dtaction ReloadActions
```

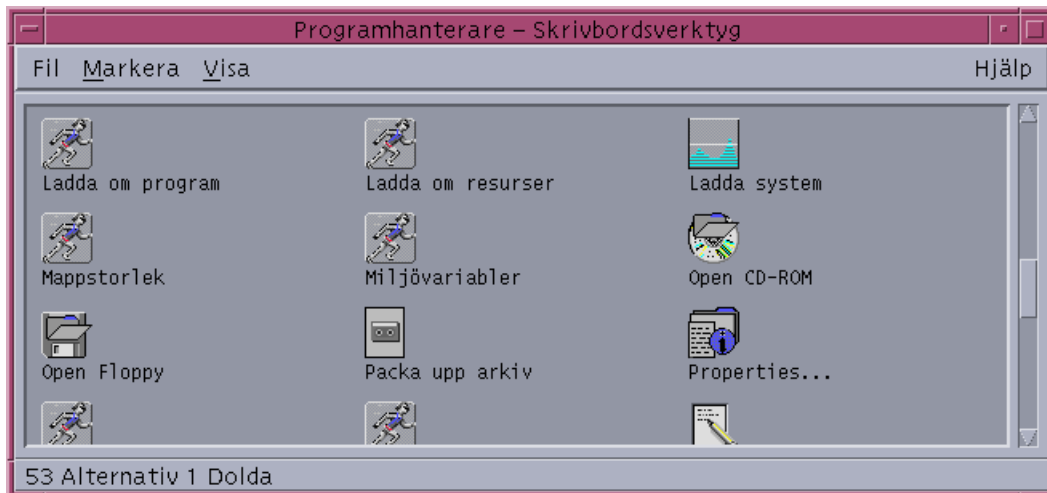
ReloadActions är namnet på den åtgärd vars ikon heter "Ladda om åtgärder".

Åtgärdsdatabasen läses också om när användaren:

- Loggar in
- Startar om arbetsyttehanteraren
- Sparar en åtgärd i fönstret Skapa åtgärd genom att välja Spara på Fil-menyn.

Skapa åtgärdsfil (ikon) för åtgärd

En *åtgärdsfil* är en fil som skapas för att visuellt representera åtgärden i filhanteraren eller programhanteraren.



Figur 12-1 Åtgärdsfiler (åtgärdsikoner) i programhanteraren

Eftersom en åtgärdsfils ikon representerar en åtgärd, kallas den ibland *åtgärdsikon*. Om den underliggande åtgärden startar ett program, kallas åtgärdsfilsikonen för *programikon*.

Om du dubbelklickar på åtgärdsikonen körs åtgärden. Åtgärdsikonen kan också vara ett släppområde.

Skapa en åtgärdsfil (åtgärdsikon)

- ♦ **Skapa en körbar fil med samma namn som åtgärdsnamnet. Innehållet i filen har ingen betydelse.**

Om åtgärdsdefinitionen är:

```
ACTION MyFavoriteApp
{
    EXEC_STRING    Mfa -file %Arg_1%
    DESCRIPTION    Kör MyFavoriteApp
    ICON           Mfapp
}
```

blir åtgärdsfilen en körbar fil med namnet `MyFavoriteApp`. I filhanteraren och programhanteraren använder filen `MyFavoriteApp` ikonbilden `Mfapp.storlek.typ`. Om du dubbelklickar på ikonen `MyFavoriteApp` körs åtgärdens exekveringssträng, och innehållet i fältet `DESCRIPTION` utgör ikonens objekt hjälp ("Kör `MyFavoriteApp`").

Åtgärdsnamn

Om åtgärdsdefinitionen innehåller fältet LABEL, får åtgärdsfilen innehållet i detta fält som namn i filhanteraren och programhanteraren i stället för filnamnet (*åtgärdsnamn*). Om åtgärdsdefinitionen t ex innehåller:

```
ACTION MyFavoriteApp
{
  LABEL      Mitt favoritprogram
  ...
}
```

kommer åtgärdsikonen att heta "Mitt favoritprogram".

Ange åtgärdens ikonbild

Använd fältet ICON för att ange vilken ikonbild som ska användas i filhanteraren och programhanteraren för den åtgärdsikon som skapas för åtgärden.

Om du inte anger någon ikon används bildfilerna för standardåtgärdsikoner:
`/usr/dt/appconfig/icons/språk/Dtactn.*`.



Figur 12-2 Standardbild för åtgärdsikon

Standardåtgärdsikonen ändras med resursen:

```
*actionIcon: ikonfilnamn
```

där *ikonfilnamn* kan vara ett grundnamn eller en absolut sökväg.

Värdet på fältet ICON kan vara:

- Ett basfilnamn.

Basfilnamnet är namnet på filen som innehåller ikonerna minus filnamnsuffixen för storlek (m och t) och bildtyp (bm och pm). Om filerna heter t ex `GameIcon.m.pm` och `GameIcon.t.pm`, använder du `GameIcon`.

Om du använder basfilnamnet måste ikonfilerna placeras i en katalog i ikonsökvägen:

- Egna ikoner: *Hemkatalog*/`.dt/icons`
- Ikoner för hela systemet: `/etc/dt/appconfig/icons/språk`

- En absolut sökväg till ikonfilen, inklusive fullständigt filnamn.

Du ska bara använda den absoluta sökvägen om ikonfilen inte finns i ikonsökvägen. Om t ex ikonfilen `GameIcon.m.pm` finns i katalogen

/doc/projects, som inte finns i ikonsökvägen, ska värdet i fältet ICON vara /doc/projects/GameIcon.m.pm.

Tabell 12-1 visar en lista över de ikonstorlekar du ska skapa och motsvarande filnamn.

TABELL 12-1 Ikonnamn och storlek på åtgärdsikoner

Storleken i bildpunkter	Bitmappsnamn	Bildpunktsmönster
48 gånger 48	<i>namn.l</i> .bm	<i>namn.l</i> .pm
32 gånger 32	<i>namn.m</i> .bm	<i>namn.m</i> .pm
16 gånger 16	<i>namn.t</i> .bm	<i>namn.t</i> .pm

▼ Ändra en befintlig åtgärdsdefinition

Du kan ändra alla åtgärder som finns tillgängliga i systemet, inklusive inbyggda åtgärder.

Obs! Var försiktig när du ändrar databasen för inbyggda åtgärder. De inbyggda åtgärderna är utformade speciellt för att fungera med skrivbordsprogrammen.

1. Sök reda på definitionen till åtgärd du vill ändra.

Standardsökvägen för åtgärdsdefinitioner är:

- Inbyggda åtgärder: /usr/dt/appconfig/types/*språk*
- Åtgärder för hela systemet: /etc/dt/appconfig/types/*språk*
- Egna åtgärder: *Hemkatalog*/.dt/types

Det kan finnas fler sökvägar i systemet. Om du vill visa en lista med de sökvägar som systemet använder för åtgärder, skriver du kommandot:

```
dtsearchpath -v
```

Systemet använder katalogerna som visas i listan under DTDATABASESEARCHPATH.

2. Om det behövs kopierar du texten till åtgärdsdefinitionen till en ny eller befintlig fil i någon av dessa kataloger:

- Åtgärder för hela systemet: /etc/dt/appconfig/types/*språk*

- Egna åtgärder: *Hemkatalog*/.dt/types
Du måste kopiera inbyggda åtgärder eftersom du inte bör redigera filer i katalogen /usr/dt/appconfig/types/språk.

3. Redigera åtgärdsdefinitionen.

4. När du är färdig med redigeringen sparar du filen.

5. Dubbelklicka på Ladda om åtgärder i programgruppen Skrivbordsverktyg.

Prioritet i åtgärdsdefinitioner

När användarna anropar en åtgärd söker systemet i databasen efter ett matchande åtgärdsnamn. När det finns fler än en åtgärd med samma namn används prioritetsregler för att bestämma vilken åtgärd som ska användas.

- Om inga andra prioritetsregler gäller, baseras prioriteten på definitionens sökväg. Följande lista är rangordnad från högsta till lägsta prioritet:

- Egna åtgärder: *Hemkatalog*/.dt/types
- Lokala åtgärder för hela systemet: /etc/dt/appconfig/types/språk
- Fjärråtgärder för hela systemet: *värddamn*:/etc/dt/appconfig/types/språk.
De fjärranslutna värddatorer som genomsöks är de som listas i programmets sökväg.
- Inbyggda åtgärder: /usr/dt/appconfig/types/ språk

- Inom en viss katalog läses filerna *.dt i alfabetisk ordning.

- Åtgärder som begränsas av ARG_CLASS, ARG_TYPE, ARG_MODE eller ARG_COUNT prioriteras framför obegränsade åtgärder. (Standardvärdet för dessa fyra fält är *.)

När fler än en begränsning gäller är prioritetsordningen (från hög till låg):

- ARG_CLASS
- ARG_TYPE
- ARG_MODE
- ARG_COUNT

När det finns fler än en begränsad ARG_COUNT är prioritetsordningen (från hög till låg):

- Specifikt heltalsvärde *n*
- <*n*
- >*n*
- *

Ta följande del av en åtgärdsdefinition som exempel:

```
ACTION EditGraphics
# EditGraphics-1
{
  ARG_TYPE    XWD
  ...
}
```

```
ACTION EditGraphics
# EditGraphics-2
{
  ARG_COUNT    0
  ...
}
```

```
ACTION EditGraphics
# EditGraphics-3
{
  ARG_TYPE    *
  ...
}
```

När du dubbelklickar på åtgärdsikonen för EditGraphics startar EditGraphics-2 eftersom inga argument har angivits och ARG_COUNT 0 har företräde. När filargument av XWD-typ tillhandahålls, används EditGraphics-1 eftersom det angav XWD som ARG_TYPE. EditGraphics-3 används för alla andra filargument.

Skapa en exekveringssträng för COMMAND-åtgärder

Minimikraven för en COMMAND-åtgärd är två fält: ACTION och EXEC_STRING.

```
ACTION åtgärdsnamn
{
  EXEC_STRING exekveringssträng
}
```

Exekveringssträngen är den viktigaste delen av en COMMAND-åtgärdsdefinition. Den använder syntax som liknar den på kommandoraden i ett terminalfönster, men innehåller ytterligare syntax för hantering av fil- och strängargument.

Generella egenskaper för exekveringssträngar

Exekveringssträngar kan innehålla:

- Filargument och icke-filargument
- Skalsyntax
- Absoluta sökvägar eller namn på körbara filer

Åtgärdsargument

Ett argument är information som är nödvändig för att ett kommando eller program ska kunna köras. Kommandoraden nedan kan du t ex använda för att öppna en fil i Textredigeraren:

```
dtpad filnamn
```

I det här kommandot är *filnamn* ett filargument till kommandot `dtpad`.

Åtgärder kan, på samma sätt som program och kommandon, ha argument. Det finns två typer av data som en `COMMAND`-åtgärd kan använda:

- Filer
- Strängdata

Använda skal i exekveringssträngar

I de flesta fall finns funktioner och deras program i samma system. Eftersom den normala värddatorn för exekvering av en funktion är databasvärden, behövs ingen särskild syntax.

Till exempel:

```
EXEC_STRING    \
/bin/sh -c \
'tar -tvf %(File)Arg_1% 2>&1 | \${PAGER:-more};\
echo "\\n*** Välj Stäng från Fönster-menyn för att stänga***'
```

Skapa en funktion som kör ett fjärrprogram

Om ditt program finns i en katalog som är listad i variabeln `PATH`, kan du ange den körbara filens enkla namn. Om programmet finns någon annanstans måste du använda den absoluta sökvägen till den körbara filen.

Skapa åtgärder som inte använder argument

Använd samma syntax för `EXEC_STRING` som du skulle använda för att starta programmet från kommandoraden.

Exempel

- Den här exekveringssträngen är en del av åtgärden som startar X-klienten `xcutsel`.

```
EXEC_STRING xcutsel
```

- Den här exekveringssträngen startar klienten `xclock` med en digital klocka. Kommandoraden omfattar ett kommandoradsalternativ men kräver inga argument.

```
EXEC_STRING xclock -digital
```

Skapa åtgärder som accepterar släppta filer

Använd följande syntax för filargumentet:

```
%Arg_n%
```

eller

```
%(File)Arg_n%
```

`(File)` är valfritt eftersom argument för `Arg_n` som standard antas vara filer. (I "Tolka filargument som en sträng" på sidan 183 finns information om hur du använder `%(String)Arg_n%-syntaxen`.)

Med den här syntaxen kan användarna släppa ett datafilobjekt på åtgärdsikonen för att starta åtgärden med filargumentet. Den ersätter det *n*:te argumentet på kommandoraden. Filen kan vara en lokal fil eller en fjärrfil.

Exempel

- Den här exekveringssträngen kör `wc -w` med en släppt fil som parametern `-load`.

```
EXEC_STRING wc -w %Arg_1%
```

- I det här exemplet visas en del av en definition för en åtgärd som bara arbetar med katalogargument. När katalogen släpps på åtgärdsikonen, visas en lista över alla filer i katalogen som har läs- och skrivbehörighet.

```
ACTION List_Writable_Files
{
    ARG_TYPE      FOLDER
    EXEC_STRING /bin/sh -c 's -l %Arg_1% | grep rw-'
    ...
}
```

Skapa en åtgärd som frågar efter ett filargument

Använd följande syntax för filargumentet:

```
%(File)"ledtext" %
```

Den här syntaxen skapar en åtgärd som visar en ledtext för ett filnamn när användaren dubbelklickar på åtgärdsikonen.

Den här exekveringssträngen visar t ex en dialogruta som frågar efter filargumentet till kommandot `wc -w`:

```
EXEC_STRING wc -w %(File)"Räkna ord i filen:" %
```

Skapa åtgärder som accepterar eller frågar efter en släppt fil

Använd följande syntax för filargumentet:

```
%(Arg_n)"ledtext" %
```

eller

```
%(File)Arg_n"ledtext" %
```

Syntaxen skapar en åtgärd som:

- Accepterar en släppt fil som filargument.
- Visar en dialogruta som frågar efter ett filnamn när användaren dubbelklickar på åtgärdsikonen.

Den här exekveringssträngen utför t ex `lp -oraw` på en släppt fil. Om åtgärden startas genom att någon dubbelklickar på ikonen visas en dialogruta som frågar efter filnamnet.

```
EXEC_STRING lp -oraw %(Arg_1)"Fil som ska skrivas ut:" %
```

Skapa en åtgärd som frågar efter icke-filargument

Använd följande syntax för icke-filparametern:

```
%"ledtext" %
```

eller

```
%(String)"ledtext" %
```

`(String)` är valfritt, eftersom text inom citattecken normalt tolkas som strängdata. Den här syntaxen visar en dialogruta som frågar efter icke-fildata. Använd inte syntaxen för att fråga efter ett filnamn.

Den här exekveringssträngen kör t ex kommandot `xwd` och frågar efter ett värde som ska läggas till i varje bildpunkt:

```
EXEC_STRING xwd -add %"Lägg till värde:" % -out %(Arg_1)"Filnamn:" %
```

Tolka filargument som en sträng

Använd följande syntax för argumentet:

```
%(String)Arg_n%
```

Den här exekveringssträngen skriver t ex ut en fil med en titel som innehåller filnamnet med hjälp av kommandot `lp -t titelfilnamn`.

```
EXEC_STRING lp -t%(String)Arg_1% %(File)Arg_1"Fil som ska skrivas ut:"%
```

Skalmöjligheter för åtgärder

Ange skalet i exekveringssträngen:

```
/bin/sh -c 'kommando'  
/bin/ksh -c 'kommando'  
/bin/csh -c 'kommando'
```

Exempel

- Den här exekveringssträngen illustrerar en åtgärd som använder skalprogramkoppling.

```
EXEC_STRING /bin/sh -c 'ps | lp'
```

- Det här är en mer komplicerad exekveringssträng som kräver skalbearbetning och accepterar filargument.

```
EXEC_STRING /bin/sh -c 'tbl %Arg_1"Man-sida:"% | troff -man'
```

- Den här exekveringssträngen kräver att argumentet ska vara en komprimerad fil. Åtgärden expanderar filen och skriver ut den med `lp -oraw`.

```
EXEC_STRING /bin/sh -c 'cat %Arg_1 "Fil som ska skrivas ut:"% | \  
uncompress | lp -oraw'
```

- Denna exekveringssträng startar ett skalskript:

```
EXEC_STRING /usr/local/bin/StartGnuClient
```

Skapa COMMAND-åtgärder för flera filargument

Åtgärder kan använda något av följande tre sätt att hantera flera filargument:

- Åtgärden utförs en gång för varje argument: När en `EXEC_STRING` innehåller ett enstaka filargument och flera filargument tillhandahålls genom att flera filer släpps på åtgärdsikonen, körs åtgärden separat för varje filargument.

Om det t ex finns flera filargument för följande åtgärdsdefinition:

```
ACTION DisplayScreenImage  
{  
    EXEC_STRING xwud -in %Arg_1%  
    ...  
}
```

```
}
```

körs åtgärden `DisplayScreenImage` upprepade gånger.

- Åtgärden kan använda två eller flera icke utbytbara filargument: Till exempel:

```
xsetroot -cursor markörfil maskfil
```

som kräver två unika filer i en viss ordning.

- Åtgärden kan utföra samma kommando för varje filargument i tur och ordning. Till exempel:

```
pr fil [fil ...]
```

som skriver ut en eller flera filer.

Skapa en åtgärd för icke utbytbara argument

Använd någon av följande syntaxer:

- Om du vill att åtgärden ska fråga efter filnamnen använder du den här syntaxen för varje filargument:

```
%(File)"ledtext"%
```

Använd en särskild *ledtext* för varje argument.

Följande exekveringssträng frågar efter två filer.

```
EXEC_STRING xsetroot -cursor %(File)"Markörbitmapp:"% \  
%(File)"Maskbitmapp:"%
```

- Om följande syntax används för filargument, accepteras släppta filer:

```
%Arg_n%
```

olika värden på *n* för varje argument. Till exempel:

```
EXEC_STRING diff %Arg_1% %Arg_2%
```

Skapa en åtgärd med utbytbara filargument

Använd någon av följande syntaxer:

- Om du vill skapa en åtgärd som accepterar släppta filer och som skapar ett kommando med formen *kommando fil 1 fil 2 ...*, ska följande syntax för filargument användas:

```
%Args%
```

- Om du vill skapa en åtgärd som accepterar flera släppta filer, eller visar en ledtext för en enstaka fil när du dubbelklickar på den, använder du följande syntax för filargumenten:


```
%Arg_1"ledtext"% %Args%
```

Åtgärden startar kommandot med formen *kommandofil 1 fil 2*

Exempel

- Den här exekveringssträngen skapar en åtgärd som kör:

```
pr fil 1 fil 2
```

med flera filargument.

```
EXEC_STRING pr %Args%
```

- Den här exekveringssträngen skapar en åtgärd som liknar den i föregående exempel, med undantaget att den visar en ledtext när du dubbelklickar (inga filargument).

```
EXEC_STRING pr %Arg_1"Fil(er) som ska skrivas ut:"% %Args%
```

Skapa åtgärder för flera släppta filer

För att acceptera flera argument för släppta filer och köra en kommandorad i formatet:

```
kommando fil 1 fil 2 ...
```

använder du syntaxen:

```
%Args%
```

Exempel

- Den här exekveringssträngen kör skriptet Checkout för flera filer:

```
EXEC_STRING /usr/local/bin/Checkout \  
%Arg_1"Fil som ska checkas ut:"% %Args%
```

- Den här exekveringssträngen kör `lp -oraw` med flera filer.

```
EXEC_STRING lp -oraw %Arg_1"Fil som ska skrivas ut:"% %Args%
```

Fönsterstöd och terminalemulatorer för COMMAND-åtgärder

Det finns flera sätt på vilka åtgärden COMMAND stödjer fönster på skrivbordet.

- Om programmet har ett eget fönster kan åtgärden skrivas så att den inte ger ytterligare fönsterstöd. Det här alternativet används också när en åtgärd kör ett kommando som inte kräver någon direkt användarindata och inte har några utdata.
- Om programmet måste köras i ett terminalfönster kan åtgärden skrivas så att den startar ett terminalfönster och sedan kör programmet. Det finns flera terminalalternativ.

Ange fönsterstöd för åtgärden

Använd fältet `WINDOW_TYPE` för att ange den typ av fönsterstöd som krävs av åtgärden enligt Tabell 12-2 .

TABELL 12-2 Fältet `WINDOW_TYPE` och tillgängligt fönsterstöd

<code>WINDOW_TYPE</code>	Tillgängligt fönsterstöd
<code>NO_STDIO</code>	Inget. Använd <code>NO_STDIO</code> om programmet har ett eget fönster eller om kommandot inte har synliga utdata.
<code>PERM_TERMINAL</code>	Permanent terminalemulatorfönster. Åtgärden öppnar ett terminalfönster som är öppet tills användaren stänger det. Användaren kan skriva data i fönstret. Används tillsammans med kommandon som tar emot vissa indata, skapar vissa utdata och sedan avslutas (t ex <code>ls katalog</code>).
<code>TERMINAL</code>	Temporärt terminalemulatorfönster. Åtgärden öppnar ett terminalfönster som stängs så fort kommandot är färdigt. Används tillsammans med fullskärmskommandon (t ex <code>vi</code>).

Ange kommandoradsalternativ för terminalemulatorn

Använd fältet `TERM_OPTS` i åtgärdsdefinitionen för att ange kommandoradsalternativ för terminalemulatorn.

Följande åtgärd frågar t ex efter värddatorn för exekvering:

```
ACTION OpenTermOnSystemUserChooses
{
    WINDOW_TYPE      PERM_TERMINAL
```

```
EXEC_HOST      %(String)"Fjärrterminal på:"%
TERM_OPTS     -title %(String)"Fönstertitel:"%
EXEC_STRING    $SHELL
}
```

Ange en annan standardterminalemulator

Standardterminalemulator för åtgärder är `dtterm`. Du kan ändra detta till en annan terminalemulator. Standardterminalemulatorn används när åtgärden inte särskilt anger vilken terminalemulator som ska användas.

Terminalemulatorn som används av åtgärderna måste ha dessa kommandoradsalternativ:

- `-rubrik` *fönsterrubrik*
- `-e` *kommando*

Två resurser bestämmer vilken standardterminalemulator som används av åtgärderna:

- Resursen `localTerminal` avgör vilken terminalemulator som används av lokala program.

```
*localTerminal: terminal
```

Till exempel:

```
*localTerminal: xterm
```

- Resursen `remoteTerminal` anger vilken terminalemulator som används av fjärrprogram.

```
*remoteTerminal: värd: terminal [ , värd: terminal...]
```

Till exempel:

```
*remoteTerminal: sysibml:/usr/bin/xterm,syshp2:/usr/bin/yterm
```

Begränsa åtgärder för vissa argument

När du begränsar en åtgärd till en viss typ av argument förfinas åtgärden. Du bör t ex begränsa en åtgärd som startar en läsare för PostScript-filer till endast PostScript-filargument. Med en sådan begränsning returnerar åtgärden ett felmeddelande om en icke-PostScript-fil anges.

Du kan begränsa åtgärder baserat på:

- Filargumentets datatyp.

- Antalet filargument – exempelvis inga argument jämfört med ett eller flera argument. Detta gör att åtgärdsikonen fungerar på olika sätt när någon släpper filer eller dubbelklickar på den.
- Argumentets läs- och skrivläge.

Begränsa åtgärder till en viss datatyp

Använd fältet `ARG_TYPE` för att ange vilka datatyper som åtgärden gäller för. Använd datanamnsattributet.

Du kan ange en lista med datatyper. Avgränsa posterna med kommatecken.

Följande åtgärdsdefinition antar att en gif-datatyp har skapats.

```
ACTION Open_Gif
{
    TYPE      COMMAND
    LABEL     "Visa Gif"
    WINDOW_TYPE NO_STDIO
    ARG_TYPE  Gif
    ICON      xgif
    DESCRIPTION Visar gif-filer.
    EXEC_STRING xgif
}
```

Begränsa åtgärder med antalet argument

Använd fältet `ARG_COUNT` för att ange hur många argument som åtgärden ska acceptera. Giltiga värden är:

* (standard) – hur många argument som helst. Andra värden har företräde före *.

n – vilket icke-negativt heltal som helst, även 0.

>*n* – fler argument än *n*.

<*n* – färre argument än *n*.

Använd `ARG_COUNT` för att ange olika funktioner för åtgärdsikoner, beroende på om användaren dubbelklickar på ikonen eller släpper en fil på den. Se nästa avsnitt, ”Ange olika funktioner för dubbelklickning och släppning”.

▼ Ange metod för dubbelklickning och släppfunktionen

Använd följande procedur för att skapa en åtgärd som accepterar en släppt fil, men inte frågar efter en fil när någon dubbelklickar på åtgärdsikonen.

1. Skapa en åtgärdsdefinition för dubbelklickningsfunktionalitet.

Använd fältet `ARG_COUNT` för att ange 0-argument. Använd en syntax för `EXEC_STRING` som inte accepterar släppta argument.

2. Skapa en andra åtgärdsdefinition för släppfunktionen.

Använd fältet ARG_COUNT för att ange >0-argument. Använd en syntax för EXEC_STRING som accepterar en släppt fil.

Anta att följande två kommandorader kan användas för att starta en redigerare med namnet vedit:

- Starta redigeraren utan filargument:

```
vedit
```

- Starta redigeraren med ett filargument som öppnas som skrivskyddat dokument:

```
vedit -R filnamn
```

Följande två åtgärder skapar släpp- och dubbelklickningsfunktionalitet för en funktion som heter Vedit. Den första åtgärden prioriteras när databasen letar efter en matchning, eftersom ARG_COUNT 0 är mer specifikt än ARG_COUNT * i definitionen av släppfunktionaliteten.

```
# Funktionalitet vid dubbelklickning
ACTION Vedit
{
  TYPE          COMMAND
  ARG_COUNT     0
  WINDOW_TYPE   PERM_TERMINAL
  EXEC_STRING    vedit
}

# Funktionalitet vid släppning
ACTION Vedit
{
  TYPE          COMMAND
  WINDOW_TYPE   PERM_TERMINAL
  EXEC_STRING    vedit -R %Arg_1%
}
```

Begränsa åtgärder baserat på argumentläge

Använd fältet ARG_MODE för att ange läs- och skrivläge för argumentet. Giltiga värden är:

* (standard) – vilket läge som helst

!w – skrivskyddat

w – skrivbar

Skapa åtgärder som kör program i fjärrsystem

När åtgärder och fjärrexeivering behandlas finns det två termer som ofta används:

databasvärd – systemet som innehåller åtgärdsdefinitionen

exekevingsdator – det system där den körbara filen körs

I de flesta situationer finns åtgärder och deras program i samma system. Eftersom databasvärden är standardkörvärd för en åtgärd, krävs ingen särskild syntax.

När värddatorn för exekevring däremot är en annan än databasvärden, måste åtgärdsdefinitionen ange var exekevingssträngen ska köras.

Möjligheten att placera åtgärder och program på olika system är en del av skrivbordets klient/server-arkitektur. Mer ingående information om nätverksprogram finns i "Administrera programtjänster" på sidan 114.

Skapa en åtgärd som kör ett fjärrprogram

Använd fältet EXEC_HOST i åtgärdsdefinitionen för att ange platsen för programmet.

Giltiga värden för EXEC_HOST är:

%DatabaseHost% – den datorn där åtgärden har definierats.

%LocalHost% – den värd där åtgärden anropas (*sessionsservern*).

%DisplayHost% – den värddator som kör X-servern (ej tillåtet för X-terminaler).

%SessionHost% – den värddator där den styrande inloggningshanteraren körs.

värddamn – den namngivna värden. Använd det här värdet för omgivning där åtgärden alltid ska anropas på en viss värddator.

%"*ledtext*"% – frågar användaren efter värddatornamnet varje gång åtgärden startas.

Standardvärdet är %DatabaseHost%, %LocalHost%. Det innebär att när fältet EXEC_HOST utelämnas försöker ändå åtgärden först köra kommandot på den värddator som innehåller åtgärdsdefinitionen. Om detta misslyckas försöker åtgärden köra kommandot på sessionsservern.

Exempel

- Detta fält anger värddatorn ddsyd:

```
EXEC_HOST ddsyd
```

- Detta fält frågar efter värddatornamnet:

```
EXEC_HOST  %"Värd med program:"%
```

- Detta fält anger att åtgärden försöker köra programmet på värddatorn som innehåller åtgärdsdefinitionen. Om detta misslyckas försöker åtgärden köra programmet på värddatorn ddsyd.

```
EXEC_HOST  %DatabaseHost%, ddsyd
```

Använda variabler i åtgärds- och datatypsdefinitioner

Du kan ta med strängvariabler och omgivningsvariabler i filer med åtgärds- och datatyper.

Använda strängvariabler i en åtgärd

En strängvariabeldefinition gäller från definitionens adress till slutet av filen. Det finns inga globala strängvariabler för databasen.

Om en strängvariabel och en omgivningsvariabel har samma namn används strängvariabeln.

Definiera en strängvariabel

- ◆ **Använd syntaxen:**

```
set variabelnamn=värde
```

Variabelnamn kan innehålla alla alfanumeriska tecken samt understreck (_). Varje variabeldefinition måste stå på en separat rad.

Till exempel:

```
set Remote_Application_Server=sysapp  
set Remote_File_Server=sysdata
```

Referera till en strängvariabel

- ◆ **Använd syntaxen:**

```
 ${[]variabelnamn[] }
```

Till exempel:

```
EXEC-HOST  $Remote_Application_Server  
CWD       /net/${Remote_File_Server}/doc/project
```

Använda omgivningsvariabler i åtgärder och datatyper

- ◆ Referera till en omgivningsvariabel med syntaxen:

```
 ${[]variabel[] }.
```

Variabeln ersätts med dess värde när databasen laddas. Om en strängvariabel och en omgivningsvariabel har samma namn används strängvariabeln.

Den här exekveringssträngen skriver exempelvis ut en fil med loginnamnet på försättsbladet.

```
EXEC-STRING lp -t$LOGNAME %(File)Arg_1%
```

Starta åtgärder från en kommandorad

På skrivbordet finns kommandot `dtaction` som används för att köra åtgärder från kommandoraden. Du kan använda `dtaction` för att köra åtgärder från:

- Skript
- Andra åtgärder
- Kommandoraden i en terminalemulator

Syntax för `dtaction`

```
dtaction [-user användarnamn] [-execHost värddamn] \  
  åtgärdsnamn [argument [argument]...]
```

–användare *användarnamn* – ger dig möjlighet att köra åtgärden som en annan användare. Om `dtaction` anropas av en annan användare än *användarnamn*, ombeds användaren ange lösenord.

–execHost *värddamn* – endast för COMMAND-åtgärder. Anger värddatorn där kommandot kommer att köras.

argument – argument till åtgärden, vanligen filargument.

Klienten `dtaction` har ytterligare kommandoradsalternativ. Mer information finns i direkthjälpen (man page) för `dtaction(1)`.

Skapa en åtgärd som kör en annan åtgärd

Använd `dtaction` i åtgärdens `EXEC_STRING`.

Följande åtgärd använder t ex det inbyggda åtgärdsnamnet `Spell` (åtgärden "Kontrollera stavning" i programhanteraren). Den nya åtgärden kör `Textredigeraren` och åtgärden `Spell`, samt visar stavfel i ett separat terminalemulatorfönster.

```
ACTION EditAndSpell
{
    WINDOW_TYPE    NO_STDIO
    EXEC_STRING    /bin/sh -c 'dtaction Spell \
                    %Arg_1"Fil:"%; dtpad %Arg_1%'
}
```

Skapa åtgärder som kör som andra användare

Använd följande syntax i fältet `EXEC_STRING`:

```
EXEC_STRING    dtaction -user användarnamn åtgärdsnamn [filargument]
```

Den nya användaren (*användardnamn*) måste ha visningsbehörighet för systemet via någon av följande mekanismer:

- Läsrättigheter till den inloggade användarens `.Xauthority`-fil
- *Eller* `xhost`-behörighet:

Följande åtgärder ger exempelvis möjligheten att bli superanvändare och redigera parameterfiler.

```
ACTION AppDefaults
{
    WINDOW_TYPE    NO_STDIO
    EXEC_STRING    /usr/dt/bin/dtaction -user root \
                    EditAppDefaults %Arg_1"Fil:"%
}
ACTION EditAppDefaults
{
    WINDOW_TYPE    TERMINAL
    EXEC_STRING    /bin/sh -c 'chmod +w %Arg_1%; \
                    vi %Arg_1%; chmod -w %Arg_1%'
}
```

Skapa språkpassade åtgärder

Sökvägen för datatyper omfattar språkberoende adresser. Skrivbordet använder värdet på LANG för att bestämma på vilka adresser datatypsdefinitioner söks.

Adresser för språkpassade åtgärder

Språkpassade åtgärdsdefinitioner måste placeras i rätt språkberoende katalog i åtgärdens sökväg.

Standardsökväg:

- Egna åtgärder: *Hemkatalog*/.dt/types
- Åtgärder för hela systemet: /etc/dt/appconfig/types/*språk*
- Inbyggda åtgärder: /usr/dt/appconfig/types/*språk*

▼ Språkpassa en befintlig åtgärd

1. **Skapa en fil i rätt språkberoende katalog (t ex i /etc/dt/appconfig/types/japanese).**
2. **Kopiera åtgärdsdefinitionen till den språkberoende konfigurationsfilen.**
Anta att du kopierar åtgärdsdefinitionen från

```
prgm_rot/dt/appconfig/types/C/fil.dt
```

till

```
prgm_rot/dt/appconfig/types/japanese/ny_fil.dt
```

3. **Lägg till fältet LABEL eller ändra det befintliga LABEL-fältet.**

```
LABEL sträng
```

Programhanteraren och filhanteraren använder namnsträngen för att identifiera åtgärdens ikon.

4. **Språkpassa något av följande fält i åtgärdsdefinitionen:**
 - Språkpassade ikoner: ICON
 - Språkpassad objekthjälp: DESCRIPTION

- Språkpassade ledtexter: all text inom citattecken i EXEC_STRING

Skapa åtgärder för ToolTalk-program

Obs! Följande information gäller bara för program som stöder ToolTalk-meddelanden.

Använd åtgärdstypen `TT_MSG` för att skapa en åtgärd som skickar ett ToolTalk-meddelande.

```
ACTION åtgärdsnamn
{
  TYPE TT_MSG
  ...
}
```

Fälten `addressing` och `disposition`

- ToolTalk-fältet `addressing` anges alltid som `TT_PROCEDURE`.
- ToolTalk-fältet `disposition` använder som standard specifikationen i det statiska meddelandemönstret.

Meddelanden som inte stöds

Följande stöds inte av åtgärder av `TT_MSG`-typ:

- Objektorienterade ToolTalk-meddelanden
- Sammanhangsargument i meddelanden

Nyckelord för `TT_MSG`-åtgärder

I Tabell 12-3 visas nyckelord för och användning av `TT_MSG`-åtgärder.

TABELL 12-3 TT_MSG Åtgärdsnyckelord och åtgärdsanvändning

Nyckelord	Användning
TT_CLASS	Definierar värdet på ToolTalk-meddelandefältet class
TT_SCOPE	Definierar värdet på ToolTalk-meddelandefältet scope
TT_OPERATION	Definierar värdet på ToolTalk-meddelandefältet operation
TT_FILE	Definierar värdet på ToolTalk-meddelandefältet file
TT_ARGn_MODE	Definierar värdet på ToolTalk-attributet mode för det <i>n</i> :te meddelandeargumentet
TT_ARGn_VTYPE	Definierar värdet på ToolTalk-attributet vtype för det <i>n</i> :te meddelandeargumentet
TT_ARGn_VALUE	Definierar värdet för det <i>n</i> :te meddelandeargumentet

Skapa datatyper manuellt

Du kan skapa datatypsdefinitioner på två sätt:

- Med verktyget Skapa åtgärd. Skapa åtgärd beskrivs i Kapitel 11.
- Manuellt.

När du skapar en datatyp manuellt måste du redigera en databasfil.

I det här kapitlet beskrivs hur du manuellt skapar datatypsdefinitioner i följande avsnitt:

- "Varför du måste skapa datatyper manuellt" på sidan 197
- "Komponenter i datatypsdefinitionen: Villkor och attribut" på sidan 198
- "Skapa datatyper manuellt: Allmänna steg" på sidan 198
- "Exempel på hur du skapar en egen åtgärd och datatyp" på sidan 200
- "Definiera dataattribut för en datatyp" på sidan 204
- En introduktion till datatyper finns i Kapitel 10.
- Mer information om datatypsdefinitioner finns i direkthjälpen (man) för dtddsfile(4).

Varför du måste skapa datatyper manuellt

Med manuellt skapade datatyper kan du använda alla möjligheter som finns inbyggda i datatypsdefinitionernas syntax.

Du måste skapa en datatyp manuellt om du vill kunna använda följande funktionalitet:

- Sökvägsbaserade datatyper
- Möjlighet att ange andra åtgärder än Open (Öppna) och Print (Skriv ut) som associeras med datatypen.
- Flera namn-, mönster- eller innehållsvillkor för samma datatyp – exempelvis en datatyp som är baserad på filer som heter *.abc eller *.def
- Länkbaserade datatyper

Komponenter i datatypsdefinitionen: Villkor och attribut

En datatypsdefinition består av två separata databasdefinitioner:

- Definitionen DATA_ATTRIBUTES.

Definitionen av DATA_ATTRIBUTES beskriver datatypens namn samt utseende och egenskaper hos filer av den här typen.

- Definitionen DATA_CRITERIA.

Definitionen DATA_CRITERIA beskriver typvillkoren. Varje villkorsdefinition anger vilken DATA_ATTRIBUTES -definition som villkoret gäller för.

Det måste finnas minst en DATA_CRITERIA-definition för varje DATA_ATTRIBUTES-definition. En DATA_ATTRIBUTES-definition kan ha flera associerade DATA_CRITERIA.

Du kan t ex skapa en attributdefinition för PostScript-filer som beskriver hur PostScript-filerna ser ut och fungerar i filhanteraren. Sedan kan du skapa två separata villkor för PostScript-datatypen – ett som är baserat på filnamn och ett som är baserat på filinnehåll.

Mer information finns i "Definiera dataattribut för en datatyp" på sidan 204.

Skapa datatyper manuellt: Allmänna steg

I det här avsnittet beskrivs hur du skapar en datatypskonfigurationsfil.

Konfigurationsfiler för datatyper

Kraven för konfigurationsfiler som innehåller datatypsdefinitioner är:

- Filerna måste använda namngivningskonventionen *namn.dt*
- Filerna måste finnas i databassökvägen. Standardsökväg:

Egna datatyper – *Hemkatalog/.dt/types*

Systemtäckande datatyper – */etc/dt/appconfig/types/språk*

Inbyggda datatyper – */usr/dt/appconfig/types/språk* Den här katalogen ska du inte använda.

Mer information om hur du ändrar databassökvägen finns i ”Ange värde för en sökväg” på sidan 131.

▼ Skapa en datatypsdefinition

1. Öppna en befintlig databasfil eller skapa en ny.

Mer information finns i föregående avsnitt, ”Konfigurationsfiler för datatyper” på sidan 199.

2. Definiera dataattribut för datatypen med följande syntax:

```
DATA_ATTRIBUTES datatypnamn
{
    ICON          bildnamn
    DESCRIPTION   sträng
    attributfält
    attributfält
    ...
}
```

där:

datatypnamn är ett unikt namn som ges till den här datatypen.

bildnamn är ett filnamn eller en sökväg för en ikonfil. Använd filens basnamn. Använd till exempel filnamnet *minbild* för ikonfilerna *minbild.m.pm* och *minbild.t.pm*.

attributfält definierar dataatypens utseende eller beteende.

sträng är en teckensträng. Innehållet blir objekthjälp för datatypen.

Se ”Exempel på hur du skapar en egen åtgärd och datatyp” på sidan 200.

3. Definiera datatypens datavillkor med hjälp av följande syntax:

```
DATA_CRITERIA villkorsnamn
{
    DATA_ATTRIBUTES_NAME   datatypnamn
    villkorsfält
    villkorsfält
}
```

```
}''
```

där:

villkorsnamn är ett unikt namn för den här villkorsdefinitionen

datatypnamn är det namn som används i definitionen DATA_ATTRIBUTES

villkorsfält är ett fält som används för att definiera villkoren för tilldelning av en fil till den här datatypen

Se "Definiera dataattribut för en datatyp" på sidan 204.

4. Spara databasfilen.

5. Skapa datatypens ikoner

Mer information finns i "Ange ikonbild för datatypen" på sidan 202.

6. Skapa de åtgärder som finns i fältet ACTIONS i attributdefinitionen.

7. Dubbelklicka på Ladda om åtgärder i programgruppen Skrivbordsverktyg för att läsa om databasen.

Exempel på hur du skapar en egen åtgärd och datatyp

Anta att det finns ett program i ditt system som heter `xgif` som används för att visa GIF-bilder. Normalt startas programmet med kommandot

```
xgif filnamn
```

Du vill kunna visa GIF-bilder på flera olika sätt:

- Genom att dubbelklicka på en GIF-datafil
 - Genom att markera datafilen och välja programmet från Markera-menyn
1. **Öppna den nya filen *Hemkatalog*/.dt/types/GifViewer.dt för redigering.**
 2. **Skriv datatypsdefinitionerna:**

```
DATA_ATTRIBUTES Gif
{
  DESCRIPTION    Gif-bildfil.
  ICON           GifIcon
  ACTIONS        View
}
DATA_CRITERIA Gif_Criteria
{
  DATA_ATTRIBUTES_NAME    Gif
```



```
NAME_PATTERN    *.gif
}
```

3. Skriv åtgärdsdefinitionen för åtgärden GifViewer:

```
ACTION GifViewer
{
  EXEC_STRING    xgif %(File)Arg_1"Gif-fil som ska visas:"
  WINDOW_TYPE    NO_STDIO
  DESCRIPTION    Dubbelklicka eller släpp en fil för att \
                 visa Gif-filer.
}
```

Eftersom definitionen inte omfattar något ICON-fält använder åtgärden systemets standardikon.

4. Skapa följande tilldelningsåtgärd för att koppla åtgärden GifViewer till åtgärden View i datatypsdefinitionen. Använd fältet ARG_TYPE för att begränsa denna visningsåtgärd till GIF-filer.

```
ACTION View
{
  ARG_TYPE       Gif
  TYPE           MAP
  MAP_ACTION     GifViewer
}
```

5. Spara filen.

6. Dubbelklicka på Ladda om åtgärder i programgruppen Skrivbordsverktyg.

Definiera datatypens dataattribut

DATA_ATTRIBUTES-definitionen definierar utseende och åtgärd för datatypen. Den anger namn på datatypen och ger dig möjlighet att ange:

- Ikonen filhanteraren (fältet ICON)
- Dubbelklickfunktionen och innehållet på Markera-menyn (fältet ACTIONS)
- Datatypens objekthjälp (fältet DESCRIPTION)

Ange ikonbild för datatypen

Använd fältet `ICON` för att ange vilken ikon som ska användas i filhanteraren. Om du inte anger någon ikonbild visas bara namnet i filhanteraren.

Värdet på fältet `ICON` kan vara:

- Ett basfilnamn.

Basfilnamnet är namnet på filen som innehåller ikonen minus filnamnsuffixet för storlek (`m` och `t`) och bildtyp (`bm` och `pm`). Använd till exempel `GameIcon` om filnamnet är `GameIcon.m.pm` eller `GameIcon.t.pm`.

Om du använder basfilnamnet måste ikonfilerna placeras i en katalog i ikonsökvägen:

- Egna ikoner: `Hemkatalog/.dt/icons`
- Systemtäckande ikoner: `/etc/dt/appconfig/icons/språk`

- En absolut sökväg till ikonen, inklusive fullständigt filnamn.

Du ska bara använda den absoluta sökvägen om ikonfilen inte finns i ikonsökvägen. Om till exempel ikonfilen `Spelikon.m.pm` finns i katalogen `/doc/projects`, som inte finns i ikonsökvägen, ska värdet i fältet `ICON` vara `/doc/projects/Spelikon.m.pm`.

I Tabell 13-1 finns en lista över de ikonstorlekar du ska skapa och motsvarande filnamn.

TABELL 13-1 Ikonnamn och storlekar för datatypsikoner

Storleken i bildpunkter	Bitmappsnamn	Bildpunktsmönster
32 gånger 32	<code>namn.m.bm</code>	<code>namn.m.pm</code>
16 gånger 16	<code>namn.t.bm</code>	<code>namn.t.pm</code>

Associera datatyper med åtgärder

Du kan associera datatyper med åtgärder på två sätt:

- I fältet `ACTIONS` i definitionen `DATA_ATTRIBUTES` finns en lista över de åtgärder som visas på filhanterarens Markera-meny. Den första åtgärden i listan är standardåtgärden (dubbeltklick).

- Åtgärder kan begränsas till angivna datatyper med åtgärdsdefinitionens ARG_TYPE-fält.

Följande datatypsdefinition skapar t ex en datatyp för särskilda "läs detta först"-filer (readme-filer) som skapas av systemadministratören och använder namnkonventionen *.rm.

```
DATA_ATTRIBUTES SysReadmeFile
{
  ICON      SysReadMe
  ACTIONS   Open,Respond
}
DATA_CRITERIA SysReadmeFileCriteria
{
  NAME_PATTERN      *.rm
  DATA_ATTRIBUTES_NAME  SysReadmeFile
}
```

Nedan definieras en speciell Respond-åtgärd för filen. Den öppnar en skrivbar kopia av filen i textredigeraren. När filen sparas och textredigeraren avslutas, skickas filen till systemadministratören (adressen sysadmin@utd).

```
ACTION Respond
{
  ARG_TYPE      SysReadmeFile
  EXEC_STRING   /bin/sh -c 'cp %Arg_1% $HOME/readme.temp;\
    chmod +w $HOME/readme.temp; \
    dtpad $HOME/readme.temp; \
    cat $HOME/readme.temp | \
    /usr/bin/mailx sysadmin@utd; \
    rm $HOME/readme.temp'
  WINDOW_TYPE   NO_STDIO
}
```

Dölja filer baserat på datatyp

Om en fil är en osynlig datatyp visas den inte i filhanteraren.

Använd fältet PROPERTIES i definitionen DATA_ATTRIBUTES för att ange att objekt av den här typen ska döljas:

```
PROPERTIES      invisible
```

Ange metoder när en fil manipuleras

I Tabell 13-2 visas DATA_ATTRIBUTES-fälten, som mest används av programmerare. De anger hur filer ska fungera när användaren utför olika skrivbordsaktiviteter.

Mer information finns i *Common Desktop Environment Programmer's Guide*, som ingår i dokumentationen till utvecklingsmiljön.

TABELL 13-2 DATA_ATTRIBUTES Fält och beskrivningar

Fält	Beskrivning
MOVE_TO_ACTION	För behållare som t ex kataloger. Anger en åtgärd som ska köras när en fil flyttas till en behållare av den här datatypen.
COPY_TO_ACTION	För behållare som t ex kataloger. Anger en åtgärd som ska köras när en fil kopieras till en behållare av den här datatypen.
LINK_TO_ACTION	Anger en åtgärd som ska köras när en fil länkas till en fil av den här datatypen.
IS_TEXT	Anger att filer av den här datatypen innehåller text som kan visas i en textruta.
MEDIA	Anger motsvarande typ av ToolTalk-medium.
MIME_TYPE	Anger motsvarande MIME-typ.
X400_TYPE	Anger motsvarande X400-typ.

Definiera dataattribut för en datatyp

Definitionen för DATA_CRITERIA anger de villkor som används för att tilldela en fil eller katalog en datatyp.

Du kan använda villkoren som definieras i Tabell 13-3 för objekttypsbestämning.

TABELL 13-3 DATA_CRITERIA Villkor och beskrivningar

Villkor	Beskrivning
Filnamn	Filnamnet måste matcha ett angivet mönster. Använd fältet NAME_PATTERN.
Filens sökväg	Sökvägen måste matcha ett angivet mönster. Använd fältet PATH_PATTERN.
Fil innehåll	En angiven del av filens innehåll måste matcha angivna data. Använd fältet CONTENT.

TABELL P-3 DATA_CRITERIA Villkor och beskrivningar (forts.)

Villkor	Beskrivning
Filläge	Filen måste ha de angivna behörigheterna (läsa, ändra, använda, katalog). Använd fältet <code>MODE</code> .
Symboliska länkar	Typen beror på filen som objektet har länkats till.

Du kan använda fler än ett villkor för en datatyp. Du bör däremot inte använda villkoren `NAME_PATTERN` eller `PATH_PATTERN` i samma datatyp.

Namnbaseerade datatyper

Använd fältet `NAME_PATTERN` för att ange namngivningskrav. Fältvärdet kan innehålla följande jokertecken:

? – matchar endast ett tecken

* – matchar noll eller flera tecken i följd

[*cc...*] – matchar vilket som helst av tecknen (*c*) inom hakparenteser

[*c-c*] – matchar vilket som helst av tecknen i intervallet *c* till *c*

Exempel

- Följande datatypsdefinition skapar en datatyp baserat på filnamnet. Filnamnet måste börja med `QS` och sluta med `.doc`.

```
DATA_ATTRIBUTES QS_Doc
{
    DESCRIPTION    Den här filen innehåller ett dokument för QS-\
                  projektet.
    ICON           Word_Doc
    ACTIONS        Open
}

DATA_CRITERIA QS_Doc_Criteria
{
    NAME_PATTERN   QS*.doc
    DATA_ATTRIBUTES_NAME QS_Doc
}
```

- Följande definition skapar en datatyp för kataloger med namnet `Demo_n`, där *n* kan motsvaras av ett tal mellan 0 och 9.

```
DATA_ATTRIBUTES Demo_directory
{
```

```

DESCRIPTION    Det här är en katalog. Dubbelklicka för att öppna den.
ICON           Demo
ACTIONS        OpenInPlace,OpenNewView
}

DATA_CRITERIA Demo_directory_criteria
{
  NAME_PATTERN    Demo_[0-9]
  MODE            d
  DATA_ATTRIBUTES_NAME    Demo_directory
}

```

Sökvägsbaserade datatyper

Använd fältet `PATH_PATTERN` för att ange sökväg. Du kan använda samma jokertecken som med `NAME_PATTERN`.

Följande datatyp använder ett villkor baserat på sökvägen.

```

DATA_ATTRIBUTES Project_Graphics
{
  DESCRIPTION    Grafikfil för QS-projektet. Dubbelklicka på \
                 ikonerna för att visa bilden.
  ICON           QSgraphics
}
DATA_CRITERIA Project_Graphics_Criteria
{
  DATA_ATTRIBUTES_NAME    Project_Graphics
  PATH_PATTERN             */projects/QS/graphics/*
}

```

Datatyper baserade på namn och sökväg

Om du vill skapa en datatyp baserat på både filnamn och sökväg, tar du med namnet i värdet `PATH_PATTERN`. Du kan inte använda både `NAME_PATTERN` och `PATH_PATTERN` i samma villkorsdefinition.

Exempel

- Datatypen `QS_Source_Files` som definieras nedan gäller för alla filer med namnet `appn.c` (där *n* ligger inom intervallet 1 till 9) som finns i underkataloger till `*/projects/QS`.

```

DATA_ATTRIBUTES QS_Source_Files
{
  ...
}
DATA_CRITERIA QS_Source_Files_Criteria
{
  PATH_PATTERN    */projects/QS/*/app[1-9].c
  DATA_ATTRIBUTES_NAME    QS_Source_Files
}

```

- Följande datatyp gäller för alla filer i katalogen /doc/project1 med namnet *chm.xxx*, där *n* är ett tal mellan 0 och 9 och *xxx* är ett valfritt filnamnsuffix som är tre tecken långt.

```
DATA_ATTRIBUTES ChapterFiles
{
  DESCRIPTION  Kapitelfil för projektdokumentet.
  ICON        chapter
  ACTIONS     Edit,Print
}

DATA_CRITERIA Chapter_Criteria
{
  PATH_PATTERN  /doc/project1/ch[0-9][0-9].???
  DATA_ATTRIBUTES_NAME  ChapterFiles
}
```

Använda fillägen som typvillkor

Använd fältet `MODE` för att ange vilka behörigheter som krävs.

Lägesvillkor används normalt i kombination med namnbaserade, sökvägsbaserade eller innehållsbaserade datatyper. Med dessa kan du begränsa en datatyp till en fil eller katalog, eller ange om det krävs behörighet för att läsa, skriva eller använda.

Fältet `MODE` kan innehålla logiska operatorer (Tabell 13-4) och tecken (Tabell 13-5).

TABELL 13-4 `MODE` Logiska operatorer

Operator	Beskrivning
!	NOT, logisk operator
&	AND, logisk operator
	OR, logiskt eller

TABELL 13-5 MODE Fälttecken och beskrivningar

Tecken	Beskrivning
f	Datotypen gäller bara filer
d	Datotypen gäller bara kataloger
r	Filen kan läsas av alla användare
w	Filen kan ändras av alla användare.
x	Filen kan köras av alla användare.
l	Filen är en länk

Standardinställningen för ett behörighetsläge är att läget inte spelar någon roll.

Exempel

- Följande fält begränsar datotypen enligt följande:

f&!w – skrivskyddade filer

!w – skrivskyddade filer och kataloger

f&x – körbara filer

f&r&x – filer som både kan ändras och köras

x|!w – filer som är körbara eller skrivskyddade

- Följande datatypsdefinition skapar en datatyp för skrivskyddade, icke körbara filer vilkas filnamn följer namnkonventionen *.doc. Den förutsätter att det redan har skapats en visningsåtgärd för datotypen.

```
DATA_ATTRIBUTES ReadOnlyDocument
{
    ICON          read_only
    DESCRIPTION   Det här dokumentet är skrivskyddat. Dubbel-\  
                klicka för att köra redigeraren med en \  
                skrivskyddad kopia av filen.
    ACTIONS       View
}

DATA_CRITERIA ReadOnlyDocument_Criteria
{
    NAME_PATTERN  *.doc
    MODE          !d&!x&!w
    DATA_ATTRIBUTES_NAME  ReadOnlyDocument
}
```



```
}
```

Innehållsbaserad datatypsbestämning

Använd fältet `CONTENT` för att ange datatyp baserat på filens innehåll. Innehållsbaserad datatypsbestämning kan användas i kombination med namn- eller platsbaserad datatypsbestämning.

Typbestämningen kan baseras på texten i filen eller på dess numeriska innehåll. Numret på filens första byte är 0.

- Typbestämning med hjälp av filens innehåll använder syntaxen:

```
CONTENT startbyte string sträng
```

- För numeriskt innehåll använder du syntaxen:

```
CONTENT startbyte byte tal  
CONTENT startbyte short tal  
CONTENT startbyte long tal
```

- För mappinnehåll använder du syntaxen:

```
CONTENT 0 filename "filnamn"
```

Använd den vanliga C-beteckningen för oktala (inledande `o`) och hexadecimala (inledande `0x`) tal.

Obs! Om du använder en innehållsbaserad datatyp kan systemets prestanda sjunka. Använd därför namn- eller sökvägsbaserade datatyper när det är möjligt.

Datatypen `Writable_Wingz` gäller för alla filer med skrivbehörighet som innehåller strängen `WNGZ` i början av filen.

```
DATA_ATTRIBUTES Writable_Wingz  
{  
  ...  
}  
  
DATA_CRITERIA Writable_Wingz_Criteria  
{  
  CONTENT      0 string WNGZ  
  MODE         w&!d  
  DATA_ATTRIBUTES_NAME Writable_Wingz  
}
```

▼ Skapa en datatyp med flera oberoende villkor

Du kan skapa en datatyp med flera oberoende villkor, dvs filen tilldelas datatypen om den motsvarar minst ett av villkoren.

1. Skapa en `DATA_ATTRIBUTES`-definition för datatypen.

2. Skapa en DATA_CRITERIA-definition för varje villkor.

Använd fältet DATA_ATTRIBUTES_NAME för att ansluta alla villkor till samma DATA_ATTRIBUTES-definition.

Följande definitioner skapar datatypen Mif. Typen baseras på namn eller innehåll.

```
DATA_ATTRIBUTES Mif
{
    ICON          Frame
    ACTION_LIST   Open,Print
}

DATA_CRITERIA Mif_Name_Criteria
{
    DATA_ATTRIBUTES_NAME Mif
    NAME_PATTERN          *.mif
}

DATA_CRITERIA Mif_Content_Criteria
{
    DATA_ATTRIBUTES_NAME Mif
    CONTENT               1 string MIFFile
}
```

Skapa språkpassade datatyper

Sökvägen för datatyper omfattar språkberoende adresser. Skrivbordet använder värdet på LANG för att bestämma på vilka adresser datatypsdefinitioner söks.

Adresser för språkpassade datatyper

Språkpassade datatypsdefinitioner måste placeras i rätt språkberoende katalog i åtgärdssökvägen.

Standardsökväg:

- Egna åtgärder: *Hemkatalog*/.dt/types
- Åtgärder för hela systemet: /etc/dt/appconfig/types/*språk*
- Inbyggda åtgärder: /usr/dt/appconfig/types/*språk*

▼ Språkpassa en datatyp

1. Skapa en fil i rätt språkberoende katalog (t ex i /etc/dt/appconfig/types/japanese).

2. Kopiera datatypsdefinitionen till den språkberoende konfigurationsfilen.
3. Språkänpassa ett eller flera fält i datatypsdefinitionen.

Skapa skrivbordsikoner

Skrivbordsikoner kan kopplas till:

- Funktionsfiler och datatyper i filhanteraren och programhanteraren
- Ikoner i huvudgruppen
- Minimerade programfönster
- Grafik som används av program, till exempel paletter och verktygsfält.
- Arbetsytans bakgrund
- "Ikonbilsfiler" på sidan 213
- "Associera ikoner" på sidan 216
- "Rekommendationer för hur du skapar ikoner" på sidan 219

Obs! Dokumentationen för utvecklingsmiljön innehåller ytterligare information om skrivbordsikoner. Se kapitel 4, "Visual Design," i *Common Desktop Environment: Style Guide and Certification Checklist*.

Ikonbilsfiler

Om skrivbordet ska använda ikonbilder måste filerna:

- Ha rätt format
- Använda rätt filnamnsregler
- Använda skrivbordets storlekskonventioner
- Finnas i en katalog i ikonsökvägen

- Anropas från skrivbordet med rätt syntax. Om du t ex skapar en ny kontroll för huvudgruppen ska du använda `ICON`-fältet i definitionen för huvudgruppen för att ange den ikonbild som ska användas för kontrollen.

Ikonfilformat

För en färgskärm kan du använda ikonfiler i formatet X-pixmap (XPM), som normalt har suffixet `.pm`. Annars använder du filer i formatet X-bitmap (XBM), som normalt har suffixet `.bm`. Om genomskinlighet används i pixmap-filen skapas en maskfil (`_m.bm`) när filen `.bm` skapas. Mer information om hur skrivbordssystemet hittar filerna finns i "Ikonsökväg" på sidan 137 .

Ikonfilnamn

Alla ikoner och bakgrundsbilder lagras som separata filer. Normalt visar basdelen i filnamnet att den innehåller en ikon. En ikon kan t ex refereras till med hjälp av namnet `mail` när filen lagras som:

```
/usr/dt/appconfig/icons/språk/mail.l.pm
```

Namngivningskonventionen med tillagda suffix grupperar ikoner efter storlek och typ. Ikonnamn för skrivbordskomponenter finns i följande allmänna format:

grundnamn.storlek.format

Eller

grundnamn.format

där:

grundnamn är bildens grundnamn som används som referens till bilden

storlek är en bokstav som visar storleken: `l` (stor) `m` (normal) `s` (liten) `t` (pytteliten)

format anger filformat: `pm` (bildpunktsbeskrivning) `bm` (bitmapp)

Storlekskonventioner

Tabell 14-1 visar rekommenderade bildpunktsstorlekar för skrivbordsikoner.

TABELL 14-1 Ikonstorlekar och filnamn

Ikonstorlek	Bitmappsnamn	Bildpunktsmönster
16 gånger 16 (pytteliten)	<i>namn</i> .t.bm	<i>namn</i> .t.pm
24 gånger 24 (liten)	<i>namn</i> .s.bm	<i>namn</i> .s.pm
32 gånger 32 (normal)	<i>namn</i> .m.bm	<i>namn</i> .m.pm
48 gånger 48 (stor)	<i>namn</i> .l.bm	<i>namn</i> .l.pm

Tabell 14-2 visar ikonstorlekar som används av skrivbordskomponenter. I vissa fall bestäms ikonstorleken av upplösningen på skärmen.

TABELL 14-2 Skrivbordets komponenter och deras ikonstorlekar

Skrivbordskomponent	Hög upplösning	Normal upplösning	Låg upplösning
Filhanteraren och programhanteraren (namn och ikon)	medelstor	medelstor	normal
Filhanteraren och programhanteraren (namn och liten ikon)	pytteliten	pytteliten	pytteliten
Huvudgruppens huvudkontroller	stor	stor	normal
Huvudgruppens panelundermenyer	medelstor	normal	pytteliten
Huvudgruppens arbetsytetekopplare	liten	liten	pytteliten
Minimerade fönster	stor	stor	normal

Om du t ex anger en ikon med namnet mail för en datatyp används ikonbilden *mail.t.pm* (förutsatt att du har en färgskärm och har angett filhanterarens inställningar till små ikoner).

Ikonsökväg

Skrivbordet söker efter en ikonfil eller en bild genom att leta efter filen i en lista över kataloger. Denna lista kallas *ikonsökväg* och bestäms av värdet på ett flertal omgivningsvariabler. Mer information om vilka variabler som används och hur de läggs ihop för att skapa ikonsökvägen finns i "Ikonsökväg" på sidan 137.

Standardsökväg:

- Inbyggda ikoner: `/usr/dt/appconfig/icons/språk`
- Ikoner för hela systemet: `/etc/dt/appconfig/icons/språk`
- Egna ikoner: `Hemkatalog/.dt/icons`

Komma åt ikoner via nätverket

Skrivbordet kan även använda ikoner i fjärrsystem. Mer information om hur du skapar en ikonserver finns i "Konfigurera databas-, ikon- och hjälptjänster" på sidan 117.

Associera ikoner

För att objekt snabbare ska kunna kännas igen associerar du ikoner med:

- Funktioner och datatyper
- Kontroller i huvudgruppen och på panelundermenyer
- Minimerade programfönster

Ange ikonfiler

För ikoner som används i funktioner, datatyper och i huvudgruppen eller panelundermenyer, anger du bara ikonens basnamn (inga suffix). Rätt suffix läggs till automatiskt utifrån skärmutplösning, färgstöd och alternativ för filhanterarfönstret (t ex Små ikoner).

För att åsidosätta sökvägen anger du fullständig sökväg och namn på ikonen.

▼ Associera en ikon med en funktion eller datatyp

1. Ange ikonen med fältet `ICON`.

Om du följer lämplig namngivningskonvention för ikonfiler, anger du bara ikonens basnamn. Rätt ikon visas baserat på skärmens upplösning och färgstöd.

2. Skapa följande ikonstorlekar:

- Ågärder: stor, normal och pytteliten
- Datatyper: normal och pytteliten

Exempel på en funktionsdefinition

Följande exempel är funktionsdefinitioner för att köra ritverktyget Island Paint. Ikonerna `Ipaint.l` och `Ipaint.s` associeras med åtgärden:

```
ACTION IslandPaintOpenDoc
{
  WINDOW_TYPE    NO-STDIO
  ICON           Ipaint
  EXEC_STRING     /usr/bin/IslandPaint %Arg_1"Fil som ska öppnas:"%
}
```

Om du använder färgikoner lägger skrivbordssystemet först till `.pm` vid sökning efter ikonfilen. Annars (eller om ingen matchning hittades med `.pm`) lägger skrivbordssystemet till `.pm`.

Exempel på datatypsdefinition

Följande datatypsdefinition associerar ikonerna `comprsd.l` och `comprsd.s` med komprimerade filer:

```
DATA_ATTRIBUTES COMPRESSED
{

  ICON      comprsd
  ACTIONS   Uncompress
  DESCRIPTION En KOMPRIMERAD fil har komprimerats av \
              kommandot 'compress' för att ta upp mindre plats .

}
```

▼ Visa en ikon i huvudgruppen

1. Ange bildens namn i fältet `ICON`.

Om kontrollen övervakar en fil (`MONITOR_TYPE` anges som `mail` eller `file`) använder du fältet `ALTERNATE_ICON` för att ange vilka ikoner som ska användas när en ändring upptäcks.

Du kan också ange animeringar för ikoner till knappar och målområden.

2. Skapa följande ikonstorlekar:

- Huvudgruppen och panelundermenyer: stor, normal och pytteliten

- val för arbetsyta: liten

Exempel

Följande ikon ändrar utseende när en fil med namnet `report` placeras i katalogen `/doc/ftp/pub/`. När filen inte hittas kommer `NoReport.pm`-ikonen att visas. När filen hittas kommer `Report.pm` att visas.

```
CONTROL MonitorReport
{
  CONTAINER_NAME    behållarnamn
  TYPE              ICON
  MONITOR_TYPE      file
  FILE_NAME          /doc/ftp/pub/report
  ICON               NoReport
  ALTERNATE_ICON    Report
}
```

▼ Associera en ikon med ett programfönster

1. Ange resursen `iconImage` för arbetsytahanteraren så här:

```
Dtwm*klientnamn*iconImage: ikonfilsnamn
```

För att bestämma rätt värde på `klientnamn` öppnar du programhanteraren och dubbelklickar på Fönsteregenskaper i programgruppen Skrivbordsverktyg. När du markerar ett fönster visas dess egenskaper. Egenskapen `WM_CLASS` visar fönstrets klassnamn inom citattecken.

Mer information om hur du anger resurser finns i "Ange programresurser" på sidan 265.

2. Välj Starta om arbetsytahanteraren på Arbetsyta-menyn.

För att kontrollera att ikonen har känts igen av arbetsytahanteraren minimerar du fönstret vars ikon du försöker ändra.

Obs! I vissa program kan inte standardfönsterikonen åsidosättas.

▼ Använda filhanteraren som ett ikonfönster

1. Kopiera filen `/usr/dt/examples/språk/IconBrowse.dt` till katalogen **Hemkatalog**/`.dt/types/Iconbrowse.dt`.

2. Öppna programhanteraren och dubbelklicka på Ladda om funktioner i programgruppen Skrivbordsverktyg.

När du växlar till en katalog som innehåller ikoner (.bm- och .pm-filer) visas varje ikon intill sitt namn. Om du t ex växlar till katalogen `/usr/dt/appconfig/icons/språk`, visas många av skrivbordsikonerna.

Obs! Om du aktiverar ikonfönstret i ett system med lite minne kan det hända att filhanteraren visar kataloger mycket långsammare. Bilder som är större än 256 x 256 bildpunkter visas inte i standardkonfigurationen.

Så här avaktiverar du ikonfönstret:

1. Ta bort din egen kopia av filen `IconBrowse.dt`.
2. Öppna programhanteraren och dubbelklicka på Ladda om funktioner i programgruppen Skrivbordsverktyg.

Rekommendationer för hur du skapar ikoner

Använd ett tema för närliggande ikoner. Om du t ex skapar ikoner för ett program bör programikonen och ikonen för datafilerna likna varandra.

Kontrollera att den tvåfärgsversion av färgikonen som du skapar är giltig. Om ikonen visas på en monokrom skärm eller gråskaleskärm (eller om det inte finns tillräckligt med tillgängliga färger) visas ikonen automatiskt i tvåfärgsformatet.

Om du vill sänka antalet färger kan du begränsa färgintervallet i en ikon genom att använda färgerna från skrivbordet. (Ikoner som är skapade med ikonredigeraren kommer endast att använda skrivbordsfärger.)

Information om vilka storlekar som används av skrivbordets komponenter finns i Tabell 14-1.

Hur färger används

Skrivbordet använder en palett med 22 färger, bl a:

- Åtta statiska gråskalor
- Åtta statiska färger: röd, blå, grön, cyan, magenta, gul, svart och vit

- Sex dynamiska färger: förgrund, bakgrund, övre skugga, undre skugga, markering och genomskinlig

Med den här paletten skapar du snygga och lättlästa ikoner som inte inkräktar på de färgresurser som behövs av andra program. För de flesta ikoner på skrivbordet används gråskalor som betonas med färg.

Den genomskinliga färgen använder du när du vill skapa ikoner som verkar icke-rektangulära, eftersom färgen bakom ikonerna skiner igenom.

Särskild anpassning av huvudgruppen

Användarna kan anpassa huvudgruppen och dess meny och använda funktionen Installationsikon på panelundermenyerna.

I det här kapitlet beskrivs hur du anpassar huvudgruppen genom att skapa och redigera konfigurationsfiler.

- "Konfigurationsfiler för huvudgruppen" på sidan 221
- "Administrera anpassning av användargränssnitt" på sidan 224
- "Hur huvudgruppsdefinitionen är organiserad" på sidan 225
- "Anpassa huvudpanelen" på sidan 228
- "Skapa och ändra panelundermenyer" på sidan 233
- "Definiera kontroller i huvudgruppen" på sidan 237
- "Anpassa området för växling av arbetsyta" på sidan 243
- "Allmän konfiguration av huvudgruppen" på sidan 245
- Referensinformation om huvudgruppens kontroller och konfiguration finns i direkt hjälpsavsnittet (man page) för `dtfpfile(4X)`.
- Referensinformation om arbetsyتهanteraren finns i avsnitten om `dtwm(1)` och `dtwmrc(4)` i direkt hjälpen (man page).

Konfigurationsfiler för huvudgruppen

Huvudgruppen definieras i en databas med konfigurationsfiler.

Med konfigurationsfilerna anpassar du huvudgruppen. Vissa ändringar kan du bara göra genom att redigera en konfigurationsfil, bl a:

- Lägga till en ny kontrollposition på huvudpanelen
- Lägga till speciella typer av kontroller, t ex klientfönster
- Ändra vissa standardbeteenden, t ex om enkel- eller dubbelklickning ska användas för huvudgruppens kontroller

För att konfigureringen ska vara så flexibel som möjligt kan dessa filer vara egna eller systemtäckande eller finnas i andra system.

Huvudgruppen skapas och hanteras av arbetsyتهanteraren.

Standardhuvudgruppens konfigurationsfil

Standardhuvudgruppen definieras i konfigurationsfilen

`/usr/dt/appconfig/types/språk/dtwm.fp`.

Den här filen får inte ändras.

Sökvägen till huvudgruppens konfigurationsfiler

Huvudgruppsdefinitionen kan distribueras mellan ett stort antal filer som finns lagrade lokalt eller i fjärrsystem.

Filer som används för att definiera huvudgruppen måste uppfylla följande krav:

- Filnamnet måste avslutas med `.fp`, t ex `mail.fp`.
- Filen måste finnas i åtgärdsdatabasens sökväg

Sökvägen till standardåtgärdsdatabasen omfattar följande kataloger, vilka söks i denna ordning:

Egna anpassningar – *Hemkatalog*/`.dt/types`

Anpassningar för hela systemet – `/etc/dt/appconfig/types/språk`

Inbyggda paneler och kontroller – `/usr/dt/appconfig/types/språk`

Ytterligare en katalog, *Hemkatalog*/`.dt/types/fp_dynamic`, används för egna anpassningar som har utförts med hjälp av användargränssnittet. Använd inte den här katalogen för manuella anpassningar.

Sökvägen till åtgärdsdatabasen kan innehålla ytterligare kataloger som har lagts till för att konfigurera det system som ska användas i nätverk. Ytterligare fjärradresser läggs speciellt till när systemet konfigureras för åtkomst av en programserver. Mer information finns i "Databassökväg (åtgärd/datatyper)" på sidan 135.

Hur huvudgruppen sätts samman: prioritetsregler

Huvudgruppen sätts ihop ifrån alla konfigurationsfiler som finns i åtgärdsdatabasens sökväg.

När det uppstår en konflikt mellan olika delar av definitionen, bestämmer prioritetsreglerna vilken definition som ska användas. Två komponenter krockar med varandra när de:

- Har samma kontrollnamn, CONTAINER_NAME och CONTAINER_TYPE.
- Konkurrerar om samma placering (genom att de har olika namn men samma CONTAINER_NAME, CONTAINER_TYPE och POSITION_HINTS).

Huvudgruppen använder följande prioritetsregler:

- Om olika komponenter har samma kontrollnamn, behållarnamn (container_name) och behållartyp (container_type), används den komponent som läses först

Om t ex både en systemtäckande och en inbyggd kontroll innehåller de här fälten men i övrigt är olika:

```
CONTROL TextEditor
{
  CONTAINER_TYPE    BOX
  CONTAINER_NAME    Top
  ...
}
```

prioriteras den systemtäckande kontrollen.

- Om två olika komponenter konkurrerar om samma position, placeras de i den ordning de läses.

Om en användare t ex skapar en ny egen kontroll för huvudpanelen (CONTAINER_TYPE och CONTAINER_NAME Top) och tilldelar den POSITION_HINTS 5, flyttar den egna kontrollen den inbyggda kontrollen och alla andra kontroller med högre positionsnummer ett steg åt höger.

Obs! När du ändrar en kontroll genom att skapa en ny kontroll för hela systemet eller egen version av den, måste den nya kontrolldefinitionen ange samma kontrollnamn, CONTAINER_NAME och CONTAINER_TYPE. I annat fall visas både den nya kontrollen och den befintliga.

Skapa huvudgruppsfiler dynamiskt

När användaren anpassar huvudgruppen med kontrollen Installationsikon och dess undermenyer, sparas filer i katalogen *Hemkatalog/.dt/types/fp_dynamic..*

Huvudgruppen skapar ytterligare en fil, *Hemkatalog/.dt/sessions/dtwmfp.session*, som används för att spara och återställa status för den anpassade huvudgruppen för varje session.

Administrera anpassning av användargränssnitt

Användare kan använda snabbmenyerna och kontrollerna för ikoninstallation på huvudgruppen när de vill göra betydande anpassningar av huvudgruppen.

I det här avsnittet beskrivs hur du:

- Förhindrar vissa egna anpassningar. Du kanske vill göra det omöjligt för användarna att ta bort en kontroll.
- Ångrar egna anpassningar. En användare kanske vill att du ska återställa en kontroll som har tagits bort av misstag.

▼ Förhindra egna anpassningar

1. **Om kontrollen är en inbyggd kontroll kopierar du dess definition från `/usr/dt/appconfig/types/språk/dtwm.fp` till `/etc/dt/appconfig/types/språk/namn.fp`.**

2. **Lägg till följande rad i kontrolldefinitionen:**

```
LOCKED    True
```

▼ Återställa en borttagen kontroll eller panelundermeny

Åtgärden Återställ huvudgrupp i programgruppen Skrivbords_kontroller tar bort alla anpassningar av användargränssnittet. Användarna kan använda den här åtgärden för att ta bort alla egna anpassningar som har gjorts med huvudgruppens menyer.

Använd proceduren nedan för att återställa individuella kontroller.

- ◆ **I katalogen *Hemkatalog*/`.dt/types/fp_dynamic` tar du bort filen som skapades när användaren tog bort kontrollen. Kontrollen har samma namn som den originalkontroll som togs bort.**

Om användaren t ex tog bort kontrollen för Ikonredigeraren, innehåller en fil i katalogen `fp_dynamic`:

```
CONTROL IconEditor
{
  ...
```



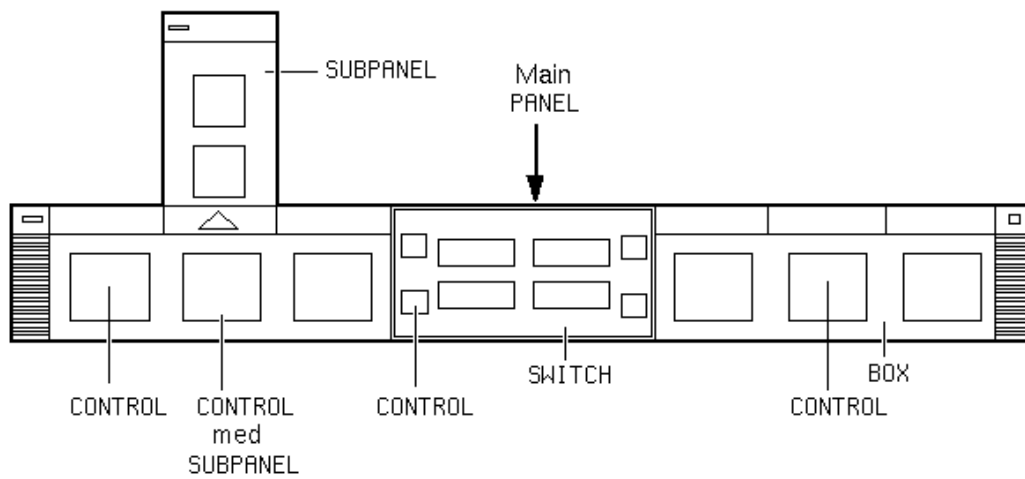
```
    DELETE    True
}
```

När användaren tar bort en panelundermeny skapas en separat dynamisk fil för panelundermenyn och för varje kontroll på den.

Hur huvudgruppsdefinitionen är organiserad

Huvudgruppen skapas genom att definitioner för dess komponenter läggs ihop. Var och en av dessa komponenter har en obligatorisk syntax som definierar var komponenten placeras i huvudgruppen, hur den ser ut och hur den fungerar.

Komponenter i huvudgruppen



Figur 15-1 Komponenter i huvudgruppen

Huvudgruppen sätts samman från utsidan och inåt:

- PANEL är behållaren på huvudgruppens översta nivå (den överordnade).
- En PANEL är en behållare för en eller flera rutor (BOX).
- En BOX är en behållare som innehåller en eller flera kontroller (CONTROL).

Det finns två specialtyper av behållare:

- En panelundermeny (SUBPANEL) är kopplad till en viss kontroll (kontrollen är behållaren för panelundermenyn). Panelundermenyer öppnas när du klickar på den kontroll som de hör till.
- En arbetsyväxlare (SWITCH) innehåller knappar för att byta arbetsyta och andra kontroller.

Allmän syntax för huvudgruppsdefinitionen

Varje komponent i huvudgruppen definieras separat med syntaxen:

```
COMPONENT namn
{
  KEYWORD      värde
  KEYWORD      värde
  ...
}
```

Vissa nyckelord är obligatoriska och andra valfria. Mer information finns i direkthjälpen (man) för dtfpfile(4X).

PANEL-definition

En PANEL är komponenten på den översta nivån. Dess definition innehåller:

- huvudgruppens namn
- Fält som beskriver allmänt utseende och allmänna metoder för hela huvudgruppen

```
PANEL front_panel_name
{
  KEYWORD      värde
  KEYWORD      värde
  ...
}
```

Front_panel_name är ett unikt namn på huvudgruppen. Standardnamnet är "huvudgruppen".

BOX-definitioner

Definitionen för en BOX beskriver:

- BOX-namnet
- Vilken PANEL som rutan ingår i (CONTAINER_NAME)
- Rutans placering i PANELEN (POSITION_HINTS)

- Fält som beskriver utseende och metod för hela rutan

```
BOX boxnamn
{
  CONTAINER_NAME      front_panel_name
  POSITION_HINTS       position
  KEYWORD              värde
  KEYWORD              värde
  ...
}
```

CONTROL, definitioner

Definitionen för en CONTROL beskriver:

- CONTROL-namnet
- Om kontrollen finns i en ruta (BOX), panelundermeny (SUBPANEL) eller arbetsytväxlare (SWITCH) (CONTAINER_TYPE)
- Vilken ruta, panelundermeny eller arbetsytväxlare kontrollen finns i (CONTAINER_NAME)
- Kontrollens (CONTROL) placering i rutan (BOX) (POSITION_HINTS)
- Fält som beskriver utseende och metod för kontrollen

```
CONTROL kontrollnamn
{
  CONTAINER_TYPE      BOX eller SUBPANEL eller SWITCH
  CONTAINER_NAME      boxnamn eller underpanelnamn eller parameternamn
  TYPE                kontrolltyp
  POSITION_HINTS       position
  KEYWORD              värde
  KEYWORD              värde
  ...
}
```

SUBPANEL, definitioner

Definitionen för en panelundermeny (SUBPANEL) beskriver:

- SUBPANEL-namnet
- Namnet på kontrollen som panelundermenyn hör till (CONTAINER_NAME)
- Fält som beskriver utseende och metod för panelundermenyn

```
SUBPANEL underpanelnamn
{
  CONTAINER_NAME      kontrollnamn
  KEYWORD              värde
  KEYWORD              värde
  ...
}
```

SWITCH, definitioner

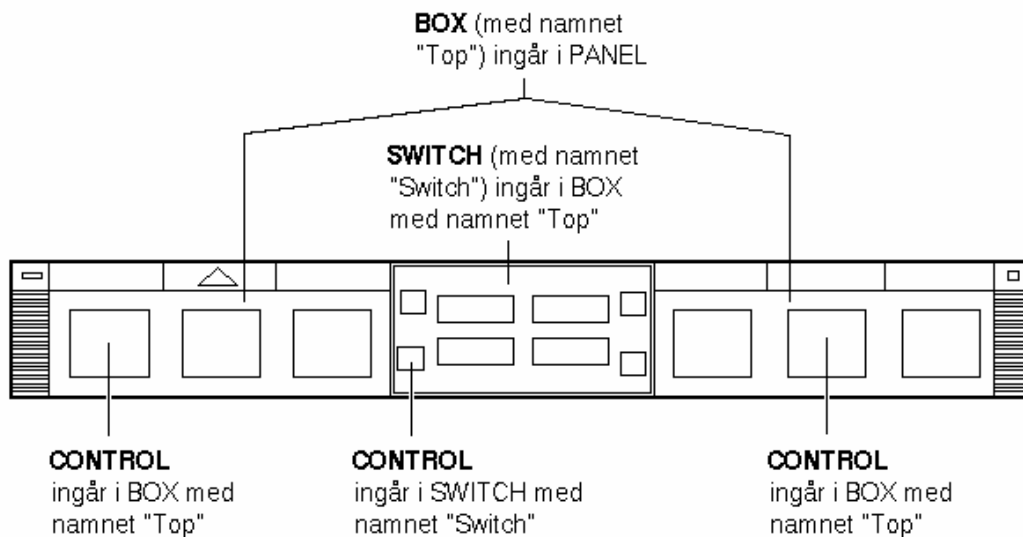
Definitionen av en arbetsyteväxlare (SWITCH) beskriver:

- SWITCH-namnet
- Vilken ruta (BOX) som arbetsyteväxlaren (SWITCH) ingår i (CONTAINER_NAME)
- Kontrollens (SWITCH) placering i rutan (BOX) (POSITION_HINTS)
- Fält som beskriver utseende och metod för arbetsyteväxlaren (SWITCH)

```
SWITCH parameternamn
{
  CONTAINER_NAME   boxnamn
  POSITION_HINTS    position
  KEYWORD          värde
  KEYWORD          värde
  ...
}
```

Anpassa huvudpanelen

Huvudpanelen är huvudgruppen förutom dess underpanelmenyer.



Figur 15-2 Behållare på huvudpanelen

De ändringar du kan göra är bl a:

- Lägga till och ta bort kontroller

- Byta plats på kontroller

▼ Lägg till en kontroll på huvudpanelen

1. Skapa huvudgruppens konfigurationsfil:

- För hela systemet: `/etc/dt/appconfig/types/språk/*.fp`
- Egen: `Hemkatalog/.dt/types/*.fp`

2. Definiera kontrollen i filen.

Använd fälten `CONTAINER_NAME` och `CONTAINER_TYPE` för att ange behållare för kontrollen:

```
CONTAINER_NAME  Top
CONTAINER_TYPE  BOX
```

Använd `POSITION_HINTS` för att ange kontrollens placering i sidled. Eftersom anpassningar har högre prioritet än inbyggda kontroller kommer den nya kontrollen att förskjuta en befintlig kontroll på samma plats en plats åt höger.

3. Spara konfigurationsfilen.

4. Skapa en ikon för huvudgruppskontrollen.

Se "Ange vilken ikonbild som ska användas av kontrollen" på sidan 232.

5. Välj Starta om arbetsytahanteraren på Arbetsyta-menyn.

Följande kontrolldefinition som finns i filen

`/etc/dt/appconfig/types/språk/audio.fp` infogar en ljudprogramkontroll mellan kontrollerna för Klocka och Kalender.

```
CONTROL AudioApplication
{
  TYPE          icon
  CONTAINER_NAME Top
  CONTAINER_TYPE BOX
  ICON          AudioApp
  POSITION_HINTS 2
  PUSH_ACTION   StartAudioApplication
  PUSH_RECALL   true
}
```

▼ Ta bort en kontroll

1. Skapa huvudgruppens konfigurationsfil:

- För hela systemet: `/etc/dt/appconfig/types/språk/namn.fp`

- Egen: *Hemkatalog*/.dt/types/*namn*.fp

2. Kopiera definitionen för kontrollen som du vill ta bort till den nya filen.

Om kontrollen är inbyggd finns definitionen i
/usr/dt/appconfig/types/*språk*/dtwm.fp.

Du behöver inte kopiera hela definitionen. Den del du kopierar måste dock innehålla fälten CONTAINER_NAME och CONTAINER_TYPE.

3. Lägg till fältet DELETE i definitionen:

```
DELETE    True
```

4. Spara konfigurationsfilen.

5. Välj Starta om arbetsythanteraren på Arbetsyta-menyn.

Följande kontrolldefinition som finns i filen
/etc/dt/appconfig/types/*språk*/TrashCan.fp tar bort kontrollen för papperskorgen från huvudgruppen.

```
CONTROL Trash
{
  CONTAINER_NAME  Top
  CONTAINER_TYPE  BOX
  DELETE          True
}
```

▼ Ändra en kontroll

Använd den här proceduren om du behöver ändra en kontrolldefinition, t ex byta dess ikonbild.

1. Kopiera hela kontrolldefinitionen från

/usr/dt/appconfig/types/*språk*/dtwm.fp **till:**

- För hela systemet: /etc/dt/appconfig/types/*språk*/*namn*.fp
- Egen: *Hemkatalog*/.dt/types/*namn*.fp.

2. Redigera fältet du vill ändra. Du kan också lägga till ytterligare fält.

3. Spara filen.

4. Välj Starta om arbetsythanteraren på Arbetsyta-menyn.

▼ Byta position för kontrollen

1. **Kopiera kontrolldefinitionerna för de kontroller som du vill byta plats på från /usr/dt/appconfig/types/*språk*/dtwm.fp till:**

- För hela systemet: /etc/dt/appconfig/types/*språk*/*namn*.fp
- Egen: *Hemkatalog*/.dt/types/*namn*.fp.

Du måste kopiera hela kontrolldefinitionen för alla kontroller som ska flyttas.

2. **Byt ut värdena på fälten POSITION_HINTS mellan kontrolldefinitionerna.**

3. **Spara filen.**

4. **Välj Starta om arbetsytahanteraren på Arbetsyta-menyn.**

Följande kontrolldefinitioner som finns i filen

/etc/dt/appconfig/types/C/MailHelp.fp byter plats på kontrollerna för post och hjälphanteraren och läser dessa kontroller för egna ändringar.

```
CONTROL Mail
{
  POSITION_HINTS      12
  LOCKED             True
  .....fortsättningen på kontrolldefinitionen
}
```

```
CONTROL Help
{
  POSITION_HINTS      5
  LOCKED             True
  .....fortsättningen på kontrolldefinitionen
}
```

▼ Ersätta en huvudgruppskontroll

- ◆ **Skapa en annan kontrolldefinition med samma namn:**

- *kontrollnamn*
- CONTAINER_NAME värde

Följande två kontroller definieras i två olika konfigurationsfiler. Dessa kontroller har samma kontrollnamn och behållarnamn, och betraktas därför som samma kontroll.

- Definition i /etc/dt/appconfig/types/C/SysControls.fp.

```
Control ImportantApplication
{
    CONTAINER_NAME      Top
    CONTAINER_TYPE      BOX
    POSITION_HINTS       2
    ...
}
```

■ Definition i *Hemkatalog*/.dt/types/MyControls.fp

```
Control ImportantApplication
{
    CONTAINER_NAME      Top
    CONTAINER_TYPE      BOX
    POSITION_HINTS       6
    ...
}
```

Den egna kontrollen har prioritet, så kontrollen läses vid position 6.

Ange vilken ikonbild som ska användas av kontrollen

Kontrolldefinitionens `ICON`-fält definierar vilken ikonbild som ska användas för kontrollen.

Värdet på fältet `ICON` kan vara:

■ Ett basfilnamn.

Grundfilnamnet är namnet på filen som innehåller ikonerna minus filnamnsuffixen för storlek (m och t) och bildtyp (bm och pm). Använd till exempel `MittSpel` om filnamnet är `MittSpel.l.pm` eller `MittSpel.m.pm`.

Om du använder basfilnamnet måste ikonfilerna placeras i en katalog i ikonsökvägen:

- Egna ikoner: *Hemkatalog*/.dt/icons
- Systemtäckande ikoner: /etc/dt/appconfig/icons/*språk*

■ En absolut sökväg till ikonerna, inklusive fullständigt filnamn.

Du ska bara använda den absoluta sökvägen om ikonfilen inte finns i ikonsökvägen.

Vilken ikonstorlek på ikonerna som behövs beror på var kontrollen placeras:

Platsstorlek

Huvudpanelen – 48 gånger 48 bildpunkter (*namn.l.pm* eller *namn.l.bm*)

Panelundermeny – 24 gånger 24 bildpunkter (*namn.s.pm* eller *namn.s.bm*)

Placera filikonen i en av följande kataloger:

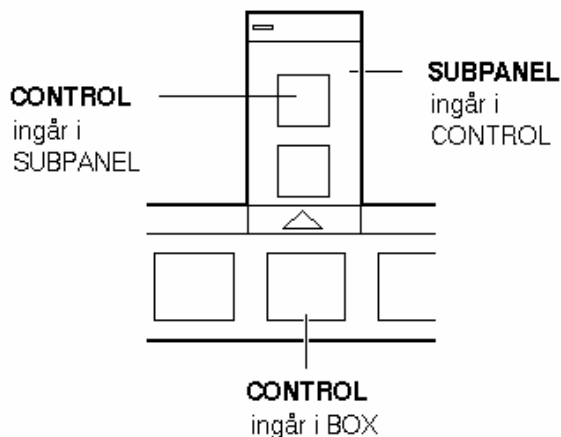
- Egna ikoner: *Hemkatalog/.dt/icons*
- Systemtäckande ikoner: */etc/dt/appconfig/icons/språk*

Skapa och ändra panelundermenyer

Användarna kan skapa och ändra panelundermenyerna med hjälp av huvudgruppens menyer.

I det här avsnittet beskrivs hur du gör systemtäckande anpassningar, vilket kräver att du ändrar huvudgruppens konfigurationsfiler.

En panelundermeny hör till en kontroll i huvudpanelen.



Figur 15-3 En panelundermenys behållare är kontrollen som den är kopplad till

Kopplingen görs i panelundermenyns definition. Fältet `CONTAINER_NAME` anger vilken kontroll som panelundermenyn hör till:

```
CONTROL kontrollnamn
{
  ...
}

SUBPANEL underpanelnamn
{
  CONTAINER_NAME      kontrollnamn
  ...
}
```

▼ Skapa en ny systemtäckande panelundermeny

1. Leta reda på *kontrollnamnet* på den kontroll i huvudpanelen som du vill koppla panelundermenyn till.

Om kontrollen är inbyggd finns definitionen i
`/usr/dt/appconfig/types/språk/dtwm.fp`.

2. Skapa en ny fil med namnet `/etc/dt/appconfig/types/språk/*.fp`.

3. Definiera panelundermenyn:

```
SUBPANEL underpanelnamn
{
  CONTAINER_NAME kontrollnamn
  TITLE värde
  KEYWORD värde
  ...
}
```

4. Spara den nya konfigurationsfilen.

5. Välj Starta om arbetsyttehanteraren på Arbetsyta-menyn.

Anpassa inbyggda panelundermenyer

Du kan ändra allmänna egenskaper för (t ex rubrik) och innehåll i inbyggda panelundermenyer.

Ändra allmänna egenskaper för en inbyggd panelundermeny

1. Skapa en ny konfigurationsfil för huvudgruppen:

- För hela systemet: `/etc/dt/appconfig/types/språk/namn.fp`
- Egen: `Hemkatalog/.dt/types/namn.fp`.

2. Kopiera hela standarddefinitionen för SUBPANEL från

`/usr/dt/appconfig/types/språk/dtwm.fp`:

```
SUBPANEL underpanelnamn
{
  ...
}
```

3. Ändra definitionen till panelundermenyn.

4. Spara den nya konfigurationsfilen.

5. Välj Starta om arbetsyterhanteraren på Arbetsyta-menyn.

Följande definition som finns i filen `/users/jenny/.dt/types/PerApps.fp` ändrar t ex namnet på panelundermenyn Egna program:

```
SUBPANEL PersAppsSubpanel
{
  CONTAINER_NAME      TextEditor
  TITLE               Jennys program
}
```

Lägga till systemtäckande kontroller i en inbyggd panelundermeny

1. Skapa en konfigurationsfil för huvudgruppen:

`/etc/dt/appconfig/types/språk/namn.fp`.

2. Definiera den systemtäckande kontrollen i filen.

Använd fälten `CONTAINER_NAME` och `CONTAINER_TYPE` för att ange behållare för kontrollen:

```
CONTROL kontrollnamn
{
  CONTAINER_NAME      underpanelnamn
  CONTAINER_TYPE      SUBPANEL
  ...
}
```

Se "Definiera kontroller i huvudgruppen" på sidan 237.

3. Spara konfigurationsfilen.

4. Välj Starta om arbetsyterhanteraren på Arbetsyta-menyn.

Följande kontroll som definieras i en ny fil `/etc/dt/appconfig/types/språk/DigitalClock.fp` lägger till `DigitalClock` (i programgruppen Skrivbordsverktyg) på panelundermenyn Egna program för alla användare.

```
CONTROL DigitalClockControl
{
  TYPE      icon
  CONTAINER_NAME      PerAppsSubpanel
  CONTAINER_TYPE      SUBPANEL
  ICON      Dtdgclk
  PUSH_ACTION      DigitalClock
  PUSH_RECALL      True
}
```

Ta bort en kontroll från en inbyggd panelundermeny

- ◆ Använd samma procedur som för att ta bort kontroller på Huvudpanelen. Se "Ta bort en kontroll" på sidan 229.

Ta bort kontrollen Installationsikon

- ◆ Lägg till följande fält i panelundermenyns definition:

```
CONTROL_INSTALL      False
```

▼ Ta bort den automatiska stängningen (Auto-Close) av panelundermenyer

Standardmetoden för panelundermenyer är att de stängs när användaren väljer en kontroll, såvida inte användaren har flyttat panelundermenyn från den ursprungliga positionen.

Huvudgruppen kan konfigureras så att panelundermenyerna hålls öppna tills användaren stänger dem.

1. Skapa en ny konfigurationsfil för huvudgruppen i:

- För hela systemet: `/etc/dt/appconfig/types/språk/*.fp`
- Egen: `Hemkatalog/.dt/types/*.fp`

2. Kopiera standarddefinitionen för PANEL från

`/usr/dt/appconfig/types/språk/dtwm.fp` till den nya filen:

```
PANEL FrontPanel
{
  ...
}
```

3. Lägg till följande fält i definitionerna för PANEL:

```
SUBPANEL_UNPOST      False
```

4. Spara den nya konfigurationsfilen.

5. Välj Starta om arbetsytahanteraren på Arbetsyta-menyn.

Definiera kontroller i huvudgruppen

Användarna kan skapa egna knappar genom att släppa ikoner på Installationsikon-kontrollerna.

Det här sättet att anpassa är enkelt, men funktionaliteten omfattar bara en del av vad som kan göras med kontroller i huvudgruppen. En kontroll som har skapats med Installationsikon kan t ex inte:

- Innehålla animeringar
- Visa ett klientfönster
- Ändra utseende när något inträffar (t ex för att ange att ny post har tagits emot)

I det här avsnittet beskrivs hur du manuellt skapar kontroller i huvudgruppen.

Referensinformation om syntaxen för huvudgruppskontroller finns i direkthjälpen (man) för dtfprofile(4X).

Definitioner till kontroller i huvudgruppen

Strukturen på en definition till en huvudgruppskontroll är:

```
CONTROL kontrollnamn
{
  TYPE kontrolltyp
  CONTAINER_NAME värde
  CONTAINER_TYPE värde
  övriga fält som definierar utseende och uppförande
}
```

Kontrolltyper

Fältet `TYPE` i kontrolldefinitionen anger grundläggande beteende för den kontrollen.

Kontrolltyp (TYPE)	Kontrollfunktion
<code>icon</code> (Standard.)	Kontrollen kör en angiven åtgärd när användaren klickar på kontrollen eller släpper en fil på den.
<code>blank</code>	Platshållare för att justera avstånd mellan kontroller.
<code>busy</code>	Upptaget-lampa. Kontrollen blinkar (växlar bilder) när en åtgärd startas
<code>client</code>	Ett klientfönster i huvudgruppen.

clock	Klocka.
date	Visar aktuellt datum.
file	Representerar en fil. När du väljer kontrollen körs standardåtgärden för den filen.

▼ Skapa en ny kontroll

Detta avsnitt beskriver hur en kontroll och olika kontrolltyper skapas.

- 1. Om kontrollen ska ha något av fälten `PUSH_ACTION` eller `DROP_ACTION` skapar du motsvarande åtgärdsdefinitioner. Dessa åtgärder körs när användaren klickar på kontrollen eller släpper en fil på den.**
- 2. Skapa ikonbildens fil för kontrollen.**
Mer information om ikonstorlek, ikonnamn och ikonplaceringar finns i "Ikonbilsfiler" på sidan 213.
- 3. Skapa en ny konfigurationsfil för huvudgruppen i:**
 - För hela systemet: `/etc/dt/appconfig/types/språk/*.fp`
 - Egen: `Hemkatalog/.dt/types/*.fp`
- 4. Lägg till kontrolldefinitionen i filen.**
- 5. Spara filen.**
- 6. Välj Starta om arbetsytahanteraren på Arbetsyta-menyn.**

Skapa en kontroll som kör en åtgärd när du klickar på den

Använd följande fält för att definiera kontrollens funktion:

- `TYPE`: Ange som ikon
- `PUSH_ACTION`: Anger namnet på åtgärden som ska köras

Följande kontroll, som placeras på panelundermenyn Egna program, kör ett spel som användaren har lagt in:

```
CONTROL Ball
{
  TYPE          icon
  CONTAINER_NAME  PersAppsSubpanel
  CONTAINER_TYPE  SUBPANEL
  ICON          ball
  PUSH_ACTION    RunBallGame
  HELP_STRING    "Välj den här kontrollen för att spela."
}
```

Följande kontroll placeras i det övre vänstra hörnet av arbetsytväxlaren. Den startar en åtgärd som heter CutDisp.

```
CONTROL StartCutDisp
{
    TYPE          icon
    CONTAINER_NAME Switch
    CONTAINER_TYPE SWITCH
    POSITION_HINTS first
    ICON          cutdisp
    HELP_STRING   "Välj den här kontrollen för att köra cutdisp."
    PUSH_ACTION   CutDisp
}
```

Skapa en kontroll som öppnar en fil

Använd följande fält för att definiera kontrollens funktion:

- TYPE: Ange som fil
- FILE_NAME: Anger sökvägen till filen som ska öppnas
- PUSH_ACTION: Ange som Open

Det måste finnas en Open-åtgärd definierad för filens datatyp.

Följande kontroll placeras längst till höger på huvudpanelen. Den startar textredigeraren med datafilen /users/ellen/PhoneList.txt. Åtgärden Open för *.txt-filer är en del av standardåtgärdsdatabasen.

```
CONTROL EditPhoneList
{
    TYPE          file
    FILE_NAME     /users/ellen/PhoneList.txt
    CONTAINER_NAME Top
    CONTAINER_TYPE BOX
    POSITION_HINTS last
    ICON          PhoneBook
    HELP_STRING   "Den här kontrollen visar Ellens telefonlista."
    PUSH_ACTION   Open
}
```

Skapa en kontroll som fungerar som ett målområde

Använd fältet DROP_ACTION för att ange vilken åtgärd som ska köras när användaren släpper en fil på kontrollen. Åtgärden måste kunna acceptera filargument.

En kontrolldefinition innehåller ofta fälten PUSH_ACTION och DROP_ACTION. Du kan använda samma åtgärd för en dra och släpp-åtgärd.

Följande kontroll på panelundermenyn Egna program kör X-klienten xwud, som använder filargument.

```
CONTROL Run_xwud
{
    CONTAINER_NAME PerAppsSubpanel
    CONTAINER_TYPE SUBPANEL
}
```

```

POSITION_HINTS    2
ICON              XwudImage
PUSH_ACTION       RunXwud
DROP_ACTION       RunXwud
}

```

Skapa en kontroll som övervakar filer

Använd följande fält för att definiera kontrollens funktion:

- **TYPE:** Ange något av följande värden:

`icon` – använd den här typen om du vill ange en `PUSH_ACTION` och/eller `DROP_ACTION` för kontrollen.

`file` – använd den här typen om du vill att kontrollen ska fungera som en fil när du dubbelklickar på filikonen i filhanteraren.

- **ICON** och **ALTERNATE_ICON:** Beskriver de bilder som används för att ange ändrad och oförändrad status för den övervakade filen.
- **MONITOR_TYPE:** Beskriver vilka villkor som gör att bilden ändras. Använd något av följande värden:

`mail` – kontrollen ändrar utseende när information läggs till i filen.

`file` – kontrollen ändras när den angivna filen inte längre är tom.

- **FILE_NAME:** Anger namnet på den fil som ska övervakas.

Följande kontroller letar t ex efter en fil med namnet `meetings` som ska överföras till ditt system med hjälp av anonym ftp. Kontrollen placeras i panelundermenyn **Egna program**.

```

CONTROL MonitorCalendar
{
    TYPE          file
    CONTAINER_NAME PersonalApps
    CONTAINER_TYPE SUBPANEL
    POSITION_HINTS first
    FILE_NAME      /users/ftp/meetings
    MONITOR_TYPE   file
    ICON           meetingsno
    ALTERNATE_ICON meetingsyes
}

```

Skapa en växlingskontroll

En kontroll för enstaka förekomster kontrollerar om den process som har startats av `PUSH_ACTION` redan körs. Om processen inte körs, körs `PUSH_ACTION`. Om processen redan körs flyttas fönstret högst upp i fönsterstacken på den aktuella arbetsytan.

Använd följande fält för att definiera kontrollens funktion:

- **PUSH_RECALL:** Ange som `True`.
- **CLIENT_NAME:** Anger namnet på kontrollens klient.

Värdet på `CLIENT_NAME` måste stämma överens med den första strängen (*res_name*) i egenskapen `WM_CLASS` i programmets primära fönster. Mer information finns i direkthjälpen (man) för `xprop(1)`.

- `PUSH_ACTION`: Beskriver åtgärden som körs när användaren klickar på kontrollen.

Följande kontroll kör en förekomst av ett program vars åtgärd heter `MyEditor`.

```
CONTROL MyEditor
{
    TYPE          icon
    CONTAINER_NAME    Top
    CONTAINER_TYPE    BOX
    POSITION_HINTS     15
    PUSH_RECALL      True
    CLIENT_NAME       BestEditor
    PUSH_ACTION       StartMyEditor
    ICON              MyEd
}
```

Skapa en klientfönsterkontroll

En klientfönsterkontroll är ett programfönster som är inbäddat i huvudgruppen. Du kan t ex placera en systembelastningsmätare i huvudgruppen genom att skapa en klientfönsterkontroll för `xload`.

1. Definiera kontrollen.

Använd följande fält för att definiera kontrollens funktion:

- `TYPE`: Ange som `client`.
- `CLIENT_NAME`: Anger vilken klient som ska startas.
Värdet på `CLIENT_NAME` måste stämma överens med den första strängen (*res_name*) i egenskapen `WM_CLASS` i programmets primära fönster. Mer information finns i direkthjälpen (man) för `xprop(1)`.
- `CLIENT_GEOMETRY`: Anger den storlek, (i bildpunkter) som behövs för klientens huvudgruppsfönster.
Direkthjälpen (man) för `xwininfo(1)` beskriver hur du tar reda på storleken på ett fönster i bildpunkter.

2. Välj Starta om arbetsytahanteraren på Arbetsyta-menyn.

3. Starta klienten från kommandoraden i ett terminalfönster.

Följande kontroll visar t ex en belastningsmätare på 30×20 bildpunkter.

```
CONTROL LoadMeter
{
    TYPE          client
    CONTAINER_NAME    Top
    CONTAINER_TYPE    BOX
    CLIENT_NAME       xload
    CLIENT_GEOMETRY   30x20
}
```

Om klienten inte sparas och återställs mellan sessionerna, vill du kanske konfigurera kontrollen så att klienten startar när användaren klickar på den. Du kan t ex konfigurera kontrollen för LoadMeter så att den startar `xload` genom att lägga till följande rad i definitionen:

```
PUSH_ACTION    StartXload
```

och skapa åtgärden:

```
ACTION StartXload
{
    WINDOW_TYPE    NO_STDIO
    EXEC_STRING    /usr/contrib/bin/X11/xload
}
```

Animera en kontroll

Du kan skapa en animeringssekvens som används när användaren väljer kontrollen eller släpper ett objekt på den.

För att kunna innehålla en animering måste kontrollen:

- Vara av typen `icon`
- Ha en `PUSH_ACTION` eller `DROP_ACTION`

1. Ange animeringssekvensen med hjälp av komponenten `ANIMATION`:

```
ANIMATION animationsnamn
{
    ANIMATION ikon1    [fördröjning_ms]
    ANIMATION ikon2    [fördröjning_ms]
    ...
}
```

där *ikon1*, *ikon2* osv är namn på ikoner och *fördröjning_ms* är tidsfördröjningen mellan animeringsikonerna i millisekunder. Standardtidsförskjutningen är 200 millisekunder.

2. Lägg till fältet `PUSH_ANIMATION` och/eller `DROP_ANIMATION` till kontrolldefinitionen. Värdet är namnet på sekvensen `ANIMATION`.

Följande rad animerar en kontroll som startar programmet `BestEditor`.

Tidsförskjutningen mellan ikonerna är 300 millisekunder. I exemplet antas det att du har skapat ikonfilerna `frame1`, `frame2` osv.

```
CONTROL BestEditor
{
    ...
    PUSH_ANIMATION BestEdAnimation
    ...
}
```

```
ANIMATION BestEdAnimation
{
  frame1 300
  frame2
  ...
}
```

Objekthjälp för kontroller på huvudgruppen

Det finns två sätt att göra hjälp till en kontroll tillgänglig:

- Lägg till en hjälpsträng i kontrolldefinitionen.

Hjälpsträngen visas i hjälpfönstret när användaren anropar objekthjälp till kontrollen. Hjälpsträngen kan inte innehålla formatering (t ex rubriker) eller länkar.

Om du vill lägga till en hjälpsträng anger du den i kontrolldefinitionen:

```
HELP_STRING      hjälpsträng
```

- Ange ett hjälpavsnitt i en registrerad hjälpvolym.

Ett hjälpavsnitt är skriven information som utnyttjar alla möjligheter som finns i hjälpsystemet. För att kunna skriva hjälpavsnitt måste du använda skrivbordets Help Developer's Kit.

När du vill lägga in ett hjälpavsnitt anger du hjälpvolym och avsnitts-ID i kontrolldefinitionen:

```
HELP_VOLUME      hjälpvolymnamn
HELP_TOPIC       avsnittsid
```

Anpassa området för växling av arbetsyta

Det finns flera sätt att anpassa området för växling av arbetsyta:

- Ändra antalet arbetsytor
- Ändra utseende på växlingsområde
- Ändra kontrollerna i växlingsområde

▼ Ändra standardantalet arbetsytor

- ◆ Ändra följande resurs för arbetsytehanteraren:

```
Dtwm*workspaceCount:      n
```

Mer information finns i "Ändra antalet systemtäckande arbetsytor" på sidan 252.

▼ Ändra antalet knapprader i växlingsområdet

- ◆ Ändra fältet NUMBER_OF_ROWS i SWITCH-definitionen.

Följande definition definierar ett treraders växlingsområde.

```
SWITCH Switch
{
  CONTAINER_NAME      boxnamn
  NUMBER_OF_ROWS      3
  ...
}
```

▼ Ändra eller lägga till kontroller i växlingsområdet

1. Skapa en konfigurationsfil för huvudgruppen med kontrolldefinitionen.

- Ange att kontrollen ska finnas inuti växlingsområdet:

```
CONTAINER_NAME      Switch
CONTAINER_TYPE      SWITCH
```

- Ange positionen i arbetsyteväxlaren:

```
POSITION_HINTS      n
```

där *n* är ett heltal. Positionerna numreras i ordning från vänster till höger, uppifrån och ned. (För den normala tvåradersknappen är positionerna mellan 1 och 4.)

2. Skapa kontrollens ikon. Storleken bör vara 16 gånger 16 bildpunkter.

Följande kontroll placerar en Terminal-kontroll i arbetsyteväxlaren.

```
CONTROL SwitchTerminal
{
  TYPE                icon
  CONTAINER_NAME      Switch
  CONTAINER_TYPE      SWITCH
  POSITION_HINTS       3
  ICON                Fpterm
  LABEL               Terminal
}
```

```
PUSH_ACTION      Dterm
HELP_TOPIC       FPOnItemTerm
HELP_VOLUME      FPanel
}
```

Kontrollen använder en inbyggd ikon och samma hjälpavsnitt som används av Terminal-kontrollen på panelundermenyn Egna program.

Allmän konfiguration av huvudgruppen

Med huvudgruppens syntax för `PANEL` kan du:

- Ändra positionen för huvudgruppen
- Ändra fönstrets detaljer
- Ange utseende och metod för kontrollerna

Standardbeskrivningen av `PANEL` finns i
`/usr/dt/appconfig/types/språk/dtwm.fp`.

Mer information finns i direkthjälpen (`man`) för `dtfpfile(4X)`.

Allmänna steg

1. **Skapa en ny konfigurationsfil för huvudgruppen i**
`/etc/dt/appconfig/types/språk eller Hemkatalog/.dt/types`.
2. **Kopiera standardbeskrivningen för `PANEL` från**
`/usr/dt/appconfig/types/språk/dtwm.fp` **till den nya filen:**
3. **Redigera `PANEL`-beskrivningen.**
Den nya `PANEL`-beskrivningen har högre prioritet än standardbeskrivningen.

▼ Ändra huvudgruppens standardposition

- ◆ Använd fältet `PANEL_GEOMETRY` i `PANEL`-definitionen för att ange positionen.

Följande panel finns i det övre högra hörnet.

```
PANEL SpecialFrontPanel
{
    PANEL_GEOMETRY      -1+1
    ...
}
```

▼ Namnge kontroller på huvudpanelen

1. **Lägg till följande rad till definitionen av PANEL:**

```
DISPLAY_CONTROL_LABELS    True
```

2. **Lägg till fältet LABEL till varje kontroll.**

Kontrollnamn används om LABEL inte anges.

▼ Ändra klickmetod för kontroller

- ◆ **Använd fältet CONTROL_BEHAVIOR i PANEL-definitionen för att ange hur användaren kör PUSH_ACTION. Tänkbara värden är:**

`single_click` - användaren klickar på kontrollen för att köra PUSH_ACTION

`double_click` - användaren dubbelklickar på kontrollen för att köra PUSH_ACTION

▼ Skapa en helt ny huvudgrupp

När du vill göra mycket omfattande ändringar kan det vara bättre att skapa en helt ny huvudgrupp.

För att undvika konflikt med huvudgruppens inbyggda komponenter bör en helt ny huvudgrupp använda nya namn för PANEL och andra behållare.

1. **Skapa PANEL-komponenten för den nya huvudgruppen. Ge den ett unikt namn:**

```
PANEL front_panel_name
{
    ...
}
```

2. **Skapa nya rutor och kontroller, och använd de nya behållarnamnen.**

Om du vill använda befintliga komponenter måste du kopiera deras definitioner och ändra värdet CONTAINER_NAME.

3. Välj Starta om arbetsyttehanteraren på Arbetsyta-menyn.

Exempel på hur du skapar en egen huvudgrupp med tre rader

I följande exempel ändras standardhuvudgruppen så att dess kontroller ordnas i tre rader.

1. **Kopiera** `/usr/dt/appconfig/types/språk/dtwm.fp` till **Hemkatalog**/`.dt/types/MyFrontPanel.fp`. **Tilldela** **filen skrivbehörighet**.
Det är den här filen du redigerar för att skapa en ny huvudgrupp.

2. **Ändra namnet på huvudgruppen:**

```
PANEL NewFrontPanel
```

3. **Ändra namnet på rutan som heter Top och redigera behållarens namn:**

```
BOX NewFrontPanelTop
{
  CONTAINER_NAME      NewFrontPanel
  POSITION_HINTS       first
  ...
}
```

4. **Lägg till rutdefinitioner för den mellersta och undre raden:**

```
BOX NewFrontPanelMiddle
{
  CONTAINER_NAME      NewFrontPanel
  POSITION_HINTS       second
}
```

```
BOX NewFrontPanelBottom
{
  CONTAINER_NAME      NewFrontPanel
  POSITION_HINTS       second
}
```

5. **Ändra CONTAINER_NAME för följande kontroller till** `NewFrontPanelTop`:

- Klocka
- Datum
- Hem
- Textredigerare

- Post
6. **Ändra** CONTAINER_NAME **för följande kontroller till** NewFrontPanelBottom:
 - Skrivare
 - Snittvariant, Typ
 - Program
 - Hjälp
 - Papperskorg
 7. **Ändra** CONTAINER_NAME **för växlaren till** NewFrontPanelMiddle:
 8. **Välj Starta om arbetsytahanteraren på Arbetsyta-menyn.**

Anpassa arbetsytehanteraren

I följande avsnitt beskrivs hur du anpassar skrivbordets arbetsytehanterare.

- "Arbetsytehanterarens konfigurationsfiler" på sidan 250
- "Anpassa arbetsytor" på sidan 252
- "Arbetsytehanterarens menyer" på sidan 254
- "Anpassa knappkopplingar" på sidan 259
- "Anpassa tangentkopplingar" på sidan 261
- "Växla mellan förinställda och egna uppsättningar" på sidan 263

Arbetsytehanteraren är en fönsterhanterare som finns på skrivbordet. På samma sätt som andra fönsterhanterare hanterar den följande:

- Hur olika delar av fönsterramen visas
- Fönstrets olika visningsinställningar, bl a staplingsordning och fokusering
- Tangentbords- och knappkopplingar
- Visning av minimerade fönster
- Arbetsyta- och Fönster-menyerna

Arbetsytehanteraren hanterar dessutom följande skrivbordskomponenter:

- *Arbetsytor.* Arbetsytehanteraren hanterar antalet arbetsytor och håller reda på vilka fönster som är öppna i varje arbetsyta.
- *Bakgrunder för arbetsytor.* Användaren ändrar bakgrund med hjälp av Inställningar. Hantering av bakgrunder är dock en funktion i arbetsytehanteraren.
- *Huvudgruppen.* Trots att huvudgruppen har egna konfigurationsfiler, skapas och hanteras den med arbetsytehanteraren.

Flera av dessa komponenter kan ändras med Inställningar. Med Inställningar går det snabbt och lätt att redigera sådant som du behöver ändra ofta. Övriga resurser måste anges manuellt.

Arbetsytehanteraren är `dtwm`. Den bygger på Motif-fönsterhanteraren.

- Om du behöver referensinformation om arbetsytehanteraren läser du `dtwm(1)` och `dtwmrc(4)` i direkthjälpen.
- Mer information om hur du anger arbetsytehanterarens resurser finns i "Ange programresurser" på sidan 265.
- Mer information om konfigurationsfiler för huvudgruppen finns i Kapitel 15.

Mer information om hur du anger resurser finns i "Ange programresurser" på sidan 265.

Arbetsytehanterarens konfigurationsfiler

Arbetsytehanteraren hämtar information om Fönster-menyerna, arbetsystemenyerna samt knapp- och tangentskopplingar från en konfigurationsfil.

Den använder någon av följande filer:

- Egen fil: *Hemkatalog*/`.dt/dtwmrc`
- Egen systemfil: `/etc/dt/appconfig/appmanager/språk/sys.dtwmrc`
- Inbyggd fil: `/usr/dt/config/språk/sys.dtwmrc`

Arbetsytehanteraren söker efter en konfigurationsfil enligt ovan nämnda ordning och använder den första som den hittar.

För användare som använder mer än ett sessionsspråk, kan en egen språkberoende konfigurationsfil med namnet *Hemkatalog*/`.dt/språk/dtwmrc` skapas. Den prioriteras över *Hemkatalog*/`.dt/dtwmrc`.

▼ Skapa och ändra en egen konfigurationsfil

Den egna konfigurationsfilen för arbetsytehanteraren är *Hemkatalog*/`.dt/dtwmrc`. Om den här filen finns är det den som används.

1. Dubbelklicka på Redigera Dtwmrc i tillbehörsguppen Skrivbordsverktyg.

Om det redan finns en egen `dtwmrc`-fil hämtas den till redigeraren. Annars kopieras `sys.dtwmrc` till *Hemkatalog*/`.dt/dtwmrc` och hämtas sedan till redigeraren.

2. Redigera filen.

3. Avsluta redigeraren.

Filen sparas som din egen `dtwmrc`, oavsett ursprungskälla.

▼ Skapa en systemtäckande konfigurationsfil

Den systemtäckande konfigurationsfilen för arbetsyterhanteraren är
`/etc/dt/config/språk/sys.dtwmrc`.

- ◆ **Kopiera** `/usr/dt/config/språk/sys.dtwmrc` till
`/etc/dt/config/språk/sys.dtwmrc`.

Obs! Den här filen används inte om *Hemkatalog*/`.dt/dtwmrc` finns.

▼ Inkludera andra filer

- ◆ **Använd syntaxen:**

```
include
{
    sökväg
    sökväg
    ...
}
```

Följande rader lägger t ex till filen `/users/ellen/mymenu`:

```
include
{
    /users/ellen/mymenu
}
```

Använd programsatserna `Include` när du vill skaffa ytterligare funktionalitet utan att hela konfigurationsfilen behöver kopieras. En användare vill kanske skapa en ny tangentkoppling utan att administrera hela konfigurationsfilen. Användaren skapar då filen *Hemkatalog*/`.dt/dtwmrc` med följande innehåll:

```
include
{
    /etc/dt/config/C/sys.dtwmrc
}
Keys DtKeyBindings
{
    Alt<Key>F5 root f.menu Applications
}
```

```
Menu Applications
{
    "GraphicsApp" f.exec "/usr/bin/GraphicsApp/GApp"
```

```
} ...
```

▼ Starta om arbetsyterhanteraren

Arbetsyterhanteraren måste startas om för att ändringarna i konfigurationsfilen ska träda i kraft.

- ◆ **Välj Starta om arbetsyterhanteraren på menyn Arbetsyta (klicka på tredje musknappen när pekaren befinner sig över bakgrunden).**

Anpassa arbetsytor

De flesta anpassningar av arbetsytor, t ex ändring av namn och antalet arbetsytor, kan göras av användaren med hjälp av skrivbordets gränssnitt. Arbetsyterhanteraren omfattar dock resurser för att ange systemtäckande standardvärden.

▼ Ändra antalet systemtäckande arbetsytor

Standardskrivbordskonfigurationen omfattar fyra arbetsytor. Användaren kan lägga till och ta bort arbetsytor med hjälp av menyerna som hör till arbetsyteromkopplaren.

I filen `/usr/dt/app-defaults/C/Dtwm` anges följande standardantal arbetsytor för resursen `workspaceCount`:

```
Dtwm*0*workspaceCount: 4
Dtwm*workspaceCount: 1
```

Flera arbetsytor anges på skärm 0. Enstaka arbetsytor anges på övriga skärmar.

Du kan skapa filen `/etc/dt/config/C/sys.resources` (eller redigera den om den redan finns) för att ändra det förvalda antalet arbetsytor för alla nya användare på en arbetstation.

- ◆ **Använd resursen `0*workspaceCount` om du vill använda den förvalda systemtäckande inställningen på den primära skärmen:**

```
Dtwm*0*workspaceCount: tal
```

Följande resurs ställer in antalet systemtäckande arbetsytor på den primära skärmen till sex:

```
Dtwm*0*workspaceCount: 6
```

Mer information om hur du anger arbetsythanterarens resurser finns i "Ange programresurser" på sidan 265 .

Följande resurs ställer in antalet arbetsytor till sex:

```
Dtwm*workspaceCount: 6
```

▼ Namnge systemtäckande arbetsytor

Arbetsytor numreras internt med numreringskonventionen wsn , där n är 0, 1, 2 osv. De fyra standardarbetsytorna numreras internt från $ws0$ till $ws3$.

♦ **Använd resursen `title` om du vill ändra namnet på en angiven arbetsyta:**

```
Dtwm*wsn: namn
```

Mer information om hur du anger arbetsythanterarens resurser finns i "Ange programresurser" på sidan 265.

Följande resurser kopplar de fyra förvalda arbetsytorna till de angivna namnen:

```
Dtwm*ws0*title: Anna  
Dtwm*ws1*title: Dan  
Dtwm*ws2*title: Julia  
Dtwm*ws3*title: Patrik
```

▼ Skapa ytterligare bakgrunder

1. **Skapa bakgrundsbilderna. De kan vara i bitmapps- eller pixmappsformat.**

2. **Placera bakgrunden i någon av följande kataloger. (Du kanske först måste skapa katalogen.)**

- Systemtäckande bakgrunder: `/etc/dt/backdrops`
- Egna bakgrunder: *Hemkatalog*/`.dt/backdrops`

3. **Välj Starta om arbetsythanteraren på menyn Arbetsyta.**

Systemtäckande och egna bakgrunder läggs till de inbyggda bakgrunderna i `/usr/dt/backdrops`.

Du kan ersätta en befintlig inbyggd bakgrund genom att skapa en egen eller systemtäckande bakgrund med samma namn.

▼ Ersätta bakgrunden med en grafisk bild

Bakgrunder läggs som skikt ovanpå skärmens rotfönster. I dialogrutan Inställningar - Bakgrund finns inställningen NoBackdrop som gör bakgrunden genomskinlig.

Det finns bara ett rotfönster bakom alla arbetsytbakgrunder. Därför finns en grafisk bild som har placerats på rotfönstret kvar för alla arbetsytor. Du kan ange vilka arbetsytor som ska täcka rotfönstret med en bakgrund. Bilden som visas när NoBackdrop är aktivt är dock densamma för alla arbetsytor.

1. Skapa den grafiska bilden.

Bilden måste vara i ett format som det finns ett verktyg till för att den ska kunna visas i rotfönstret. Om du t ex tänker använda `xsetroot`, måste du skapa en bitmapsfil.

2. Om den inte redan finns, skapar du den körbara filen

Hemkatalog/.dt/sessions/sessionetc.

Filen `sessionetc` körs varje gång som användaren loggar in.

3. Placera kommandot som visar bilden i filen `sessionetc`.

Följande kommando visar rotfönstret sida vid sida med angiven bitmapsfil:

```
xsetroot -bitmap /users/ellen/.dt/icons/root.bm
```

Arbetsytahanterarens menyer

Arbetsytahanteraren har tre standardmenyer:

menyn Arbetsyta – kallas även rotmenyn. Den visas när du trycker på musknapp 3 med pekaren placerad över bakgrunden. Menyn är kopplad till musknappen med en knappkoppling.

Fönster-menyn – menyn visas när användaren trycker ned musknapp 1 eller musknapp 3 med pekaren på fönstermenyknappen (i övre vänstra hörnet i fönsterramen). Menyn är kopplad till knappen med `windowMenu`-resursen.

Huvudgrupp-menyn – menyn visas när användaren trycker ned musknapp 1 eller musknapp 3 med pekaren på fönstermenyknappen i huvudgruppen.

Arbetsyterhanterarens menysyntax

Arbetsyterhanterarens menyer har följande syntax:

```
Menu Menynamn
{
  val1 [kortkommando] [snabbkommando] funktion [argument]
  val2 [kortkommando] [snabbkommando] funktion [argument]
  ...
}
```

där:

val är den text eller bitmappsbild som visas på menyn. Om texten innehåller blanksteg innesluter du texten med citattecken. För bitmappsbilder använder du syntaxen *@/sökväg*.

kortkommando är ett tecken som fungerar som ett snabbkommando när menyn visas. Det anges enligt mönstret *_tecken*.

snabbkommando – ett kortkommando som är aktiverat även om inte menyn visas. Snabbkommandon har följande syntax: *modifieringstangent*<Tangent> *Tangentnamn*. Modifieringstangenterna är Ctrl, Skift, Alt (tecken) eller Skiftlåås. En lista över alla tangentnamn finns i filen *keysymdef.h* i katalogen "X11 include".

funktion – den funktion som ska utföras när markeringen har gjorts. En lista över funktioner finns i direkthjälpavsnittet *dtwmrc* (4).

argument – funktionsargument. Se sidan (4) i *dtwmrc* för mer utförlig information.

Följande objekt med etiketten Återställ återställer fönstret. När menyn visas kan du återställa fönstret genom att trycka på "R". Du kan även återställa fönstret genom att trycka på Extend char F5.

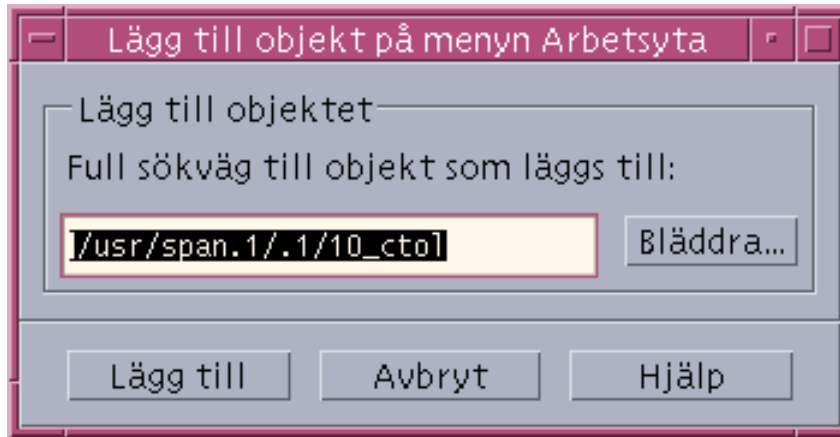
```
Restore  _R  Alt<Key> F5  f.normalize
```

Obs! Fullständig information om arbetsyterhanterarens menysyntax hittar du i direkthjälpavsnittet *dtwmrc* (4).

▼ Så här lägger du till ett nytt menyalternativ på menyn Arbetsyta

1. **Klicka på kontrollen Lägg till objekt på meny på panelundermenyn Verktyg i huvudgruppen.**

Dialogrutan Lägg till objekt på menyn Arbetsyta visas.



2. **Skriv en fullständig sökväg, eller klicka på Bläddra och markera sökvägen till den fil som du vill att det nya menyalternativet ska köra.**

Den fil som du använder bör vara tillgänglig för den aktuella värden. Den måste vara en körbar fil eller en fil av en datatyp som är registrerad hos värden (t ex ljud eller direkthjälp).

Den fullständiga sökvägen med filnamnet visas i textrutan.

3. **Klicka på Lägg till om du vill lägga till filen på menyn.**

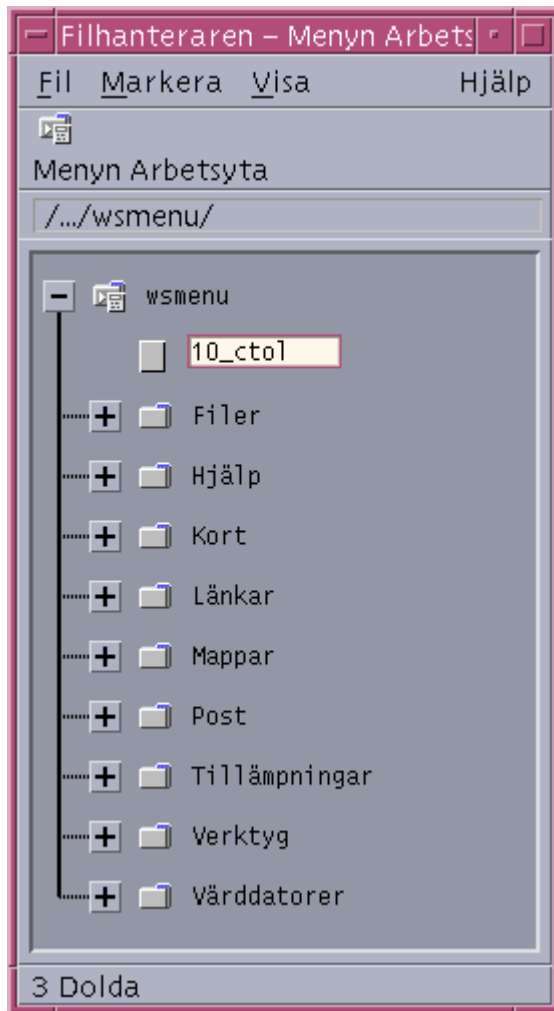
Filen läggs till på den första platsen på menyn Arbetsyta. Standardikonen för filen (om den har en sådan) och filnamnet används som ikon och text i menyalternativet.

Obs! Information om hur du ändrar menyalternativets placering på menyn Arbetsyta finns i "Så här anpassar du menyn Arbetsyta" på sidan 256.

▼ Så här anpassar du menyn Arbetsyta

1. **Klicka på kontrollen Anpassa menyn Arbetsyta på panelundermenyn Verktyg i huvudgruppen.**

I filhanteraren visas konfigurationsmappen för menyn Arbetsyta. Observera att filerna i den här mappen representerar alla menyalternativ, och undermapparna representerar undermenyerna. Genom att ordna innehållet i mappen ändrar du ordningen på menyn Arbetsyta.



2. **Dra ett menyalternativ som du vill flytta och släpp det där du vill placera det.**
Information om hur du använder filhanteraren i trädvyläge finns i *Solaris Common Desktop Environment: Användarhandbok*.
3. **Ta bort menyalternativ som du inte vill använda genom att dra motsvarande filer till papperskorgen i huvudgruppen.**
4. **Du kan också byta namn på menyalternativ genom att redigera fil- eller mappnamnen.**
5. **När du är nöjd med ändringarna väljer du Uppdatera menyn Arbetsyta på Fil-menyn i filhanteraren och avslutar sedan filhanteraren.**

På menyn Arbetsyta visas nu de ändringar som du har gjort i mappen för menyn Arbetsyta.

▼ Skapa en ny Arbetsyta-meny

1. Öppna en lämplig fil för redigering:

- Egen: *Hemkatalog*/.dt/dtwmrc
 - För hela systemet: /etc/dt/config/*språk*/sys.dtwmrc
- Information om hur du skapar de här filerna finns i "Arbetsytehanterarens konfigurationsfiler" på sidan 250.

2. Skapa den nya menyn:

```
Menu menynamn
{
    ...
}
```

Se "Arbetsytehanterarens menysyntax" på sidan 255.

3. Skapa eller redigera knappkopplingarna för att visa den nya menyn.

Om menyn ersätter den befintliga menyn redigerar du knappkopplingen som visar menyn Arbetsyta.

```
<Btn3Down> root f.menu menynamn
```

Om menyn är en tilläggsmeny skapar du en ny musknappskoppling. Den följande knappkopplingen visar menyn när Skift och musknapp 3 hålls ned över bakgrunden:

```
Shift<Btn3Down> root f.menu menynamn
```

4. Välj Starta om arbetsytehanteraren på menyn Arbetsyta.

▼ Skapa en ny Fönster-meny

Obs! Fönster-menyn är inbyggd i arbetsytehanteraren och anpassas normalt inte. Om du vill att fönstrens uppträdande i de olika programmen ska vara konstant bör du inte ändra inställningarna på Fönster-menyn.

1. Öppna en lämplig fil för redigering:

- Egen: *Hemkatalog*/ .dt/dtwmrc
 - Systemtäckande: /etc/dt/config/*språk*/sys.dtwmrc
- Information om hur du skapar de här filerna finns i "Arbetsyتهانtererens konfigurationsfiler" på sidan 250.

2. Skapa den nya menyn:

```
Menu menynamn
{
    ...
}
```

3. Använd resursen windowMenu för att ange den nya menyn:

```
Dtwm>windowMenu: menynamn
```

4. Välj Starta om arbetsyتهانterearen på menyn Arbetsyta.

Anpassa knappkopplingar

Med en *musknappkoppling* kopplas en musknapp, eventuellt tillsammans med en modifieringstangent, till en fönsterhanterarfunktion. Knappkopplingar gäller i alla arbetsytor.

Skrivbordets standardknappkopplingar definieras i konfigurationsfilen för arbetsyتهانterearen i en knappkopplingsuppsättning som heter

DtButtonBindings:

```
Buttons DtButtonBindings
{
    ...
}
```

Syntax för knappkoppling

Syntaxen för knappkoppling är:

```
Buttons Knappkopplingsnamn
{
    [ändringsbar_parameter]<knappnamnMusätgård> kontext funktion [argument]
    [ändringsbar_parameter]<knappnamnMusätgård> kontext funktion [argument]
```

där:

knappnamn – Btn1 är vänster musknapp, Btn2 är mittenknappen (mus med tre knappar) eller båda knapparna (mus med två knappar), Btn3 är höger musknapp, Btn4 är knapparna 1 och 2 tillsammans på en mus med tre knappar och Btn5 är knapparna 2 och 3 tillsammans på en mus med tre knappar

ändringsbar_parameter – Ctrl, Shift (Skift), Alt och Lock (Skiftlås)

musåtgärd – Down – hålla ned en musknapp, Up släppa en musknapp, Click trycka på och släppa en musknapp, Click2 dubbelklicka med en musknapp och Drag dra musen medan du håller ned musknappen

kontext – visar var pekarens måste vara placerad för att kopplingen ska fungera. Dela upp innehållet med tecknet "|".

root – arbetsytetjänstfönstret, window – klientfönster eller klientfönsterram, frame – fönsterram utan innehåll, icon – ikon, title – namnlist och app – klientfönster (utan ramen)

funktion – en av arbetsyttehanterarens funktioner. En lista över giltiga funktioner finns i direkt hjälpsavsnittet dtwmrc (4).

argument – funktionsargument för fönsterhanteraren som behövs. Mer information finns i direkt hjälpsavsnittet dtwmrc (4).

Följande rad gör t ex att menyn som beskrivs i DtRootMenu visas när musknapp 3 trycks ned och pekaren är i arbetsytetjänstfönstret (men inte i ett klientfönster).

```
<Btn3Down>      root      f.menu      DtRootMenu
```

Obs! Fullständig information om syntax för knappkopplingar hittar du i dtwmrc(4) i direkt hjälpen.

▼ Lägga till knappkopplingar

1. Öppna en lämplig fil för redigering:

- Egen: *Hemkatalog*/.dt/dtwmrc
- Systemtäckande: /etc/dt/config/*språk*/sys.dtwmrc
Information om hur du skapar de här filerna finns i "Arbetsyttehanterarens konfigurationsfiler" på sidan 250.

2. Lägga till knappkopplingen i DtButtonBindings-definitionen

Koppla inte samma knapp till olika funktioner och koppla inte fler än en funktion till samma knapp eller sammanhang.

3. Välj Starta om arbetsyttehanteraren på menyn Arbetsyta.

▼ Skapa nya knappkopplingsuppsättningar

1. Öppna en lämplig fil för redigering:

- Egen: *Hemkatalog*/.dt/dtwmrc
- Systemtäckande: /etc/dt/config/*språk*/sys.dtwmrc

Information om hur du skapar de här filerna finns i "Arbetsytehanterarens konfigurationsfiler" på sidan 250.

2. Skapa en ny knappkopplingsuppsättning. Se "Syntax för knappkoppling" på sidan 259.

3. Ange det nya namnet i resursen `buttonBindings`:

```
Dtwm*buttonBindings: Knappbindningsnamn
```

4. Välj Starta om arbetsytehanteraren på menyn Arbetsyta.

Obs! De nya knappkopplingarna ersätter de befintliga. Kopiera de knappkopplingar som du vill behålla från `DtButtonBindings`.

Anpassa tangentskopplingar

Med en *tangentskoppling* (kallas också *tangenttilldelning*) kopplas olika kombinationer av tangentsnedtryckningar till funktioner i arbetsytehanteraren. Tangentskopplingar gäller för alla arbetsytor.

Obs! Använd inte en vanlig tangentskombination som tangentskoppling. Om du trycker på Skift+A infogas vanligen bokstaven "A" i det aktuella fönstret. Om du kopplar Skift+A till en funktion förlorar du möjligheten att skriva "A".

Förinställda tangentskopplingar för skrivbordet

De förinställda tangentskopplingarna anges i konfigurationsfilen för arbetsytehanteraren i en tangentskopplingsuppsättning som heter `DtKeyBindings`:

```
Keys DtKeyBindings  
{
```

```
} ...
```

Syntax för tangentkopplingar

Syntaxen för tangentkopplingar är:

```
Keys Tangentbindningsnamn
{
  [ändringsbara_parametrar]<Key>tangentnamn   kontext   funktion   [argument]
  [ändringsbara_parametrar]<Key>tangentnamn   kontext   funktion   [argument]
} ...
```

där:

Ändringsbara_parametrar – Ctrl, Shift (Skift), Alt och Lock (Skiftlås). Flera styrtangenter är tillåtna, dela av dem med blanksteg.

tangentnamn är tangenten som funktionen kopplas till. Namnet på tangenter med bokstäver eller siffror står oftast på tangenten. Tangenten "a" heter till exempel "a" och "2"-tangenten heter "2". Tab-tangenten heter "Tab". F3-tangenten heter "F3".

För övriga tangenter skrivs namnet oftast ut, till exempel plus för "+"-tangenten. Filen `keysymdef.h`, som finns i en systemberoende katalog, innehåller mer information om tangentnamn.

kontext – det element som måste vara aktivt för att operationen ska fungera. De kan slås ihop om kopplingen ska gälla i mer än ett sammanhang. Olika sammanhang delas upp med tecknet "|".

root – arbetsytbakgrund, *window* – klientfönster och *icon* – ikon

funktion – en av arbetsytahanterarens funktioner. En lista över giltiga funktioner finns i direkt hjälpsavsnittet `dtwmrc(4)`.

argument – funktionsargument för fönsterhanteraren som behövs. Mer information finns i direkt hjälpsavsnittet `dtwmrc(4)`.

Följande tangentkoppling gör att användaren kan byta tangentbordsfokus till nästa fönster i ett program genom att trycka på Alt+F6.

```
Alt<Key>F6      window      f.next_key      transient
```

Obs! Om du behöver mer information om syntax för tangentkopplingar läser du `dtwmrc(4)` i direkt hjälpen.

▼ Skapa en anpassad tangentkopplingsuppsättning

1. Öppna en lämplig fil för redigering:

- Egen: *Hemkatalog*/.dt/dtwmrc

- Systemtäckande: `/etc/dt/config/språk/sys.dtwmrc`
Information om hur du skapar de här filerna finns i "Arbetsytehanterarens konfigurationsfiler" på sidan 250.

2. **Skapa en ny tangentkopplingsuppsättning med ett unikt namn (*Tangentbindningsnamn*). Använd skrivbordets förinställda tangentkopplingsuppsättning, `DtKeyBindings`, som förebild.**

3. **Ange namnet på den nya uppsättningen i resursen `keyBindings`:**

```
Dtwm*keyBindings: Tangentbindningsnamn
```

4. **Välj Starta om arbetsytehanteraren på menyn Arbetsyta.**

Obs! De nya tangentkopplingarna ersätter de befintliga tangentkopplingarna. Kopiera de tangentkopplingar som du vill behålla från `DtKeyBindings` till den nya uppsättningen.

Växla mellan förinställda och egna uppsättningar

Växla mellan Motif-standardfunktioner och CDE-funktioner för skrivbordsfönster:

1. **Tryck på `Alt+Skift+Ctrl+!`**

2. **Klicka på OK i dialogrutan.**

Om du växlar till den förinställda metoden tas huvudgruppen samt de anpassade tangent- och knappkopplingarna bort.

Hantera programresurser, teckensnitt och färger

Du kan välja mellan ett flertal färger och teckensnitt för skärmen med hjälp av Inställningar eller genom att anpassa ytterligare teckensnitts- och färgresurser. I det här kapitlet beskrivs hur du anpassar teckensnitt och färgresurser.

I det här kapitlet beskrivs också hur du anger stildefinitioner för DtEditor-bildmodulsprogram, t ex skrivbordets textredigerare (dtpad) och Postverktyget (dtmail), samt hur du ändrar snabbval för DtEditor-bildmodulsprogram som kolliderar med dessa definitioner.

- "Ange programresurser" på sidan 265
- "Definiera UNIX-kopplingar" på sidan 267
- "Hantera teckensnitt" på sidan 271
- "Hantera färger" på sidan 277
- "Ange skuggtjocklek för programfönster" på sidan 285

Ange programresurser

Resurser används av program för att ange vissa aspekter i utseende och metod. I Inställningar (dtstyle) finns t ex resurser som gör att du kan ange var systemet ska leta efter filer som innehåller information om färgpaletter:

```
dtstyle*paletteDirectories: /usr/dt/palettes/C Hemkatalog/.dt/palettes
```

Programstandardfiler för skrivbordsprogram finns i katalogen /usr/dt/app-defaults/*språk*.

▼ Ange systemtäckande resurser

- ◆ **Lägg till resurserna i filen** `/etc/dt/config/språk/sys.resources`. (Först måste du skapa filen.)

Om du t ex anger följande i `/etc/dt/config/C/sys.resources`:

```
AnApplication*resource: värde
```

anges resursen `AnApplication*resource` i varje användares `RESOURCE_MANAGER`-egenskap vid nästa inloggning.

Obs! Mer information om datatypsdefinitioner finns i direkthjälpavsnittet (man page) för `dtstyle`. Mer information om Postverktygsresurser finns i direkthjälpavsnittet (man page) för `dtmail`.

▼ Anger egna resurser

1. **Lägg till resurserna i filen** *Hemkatalog*/`.Xdefaults`.
2. **Dubbelklicka på Ladda om resurser i programgruppen Skrivbordsverktyg.**

Hur skrivbordet startar resurser

Resurser laddas när sessionen startas av sessionshanteraren. Mer information om hur sessionshanteraren laddar resurserna i `RESOURCE_MANAGER` finns i "Ladda sessionsresurser" på sidan 27.

Processhanteraresurser

Följande processhanteraresurser är tillgängliga:

- `sampleNowTR`
- `postPopupMenuTR`
- `selectNextProcessTR`
- `selectPrevProcessTR`
- `selectFirstProcessTR`
- `selectLastProcessTR`
- `killSelectedProcessTR`

Definiera UNIX-kopplingar

UNIX-kopplingar är normalt inte aktiverade.

▼ Ange EMACS-stildefinitioner

Följande procedur:

- EMACS-tangentbindningar för DtEditor-bildmodulsprogram, t ex texthanteraren (dtpad) och Postverktyget (dtmail)
- Ändrar snabbval för DtEditor-bildmodulsprogram som krockar med dessa definitioner.

1. Lägg till följande rad i filen *Hemkatalog*/.Xdefaults:

```
#include "/usr/dt/app-defaults/language/UNIXbindings"
```

där *språk* är värdet för omgivningsvariabeln LANG.

2. Starta om sessionen.

▼ Ändra EMACS-stildefinitioner

1. Infoga innehållet i filen /usr/dt/app-defaults/*språk*/UNIXbindings i *Hemkatalog*/.Xdefaults.

2. Redigera kopplingarna i filen .Xdefaults.

3. Starta om sessionen när du är klar.

UNIX-kopplingar i filen UNIXbindings

I filen /usr/dt/app-defaults/*språk*/UNIXbindings finns de kopplingar som beskrivs i tabellen nedan.

Obs! Tangenten Delete tar bort föregående tecken när UNIX-kopplingar aktiveras och Skift+Delete tar bort nästa tecken.

Tabell 17-1 innehåller en lista över de dtpad-ersättningar för menysnabbval och den snabbvalstext som krockar med UNIX-kopplingarna.

TABELL 17-1 dtpad Åsidosätter

Snabbval för menyer och snabbvalstext	Åsidosätter
Dtpad*fileMenu.print.acceleratorText:	
Dtpad*fileMenu.print.accelerator:	
Dtpad*editMenu.undo.acceleratorText:	Ctrl+_
Dtpad*editMenu.undo.accelerator:	Ctrl<Key>_
Dtpad*editMenu.paste.acceleratorText:	Skift+Insert
Dtpad*editMenu.paste.accelerator:	Skift<Key>osfInsert
Dtpad*editMenu.findChange.acceleratorText:	Ctrl+S
Dtpad*editMenu.findChange.accelerator:	Ctrl<Key>s

Tabell 17-2 innehåller en lista över de dtmail-ersättningar för menysnabbval och den snabbvalstext för Skrivfönster som krockar med UNIX-kopplingarna.

TABELL 17-2 dtmail Åsidosättning i Skrivfönstret

Snabbval för menyer och snabbvalstext	Åsidosätter
Dtmail*ComposeDialog*menubar*Edit.Undo.acceleratorText:	Ctrl+_
Dtmail*ComposeDialog*menubar*Edit.Undo.accelerator:	Ctrl<Key>_
Dtmail*ComposeDialog*menubar*Edit.Paste.acceleratorText:	Skift+Insert
Dtmail*ComposeDialog*menubar*Edit.Paste.accelerator:	Skift<Key>osfInsert
Dtmail*ComposeDialog*menubar*Edit.Find/Change.acceleratorText:	Ctrl+S
Dtmail*ComposeDialog*menubar*Edit.Find/Change.accelerator:	Ctrl<Key>s

Följande tabell tillhandahåller meta- och ctrl-kopplingar enligt GNU EMACS-formatet samt några ytterligare kopplingar. När det behövs kan också tangenten Skift användas i kombination med den normala kopplingen för att skapa en motsatt effekt. Exempelvis flyttar Ctrl+Skift+F markören bakåt ett tecken eftersom Ctrl+F normalt flyttar den framåt ett tecken.

Ytterligare kopplingar:

Ctrl+komma – ett ord bakåt

Ctrl+Skift+komma – ett ord framåt

Ctrl+punkt – ett ord framåt

Ctrl+Skift+punkt – ett ord bakåt

Ctrl+Retur – slutet av filen

Ctrl+Skift+Retur – början av filen

GNU EMACS kopplar delete-previous-character() snarare än delete-next-character() till tangenten Delete. Meta+F är vanligen kortkommando för Fil-menyn, så kopplingen till forward-word() kommer att ignoreras. Använd någon av de andra kopplingarna för forward-word (exempelvis Ctrl+punkt).

Tabell 17-3 innehåller en lista över DtEditor.text-översättningar

TABELL 17-3 DtEditor.text översättningar

Ändringstangent	Tangent	Åtgärd
c ~s	<Key>a:	beginning-of-line()\n\
c s	<Key>a:	end-of-line()\n\
c ~s	<Key>b:	backward-character()\n\
c s	<Key>b:	forward-character()\n\
c ~s	<Key>b:	backward-character()\n\
c s	<Key>b:	backward-word()\n\
m ~s	<Key>b:	backward-word()\n\
m s	<Key>b:	forward-word()\n\
c ~s	<Key>d:	delete-next-character()\n\
c s	<Key>d:	delete-previous-character()\n\

TABELL P-3 DtEditor.text översättningar (forts.)

Ändringstangent	Tangent	Åtgärd
m ~s	<Key>d:	kill-next-word()\n\
m s	<Key>d:	kill-previous-word()\n\
c ~s	<Key>e:	end-of-line()\n\
c s	<Key>e:	beginning-of-line()\n\
c ~s	<Key>f:	forward-character()\n\
c s	<Key>f:	backward-character()\n\
m ~s	<Key>f:	forward-word()\n\
m s	<Key>f:	backward-word()\n\
c	<Key>j:	newline-and-indent()\n\
c ~s	<Key>k:	kill-to-end-of-line()\n\
c s	<Key>k:	kill-to-start-of-line()\n\
c	<Key>l:	redraw-display()\n\
c	<Key>m:	newline()\n\
c s	<Key>n:	process-up()\n\
c ~s	<Key>n:	process-down()\n\
c	<Key>o:	newline-and-backup()\n\
c ~s	<Key>p:	process-up()\n\
c s	<Key>p:	process-down()\n\
c ~s	<Key>u:	kill-to-start-of-line()\n\
c s	<Key>u:	kill-to-end-of-line()\n\
c ~s	<Key>v:	next-page()\n\
c s	<Key>v:	previous-page()\n\
m ~s	<Key>v:	previous-page()\n\

TABELL P-3 DtEditor.text översättningar (forts.)

Ändringstangent	Tangent	Åtgärd
m s	<Key>v:	next-page()\n\
c	<Key>w:	kill-selection()\n\
c ~s	<Key>y:	unkill()\n\
m	<Key>]:	forward-paragraph()\n\
m	<Key>[:	backward-paragraph()\n\
c ~s	<Key>comma:	backward-word()\n\
c s	<Key>comma:	forward-word()\n\
m	<Key>\\<:	beginning-of-file()\n\
c ~s	<Key>period:	forward-word()\n\
c s	<Key>period:	backward-word()\n\
m	<Key>\\>:	end-of-file()\n\
c ~s	<Key>Return:	end-of-file()\n\
c s	<Key>Return:	beginning-of-file()\n\
~c ~s ~m ~a	<Key>osfDelete:	delete-previous-character()\n\
~c s ~m ~a	<Key>osfDelete:	delete-next-character()

Hantera teckensnitt

Med dialogrutan Teckensnitt i Inställningar kan du välja den teckensnittsgrupp och teckensnittstorlek du vill ha för alla program. Du kan också ange teckensnitt på kommandoraden eller använda resurser för att:

- Ange teckensnittsresurser för enstaka program
- Tilldela de olika teckensnitt som ska användas av dialogrutan Teckensnitt

Ett *teckensnitt* är en typstil som tecken skrivs ut och visas med. Skrivbordet omfattar en mängd teckensnitt med olika stil och storlek.

Ett *bitmappsteckensnitt* skapas från en punktmatris. (I Inställningar konfigureras som standard bara bitmappsteckensnitt.) Hela teckensnittet ingår i en fil. Många filer behöver en fullständig uppsättning storlekar, lutningar och tjocklekar.

Teckensnitt anges som värden på resurser och parametrar till kommandon. Namnet X Logical Font Description (XLFD) är den metod som ett teckensnitt efterfrågas med. Systemet hittar det teckensnitt som bäst motsvarar den angivna beskrivningen.

Ange teckensnittsresurser för skrivbordet

Med dialogrutan Teckensnitt i Inställningar kan du markera teckensnitt (upp till sju storlekar) för t ex textposter och etiketter. Du kan också lägga till eller ta bort teckensnittsgrupper.

Resurser som anges i dialogrutan Teckensnitt

När ett teckensnitt markeras skrivs följande resurser till egenskapen RESOURCE_MANAGER:

- SystemFont används för systemområden, t ex menyrader, menyror, knappar, växlingsknappar och etiketter. Följande resurs anges av SystemFont:
 - *FontList – visas i systemområden på skrivbordsklienter och andra klienter som är skapade med verktygslådan OSF/Motif.
- UserFont används för text som skrivs i fönster. Följande resurser anges av UserFont:
 - *Font – stöder tidigare versioner av X-program
 - *FontSet – den primära inställningen
 - *XmText*FontList – visas i textinmatningsrutor
 - *XmTextField*FontList – visas i textinmatningsrutor

Resurser som används i dialogrutan Teckensnitt

Teckensnitt som används för varje val i dialogrutan Teckensnitt anges i resursfilen /usr/dt/app-defaults/Dtstyle. Upp till sju storlekar kan anges.

NumFonts – antal teckensnittsstorlekar i dialogrutan Teckensnitt

SystemFont[1-7] – upp till sju resurser som tilldelar ett specifikt teckensnitt till en markering i dialogrutan Teckensnitt för SystemFont

UserFont[1-7] – upp till sju resurser som tilldelar ett specifikt teckensnitt till en märkning i dialogrutan Teckensnitt för UserFont

Obs! Standardteckensnitten för dessa resurser har valts för läsbarheten på olika skärmar. Om du vill använda ett visst teckensnitt i ett program, anger du det med programmets teckensnittsresurs i stället för att ändra skrivbordets teckensnitt.

Du får mer information om programteckensnitt i kommandobeskrivningarna till DtStdAppFontNames(5) och DtStdInterfaceFontNames(5).

▼ Visa tillgängliga teckensnitt

1. Skriv följande:

```
xlsfonts [-alternativ] [-fn mönster]
```

En lista över XLFD-namn och teckensnittalias som är tillgängliga i systemet visas. Bitmappsteckensnitt visar värden i alla fjorton XLFD-fält. För skalbara teckensnitt visas nollor i positionerna *PixelSize*, *PointSize*, *ResolutionX* och *ResolutionY*.

2. Om du letar efter vissa teckensnitt använder du mönstermatchningen för `xlsfonts`. Använd jokertecken för att ersätta den del av mönstret som du inte försöker matcha.

3. Om `xlsfonts` inte visar några teckensnittsnamn som börjar på `dt`, innehåller inte teckensnittssökvägen skrivbordsteckensnitt. Skriv följande kommando för att få med skrivbordsteckensnitten bland de tillgängliga teckensnitten:

```
xset +fp katalognamn
```

där *katalognamn* är den katalog som innehåller skrivbordsteckensnitten. Standardplatsen som anges vid sessionsstarten är `/usr/dt/config/xfonts/språk`.

Ytterligare information hittar du i:

- Kommandobeskrivningarna till `xset` och `xlsfonts`, som listar tillgängliga alternativ.
- *Using the X Window System* som beskriver teckensnittalias och `xset`-klienten.

▼ Ange teckensnitt på kommandoraden

♦ Ange en teckensnittsresurs för en viss klient med kommandoradsalternativet `-xrm`. Till exempel:

XLFD (X Logical Font Description)

Ett teckensnitt anges genom att fjorton olika egenskaper listas, avgränsade med bindestreck (-). Detta kallas XLFD (X Logical Font Description). I vissa fall kan en egenskap i listan ersättas av ett jokertecken (*), och ett tecken i en egenskap kan ersättas av jokertecknet ?. Tabell 17-4 listar specifikationer för teckensnittegenskapssträngar.

Formatet på egenskapssträngarnas specifikationer är:

"-*Foundry-FamilyName-WeightName-Slant-SetwidthName-AddStyleName-PixelSize-PointSize-ResolutionX-ResolutionY-Spacing-AverageWidth-CharSetRegistry-CharSetCoding*"

TABELL 17-4 Specifikationer till teckensnittens egenskapssträngar

Egenskapssträng	Beskrivning
<i>Foundry</i>	En sträng som anger vem som har utformat teckensnittet
<i>FamilyName</i>	En sträng som anger varumärkesnamnet på teckensnittet
<i>WeightName</i>	En sträng som anger relativ tjocklek på teckensnittet, t ex fetstil
<i>Slant</i>	En kod som beskriver riktning på lutningen R (Romersk – ingen lutning) I (Kursiv – lutning åt höger) O (Snedställd – lutning åt höger) RI (Omvänd kursiv – lutning åt vänster) RO (Omvänd snedställd – lutning åt vänster)
<i>SetwidthName</i>	En sträng som beskriver bredd, t ex komprimerad eller utökad
<i>AddStyleName</i>	En sträng som innehåller eventuell ytterligare information som behövs för unik identifiering av teckensnittet
<i>PixelSize</i>	Ett heltal som anger storleken på en så kallad helfyrkant i bildpunkter
<i>PointSize</i>	Ett heltal som anger storleken på en så kallad helfyrkant i decipunkter

TABELL P-4 Specifikationer till teckensnittens egenskapssträngar (forts.)

Egenskapssträng	Beskrivning
<i>ResolutionX</i>	Ett heltal som anger vågrät upplösning i bildpunkter
<i>ResolutionY</i>	Ett heltal som anger lodrät upplösning i bildpunkter
<i>Spacing</i>	En kod som anger mellanrum mellan enheter M (Fast teckenindelning-fast breddsteg) P (Proportionellt breddsteg-proportionellt breddsteg) C (Teckencell)
<i>AverageWidth</i>	Ett heltal som anger genomsnittsbredden i 1/10 bildpunkter
<i>CharSetRegistry</i>	En sträng som anger den myndighet som har auktoriserat teckensnittets kodning
<i>CharSetEncoding</i>	En sträng som anger registrets teckenuppsättning

Exempel

Följande XLFD-namn beskriver ett teckensnitt med namnet charter som har skapats av Bitstream och stöder standardkod ISO8859-1:

```
-bitstream-charter-medium-r-normal--8-80-75-75-p-45-iso8859-1
```

Teckensnittet har normal tjocklek utan någon speciell lutning och normal bredd. Teckensnittet är proportionellt med en helfyrkant på 8 bildpunkter eller 8,0 punkter. Den vågräta och lodräta upplösningen är båda på 75 bildpunkter. Snittbredden på tecknen är 45 tiondelsbildpunkter eller 4,5 bildpunkter.

Delar av den här strängen kan ersättas med jokertecken. Systemet använder det första teckensnitt som påträffas som matchar de delar du anger.

Om du bara vill ha ett åtta bildpunkters charter-teckensnitt använder du:

```
**charter-*-**-8-*
```

Visa markerade teckensnittsgruppsattribut

Genom att välja knappen Attribut i dialogrutan Teckensnitt i Inställningar, kan du visa följande teckensnittsgruppsattribut:

- Teckensnittsgrupp

- Storlek, Sidformat, Grad
- Alias
- Alias XLFD
- Aliasadress
- Teckensnitt
- Font XLFD

Lagring av användares filsystem för teckensnittsgrupper

När en användare lägger till en teckensnittsgrupp lagras den på följande adress:

Hemkatalog/ .dt/sdtfonts/värd/språkområde/teckensnitt-nnnnnn

värddator är den lokala arbetsstationens värddatornamn.

språkområde är användarens aktuella språkområde, t ex "C" eller "ja".

teckensnitt-nnnnnn är ett namn som fås från det markerade teckensnittet och ett unikt genererat tal.

Den här teckensnittsgruppskatalogen innehåller följande tre filer

- `fonts.alias`
- `fonts.dir`
- `sdtfonts.group`

Filerna `fonts.alias` och `fonts.dir` är normala X11-teckensnittsfiler som kan läggas till i X-serverns teckensnittssökväg. Filen `sdtfonts.group` innehåller det användardefinierade teckensnittsgruppsnamnet.

Skapa teckensnittsgrupp för systemadministratör

För att andra användare ska kunna komma åt teckensnittsgrupper på en arbetsstation kan systemadministratören kopiera teckensnittsgrupper till katalogen `/etc/dt/sdtfonts/språkområde` eller till katalogen `/usr/openwin/lib/X11/stdfonts/språkområde`. Sessionshanteraren letar först i *Hemkatalogen* `/ .dt/stdfonts/värd/språkområde`, därefter i `/etc/dt/sdtfonts/språkområde` och slutligen i `/usr/openwin/lib/X11/stdfonts/språkområde`.

Hantera färger

I det här avsnittet beskrivs:

- Hur du anger skärmfärger i Inställningar
- Resurser som används av Inställningar för att kontrollera hur skrivbordet använder färger

Färgpaletter

En palett består av en grupp färguppsättningar. Färguppsättningen för den aktuella paletten visas i dialogrutan Inställningar - Färg.

För varje palett finns en fil. Resursen `paletteDirectories` anger de kataloger som innehåller palettfiler. Normalt innehåller den här resursen:

- Inbyggda paletter: `/usr/dt/palettes`
- Paletter för hela systemet: `/etc/dt/palettes`
- Egna paletter: *Hemkatalog*/`.dt/palettes`

Färguppsättningar

Varje färguppsättning i aktuell palett representeras av en färgknapp i dialogrutan Inställningar - Färg. Varje färg identifieras av ett färguppsättnings-ID (ett tal mellan 1 och 8).



Figur 17-1 Värden på färguppsättnings-ID för HIGH_COLOR

Varje färguppsättning utgörs av upp till fem färger. Varje färgknapp visar bakgrundsfärgen på färguppsättningen. De fem färgerna i varje färguppsättning representerar följande komponenter:

foreground – förgrunden i ett programfönster eller en fönsterram. Den är alltid svart eller vit. Den används ofta för text i fönster och rubriker.

background – bakgrunden i ett programfönster eller en fönsterram.

topShadowColor – färgen på övre och vänstra ytan för programkontroller (t ex knappar) och fönsterramar.

bottomShadowColor – färgen på nedre och högra ytan för programkontroller och fönsterramar.

selectColor – färgen som anger aktivt läge för vissa kontroller, t ex aktiva växlar och knappar.

Antalet färguppsättningar som används av varje palett bestäms av resursen `colorUse`, som användaren kan ange med hjälp av dialogrutan `Antal färger` i `Inställningar`.

Ange färgvärden

Inställningar använder RGB-värden när information skrivs till palettfilerna. Syntaxen för RGB-tal är:

`#RödGrönBlå`

Röd, *Grön* och *Blå* är hexadecimala tal som vart och ett är 1 till 4 siffror långt. De anger vilken mängd av färgen som ska användas. Du måste ange samma antal siffror för var och en av färgerna. Giltiga färgvärden består därför av 3, 6, 9 eller 12 hexadecimala siffror.

Vit kan t ex anges på något av följande sätt:

```
#fff
#ffffff
#ffffffff
#ffffffffffffff
```

Om du anger färgresurser direkt använder du antingen färgnamnet eller RGB-värdet. Filen `/usr/lib/X11/rgb.txt` innehåller en lista över alla namngivna färger.

Hur färguppsättningar tilldelas resurser

Skrivbordssystemet kopplar färguppsättningar till olika visningsselement via resurser. Tilldelningarna visas i Tabell 17-5.

TABELL 17-5 Färguppsättningar som tilldelats resurser

Resurs	Visningsselement
<code>activeColorSetId</code>	Ramfärg för det aktiva fönstret.
<code>inactiveColorSetId</code>	Ramfärg för inaktiva fönster
<code>textColorSetId</code>	Områden för textinmatning
<code>primaryColorSetId</code>	Huvudbakgrunder för program
<code>secondaryColorSetId</code>	Menyfält, menyer och dialogrutor för program

Dessa resurser använder ett färguppsättnings-ID som värde. Genom att skärmelement anges med färguppsättnings-ID, kan elementet dynamiskt ändras till det nya färgschemat när en ny palett markeras i Inställningar.

Du använder de här resurserna för enstaka program. Följande rad visar hur du visuellt grupperar alla `dtterm`-fönster genom att använda färguppsättning 8 för huvudfärgen.

```
dtterm*primaryColorSetId: 8
```

Standardtilldelningar av färguppsättningar

Färguppsättnings-ID som används till skärmelement är beroende av inställningen Ange antal färger i Inställningar:

Tabell 17-6 visar färguppsättnings-ID för många färger (färguppsättningar med åtta färger) – inställningen Fler skrivbordsfärger i Inställningar.

TABELL 17-6 Fler skrivbordsfärger

Färguppsättnings-ID	Visningselement
1	Ramfärg för det aktiva fönstret.
2	Ramfärg för inaktiva fönster
3	Oanvända (standard)
4	Ytor för textinmatning
5	Huvudbakgrunder för program
6	Menyfält, menyer och dialogrutor för program
7	Oanvänd som standard
8	Huvudgruppens bakgrund

Tabell 17-7 visar färguppsättnings-ID för normala färger (fyrfärgsuppsättningar) – Inställningen Fler programfärger i Inställningar.

TABELL 17-7 Fler programfärger

Färguppsättnings-ID	Visningselement
1	Ramfärg för det aktiva fönstret.
2	Ramfärg för inaktiva fönster

TABELL P-7 Fler programfärger (forts.)

Färguppsättnings-ID	Visningsselement
3	Bakgrundsfärg för program och huvudgrupp
4	Ytor för textinmatning

Tabell 17-8 visar färguppsättnings-ID för få färger (tvåfärgsuppsättningar) – inställningen Ännu fler programfärger i Inställningar.

TABELL 17-8 Få färger

Färguppsättnings-ID	Visningsselement
1	Aktiv fönsterram, knappar för markering av arbetsyta
2	Alla andra skärmelement

Kontrollera färg med Inställningar

Du kan dynamiskt ändra färg på skrivbordsprogram och andra samverkande program via Inställningar. De förgrunds- och bakgrundsfärger som anges med Inställningar är även tillgängliga för icke samverkande program.

För att klienten ska reagera på färgändringar i Inställningar, måste den använda skrivbordsbiblioteket Motif. Klienter som skrivs med andra verktyg kan inte ändra färg dynamiskt efter ändringar i Inställningar. Färgändringar för dessa klienter träder inte i kraft förrän klienten startas om.

Det får inte finnas några andra specifika färgresurser för klienten. Detta omfattar användarspecificerade resurser, programstandardfiler och resurser som är inbyggda i programmet.

Klienter kan ange resurserna `primaryColorSetId` och `secondaryColorSetId` för att använda vissa färger i en skrivbordspalette.

Antal färger som används i Inställningar

Antalet färger som används i Inställningar är beroende av värdena på följande resurser:

`colorUse` – konfigurerar antalet färger som skrivbordet använder

`shadowPixmaps` – instruerar skrivbordssystemet att ersätta de två skuggningsfärgerna med bildpunktsbeskrivningar

`foregroundColor` – anger om förgrundsfärgen ska ändras dynamiskt

`dynamicColor` – bestämmer om program ska ändra färg när du växlar paletter

Tabell 17–9 innehåller en lista över det högsta antalet färger som kan tilldelas av skrivbordet.

TABELL 17–9 Antal skrivbordsfärger

Skärm	Maximalt antal färger	Antalet är härlett från
B_W	2	Svart och vitt
LOW_COLOR	12	Två färguppsättningar gångar fem färger plus svart och vitt
MEDIUM_COLOR	22	Fyra färguppsättningar gångar fem färger plus svart och vitt
HIGH_COLOR	42	Åtta färguppsättningar gångar fem färger plus svart och vitt.

Avgöra maximalt antal färger

1. Multiplicera antalet färguppsättningar i paletten med antalet färger i varje färguppsättning.

2. Lägg till 2 (för svart och vitt).

Med följande konfiguration har du emellertid bara tio färger i paletten: fyra färguppsättningar gångar två färger i varje uppsättning: (`background` och `selectColor`) plus svart och vitt:

```
*colorUse: MEDIUM_COLOR
*shadowPixmaps: True
*foregroundColor: White
```

Obs! Flerfärgade ikoner använder ytterligare fjorton färger.

Resursen colorUse

Standardvärdet på resursen `colorUse` är `MEDIUM_COLOR`. Värdet på den här resursen påverkar antalet färger som används i paletten. Andra resurser påverkar antalet färger som används för att skapa skuggor. Värdet på resursen `colorUse` påverkar också användningen av flerfärgade ikoner.

Värde	Beskrivning
<code>B_W</code>	"Svartvit" -inställning i Inställningar. Bildskärmar med 1 till 3 färgplan. Antal färguppsättningar: 2. Maximalt antal färger 2. Förvalt antal färger: 2. Inga flerfärgsikoner.
<code>LOW_COLOR</code>	"Ännu fler programfärger" -inställning i Inställningar. Bildskärmar med 4 till 5 färgplan. Antal färguppsättningar: 2. Maximalt antal färger 12. Förvalt antal färger: 12. Inga flerfärgsikoner.
<code>MEDIUM_COLOR</code>	"Fler programfärger" -inställning i Inställningar. Bildskärmar med 6 färgplan. Antal färguppsättningar: 4. Maximalt antal färger 22. Förvalt antal färger: 22. Flerfärgsikoner.
<code>HIGH_COLOR</code>	"Fler skrivbordsfärger" -inställning i Inställningar. Bildskärmar med 7 färgplan eller fler. Antal färguppsättningar: 8. Maximalt antal färger 42. Förvalt antal färger: 42. Flerfärgsikoner.
<code>standardval</code>	Skrivbordet väljer korrekt värde för den skärmen. (Om du vill minska antalet färger som används av skrivbordet för skärmar med många färger anger du standardresursen <code>colorUse</code> till <code>MEDIUM_COLOR</code> .)

Resursen shadowPixmaps

Resursen `shadowPixmaps` gör så att skrivbordet ersätter de två skuggfärgerna med bildpunktsmönster. Dessa bildpunktsmönster blandar bakgrundsfärgen med svart eller vitt för att simulera övre och undre skugga. Detta minskar antalet färger som behövs med två, eftersom färgceller inte måste allokeras för skuggfärgerna.

Värde	Beskrivning
<code>True</code>	Skrivbordssystemet skapar en <code>topShadowPixmap</code> och en <code>bottomShadowPixmap</code> som används i stället för skuggningsfärgerna.
<code>False</code>	<code>topShadowColor</code> och <code>bottomShadowColor</code> från paletten används.

Standardvärdet för `shadowPixmaps` beror på din `colorUse`-resurs och på maskinvarustödet för skärmen.

Resursen foregroundColor

Resursen foregroundColor anger hur förgrunden är konfigurerad i en palett.

Inställning	Resultat
White	Förgrunden anges till vit.
Black	Förgrunden anges till svart.
Dynamic (Standard)	Förgrunden anges dynamiskt till svart eller vit, beroende på värdet för background. Vita tecken på gul bakgrund är svåra att läsa så systemet väljer automatiskt alternativet svart.

Om foregroundColor anges till antingen Black eller White, minskas antalet färger i färguppsättningen med en, och förgrunden ändras inte när bakgrundsfärgen ändras.

Standardvärdet för foregroundColor är Dynamic, utom när värdet för colorUse är B_W.

Resursen dynamicColor

Resursen dynamicColor styr om program ändrar färg dynamiskt, dvs om klienter ändrar färg när du växlar palett.

Värde	Beskrivning
True	Klienter ändrar färg dynamiskt när en ny palett markeras. Det här är standardvärdet.
False	Klienter ändrar inte färg dynamiskt. När en ny palett markeras använder klienter de nya färgerna när sessionen startas om.

När värdet på resursen dynamicColor är True, tilldelas klienter som inte kan ändra färger dynamiskt (icke-Motif-program) andra celler i färgkartan än klienter som kan ändra färger dynamiskt, även om du ser samma färg.

Obs! Eftersom alla klienter kan dela samma färgceller kan du minska antalet färger som ditt skrivbord använder genom att ange dynamicColor till False.

Ange skuggtjocklek för programfönster

Skrivbordet definierar standardtjockleken för skugga till en bildpunkt för komponenterna i programfönstret, t ex knappskuggor och fokusmarkering. Motif 1.2-program använder det här resursvärdet. Andra program kan ha svårt att hitta resursvärdet och visas därför annorlunda på skärmen.

Om du vill ange skuggans tjocklek till en bildpunkt för icke-Motif 1.2-program gör du så här:

1. **Logga in som root.**
2. **Skapa filen** `/etc/dt/config/språk/sys.resources`.
3. **Ange den programspecifika resursen i**
`/etc/dt/config/språk/sys.resources` **så här:**

```
programklassnamn*XmCascadeButton*shadowThickness: 1
```

Mer information om hur du åsidosätter systemets standardresurser och anger ytterligare resurser för alla skrivbordsanvändare finns i "Ladda sessionsresurser" på sidan 27.

Konfigurera språkpassade skrivbordssessioner

Om du vill konfigurera språkpassade skrivbordssessioner måste du:

- Ställa in omgivningsvariabeln `LANG` och övriga omgivningsvariabler för NLS (National Language Support)
- Använda språkberoende meddelandekataloger och resursfiler
- Köra fjärrprogram via internationella system
- "Hantera omgivningsvariabeln `LANG` " på sidan 287
- "Söka efter teckensnitt" på sidan 290
- "Språkpassa resursfilerna `app-defaults`" på sidan 290
- "Språkpassa åtgärder och datatyper" på sidan 291
- "Språkpassa ikoner och bitmappar" på sidan 291
- "Språkpassa hjälpolymer" på sidan 293
- "Språkpassa meddelandekataloger" på sidan 293
- "Köra språkpassade fjärranslutna skrivbordsprogram" på sidan 293
- "Återställa tangentbordstabellen" på sidan 294

Hantera omgivningsvariabeln `LANG`

Omgivningsvariabeln `LANG` måste vara inställt för användning av operativsystemets språkpassade rutiner. Skrivbordet stöder:

- Västeuropeiska språk som skrivs med latinska bokstäver

- Japanska
- Traditionell kinesiska
- Förenklad kinesiska
- Koreanska

Obs! Stöd för andra språk kan ha lagts till av skrivbordsleverantören.

Du kan ställa in `LANG` på alla värden som stöds av operativsystemet. Alternativ-menyn på inloggningsbilden visar en lista över språk och områden som stöds.

Det finns fyra olika sätt att ange `LANG` för skrivbordet. Du kan:

- Redigera en resurs i filen `Xconfig`
- Använda Alternativ-menyn på inloggningsbilden
- Skapa en körbar `sh-` eller `ksh-` `Xsession.d`-kommandofil. (Mer information om hur man använder en `Xsession.d`-kommandofil finns i "Läsa in och köra skriptet `Xsession.d`" på sidan 24.)
- Redigera användarens `.dtprofile`-fil

När `LANG` har ställts in använder skrivbordet följande språkberoende filer för att bestämma det språkpassade gränssnittet.

Färger – `/usr/dt/palettes/desc.språk`

Bakgrunder – `/usr/dt/backdrops/desc.språk`

Ange språk för flera användare

Om du anger språk med hjälp av en `Xconfig`-fil, används en översatt inloggnings-skärm, och `LANG` anges för alla användare. Detta är det enda sättet att ange `LANG` för alla visningsenheter i ett system med flera visningsenheter. (Om du vill ändra `Xconfig`, kopierar du `/usr/dt/config/Xconfig` till `/etc/dt/config/Xconfig`.)

Språket anges genom att du lägger till följande rad i `/etc/dt/config/Xconfig`:

```
dtlogin.värd_skärm.language: språk
```

Följande rad anger till exempel `LANG` till `Swedish_locale` på visningsenheten `my_host:0`.

```
dtlogin.my_host_0.language: Swedish_locale
```

Klienten `dtlogin` läser in den lämpliga meddelandekatalogen för det språket och visar den språkpassade inloggningsbilden. Klienten `dtlogin` anger en lista över språk som använder de följande resurserna i resursfilen `/etc/dt/config/Xresources`:

- `dtlogin*language`
- `dtlogin*languageList`
- `dtlogin*languageName`

Filen `Xconfig` kan behöva ställa in omgivningsvariabeln `NLSPATH` för det valda språket. Om detta inte behövs eller om du vill ställa in `NLSPATH` själv, kan du läsa mer om detta i "Omgivningsvariabeln `NLSPATH`" på sidan 290.

Ange språk för en session

Om du vill ange språk för en session använder du Alternativ-menyn på inloggningsskärmen. Inloggningsskärmen är översatt och `LANG` är inställd för användaren. `LANG` återgår till standardinställningen (som anges i `dtlogin`) när sessionen avslutas.

Ange språk för en användare

En användare kan åsidosätta `LANG` -inställningen för inloggningen i filen *Hemkatalog*/`.dtprofile`. Inloggningsskärmen är inte översatt och `LANG` är inställd för användaren.

- Om du använder `sh` eller `ksh`:

```
LANG=språk
export LANG
```

- Om du använder `csh`:

```
setenv LANG språk
```

Omgivningsvariabeln `LANG` och sessionskonfiguration

Omgivningsvariabeln `LANG` ändrar namnet på den katalog som genomförs efter konfigurationsfilerna för din session.

Den språkanpassade sessionens konfigurationsfiler är:

- `/usr/dt/config/språk/Xresources` (resursfil till inloggningsskärmen)
- `/usr/dt/config/språk/sys.font` (resursfil till sessionskärmen)
- `/usr/dt/config/språk/sys.resources` (resursfil till sessionskärmen)
- `/usr/dt/config/språk/sys.session` (körbart skal till sessionskärmen)
- `/usr/dt/config/språk/sys.dtwmrc` (resursfil till fönsterhanteraren)

- `/usr/dt/appconfig/types/språk/dtwm.fp` (huvudgrupp till fönsterhanteraren)

Ange andra NLS-omgivningsvariabler

Förutom `LANG` finns det andra NLS-omgivningsvariabler, t ex `LC_CTYPE` och `LC_ALL`. Dessa variabler påverkas inte av språkresursen `dtlogin` eller av Alternativ-menyn i inloggningsbilden. De måste anges i följande filer:

- Systemtäckande variabler: `/etc/dt/config/Xsession.d`
- Egna variabler: *Hemkatalog*/`.dtprofile`

Omgivningsvariabeln NLSPATH

Omgivningsvariabeln `NLSPATH` bestämmer vilka sökvägar som program ska söka efter meddelandekataloger i. Både `LANG` och `NLSPATH` måste anges för att det ska gå att använda dessa meddelandekataloger. Mer information om var du hittar översatta meddelanden finns i "Språkanpassa meddelandekataloger" på sidan 293. De flesta skrivbordsklienter kommer att använda sökvägsprefixet `NLSPATH` vid start.

Söka efter teckensnitt

Skrivbordets teckensnitt finns i katalogen `/usr/lib/X11/fonts`. Varje katalog innehåller en katalogfil (`fonts.dir`) och en aliasfil (`fonts.alias`). I direkt hjälpen `mkfontdir` finns information om hur du skapar filerna `fonts.dir` och `fonts.alias`.

Om du vill visa en lista med alla tillgängliga teckensnitt på servern använder du kommandot `xlsfonts`. Om du vill lägga till eller ta bort teckensnitt på servern använder du kommandot `xset`.

Språkanpassa resursfilerna app-defaults

Standardsökvägen för filen `app-defaults` för skrivbordsklienten är `/usr/dt/app-defaults/språk`. Om till exempel `LANG` är angiven som `Swedish_locale`, kommer program att söka efter filen `app-defaults` i `/usr/dt/app-defaults/Swedish_locale`. Om `LANG` inte är angiven, ignoreras `språk`, och programmen söker efter filen `app-defaults` i `/usr/app-defaults/C`.

Om du vill flytta `app-defaults`, använder du omgivningsvariabeln `XFILESEARCHPATH`. Om du till exempel vill flytta `app-defaults` till `/users`, ställer du in `XFILESEARCHPATH` på `/usr/app-defaults/språk/klassnamn`.

Om du ställer in `XFILESEARCHPATH` på `Hemkatalog/.dtprofile`, gäller detta värde för alla skrivbords- och X-klienter som du kör. Icke-klienter kommer inte att hitta sina resursfiler om du inte länkar eller kopierar dem till katalogen som anges i `XFILESEARCHPATH`.

Språkanpassa åtgärder och datatyper

Obs! Om du vill anpassa en fil i katalogen `/usr/dt/appconfig`, kopierar du filen till katalogen `/etc/dt/appconfig` innan du anpassar den.

Sökvägen för åtgärds- och datatypsdefinitionsfiler omfattar språkberoende kataloger i:

- Egen: `Hemkatalog/dt/types`
- Systemtäckande: `/etc/dt/appconfig/types/språk`
- Inbyggt: `/usr/dt/appconfig/types/språk`

Sökvägen för programhanterarens konfigurationsfiler är:

- Egen: `Hemkatalog/dt/appmanager`
- Systemtäckande: `/etc/dt/appconfig/appmanager/språk`
- Inbyggt: `/usr/dt/appconfig/appmanager/språk`

Fil- och katalognamn i den här katalogen är språkanpassade.

Språkanpassa ikoner och bitmappar

Om du vill språkanpassa en ikon redigerar du den i ikonredigeraren och sparar den i:

`/etc/dt/appconfig/icons/språk`

Om du sparar den i en annan katalog inkluderar du den nya katalogen i omgivningsvariabeln `XMICONSEARCHPATH`. Omgivningsvariabeln `XMICONBMSEARCHPATH` anger vilken sökväg som ska användas vid sökning efter ikoner.

Språkanpassa bakgrundsnamn

Bakgrunder språkanpassar du med hjälp av beskrivningsfilerna (`desc.språk` och `desc.backdrops`). Det finns ingen språkanpassad katalog (till exempel `/usr/dt/backdrops/språk`) för bakgrundsfiler. Alla språk använder samma uppsättning bakgrundsfiler men har sina egna `desc.språk`-filer som innehåller de översatta namnen på bakgrunderna.

Beskrivningsfilen innehåller resursspecifikationer för de översatta bakgrundsnamnen. Till exempel:

```
Backdrops*Corduroy.desc:    Velours
Backdrops*DarkPaper.desc:  PapierKraft
Backdrops*Foreground.desc: AvantPlan
```

Filen `desc.språk` används för att hämta beskrivningsfilen för detta språk så att bakgrunden kan visas i Inställningar. Om det finns någon beskrivningsspecifikation visas den i bakgrundslistan i Inställningar. I annat fall används bakgrundens filnamn.

Användarna kan lägga till egna bakgrundsbeskrivningar i filen `Hemkatalog/.dt/backdrops/desc.backdrops`. Filen används för att hämta bakgrundsbeskrivningar för alla bakgrunder oavsett språk.

Sökvägen för filen `description` är:

- Egen: `Hemkatalog/.dt/backdrops/desc.backdrops`
- Systemtäckande: `/etc/dt/backdrops/desc.språk`
- Inbyggt: `/usr/dt/backdrops/desc.språk`

Språkanpassa palettnamn

Paletter språkanpassar du med hjälp av beskrivningsfilerna (`desc.språk` och `desc.palettes`). Det finns ingen språkanpassad katalog (till exempel `/usr/dt/palettes/språk`). Alla språk använder samma uppsättning palettfiler, men de har sina egna `desc.palettes`-filer som innehåller de översatta namnen på paletterna.

Beskrivningsfilen innehåller resursspecifikationer för de översatta palettnamnen. Till exempel:

```
Palettes*Cardamon.desc:    Cardamone
Palettes*Cinnamon.desc:   Cannelle
Palettes*Clove.desc:       Brun
```

Filen `desc.språk` används för att hämta beskrivningsfilen för detta *språk* så att paletten kan visas listan i Inställningar. Om det finns en beskrivningsspecifikation visas den i palettlistan i Inställningar. I annat fall används palettens filnamn.

Användarna kan lägga till egna palettbeskrivningar i filen `Hemkatalog/.dt/palettes/desc.palettes`. Filen används för att hämta palettbeskrivningar för alla paletter oavsett språk.

Sökvägen för beskrivningsfilerna är:

- Egen: `Hemkatalog/.dt/palettes/desc.palettes`
- Systemtäckande: `/etc/dt/palettes/desc.språk`
- Inbyggt: `/usr/dt/palettes/desc.språk`

Språkanpassa hjälpvolymen

Om du har språkanpassat en hjälpvolym måste du lagra den i någon av följande kataloger. Den första hjälpvolymen som påträffas används. Katalogerna söks igenom i följande ordning:

- Egen: `Hemkatalog/.dt/help`
- Systemtäckande: `/etc/dt/appconfig/help/språk`
- Inbyggt: `/usr/dt/appconfig/help/språk`

Språkanpassa meddelandekataloger

Om du har språkanpassat en meddelandekatalog lagras du den i följande katalog:

`/usr/dt/lib/nls/msg/språk`

De här katalogerna innehåller *.cat-filer.

Köra språkanpassade fjärranslutna skrivbordsprogram

Du kan anropa språkanpassade skrivbordsprogram på alla värddatorer för fjärrkörning som har en liknande språkanpassad skrivbordsinstallation. Värdena på

de NLS-relaterade omgivningsvariablerna på värddatorn som anropar programmet skickas till fjärrvärddatorn när programmet startas. Omgivningsvariablerna innehåller dock ingen värddatorinformation.

Återställa tangentbordstabellen

Om tecken ser konstiga ut och oväntade saker händer på skärmen, eller om tecken inte kan visas eller skrivas, måste du kanske återställa eller installera om tangentbordstabellen eller ändra inmatningsmetoden.

Inmatningsmetoden avgörs av omgivningsvariablerna `LC_CTYPE`, `LANG` eller `LC_ALL` eller det språk som är angivet i alternativet `-lang`.

Om användaren till exempel vill öppna en terminal med hjälp av språket `C` i ett POSIX-skal:

```
LANG=C dtterm
```

Den nya terminalen använder språket `C` med `C`-inmatningsmetoder och `C`-teckensnitt. Om du använder ett språkanpassat tangentbord är det möjligt att inmatningsmetoden inte stöder utökade tecken. När du använder språkområdet `C` med ett språkanpassat tangentbord måste du ange ett lämpligt värde i någon av omgivningsvariablerna `LC_CTYPE`, `LANG` eller `LC_ALL` innan du anropar terminalen.

Om du till exempel vill använda språket `C` med ett tyskt tangentbord, skriver du:

```
LANG=C LC_CTYPE=DeDE dtterm
```

Om X-servern har startats om och tangenttabeller har initierats, kan du återställa den korrekta tangenttabellen på servern med kommandot `xmodmap`.

Direkthjälp (man page) för dtconfig(1)

NAME
dtconfig - desktop configuration utility

SYNOPSIS
dtconfig [-d |-e |-kill |-reset |-p |-inetd |-inetd.ow]

DESCRIPTION
Desktop configuration utility. Integrates CDE with the operating system of the underlying platform. System root login privilege is required to use dtconfig.

OPTIONS

- d Disables desktop auto-start feature. At end of boot cycle, platform's native text based login mechanism will be used.
- e Enable's desktop auto-start feature. Desktop login
- kill Kill desktop (window based) login process and any user sessions associated with it. Return control to system's native text based console.
- reset Tell desktop (window based) login process to reread its configuration file to incorporate any changes.
- p Printer actions for any printer known to platform will be created if such print actions do not already exist in the platform's actions database. This option is executed automatically at boot time if desktop auto-start has been enabled.
- inetd Adds /usr/dt/bin daemons to the /etc/inetd.conf file. Specific CDE background daemon setup includes rpc.ttdbserverd (ToolTalk), rpc.cmsd (Calendar Manager), and dtspcd (subprocess control). This -inetd option is called automatically by Solaris CDE package installs. This -inetd option is also useful for CDE daemon setup outside of normal Solaris CDE install, including system setup where /usr/dt has simply been mounted from some remote fileserver exporting the

(forts.)

/usr/dt directory.

-inetd.ow
Switches the ToolTalk and Calendar Manager daemons (rpc.ttdbserverd & rpc.cmsd) start lines in /etc/inetd.conf back to the older /usr/openwin/bin area. This option is called automatically by Solaris CDE package remove scripts when needed. It is also useful outside of normal Solaris package remove operations when /usr/dt is about to be manually removed or unmounted.

RETURN VALUES
0 Successful completion
>0 Error condition

FILES
/usr/dt/bin/dtconfig location of dtconfig utility

SEE ALSO
dtlogin (1), dtprintinfo (1)

Index

Special Characters

- * , jokertecken, 205
- ? jokertecken, 205

A

- actionIcon-resurs, 176
- ACTIONS, fält, 203
- activeColorSetId-resurs, 279
- aktuell session, 22
- ALTERNATE_ICON, fält, 240
- Alternativ-menyn
 - språk, 93
- AND, operator i fältet MODE, 207
- ANIMATION, definition, 242
- animering för huvudgruppen, 242
- anpassa utskrift av brev, 96
- AnswerBook-dokumentation, lägga till i nätverk, 97
- använda X-terminaler, 92
- användar-ID, 107
- användare, ändra för åtgärd, 193
- användarkonton, 107
- app-defaults
 - skrivbordsprogram, 265
 - språkoberoende, 290
- /appconfig, 84
- /app-defaults, 84
- Apple Macintosh-programomgivning, 99
- arbetsstationer, som X-terminaler, 94
- Arbetsyta, menyn, *Se* menyn Arbetsyta, arbetsytahanterare
 - anpassningar för hela systemet, 251
 - definition, 250
 - egen anpassning, 250
 - funktioner, 255
 - hantera huvudgruppen, 250
 - konfigurationsfiler, 250
 - lägga till andra filer, 251
 - menyer, 254
 - musknappskopplingar, 259
 - starta, 23, 29
 - starta om, 252
 - ändra till Motif, 263
- arbetsytor
 - anpassa, 252
 - antal, 252
 - namn, 253
 - ändra standardantal, 244
- arbetsytor, *Se* bakgrunder bakgrunder,
- argument
 - antal för åtgärder, 188
 - begränsa för åtgärder, 187
 - efterfråga, 181
 - flera för åtgärder, 183
 - för åtgärder, 148, 180
 - icke utbytbara, för åtgärder, 184
 - icke-fil, 182
 - utbytbara för åtgärder, 184
- Arg_1, syntax, 148
- ARG_CLASS, fält, 178
- ARG_COUNT, fält, 178, 188
- ARG_MODE, fält, 178
- Arg_n syntax, 181
- ARG_TYPE, fält, 178, 188, 203

skriva ut, 127
avmontera en monterad CDE-katalog, 89

B

%B, 131
background-resurs, 278
bakgrunder, 250

- använda grafiska bilder, 254
- filplaceringar, 253
- lägga till, 253

basfilnamn, 202
behörighet, X, 109
behörighetsmönster, angivet med Skapa åtgärd, 163
bilder, *Se* ikoner,
bildpunktsbeskrivningar

- hur filer hittas, 214, 216
- namnkonventioner, 214
- sökvägar, 216

/bin, 84
bitmappar, 214

- hur filer hittas, 214, 216
- namnkonventioner, 214
- sökvägar, 216

.bm, filnamnstillägg, 214
bottomShadowColor-resurs, 278
BOX, definition, 226

- syntax, 226

BROADCAST, används i indirekt XDMCP, 8
-broadcast-flagga, 95
buttonBindings, resurs, 261

C

C, språkområde, 94
CDE-MIN-filer, 110
CDE-TT-filer, 110
CHOOSER, sträng, 8
CLIENT_GEOMETRY, fält, 241
CLIENT_NAME, fält, 241
colorUse-resurs, 28, 282, 283
COMMAND-åtgärd, 170

- exekveringssträng, 179
- exempel, 172
- obligatoriska fält, 179

/config, 84
CONTAINER_NAME, fält, 226 to 228, 231

CONTAINER_TYPE, fält, 227
CONTENT, fält, 204, 209
CONTROL-definition, syntax, 227
CONTROL_BEHAVIOR, fält, 246
COPY_TO_ACTION, 204
cpp-satser, 31
current.old, katalog, 33

D

databas

- ladda om åtgärder, 174
- öppna igen, 174

%DatabaseHost%, 190
DataBaseHost, nyckelord, 190
databasens sökväg

- påverkan på EXEC_HOST, 119

databasservrar, 105, 119

- klient till, 118
- konfigurera, 110, 117
- skapa, 117

databassökväg, 135, 171

- jämfört med programsökväg, 134, 136
- omgivningsvariabler, 136
- standardval, 136
- syntax, 136
- sätta samman, 137

databasvärd, 119, 190
datatyper

- associera ikon med, 217
- associera med åtgärder, 202
- attribut, 201
- begränsa åtgärder baserat på, 188
- begränsningar för Skapa åtgärd, 157
- behörighetsmönster, 164
- definiera, 199
- dolda, 203
- dubbelklicksfunktion, 152
- endast läsning, 208
- exempel, 200
- flera villkor, 209
- förhållande till åtgärder, 150
- hjälp om, 201
- ikoner för, 66, 166, 202
- indelningsvillkor, 204
- innehållsbaserade, 209
- integrerade med dtappintegrate, 75

- introduktion, 144, 149
- konfigurationsfiler, 199
- krav för att skapa manuellt, 157, 197
- krävs för registrering, 62
- körbara, 208
- Ladda om åtgärder, 174
- lägesvillkor, 207
- namnbaserade, 162, 205
- platsbaserade, 206
- server för, 117
- skapa manuellt, 197
- skapade med Skapa åtgärd, 149, 160
- skilja mellan, 204
- skriva ut, 153
- släppfunktion, 152
- språkanpassade, 210
- sökvägsbaserade, 206
- variabler i definitioner, 191
- villkor, 204
- ändamål för att skapa, 55
- öppna igen, 174
- datatyper, *Se* databassökväg
sökväg,
- Datatypfamiljens namn, textfält, 162
- datatyplista i Skapa åtgärd, 161
- DATA_ATTRIBUTES
 - definiera, 201
 - definition, 198
 - syntax, 199
- DATA_CRITERIA
 - definiera, 204
 - definition, 198
 - flera, 210
 - i par med DATA_ATTRIBUTES, 198
 - syntax, 200
- DELETE, fält, 225
- DESCRIPTION, fält, 175, 201
- /dev/console, 86
- direkthjälp (man page) för dtconfig(1), 295
- DISPLAY-variabel, ange med
inloggningshanteraren, 16
- %DisplayHost%, 190
- DisplayHost, nyckelord, 190
- DISPLAY_skärmmamn, makro, 31
- DROP_ACTION, fält, 239
- DROP_ANIMATION, fält, 242
- dt-filer, 171
- dtaction
 - används för att ändra användare, 193
 - syntax, 192
- dtappgather, 26, 41
- dtappintegrate, 73
 - funktioner, 75
 - syntax, 74
 - ta bort program, 49
- DTAPPSEARCHPATH, variabel
 - definition, 130
 - sammanlagd, 133
 - sätta samman, 133
- DtButtonBindings, 259
- dtchooser, fil, 19
- dtconfig, kommando, 9
- dtCreate , *Se* åtgärd,
- DTDATABASESEARCHPATH, variabel
 - användning, 177
 - definition, 130
 - sätta samman, 136, 137
- DtEditor, tangentbindningar för, 267
- Dterrors, fil, 9
- dtgreet, fil, 19
- DTHELPPSEARCHPATH, variabel
 - definition, 130
 - sätta samman, 140
- dtlogin, 85
- Dtlogin*, språkresurs, 14
- dtlogin, *Se* inloggningshanteraren,
- dtlp, 97
- dtmailpr, 97
- DTMOUNTPOINT, variabel
 - inställning, 112
 - processer som använder, 112
 - processer som kräver, 112
 - ärvd av användare, 112
- Dtpid, fil, 3
- .dtprofile, 35 to 37
- .dtprofile, fil
 - skapa, 24
 - ange LANG, 289
 - anger omgivningsvariabler i, 30
 - läsa in och köra i, 24
 - läsa in och köra i, 23
 - syntax, 30
- DtRootMenu, 257
- dtsearchpath, 25, 131, 177
- dtsmcmd, kommando, 32

DTSOURCEPROFILE, variabel, 26
 dtspcd, 110, 112, 113
 konfigurera, 113
 verifieringskatalog, 108, 113
 DTSPSYSAPPHOSTS, variabel
 definition, 130
 syntax, 133
 ändra, 48
 DTSPSYSDATABASEHOSTS, variabel
 definition, 130, 136
 påverkan på EXEC_HOST, 119
 syntax, 136
 DTSPSYSHELP, variabel, 140
 definition, 130
 syntax, 140
 DTSPSYSICON, variabel
 definition, 130
 syntax, 138
 DTSPUSERAPPHOSTS, variabel
 definition, 130
 syntax, 133
 ändra, 48
 DTSPUSERDATABASEHOSTS, variabel
 definition, 130, 136
 syntax, 136
 DTSPUSERHELP, variabel
 definition, 130
 syntax, 140
 DTSPUSERICON, variabel
 definition, 130
 syntax, 138
 dtstart_appgather, variabel, 26
 dtstart_searchpath, variabel, 26
 dtstart_ttsession, variabel, 27
 dtterm, 38
 dtwm, *Se* arbetsyttehanterare,
 dtwm.fp, fil, 222
 dtwmfp.session, fil, 223
 dtwmrc, fil, 250
 redigera, 250
 dynamicColor-resurs, 28, 284, 282, 284
 dölja filer med datatyp, 203

E

e-post, konfigurera, 109
 egen programgrupp, 46
 Egenskaper

dialogruta, 162
 fält, 165
 egna datatyper och åtgärder, skapa, 200
 EMACS-tangentbindningar, 267
 /etc/dt, 84
 /etc/rmmount.conf, 99
 etiketter
 Ikoner i huvudgruppen, 246
 åtgärder, 176, 194
 etiketter på kontroller, 246
 /examples, 84
 EXEC_HOST, fält, 190
 flera värden, 119
 påverkad av databsens sökväg, 119
 standardvärde, 119, 190
 EXEC_HOST, *Se* exekveringsdator,
 EXEC_STRING, 97
 EXEC_STRING, fält, *Se* exekveringssträng,
 exekveringsdator
 ange, 119
 angiven i fältet EXEC_HOST, 190
 konfigurera, 119
 skapa åtgärder för, 190
 exekveringssträng, 179
 absolut sökväg i, 180
 allmänna funktioner, 179
 ange körbar fil, 180
 efterfråga en fil, 181
 efterfråga en sträng, 182
 flera filargument, 183
 inga argument, 180
 skalsyntax, 180
 släppta filer, 181
 extern skärmtyp, 5

F

fellogg
 inloggning, 36
 platser, 36
 sessionshanteraren, 36
 start, 36
 Xsession, 36
 Felsäker session, alternativ, 36
 inloggningsskärm, 88
 felsäkra sessioner, 15
 felsöka skrivbordsstart, 35

- fil, datatypsvillkor, 208
- filargument
 - angivna med Skapa åtgärd, 159
 - används i åtgärder, 148
- filer
 - dölja baserat på datatyp, 203
 - fjärrdata, 111
 - fjärråtkomst, 107
 - konsekventa namn, 108
 - krav för nätverkshantering, 110
 - montera, 107
 - monteringspunkt, 111
 - åtkomst till distribuerade, 107
- filer, inloggningsstart, 35
- FILE_NAME, fält, 239, 240
- filhanteraren, använda som ikonhanterare, 218
- filledtext, angiven med Skapa åtgärd, 160
- filnamn, ikon, 214
- filnamnsavbildning, 111
- filnamnsdatabasserver, 110
- filserverar, 102, 103
- filtyper, *Se* datatyper,
- fjärrkörning
 - enligt åtgärd, 190
 - konfigurera programservern, 115
 - med åtgärden på en annan plats än programmet, 119
 - stöd för eget språk, 293
- flagga, -broadcast, 95
- flera skärmar, 89
 - inloggningshanteraren, 13
- FontSet, resurs, 272
- foreground-resurs, 278
- foregroundColor-resurs, 28, 284
- fp_dynamic, katalog, 222
- funktionen för automatisk montering, 111
- färg
 - ange förgrund, 284
 - använda i ikoner, 219
 - färguppsättningar, 278
 - högsta antal tilldelade, 282
 - kontrollera, med Inställningar, 281
 - skapa toningar med
 - shadowPixmaps-resurser, 283
 - styra, med dynamicColor-resursen, 284
- färger
 - administrera, 277
 - aktiv fönsterram, 279
 - antal använda, 281
 - färguppsättningar, 277
 - inaktiv fönsterram, 279
 - områden för textinmatning, 279
 - paletter, 277
 - programfönster, 279
 - resurser, 278
 - standardval, 280
 - värden, 278
- färghanterare, 23
 - resurser, 28
 - starta, 28
- färgpaletter, 277
- färgresurser, ändra för registrering, 59
- färguppsättningar, 277
 - avbildning till visningselement, 279
 - standardval, 280
- Fönster-menyn
 - definition, 254
 - ny, 258
 - syntax, 255
- fönsterhanterare, 250
 - ändra, 29
- fönsterstöd för åtgärder, 159
- företräde
 - åtgärdsdatabassembler, 178

G

- gemensamma filer, 107
- getty, 5, 23
- GID, 107
- grundfilnamn, 176
- grundfönster, 254
- grupp-ID, 107

H

- %H, 131
- hantera ikoner i filhanteraren, 218
- Help Developer's Kit, 65
- HELP_STRING, fält, 243
- HELP_TOPIC, fält, 243
- HELP_VOLUME, fält, 243
- hemkatalog, 85
 - gemensam, 107
 - nätverksansluten, 108

- hemsession, 22
- HIGH_COLOR, 283
- hjälp, 65
 - angivna med Skapa åtgärd, 159
 - delvis integrering, 65
 - fullständig integrering, 65
 - huvudgruppen, 243
 - om datatyper, 202
 - om skrivarikoner, 126
 - om åtgärdsfil, 175
- hjälpfiler
 - i registreringspaket, 65
 - integrerade med dtappintegrate, 75
- hjälpserverar, 105
 - klient till, 118
 - konfigurera, 110, 117
 - skapa, 117
- hjälp sökväg, 130, 139
 - jämfört med programsökväg, 134, 139
 - omgivningsvariabler, 140
 - standardval, 139
 - syntax, 140
 - sätta samman, 140
- hjälpvolym
 - huvudhjälpfil, 65
 - integreringsnivåer, 65
 - placering i registreringspaket, 65
- hjälpvolym
 - lokalanpassa, 293
- HOME, variabel, 17
- home.old, katalog, 33
- huvudgruppen, 246
 - animering, 242
 - anpassa, 221
 - använda åtgärder, 145
 - definitionens organisation, 225
 - dynamiska anpassningar, 222, 223
 - hanteras av arbetsytahanteraren, 250
 - hjälp, 243
 - klient i, 241
 - komponenter, 225
 - konfigurationsfiler, 221
 - kontroll där du kan släppa objekt, 239
 - kontroll i registreringspaket, 73
 - lägga till rader, 247
 - meny, 255
 - namnkonventioner för filer, 222
 - ny, 246

- placering på skärmen, 245
- sammansättningsprioritet, 223
- styra egna anpassningar, 224
- syntax, 226
- sökvägar, 222
 - visa ikon i, 217
 - ändra, 228
- huvudgruppen, *Se* kontroll
- kontroll,
- huvudkonfigurationsfiler, 85

I

- ICON, fält
 - för datatyper, 202
 - giltiga värden, 202
 - i huvudgruppen, 232
 - tillåtna värden, 176
- ikoner
 - använda filhanteraren som ikonhanterare, 218
 - associera med programfönster, 218
 - associera med åtgärds- eller datatyp, 216
 - basfilnamn, 202
 - bläddra i filhanteraren, 218
 - datatyp, 66
 - datatyper, 166, 202
 - dialogrutan Sök ikonuppsättning, 165
 - filformat, 214
 - filnamn, 214
 - färganvändning, 219
 - för program, 146
 - hur filer hittas, 214
 - huvudgruppen, 217, 232
 - integrerade med dtappintegrate, 75
 - krävs för registrering, 65
 - lokalanpassade, 292
 - namnkonventioner, 214
 - programgrupper, 66, 69
 - rekommendationer för hur du skapar ikoner, 219
 - representera program, 71
 - skapa associationer, 216
 - skrivarikoner, 125
 - språkanpassa, 291
 - språkanpassade, 291, 292
 - starta program, 66

- storlekskonventioner, 214
- ätgärder, 166, 176
- ätgårdsikoner, 174
- ikoner, *Se* ikonservrar
- servrar,
- ikoner, *Se* ikonsökväg
- sökväg,
- ikonplacering, 38
- ikonservrar, 105
 - klient till, 118
 - konfigurera, 110, 117
 - skapa, 117
- ikonstorlek, 214
- ikonsökväg, 137
 - jämfört med programsökväg, 134, 138
 - omgivningsvariabler, 138
 - standardval, 137
 - syntax, 138
 - sätta samman, 138
- inactiveFrameColorId-resurs, 279
- /include, 84
- include-sats, i filer i arbetsytehanetaren, 251
- indatametod, internationalisering, 294
- indatavariabler, 130
- indirect, alternativ, 6
- inetd.conf, 112
- inloggningsfellogg, 36
- inloggningshanteraren, 2
 - administration, 18
 - anpassa, 2
 - definition, 2
 - fel, 8
 - ge kommandon, 15
 - konfigurationsfiler, 19
 - resurser, 11, 12
- inloggningsserver
 - avbryta, 87
 - inaktivera vid systemstart, 86
 - konfigurationer, 3
 - starta från kommandoraden, 86
 - starta vid systemstart, 86
- inloggningsservrar
 - ansluta skärmar, 2
 - användarsökväg, 17
 - felsökning, 8
 - inaktivera, 9
 - ingen punktskärm, 4
 - kommandoradsinloggning, 5
 - konfigurera, 109
 - köra utan lokal skärm, 4
 - omgivningar, 16, 17
 - process-ID, 3
 - starta, 2
 - starta en session, 2
 - starta från kommandoraden, 2
 - stoppa, 9
 - styra åtkomst, 7
 - systemskal, 17
 - terminera process-ID, 9
 - textskärmssystemfönster, 5
 - verifiera användare, 2
 - visa inloggningsskärm, 2
 - ändra tidszonen, 18
 - översikt, 2
- inloggningsskärm
 - anpassa, 10
 - avslutande, 87
 - Felsäker session, alternativ, 88
 - hälsning, 11
 - lokalanpassa, 12
 - resurser, 11
 - skärmberoende funktion, 13
 - teckensnitt, 12
 - visa på en nätverksskärm, 6
 - X-serveromgivning, 14
 - X-serveråtkomst, 13
 - ändra funktioner, 12
 - ändra innehållet på språkmenyn, 14
 - ändra standardspråk, 14
 - ändra utseende, 10
 - ändra välkomstmeddelandet, 11
- inloggningsspråkområde, 93
- inloggningsstartfiler, 35
- inloggningsverifiering, 35
- innehållsbaserade datatyper, 209
- installation
 - katalogadress, 83
- Installationsikon, kontroll, ta bort, 236
- Inställningar
 - använda för att ange färger, 279
 - färgintegrering med, 59
 - teckensnittsintegrering med, 58
- internationalisering
 - felsöka, 294
- IS_TEXT, 204

J

jokertecken i datatyper, 205

K

kalenderbakgrundsprogram, 114

katalog

datatypsvillkor, 208

keyBindings, resurs, 263

klienter

definition, 102

fönster i huvudgruppen, 241

till server, konfigurera, 109

klientserver-konfiguration, *Se*

nätverkshantering,

kommandorad för åtgärder, 179

kommandoradsinloggning, 5

konfigurationsfiler, 85

arbetsytehanterare, 250

datatyper, 199

fönsterhanterare, 250

huvudgruppen, 221

i registreringspaket, 56

inloggningshanteraren, 19

plats, 84

sessionshanteraren, 34

åtgärd, 171

konsekventa filnamn, 108

kontroll

animering, 242

bildskärm, 240

byta ut, 231

byta ut mot annan kontroll, 231

definiera, 237

en förekomst, 240

enkel- eller dubbelklick, 246

etiketter, 246

hjälp om ett visst objekt, 243

i arbetsyteväxlaren, 244

ikon, 232

klient, 241

lägga till i huvudpanelen, 229

läsa, 224

område där du kan släppa objekt, 239

skapa, 238

som öppnar en fil, 239

ta bort från huvudgruppen, 229

typer, 237

utseende, 232

växla, 240

ändra, 230

återställa, 224

kontroll av datumtyp, 238

kontroll av filtyp, 238, 240

kontroll av ikontyp, 240

kontroll av klienttyp, 238

kontroll av klocktyp, 238

kontroll av posttyp, 240

kontroll av tom typ, 237

kontroll av övervakartyp, 240

körbar fil, datatypsvillkor, 208

L

%L, 131

LABEL, åtgärdsfält, 176

Ladda om program, åtgärd, 49

Ladda om resurser, åtgärd, 28

Ladda om åtgärder, åtgärd, 174

LANG, variabel, 287

ange med inloggningshanteraren, 16

i dtprofile, 289

påverkan på datatyper, 210

/lib, 84

LINK_TO_ACTION, 204

%LocalHost%, 190

localTerminal-resurs, 187

LOCKED, fält, 224

loggfild för wscon-systemfönster, 36

.login, 37

.login, fil, 23

.login, fil

läses inte av inloggningshanteraren, 30

läsa in och köra, 26

LOGNAME, variabel, 17

lokal skärmtyp, 5

lokalanpassning, 292

datatyp, 210

ikoner, 291, 292

inloggningsskärm, 12

meddelandekatalog, 293

palettnamn, 292

åtgärder, 194

åtgärdsetikett, 194

LOW_COLOR, 283

lp
 kommando, 108
 skrivarkö, skrivarbuffert, 108
 LPDEST, variabel, 126
 länk, datatypsvillkor, 208

M

%M, 131
 mailx, 109
 /man, 84
 MAP-åtgärder, 152
 definition, 170
 exempel, 173
 Markera-menyn, 145, 151
 meddelandekataloger, 290
 MEDIA, 204
 MEDIUM_COLOR, 283
 menyer
 använda åtgärder, 145
 arbetsytehanterare, 254
 menyn Arbetsyta
 definition, 254
 lägga till menyalternativ, 255
 skapa, 258
 syntax, 255
 ändra, 256
 MIME_TYPE_MEDIA, 204
 mkfontdir (kommando), kompilera filer, 290
 MODE, fält, 205
 syntax, 207
 MONITOR_TYPE, fält, 240
 montering
 en installerad CDE, 88
 monteringar, köra program över, 120
 monteringspunkt för fjärrfiler, 111
 Motif-fönsterhanterare,, Se
 arbetsytehanterare,
 MOVE_TO_ACTION, 204
 musknappskoppling, 259
 lägga till, 260
 skapa ny uppsättning, 261
 syntax, 259
 muskoppling, Se musknappskoppling,

N

NAME_PATTERN, fält, 204

namnbaserade datatyper, 205
 NFS, 107
 NLS-fjärrkörning, 293
 NLS-omgivningsvariabler, 290
 NoBackdrop, inställning, 254
 NoPrint, åtgärd, 153
 NOT, operator i fältet MODE, 207
 NO_STDIO, fönsterstöd, 186
 NUMBER_OF_ROWS, fält, 244
 Nytt
 alternativ på menyn Arbetsyta, 255
 nätverksanslutna skrivbord, 91
 nätverkshantering, 101
 allmänna konfigurationssteg, 106
 elektronisk post, e-post, 109
 filer som krävs för, 110
 grundkonfiguration, 106
 konfigurera klienter och servrar, 109
 köra program över monteringar, 120
 typer av tjänster, 102
 X-behörighet, 109
 översikt, 101

O

omgivningsfil, skrivbordet, 98
 omgivningsvariabler
 bitmappsökväg, 138
 databassökväg, 136
 egna, 30
 i dtprofile, 24
 exportera, 30
 fjärrkörning, 113
 hjälpsökväg, 140
 i dtprofile, 24
 ikonsökväg, 138
 inloggningshanteraren, 17
 inställning, 30
 läsa in och köra .login eller .profile, 26
 programsökväg, 133
 standard, 24
 systemtäckande, 30
 sökväg för bildpunktsbeskrivning, 138
 sökvägar, 130, 205
 åtgärdsdefinitioner, 192
 område där du kan släppa objekt
 huvudgruppskontroll, 239

- ätgärdsikon, 181
- once, alternativ, 95
- OpenWindows
 - omgivning, 86
- OR, operator i fältet MODE, 207

P

- paletter, 277
 - språkanpassa namn, 292
- /palettes, 84
- PANEL, definition, 225
 - syntax, 226
- panelundermeny
 - anpassa inbyggda paneler, 234
 - anpassningar för hela systemet, 234
 - behållare, 227
 - definition, 227
 - koppling till huvudpanelen, 233
 - ny, 234
 - skapa, 233
 - syntax, 227
 - ändra, 233
 - ändra beteende vid automatisk stängning, 236
 - återställa borttagen, 224
- PANEL_GEOMETRY, fält, 245
- PATH, variabel, 180
 - ange med inloggningshanteraren, 16
- PATH_PATTERN, fält, 204
 - syntax, 206
- PERM_TERMINAL, fönsterstöd, 186
- placering, ikoner, 38
- platsbaserade datatyper, 206
- .pm, filnamnstillägg, 214
- POSITION_HINTS, fält, 231
- primaryColorSetId-resurs, 279, 281
- prioritet
 - huvudgrupp, sammansättning, 223
- problem, skrivbordsstart, 35
- .profile, 37
- .profile, fil, 23
- .profile, fil
 - läses inte av inloggningshanteraren, 30
 - läsa in och köra, 26
- program
 - avregistrering, 49
 - datatyper som krävs, 62

- köra lokalt över monteringar, 120
- lägga till i befintliga programgrupper, 45
- lägga till i programhanteraren, 43
- lägga till utan registrering, 44
- olika sätt att lägga till, 43
- programrot, katalog, 59
- registrerat, definition, 44
- registreringsfunktioner, 54
- rotkatalog, 59
- samla, 42
- samla i programhanteraren, 40
- samlade med sessionshanteraren, 26
- skapa ikoner för, 146
- skrivbordsanpassat, 44
- starta vid inloggning, 23, 29
- sökvägar, 40
- ta bort, 49
- ändamål för datatyper, 55
- åtgärder som krävs, 62
- öppna igen, 49
- program, *Se* registrering
 - registrera,
- programfönster, associera ikon med, 218
- programgrupper
 - administrera, 46
 - anpassa, 46
 - datatyper för, 70
 - definition, 39
 - egna, 46
 - exempel på hur du skapar, 81
 - företrädare, 41
 - ikoner för, 66, 69
 - innehåll, 70
 - integrerade med dtappintegrate, 76
 - kataloger för, 67
 - namn, 68
 - namnge, 46
 - readme-filer, 72
 - samla, 40
 - skapa i registreringspaket, 67
 - standardval, 41
 - systemtäckande, 46
 - åtgärd för, 70
- programhanteraren
 - allmän systemadministration, 49
 - beskrivning, 39
 - företrädesregler, 41

- integrera program med, 54
- lägga till program, 43
- plats i filsystemet, 40
- samla ihop program, 26
- samla program, 40
- symboliska länkar, 41
- uppdatera, 49
- programikon, 172
 - dubbelklicka, 148
 - krävs av skrivbordssystemet, 66
 - med Skapa åtgärd, 156
 - skapa, 146, 174
 - släppta filer, 148
- programpaket, *Se* registreringspaket,
- programregistrering, *Se* registrering,
- programresurser, *Se* resurser,
- programrot, *Se* programrotkatalog,
- programrotkatalog, 59
- programservrar
 - administrera, 114
 - definition, 102
 - klient till, 116
 - konfigurera, 110, 116
 - konfigurera klient till, 116
 - lägga till, 47
 - programmets tillgänglighet, 57
 - standardkonfiguration, 115
- programsökväg
 - anledningar till att ändra, 47
 - används för att samla program., 40
 - definition, 132
 - egen, 48
 - omgivningsvariabler, 133
 - språkanpassade, 141
 - standardval, 47, 132
 - syntax, 133
 - systemtäckande, 48
 - sätta samman, 133
 - ändra, 48
 - ändra företräde, 134
- programsökväg, *Se* databassökväg,
- protokoll, XDM, 92
- punktskärm, köra inloggningsservern utan, 4
- PUSH_ACTION, fält, 238
- PUSH_ANIMATION, fält, 242

Q

- query, alternativ, 6

R

- readme-filer, 72
- Redigera dtwmrc, åtgärd, 250
- redigera åtgärder, 177
- registrering, 71
 - allmänna steg, 57
 - datatyper som krävs, 62
 - definition, 44, 57
 - dtappintegrate, 73
 - exempel, 76, 83
 - funktioner i, 54
 - färgändringar, 59
 - hjälpfiler, 65
 - ikonkrav, 65
 - programgrupp, 67
 - programrotkatalog, 59
 - resursändring, 58
 - teckensnittsändringar, 58
 - ändamål med, 56
 - ändra resurser, 58
 - åtgärder som krävs, 62
 - översikt, 54
- registreringspaket, 44, 71
 - definition, 56
 - exempel på hur du skapar, 76
 - huvudgruppskontroll, 73
 - innehåll i programgruppen, 70
 - kataloger, 59
 - programikon, 71
 - readme-filer, 72
 - ändamål med, 54
- RESOURCE_MANAGER, egenskap, 28, 30
- resurser
 - ange, 30, 265
 - app-defaults, 265
 - colorUse, 283
 - egna, 266
 - foregroundColor, 284
 - för hela systemet, 266
 - session, 23
 - shadowPixmaps, 283
 - skuggtjocklek för fönster, 285
 - skärmspecifik, 31

- språkoberoende, 290
- standardskrivbords-, 27
- teckensnitt, 272
- öppna, 27
- RGB-färgvärde, 279
- RGB-värden, 278
- rgb.txt, fil, 279
- rpc.cmsd, 114
- rpc.ttdbserver, 110, 112

S

- .sdl-filer, 65
- sdtcm_convert, kommandofil, 97
- secondaryColorSetId-resurs, 279, 281
- selectColor-resurs, 278
- servrar
 - databas, 110
 - datatyper, 117
 - definition, 102
 - fil, 102
 - hjälp, 105, 110
 - ikon, 105, 110
 - inloggning, 110
 - konfigurera, 109
 - program, tillämpning, 102, 110
 - session, 110
 - servrar, 102
 - typer, 105
 - åtgärder, 117
- servrar, *Se* databasservrar,
- session
 - aktuell, 22
 - definition, 21
 - felsäker, 15
 - första, 32
 - hem, 22
 - inledande, 22
 - kommandofil som körs vid avslutning, 16
 - köra kommandon vid starten, 33
 - köra kommandon vid utloggning, 33
 - loggningsfel, 34
 - resurser, 23
 - skärmspecifik, 22, 32
 - standardval, 22
 - starta, 23
 - säkerhetskopiering, 33
 - återställa, 33

- sessionetc, fil, 33
- sessionexit, fil, 33
- %SessionHost%, 190
- SessionHost, nyckelord, 190
- sessions, katalog, 32
- sessionshanteraren, 23
 - ange sökvägar, 25
 - anpassa programstart, 29
 - anpassningar för hela systemet, 24
 - fellogg, 34, 36
 - felsökning, 34
 - filer, 34
 - introduktion, 21
 - kataloger, 34
 - klient, 27
 - köra kommandon vid utloggning, 33
 - köra ytterligare kommandon, 33
 - samla ihop program, 26
 - starta, 23
 - starta arbetsytehanteraren, 29
 - starta program, 29
 - säkerhetskopiera sessioner, 33
 - välkomstmeddelande, 23
 - öppna resurser, 27
- sessionsservrar, *Se* inloggningsservrar,
- shadowPixmaps-resurs, 28, 283
- /share, 84
- SHELL-variabel, ange med
 - inloggningshanteraren, 16
- skal
 - anpassningar för hela systemet, 24
 - används i åtgärd, 183
 - egen anpassning, 24
 - i åtgärder, 183
 - läsa in och köra .profile eller .login, 26
 - Se* även omgivningsvariabler, .profile,
 - .login, .dtprofile, 30
 - syntax i exekveringssträng, 180
- Skapa åtgärd, 155
 - ange filargument, 159
 - ange ikoner, 165
 - använda, 157
 - begränsningar, 156
 - datatypsnamn, 162
 - filledtext, 160
 - funktioner, 155
 - huvudfönstret, 158

- Introduktion, 155
- skapa datatyp, 149, 160
- skapas konfigurationsfil, 155
- starta, 158
- syntax för åtgärdskommandon, 159
- åtgärdsnamn, 159
- skicka post, 109
- skriva ut
 - administration, 123
 - grunderna, 127
 - konfigurera för datatyp, 153
 - olika datatyper, 127
 - standardskrivare, 126
 - testa, 108
- skrivare
 - enhetsnamn, 108
 - fjärråtkomst, 108
 - hjälp om ett visst objekt, 126
 - ikonbilder, 125
 - lägga till, 124
 - namn, 126
 - standard, 126
 - ta bort, 124
 - uppdateringsintervall för utskrifter, 125
- skrivarkö, lp, 108
- skrivbord, nätverksanslutna, 91
- skrivbordsanpassat program, 44
- skrivbordsomgivningsfil, 98
- skrivbordsstart
 - flera skärmar, 90
 - problem, 35
- skrivbordssökvägar, 23
- Skrivbordsverktyg, programgrupp, ändra, 46
- skrivskydd, datatypsvillkor, 208
- skuggtjocklek, fönster, 285
- skärmspecifik session, 32
- skärmspecifika resurser, 31
- släppta filer, åtgärder som accepterar, 181
- Solaris CDE
 - avbryta skrivbordet, 87
- SPC, 113
 - säkerhet, 113
- Språk, menyalternativ, 93
- språk, ställa in med filen Xconfig, 288
- Språk-menyn, anpassa, 14
- språkanpassa, internationalisera
 - ange språk, 288
 - app-defaults, 290
 - LANG, variabel, 287
 - NLS-omgivningsvariabler, 290
 - teckensnitt, 290
- språkområde, C, 94
- startfellogg, 36
- startlog, fil, 34
- strängvariabler i åtgärdsdefinitioner, 191
- strängåtgärdsargument, 182
- stty, 37
- styrbakgrundsprogram för underprocesser, *Se* dtspcd,
- styr tjänst för underprocesser, *Se* SPC,
- stöd för olika språk
 - internationalisering, 287
- SUBPANEL, definition, 226
- SWITCH, definition, 228
- symboliska länkar
 - datatypsvillkor, 205
 - konsekventa filnamn, 108
 - skapas vid registreringen, 75
- sys.dtprofile, fil, 24
- sys.dtwmrc, fil, 250, 251
- sys.resources, fil, 27, 28, 30, 266
- sys.session, fil, 29, 32
- systemPath-resurs, 17
- Sök ikonuppsättning, dialogrutan, 165
- sökväg
 - användare, 17
 - system, 17
- sökvägar
 - aktuellt värde, 131
 - angivna med sessionshanteraren, 25
 - definierade av skrivbordet, 130
 - definitioner för huvudgruppen, 222
 - hjälp, 139
 - ikoner, 216
 - indatavariabler, 130
 - inställning, 131
 - omgivningsvariabler, 130, 205
 - program, 40, 132
 - språkanpassade, 141
 - utdatavariabler, 130
 - åtgärder, 171
- sökvägsbaserade datatyper, 206

T

tangentbordsinställning, återställa, 294
tangenttilldelning
 skapa ny uppsättning, 262
 standardval, 261
 syntax, 262
teckensnitt
 administrera, 271
 ange egenskapssträng, 274
 antal färger i Inställningar, 272
 bitmapp, 272
 primärkatalog, 290
 resurser, ange, 272
 resurser, ändra för registrering, 58
 system, i Inställningar, 273
 söka med aliasfiler, 290
 söka med katalogfiler, 290
 söka med mkfontdir, kommando, 290
 visa en lista över tillgängliga, 273
 X Logical Font Description, 272, 274
 xlsfonts, kommando, 290
Teckensnitt, dialogruta, 272
Teckensnittsresurs, 272
teckensnittssökväg, 93
TERMINAL, fönsterstöd, 186
terminalemulator
 automatisk stängning av åtgärder,
 alternativ, 159
 för åtgärder, 185
 kommandoradsalternativ för åtgärder, 186
 standard för åtgärder, 187
 ändra, 50
terminalemulering, 38
textColorSetId-resurs, 279
texthanteraren, ändra, 50
textskärmssystemfönster, 5
tidszon, ändra, 18
timeZone-resurs, 18
title-resurs, 253
ToolTalk
 bakgrundsprogram för meddelanden, 23,
 27
 program, åtgärder för, 195
ToolTalk, *Se* rpc.ttdbserver
 Databasserver,
ToolTalk, *Se* TT_MSG-åtgärd
 åtgärd,

ToolTalk-meddelandeserver, *Se* ttsession,
topShadowColor-resurs, 278
tset, 37
ttsession, 114
 starta, 27
tty-inställningar, 38
ttyModes, 38
TT_MSG-åtgärd
 nyckelord, 195
 skapa, 195
TYPE, fält, 237
TZ, variabel, 16, 18

U

UID, 107
UNIX-tangenttilldelningar, 265, 267
UNIXbindings, fil, 267
uppdateringsintervall för utskrifter, 125
USER, variabel, 16
user-prefs.dt, fil, 52
userPath-resurs, 17
 /usr/dt, 83
utdatavariabler, 130
utskrift av brev, anpassa, 96
Utskriftsverktyg, 124
 uppdateringsintervall för utskrifter, 125

V

val för arbetsyta
 anpassa, 243
 antal arbetsytor, 244
 antal rader, 244
 definition, 226
 definitionens syntax, 228
 lägga till kontroller, 244
 /var/dt, 84
variabler
 i åtgärdsdefinitioner, 191
variabler, *Se* omgivningsvariabler
 omgivning,
verifiering, inloggning, 35
verifieringskatalog, 107, 113
verifieringsresurs, 14
visningsfärger
 högsta antal tilldelade, 282

Väljaren, 91, 92
välkomstmeddelande
 anpassa, 25
 standard, 11
 visa, 23, 25
 ändra, 11
växlare, *Se* val för arbetsyta,

W

windowMenu, resurs, 254
WINDOW_TYPE, fält, 186
wmStartupCommand-resurs, 29
WM_CLASS, egenskap, 218
workspaceCount-resurs, 252
writeXrdbColors-resurs, 28

X

X Logical Font Description, 272
X server
 byta omgivning, 14
X-behörighet, 109
X-resurser, *Se* resurser,
X-server
 ändra åtkomst, 13
X-terminaler, 91, 94, 110
 använda, 92
 CHOOSER, sträng, 8
 direkt XDMCP, 6, 7
 erhålla sessionstjänster, 103
 indirekt XDMCP, 6, 8
 möjliga konfigurationer av
 inloggningsserver, 3
 skärmar utan XDMCP, 6
 som arbetsstationer, 94
 Xaccess, lista, 7
X11-server, 86
X400_TYPE, 204
Xaccess, fil, 7
XAUTHORITY-variabel, ange med
 inloggningshanteraren, 16
Xconfig, 85, 89
Xconfig, fil
 ange resurser i, 12
 ange språk, 288
 ändra, 3
.Xdefaults, 37, 38

.Xdefaults, fil, 28, 266
XDM, 85, 92, 95
 protokoll, 92
XDMCP, 2
 definition, 6
 direktförfrågningar, 6
 direktåtkomst, 7
 frågeläge, 6
 indirekt åtkomst, 8
 indirekta förfrågningar, 6, 18
Xerrors, fil, 8
Xfailsafe, fil, 15, 17, 18
xlsfonts, kommando
 installation, 290
 visa teckensnitt på servern, 290
XMICONBMSEARCHPATH, variabel
 användning, 138
 definition, 130
 sätta samman, 138
XMICONSEARCHPATH, variabel
 användning, 138
 definition, 130
 sätta samman, 138
XmText*FontList, resurs, 272
Xpid, fil, 3
Xreset, fil, 16
Xresources, fil, 10, 12
Xserver, 95
 avbryta, 87
Xservers, fil, 85
 hantera lokal skärm, 18
 standard, 4
 starta en server, 4
 syntax, 4
Xsession
 kommandofil, 35
Xsession, fil, 24
 ange PATH, 17
 anpassningar för hela systemet, 24
 körs av inloggningsserver, 15
 starta sessionshanteraren, 23
Xsession.d, katalog, 30
 anpassa, 24
 kommandofiler i, 24
Xsession.d, katalog, 23
Xsetup, fil, 15
Xstartup, fil, 15

xterm, 38
starta, 88
XUSERFILESEARCHPATH, variabel, 290

Special Characters

Återställ huvudgrupp , åtgärd, 224
Åtgärdsnamn, fält i dialogrutan Skapa åtgärd, 159
Öppna-åtgärd, 151
åtgärder
acceptera en släppt fil eller visa en ledtext, 182
acceptera flera släppta filer, 185
acceptera släppta filer, 148, 181
anledning att skapa manuellt, 170
användningsområden, 149
används av Huvudgruppen, 145
används i menyer, 145
argument, 180
argument, icke-fil, 182
argumentantalbegränsningar, 188
associera ikon med, 216
associera med datatyper, 202
avbildning, 151
begränsa argument, 187
begränsad enligt datatyp, 152, 188
begränsningar för Skapa åtgärd, 156
COMMAND, 170
efterfråga en fil, 181
etiketter, 176, 194
exekveringssträng, 179
exempel, 172, 173
filargument, 148
fönsterstöd, 159
fönsterstöd för, 185
företrädesregler, 178
förhållande till datatyper, 150
icke utbytbara filargument, 184
icke-filargument, 156, 182
ikoner för, 166, 176
ikoner som representerar, 174
inga argument, 180
ingen visning, 159
integrerade med dtappintegrate, 75
introduktion, 144

konfigurationsfiler, 171
krävs för registrering, 62
köra andra åtgärder, 192, 193
köra fjärrprogram, 119, 190
köra som en annan användare, 193
MAP, 170
namn, 156, 175
olika funktioner för dubbelklickning och släppning, 188
omgivningsvariabler, 192
redigera, 177
server för, 117
skapa ikoner för program, 146
skapa manuellt, 169, 171
språkanpassade, 194
standardikon, 176
standardterminal, 187
strängvariabler i, 191
terminalalternativ, 186
terminalstöd för, 187
tillhandahålla skal, 183
TT_MSG, 171
typer av, 170, 172
utan data, 180
utbytbara argument, 184
variabler i definitioner, 191
ändra, 177
öppna igen, 174
åtgärder efterfrågas, 181
åtgärder, *Se* argument parametrar,
åtgärder, *Se* databassökväg sökväg,
åtgärder, *Se* åtgärdsfil filer som representerar,
åtgärdsdefinitionsfil, skapad av Skapa åtgärd, 155
åtgärdsfil, 155, 174
definition, 146
innehåll, 147
skapa, 71, 174
åtgärdsikon, 146, 174, 175
krävs av skrivbordssystemet, 66
skapa, 174
åtgärdsikonkontroller, i Skapa åtgärd, 159