



Solaris Common Desktop
Environment: Benutzerhandbuch
für Fortgeschrittene und
Systemverwalter

Sun Microsystems, Inc.
901 San Antonio Blvd.
Palo Alto, CA 94303
U.S.A.

Bestellnummer: 805-0018-10
September 1997

Copyright 1997 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California 94303-4900 U.S.A. All rights reserved.

Dieses Produkt oder Dokument ist urheberrechtlich geschützt und wird unter Lizenzen vertrieben, die seine Verwendung, Vervielfältigung, Verteilung und Dekompilierung einschränken. Das Produkt oder Dokument darf weder vollständig noch in Teilen in irgendeiner Form oder mit irgendwelchen Mitteln ohne die vorherige schriftliche Einwilligung von Sun und gegebenenfalls seiner Lizenzgeber reproduziert werden. Die Software von Fremdherstellern, auch die Font-Technologie, ist urheberrechtlich von den Zulieferern von Sun geschützt und wird von ihnen lizenziert.

Teile des Produkts wurden aus den Berkeley BSD-Systemen abgeleitet, die von der University of California lizenziert werden. UNIX ist ein eingetragenes Warenzeichen in den USA und in anderen Staaten, das exklusiv von X/Open Company, Ltd. lizenziert wird.

Sun, Sun Microsystems, das Sun-Logo, SunSoft, SunDocs, SunExpress und Solaris sind Warenzeichen, eingetragene Warenzeichen oder Dienstleistungszeichen von Sun Microsystems, Inc. in den USA und in anderen Ländern. Alle SPARC-Warenzeichen werden unter einer Lizenz verwendet und sind Warenzeichen oder eingetragene Warenzeichen von SPARC International, Inc. in den USA und in anderen Ländern. Produkte, die mit SPARC-Warenzeichen versehen sind, basieren auf einer von Sun Microsystems, Inc. entwickelten Architektur. Code und Dokumentation für die Widgets "DtComboBox" und "DtSpinBox" stammen von Interleaf, Inc. Copyright 1993, Interleaf, Inc.

OPEN LOOK und Sun™ Graphical User Interface wurden von Sun Microsystems, Inc. für Anwender und Lizenznehmer entwickelt. Sun anerkennt die Pionierarbeit, die Xerox bei der Erforschung und Entwicklung des Konzepts der visuellen und grafischen Benutzeroberflächen für die Computerbranche geleistet hat. Sun hat eine nicht exklusive Lizenz von Xerox für Xerox Graphical User Interface. Die Lizenz gilt auch für die Lizenznehmer von Sun, die OPEN LOOK-Benutzeroberflächen implementieren und die schriftlichen Lizenzverträge von Sun einhalten.

EINGESCHRÄNKTE RECHTE: Einsatz, Duplizierung oder Preisgabe durch die Regierung der Vereinigten Staaten unterliegt den Einschränkungen von FAR 52.227-14(g)(2)(6/87) und FAR 52.227-19(6/87) oder DFAR 252.227-7015(b)(6/95) und DFAR 227.7202-3(a).

DIE DOKUMENTATION WIRD UNTER AUSSCHLUSS JEDER SACHMÄNGELHAFTUNG AUSGELIEFERT. ALLE AUSDRÜCKLICHEN ODER IMPLIZIERTEN BEDINGUNGEN, DARSTELLUNGEN UND GEWÄHRLEISTUNGEN, EINSCHLIESSLICH ALLER IMPLIZIERTEN GEWÄHRLEISTUNGEN DER MARKTGÄNGIGEN QUALITÄT, ZWECKDIENLICHKEIT ODER NICHTVERLETZUNG, WERDEN NICHT ANERKANNT. AUSGENOMMEN SIND DIE FÄLLE, IN DENEN HAFTUNGSABLEHNUNGSERKLÄRUNGEN RECHTLICH UNZULÄSSIG SIND.

Copyright 1997 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.



Inhalt

Vorwort xix

1. Den Anmeldemanager konfigurieren 1

Den Anmeldeserver starten 2

Lokale und Netzwerkanzeigen verwalten 2

Die Prozeß-ID des Anmeldeservers ermitteln 3

Eine Anmeldeanzeige auf einem lokalen Bildschirm anzeigen 4

Den Anmeldeserver ohne lokale Anzeige ausführen 4

Auf die Befehlszeilenanmeldung in einer lokalen Anzeige zugreifen 5

Eine zeichenorientierte Anzeige anpassen 6

Eine Anmeldeanzeige auf einer Netzwerkanzeige anzeigen 6

Den Zugriff auf den Anmeldeserver steuern 7

Auf Fehler prüfen 9

Den Anmeldeserver beenden 9

Die Anmeldeanzeige 10

Das Erscheinungsbild der Anzeige ändern 11

▼ Das Logo ändern 12

▼ Die Begrüßungsmeldung ändern 12

▼ Die Schriftarten ändern 12

▼ Für jede Sprache einen anderen Text anzeigen 13

	Die Funktionsweise des Anmeldefensters für jede Anzeige einzeln ändern	14
	Zugriffsrechte für den X-Server ändern	14
	Befehle vor dem Anzeigen des Anmeldefensters ausführen	16
	Eine Failsafe-Sitzung starten	17
	Befehle nach dem Beenden der Benutzersitzung ausführen	17
	Die Umgebung des Anmeldeservers	17
	Den Benutzer- oder Systempfad ändern	18
▼	Die System-Shell ändern	19
▼	Die Zeitzone ändern	19
	Den Anmeldemanager verwalten	19
	Dateien des Anmeldemanagers	20
2.	Session Manager konfigurieren	23
	Was ist eine Sitzung?	23
	Die Startsituation	24
	Die aktuelle Sitzung	24
	Die Home—Sitzung	24
	Bildschirmspezifische Sitzungen	24
	Sitzungen starten	25
	Zu Beginn einer Sitzung	25
	Das Skript .dtprofile ausführen	26
	Das Skript Xsession.d ausführen	26
	Begrüßungsmeldung anzeigen	27
	Desktop-Suchpfade einrichten	27
	Verfügbare Anwendungen erfassen	28
	Optionale Ausführung der Skripts .profile und .login	28
	Den ToolTalk—Nachrichtendämon starten	29
	Client des Session Managers starten	29

Sitzungsressourcen laden	29
Farbserver starten	31
Workspace Manager starten	31
Anwendungen einer Sitzung starten	31
Zusätzliche Anpassungen beim Start einer Sitzung	32
▼ Umgebungsvariablen einstellen	32
▼ Ressourcen einstellen	33
▼ Bildschirmspezifische Ressourcen einstellen	33
▼ Anwendungen für die Startsituation ändern	34
▼ Bildschirmspezifische Sitzungen einrichten	35
Zusätzliche Befehle beim Start der Sitzung und bei der Abmeldung ausführen	35
▼ So führen Sie beim Start der Sitzung zusätzliche Befehle aus	36
▼ So führen Sie beim Abmelden zusätzliche Befehle aus	36
▼ Gespeicherte Sitzungen wiederherstellen	36
▼ Probleme beim Start einer Sitzung	37
Dateien und Verzeichnisse des Session Managers	37
3. Fehlerbeseitigung bei Anmelde- und Sitzungsstartproblemen	39
Anmeldedateien	39
Fehlerprotokollverzeichnisse	40
Benutzer-Startdateien	41
Solaris CDE-Startbeispiele	41
4. Anwendungen hinzufügen und verwalten	43
Struktur des Anwendungsmanagers	43
Verzeichnisposition des Anwendungsmanagers	44
Wie der Anwendungsmanager Anwendungen sucht und zusammenstellt	44
Vorrangsregeln beim Zusammenstellen von Anwendungen	45

Gruppen von Anwendungen, die mit dem Standard-Desktop zur Verfügung gestellt werden 46

Beispiel für die Zusammenstellung von Gruppen von Anwendungen 46

Anwendungen zum Anwendungsmanager hinzufügen 47

Verschiedene Möglichkeiten, Anwendungen zum Anwendungsmanager hinzuzufügen 48

▼ Desktop-fähige Anwendungen zum Anwendungsmanager hinzufügen 49

▼ Eine bereits vorhandene oder eine nicht Desktop-fähige Anwendung registrieren 50

▼ Anwendungssymbole einer bereits vorhandenen Gruppe von Anwendungen hinzufügen 50

Allgemeine Gruppen von Anwendungen erstellen und verwalten 51

▼ Eine für das System definierte allgemeine Gruppe von Anwendungen erstellen 51

▼ Eine für den einzelnen Benutzer definierte allgemeine Gruppe von Anwendungen erstellen 51

▼ Eine integrierte Gruppe von Anwendungen anpassen 52

Suchpfad für Anwendungen ändern 52

Standardsuchpfad 52

Anwendungs-Server dem Anwendungssuchpfad hinzufügen 53

Allgemeine Anwendungsmanagerverwaltung 54

▼ Anwendung entfernen 54

▼ Anwendungsmanager während einer Session aktualisieren 55

Texteditor und den Terminal-Emulator ändern 55

▼ Den standardmäßigen Texteditor oder Terminal-Emulator ändern 55

5. Anwendungen registrieren 59

Übersicht zur Anwendungsregistrierung 60

Durch die Anwendungsregistrierung bereitgestellte Funktionen 60

Zweck der Anwendungsregistrierung 62

Allgemeine Schritte zur Registrierung einer Anwendung 63

Schritt 1: Schriftarten- und Farbressourcen ändern	64
Schritt 2: Desktop-Anwendungsstammverzeichnis erstellen	66
Schritt 3: Registrierungspaketverzeichnisse erstellen	66
Schritt 4: Aktionen und Datentypen für die Anwendung erstellen	69
Schritt 5: Hilfedateien in das Registrierungspaket einfügen	72
Schritt 6: Symbole für die Anwendung erstellen	73
Schritt 7: Die Anwendungsgruppe erstellen	74
Schritt 8: Anwendungen mit dtappintegrate registrieren	81
Beispiel zur Erstellung eines Registrierungspakets	84
Erforderliche Informationen zu "BesterTextEditor"	84
Schritte zum Registrieren von "BesterTextEditor"	85
6. Hinweise zur Konfiguration	93
Verzeichnisstruktur des Solaris CDE	93
/usr/dt	94
/etc/dt	94
/var/dt	95
Home-Verzeichnis	95
Zentrale Konfigurationsdateien	95
Xconfig	95
Xservers	96
Login Server starten	96
Ein installiertes CDE von einer anderen Workstation- oder Netzwerkserver-Installationsposition einhängen	99
▼ Ein installiertes CDE einhängen	99
▼ Ein CDE-Verzeichnis aushängen	100
Den Desktop für mehrere Bildschirme konfigurieren	100
▼ Den Desktop auf mehreren Bildschirmen starten	100
Desktops im Netzwerkbetrieb	102

	XTerminals verwenden	103
	Schriftartenpfad und Sprachtreiber für die Anmeldung	104
	Workstations als XTerminals einsetzen	105
▼	Mit Hilfe der Auswahl eine Host—CDE—Anmeldung bestimmen	106
▼	Einen bestimmten Host— CDE—Anmeldebildschirm verwenden	106
▼	Den ersten verfügbaren Host-Anmeldebildschirm verwenden	106
	Spezielle CDE-Konfigurationen	107
	Den Druck von E-Mail-Nachrichten anpassen	107
	Neues Datumsformat für Kalender	108
	Das AnswerBook—Package über das Netz hinzufügen	109
	Das CDE—Environment außerhalb des CDE-Desktop einrichten	109
	Desktop—Umgebungsdatei	110
	Disketten und CDs im Apple Macintosh Application Environment einsetzen	110
7.	Desktop in einem Netzwerk konfigurieren	113
	Übersicht über das Desktop im Netzwerk	114
	Typen von Desktop Services im Netzwerk	114
	Typische Netzwerksituationen	115
	Weitere Netzwerksituationen	117
	Zusammenfassung—Servertypen	118
	Allgemeine Schritte für die Konfiguration eines Desktops im Netzwerk	119
	Netzwerk des Basisbetriebssystems für das Desktop konfigurieren	119
	Anmeldeabrechnungen für Benutzer einrichten	120
	Zugriff auf verteilte Dateisysteme konfigurieren	120
	Zugriff auf ferne Drucker konfigurieren	122
	Elektronische Post konfigurieren	122
	X-Berechtigung konfigurieren	122
	Clients und Server für das Desktop konfigurieren	123

- Anmelde- und Session-Services konfigurieren 123
- Weitere anwendungsbezogene Services konfigurieren 124
- Anwendungs-Services verwalten 129
 - Suchpfadumgebungsvariablen 129
 - Anwendungs-Server und dessen Clients konfigurieren 129
 - Datenbank-, Symbol- und Hilfe-Services 131
 - Spezielle Konfigurationen für Anwendungen im Netzwerk 133
- 8. Verwaltung und Drucken über das Desktop konfigurieren 139**
 - Drucker hinzufügen und löschen 139
 - ▼ Drucker dem Desktop hinzufügen 140
 - ▼ Drucker vom Desktop löschen 140
 - Intervall, in dem Jobs aktualisiert werden, ändern 141
 - Druckersymbolbilder 141
 - Symboldateinamen und -größen 141
 - ▼ Symbol, Druckerbezeichnung oder Beschreibung eines Druckers global ändern 142
 - Standarddrucker konfigurieren 142
 - ▼ Standarddrucker ändern 142
 - Druckerkonzepte 143
- 9. Desktop-Suchpfade 145**
 - Desktop-Suchpfade und ihre Umgebungsvariablen 146
 - Wert eines Suchpfads setzen 147
 - ▼ Aktuellen Wert für einen Suchpfad anzeigen (Ausgabevariable) 147
 - ▼ Persönliche Änderungen an einem Suchpfad vornehmen 148
 - ▼ Systemweite Änderungen an einem Suchpfad vornehmen 148
 - Anwendungssuchpfad 149
 - Standardanwendungssuchpfad 149
 - Umgebungsvariablen des Anwendungssuchpfads 150

Syntax für die Eingabevariablen des Anwendungssuchpfads	150
Wie der Wert des Anwendungssuchpfads assembliert wird	151
Rangfolge der systemweiten lokalen Positionen ändern	151
Wie der Anwendungssuchpfad sich auf die Datenbank-, Symbol- und Hilfesuchpfade auswirkt	152
Datenbanksuchpfad (Aktionen/Datentypen)	153
Standarddatenbanksuchpfad	154
Wie der Anwendungssuchpfad sich auf den Datenbanksuchpfad auswirkt	154
Umgebungsvariablen des Datenbanksuchpfads	154
Syntax für die Eingabevariablen des Datenbanksuchpfads	155
Wie der Datenbanksuchpfad assembliert wird	156
Symbolsuchpfad	156
Standardsymbolsuchpfad	156
Wie der Anwendungssuchpfad sich auf den Symbolsuchpfad auswirkt	157
Umgebungsvariablen des Symbolsuchpfads	157
Syntax für die Eingabevariablen des Symbolsuchpfads	158
Wie der Symbolsuchpfad assembliert wird	158
Hilfesuchpfad	159
Standardhilfesuchpfad	159
Wie der Anwendungssuchpfad sich auf den Hilfesuchpfad auswirkt	160
Umgebungsvariablen des Hilfesuchpfads	160
Syntax für die Eingabevariablen des Hilfesuchpfads	160
Wie der Hilfesuchpfad assembliert wird	161
Positionen der Suchpfade	161
10. Einführung in Aktionen und Datentypen	163
Weiterführende Informationen	163

Einführung in Aktionen	164
Wie Aktionen Symbole für Anwendungen erstellen	168
Wie Aktionen Datendateien als Argumente verwenden	171
Zusätzliche Verwendung für Aktionen	172
Einführung in Datentypen	172
Was ist ein Datentyp?	172
Wie Datentypen Datendateien mit Aktionen verbinden	174
Drucken über das Desktop für einen Datentyp einrichten	178
11. Aktionen und Datentypen mit 'Aktion erstellen' erstellen	181
Funktionsweise von 'Aktion erstellen'	181
Einschränkungen von 'Aktion erstellen'	182
Einschränkungen bei Aktionen	182
Einschränkungen bei Datentypen	183
Aktionen und Datentypen für eine Anwendung mit 'Aktion erstellen' erstellen	183
▼ Eine Aktion für eine Anwendung erstellen	184
▼ Einen oder mehrere Datentypen für eine Anwendung erstellen	188
Symbol über das Dialogfenster 'Gruppe suchen' angeben	196
12. Aktionen manuell erstellen	199
Gründe, eine Aktion manuell zu erstellen	200
COMMAND-Aktionen	200
MAP-Aktionen	200
TT_MSG-Aktionen (ToolTalk-Meldung)	201
Eine Aktion manuell erstellen: Allgemeine Schritte	201
Konfigurationsdateien für Aktionen	201
▼ Eine Aktion manuell erstellen	202
Beispiel für das Erstellen einer COMMAND-Aktion	203
Beispiel für das Erstellen einer MAP-Aktion	204
▼ Datenbank für Aktionen und Datentypen neu laden	205

- Aktionsdatei (Symbol) für eine Aktion erstellen 205
- Das Symbol für eine Aktion festlegen 207
- ▼ Eine vorhandene Aktionsdefinition ändern 208
 - Reihenfolge der Aktionsdefinitionen 209
- Ausführungszeichenfolge für eine COMMAND-Aktion erstellen 211
 - Allgemeine Merkmale von Ausführungszeichenfolgen 211
 - Eine Aktion ohne Argumente erstellen 212
 - Eine Aktion erstellen, die eine abgelegte Datei akzeptiert 213
 - Eine Aktion erstellen, die zur Angabe eines Dateiarguments auffordert 214
 - Eine Aktion erstellen, die eine abgelegte Datei akzeptiert oder zur Angabe eines Dateiarguments auffordert 214
 - Eine Aktion erstellen, die zur Angabe eines Nicht-Dateiarguments auffordert 215
 - Ein Dateiargument als Zeichenfolge interpretieren 215
 - Shell-Funktionen in einer Aktion bereitstellen 216
 - COMMAND-Aktionen für mehrere Dateiargumente erstellen 216
- Fensterunterstützung und Terminal-Emulationen für COMMAND-Aktionen 219
 - Fensterunterstützung für die Aktion festlegen 219
 - Befehlszeilenoptionen für den Terminal-Emulator angeben 220
 - Einen anderen Standard-Terminal-Emulator festlegen 220
- Aktionen auf bestimmte Argumente beschränken 221
 - Eine Aktion auf einen bestimmten Datentyp beschränken 222
 - Eine Aktion anhand der Anzahl der Argumente beschränken 222
- ▼ Unterschiedliche Funktionsweisen für Doppelklicken oder Ablegen implementieren 223
 - Eine Aktion über den Argumentmodus beschränken 224
- Aktionen erstellen, die Anwendungen auf entfernten Systemen ausführen 224
 - Eine Aktion erstellen, die eine entfernte Anwendung ausführt 225

Variablen in Aktions- und Datentypdefinitionen verwenden	226
Zeichenfolgevariablen in einer Aktion verwenden	226
Umgebungsvariablen in Aktionen und Datentypen verwenden	227
Aktionen an der Befehlszeile aufrufen	227
Syntax von <code>dtaction</code>	227
Eine Aktion erstellen, die andere Aktionen ausführt	228
Eine Aktion erstellen, die für einen anderen Benutzer ausgeführt wird	228
Lokalisierte Aktionen erstellen	229
Verzeichnisse für lokalisierte Aktionen	229
▼ Eine vorhandene Aktion sprachabhängig einrichten	230
Aktionen für ToolTalk-Anwendungen erstellen	230
Die Felder <code>addressing</code> und <code>disposition</code>	231
Nicht unterstützte Meldungen	231
13. Datentypen manuell erstellen	233
Weiterführende Informationen	233
Gründe, einen Datentyp manuell zu erstellen	234
Komponenten einer Datentypdefinition: Kriterien und Attribute	234
Datentyp manuell erstellen: Allgemeine Schritte	235
Konfigurationsdateien für Datentypen	235
▼ Datentypdefinition erstellen	235
Beispiel für die Erstellung einer persönlichen Aktion und eines Datentyps	237
Datenattribute eines Datentyps definieren	239
Symbolbild angeben, das für einen Datentyp verwendet wird	239
Datentypen Aktionen zuordnen	240
Dateien auf der Grundlage des Datentyps verdecken	241
Verhalten der Datei bei Bearbeitung angeben	241
Datenkriterien für einen Datentyp definieren	242

- Datentypenzuordnung auf der Grundlage des Dateinamens 243
- Datentypenzuordnung auf der Grundlage der Position 245
- Datentypenzuordnung auf der Grundlage des Dateinamens und der Position 245
- Dateimodi als Kriterium der Datentypzuordnung verwenden 246
- Datentypzuordnung auf der Grundlage des Inhalts 248
- ▼ Datentypen mit mehreren unabhängigen Kriterien erstellen 249
- Sprachenabhängige Datentypen erstellen 250
 - Positionen für sprachenabhängige Datentypen 251
- ▼ Datentypen sprachenabhängig erstellen 251
- 14. Symbole für das Desktop erstellen 253**
 - Symbolbilddateien 253
 - Symboldateiformate 254
 - Symboldateinamen 254
 - Symbolgrößenkonventionen 255
 - Symbolsuchpfad 256
 - Auf Symbole über das Netzwerk zugreifen 256
 - Symbolzuordnungen 256
 - Symboldateien angeben 256
 - ▼ Symbole einer Aktion oder einem Datentyp zuordnen 257
 - ▼ Symbole in einem Bedienelement im Bedienfeld anzeigen 258
 - ▼ Symbole einem Anwendungsfenster zuordnen 259
 - ▼ Dateimanager zur Anzeige von Symbolen verwenden 259
 - Empfehlungen zur Erstellung von Symbolen 260
 - Farbpalette 260
- 15. Erweiterte Anpassung des vorderen Steuerbereichs 263**
 - Konfigurationsdateien des vorderen Steuerbereichs 264
 - Standardkonfigurationsdatei für den vorderen Steuerbereich 264

Suchpfad für die Konfigurationsdateien des vorderen Steuerbereichs	264
Den vorderen Steuerbereich konfigurieren: Vorrangsregeln	265
Dynamisch erstellte Dateien für den vorderen Steuerbereich	266
Die Anpassungen der Benutzeroberfläche verwalten	266
▼ Persönliche Anpassungen verhindern	267
▼ Eine gelöschte Steuerung oder einen gelöschten zusätzlichen Steuerbereich wiederherstellen	267
Aufbau der Definition des vorderen Steuerbereichs	268
Komponenten des vorderen Steuerbereichs	268
Allgemeine Syntax der Definition des vorderen Steuerbereichs	269
Den Hauptsteuerbereich ändern	272
▼ Eine Steuerung dem Hauptsteuerbereich hinzufügen	273
▼ Eine Steuerung entfernen	274
▼ Eine Steuerung ändern	275
▼ Die Position von Steuerungen ändern	275
▼ Eine Steuerung im vorderen Steuerbereich ersetzen	276
Das von einer Steuerung verwendete Symbol festlegen	277
Zusätzliche Steuerbereiche erstellen und ändern	278
▼ Einen neuen systemweiten zusätzlichen Steuerbereich erstellen	279
Integrierte zusätzliche Steuerbereiche anpassen	279
▼ Die Selbstschließfunktion zusätzlicher Steuerbereiche ändern	282
Steuerungen im vorderen Steuerbereich definieren	283
Steuerungsdefinitionen	283
Steuerungstypen	283
▼ Eine neue Steuerung erstellen	284
Den Wechselbereich anpassen	292
▼ Den Standardwert für die Anzahl der Arbeitsbereiche ändern	292
▼ Die Zeilenanzahl des Wechselbereichs ändern	292

- ▼ Steuerungen im Wechselbereich ändern oder hinzufügen 292
- Allgemeine Konfiguration des vorderen Steuerbereichs 294
 - Allgemeine Arbeitsschritte 294
- ▼ Die Standardposition des vorderen Steuerbereichs ändern 294
- ▼ Steuerungen im Hauptsteuerbereich beschriften 295
- ▼ Das Klickverhalten von Steuerungen ändern 295
- ▼ Einen vorderen Steuerbereich neu erstellen 295
 - Beispiel zur Erstellung eines persönlichen vorderen Steuerbereichs mit drei Zeilen 296
- 16. Workspace Manager konfigurieren 299**
 - Konfigurationsdateien des Workspace Managers 300
 - ▼ Persönliche Konfigurationsdateien anlegen und ändern 301
 - ▼ Systemweite Konfigurationsdateien anlegen 301
 - ▼ Weitere Dateien einbinden 301
 - ▼ Workspace Manager neu starten 302
 - Arbeitsbereiche anpassen 303
 - ▼ Anzahl von Arbeitsbereichen für das gesamte System ändern 303
 - ▼ Systemweit gültige Namen von Arbeitsbereichen 304
 - ▼ Zusätzliche Hintergrund-Ablagen erzeugen 304
 - ▼ Hintergrund-Ablagen durch Grafiken ersetzen 305
 - Menüs im Workspace Manager 306
 - Menüsyntax im Workspace Manager 306
 - ▼ Das bestehende Menü 'Arbeitsbereich' (Root-Menü) ändern 307
 - ▼ Ein neues Menü 'Arbeitsbereich' (Root-Menü) erstellen 308
 - ▼ Ein neues Menü 'Fenster' erstellen 308
 - Maustastenbelegung ändern 309
 - Syntax für die Maustastenbelegung 309
 - ▼ Neue Maustastenbelegungen hinzufügen 310

▼	Neue Sets für Maustastenbelegungen definieren	311
	Tastaturbelegung ändern	311
	Standard-Tastaturbelegungen für den Desktop	312
	Syntax für Tastaturbelegungen	312
▼	Benutzerdefinierte Sets für die Tastaturbelegung erstellen	313
	Wechseln zwischen Standard- und benutzerdefinierter Funktionsweise	314
17.	Anwendungsressourcen, Schriftarten und Farben verwalten	315
	Anwendungsressourcen einstellen	315
▼	Systemweite Ressourcen einstellen	316
▼	Persönliche Ressourcen einstellen	316
	Wie der Desktop Ressourcen lädt	316
	UNIX-Belegungen definieren	316
▼	EMACS-Umsetzungen angeben	316
▼	EMACS-Umsetzungen ändern	317
	UNIX-Belegungen über die UNIX-Belegungsdatei	317
	Schriftarten verwalten	322
	Schriftartressourcen des Desktop einstellen	322
▼	Verfügbare Schriftarten auflisten	323
▼	Schriftarten auf der Befehlszeile angeben	324
	X Logical Font Description (XLFD)	324
	Benutzer-Schriftartengruppen im Dateisystem	327
	Erstellung von Schriftartengruppen durch den Systemverwalter	327
	Farben verwalten	327
	Farbpaletten	328
	Farbskalen	328
	Farbe mit dem Style-Manager steuern	332
	Anzahl der vom Style-Manager benutzten Farben	332
	Schattenstärken für Anwendungsfenster einstellen	336

18.	Länderspezifische Desktop-Sitzungen konfigurieren	337
	Umgebungsvariable LANG verwalten	337
	Sprache für mehrere Benutzer einstellen	338
	Sprache für eine einzige Sitzung einstellen	339
	Sprache für einen bestimmten Benutzer einstellen	339
	Umgebungsvariable LANG und die Sitzungskonfiguration	339
	Weitere NLS-Umgebungsvariablen einstellen	340
	Schriftarten finden	340
	Ressourcendatei app-defaults länderspezifisch anpassen	341
	Aktionen und Datentypen länderspezifisch anpassen	341
	Symbole und Bitmaps länderspezifisch anpassen	342
	Namen von Hintergrund-Ablagen länderspezifisch anpassen	342
	Palettennamen länderspezifisch anpassen	343
	Hilfebände länderspezifisch anpassen	344
	Meldungskataloge länderspezifisch anpassen	344
	Fernausführung länderspezifischer Desktop-Anwendungen	344
	Tastaturbelegung zurücksetzen	345
A.	Hilfeseite dtconfig(1)	347
	Index	351

Vorwort

Dieses Handbuch behandelt die fortgeschrittene Anpassung des Erscheinungsbilds und des Verhaltens der allgemeinen Desktop-Umgebung. Es enthält Kapitel über:

- die Anpassung der Systeminitialisierung und der Anmeldung und Session-Start
- das Hinzufügen von Anwendungen das Bereitstellen von Schnittstellendarstellungen für Anwendungen und ihre Daten
- das Konfigurieren von Desktop-Prozessen, Anwendungen und Daten über das Netzwerk
- die Anpassung von Desktopservices, wie z.B. Fensterverwaltung, Drucken, Farben und Schriftarten

Hinweis - Der Begriff "x86" wird zur Bezeichnung der Intel-Mikroprozessoren der Familie 8086 verwendet, einschließlich der Prozessoren Pentium und Pentium Pro und kompatibler Mikroprozessoren von AMD und Cyrix. In diesem Dokument wird "x86" zur Bezeichnung der Architektur verwendet, während "*Intel Platform Edition*" im Produktnamen verwendet wird.

Wer sollte dieses Buch benutzen?

- Systemadministratoren. Viele der Aufgaben in diesem Buch benötigen Zugriffsberechtigung als Benutzer 'root'.
- Fortgeschrittene Benutzer, die Anpassungen durchführen wollen, die nicht mit der Benutzerschnittstelle des Desktop durchgeführt werden können. Das Desktop verfügt über benutzerspezifische Positionen für viele Konfigurationsdateien.

Bevor Sie dieses Buch lesen

Benutzer sollten die folgenden Bücher kennen:

- *Solaris Common Desktop Environment: User's Guide*
- *Solaris Common Desktop Environment: User's Transition Guide*

Wie dieses Buch aufgebaut ist

Dieses Handbuch enthält folgende Kapitel:

Kapitel 1 beschreibt, wie das Erscheinungsbild und das Verhalten des Desktop-Anmeldemanagers konfiguriert wird.

Kapitel 2 beschreibt, wie das Desktop Sessions speichert und abrufen und wie der Session-Start angepasst wird.

Kapitel 3 beschreibt Solaris CDE-Startdateien und mögliche Solaris CDE-Startprobleme und enthält Vorschläge zur Behebung dieser Probleme.

beschreibt, wie der Anwendungsmanager Anwendungen zusammenstellt und erklärt, wie Anwendungen hinzugefügt werden.

Kapitel 5 beschreibt, wie ein Registrierungspaket für eine Anwendung erstellt wird.

Kapitel 6 befasst sich mit erweiterten Konfigurationsarbeiten wie benutzerdefinierten Anmeldungen, Einrichtung mehrerer Bildschirme, Desktops und X-Terminals im Netzwerk, Änderung der Benutzer-Punktdateien, Anpassung des Mail-Drucks, Setup der Desktop-Umgebung und Fehlerprotokolltypen.

beschreibt, wie Desktop-Services, Anwendungen und Daten über ein Netzwerk verteilt werden.

beschreibt, wie Desktop-Drucker hinzugefügt und entfernt werden und wie der Standarddrucker angegeben wird.

Kapitel 9 beschreibt, wie das Desktop Anwendungen, Hilfedateien, Symbole und andere Desktopdaten über das Netzwerk findet.

gibt eine Einführung in die Konzepte von Aktionen und Datentypen und erklärt, wie sie als Benutzerschnittstelle für Anwendungen benutzt werden.

beschreibt, wie die Anwendung 'Aktion erstellen' benutzt wird, um Aktionen und Datentypen zu erstellen.

Kapitel 12 beschreibt, wie Aktionsdefinitionen durch das Bearbeiten einer Datenbankkonfigurationsdatei erstellt werden.

Kapitel 13 beschreibt, wie Datentypdefinitionen durch das Bearbeiten einer Datenbankkonfigurationsdatei erstellt werden.

Kapitel 14 beschreibt, wie der Symboleditor verwendet wird und die Namenskonventionen, Größen und Suchpfade für Desktopsymbole.

Kapitel 15 beschreibt, wie neue systemweite Bedienelemente und Bedientafeln erstellt werden und andere Anpassungen von Bedientafeln.

Kapitel 16 beschreibt, wie Fenster, Maustastenbelegungen, Tastaturbelegungen und Arbeitsbereichsmanagemenüs angepasst werden.

Kapitel 17 beschreibt, wie Anwendungsressourcen eingestellt werden und wie das Desktop Schriftarten und Farben verwendet.

Kapitel 18 beschreibt Systemverwaltungsaufgaben für Systeme, die internationale Sessions ausführen.

Anhang A ist eine Kopie der Hilfeseite dtconfig(1).

Sun-Dokumente bestellen

Im Rahmen des Programms SunDocssm sind mehr als 250 Handbücher von Sun Microsystems, Inc verfügbar. Wenn Sie in den USA, in Kanada, Europa oder Japan leben, können Sie einzelne Dokumente oder vollständige Dokumentationssätze bestellen.

Eine Liste der Dokumente und Informationen zur Bestellung finden Sie im Katalogbereich von "SunExpresstm On The Internet" unter der Adresse <http://www.sun.com/sunexpress>.

Bedeutung typographischer Kennzeichnungen

Die folgende Tabelle beschreibt die typographischen Kennzeichnungen, die in diesem Buch verwendet werden.

TABELLE P-1 Typografische Konventionen

Schriftart oder Symbol	Bedeutung	Beispiel
AaBbCc123	Die Namen von Befehlen, Dateien, Verzeichnissen und Schlüsselwörtern.	Die Datei <code>.dtprofile</code> bearbeiten. <code>ls -a</code> verwenden, um alle Dateien aufzulisten.
AaBbCc123	Zeichen, die von Ihnen eingegeben werden (im Unterschied zu den Zeichen, die auf dem Bildschirm angezeigt werden).	Rechnername% su Paßwort :
<i>AaBbCc123</i>	Parameter oder Variable, durch realen Wert ersetzen.	Um eine Datei zu löschen, <code>rm <i>Dateiname</i></code> eingeben.
<i>AaBbCc123</i>	Buchtitel, neue Wörter oder Begriffe oder Wörter hervorheben.	Kapitel 6 im <i>Benutzerhandbuch lesen</i> . Diese Optionen werden <i>Klasse</i> genannt . Der Benutzer <i>muß</i> root sein, um dies zu tun.

Shell-Eingabeaufforderungen in Befehlsbeispielen

Die folgende Tabelle zeigt die Standard-Systemeingabeaufforderung und die Superuser-Eingabeaufforderung für die C-Shell, die Bourne-Shell und die Korn-Shell.

TABELLE P-2 Shell-Eingabeaufforderungen

Shell	Eingabeaufforderung
Eingabeaufforderung der C-Shell	Rechnername%
Superuser-Eingabeaufforderung der C-Shell	Rechnername#

TABELLE P-2 Shell-Eingabeaufforderungen *(fortgesetzt)*

Shell	Eingabeaufforderung
Eingabeaufforderung der Bourne- und der Korn-Shell	\$
Superuser-Eingabeaufforderung der Bourne- und der Korn-Shell	#

Den Anmeldemanager konfigurieren

Der Anmeldemanager ist ein Server, der einen Anmeldebildschirm anzeigt, die Zugriffsberechtigungen der Benutzer prüft und Benutzersitzungen startet. Die grafische Anmeldung ist auf grafikfähigen Bildschirmen eine attraktive Alternative zur Anmeldung im herkömmlichen Zeichenmodus. Die vom Anmeldeserver verwalteten Anzeigen können direkt dem Anmeldeserver zugeordnet oder mit einem X-Terminal bzw. einer Workstation im Netzwerk verbunden werden.

Hinweis - Der Anmeldeserver kann nur vom Benutzer 'root' gestartet, angehalten oder konfiguriert werden.

- „Den Anmeldeserver starten“ auf Seite 2
- „Lokale und Netzwerkanzeigen verwalten“ auf Seite 2
- „Auf Fehler prüfen“ auf Seite 9
- „Den Anmeldeserver beenden“ auf Seite 9
- „Das Erscheinungsbild der Anzeige ändern“ auf Seite 11
- „Den Anmeldemanager verwalten“ auf Seite 19
- „Dateien des Anmeldemanagers“ auf Seite 20

Der Anmeldeserver führt folgende Aktion aus:

- Anzeigen einer Anmeldeanzeige uneingeschränkt oder auf Anforderung auf lokalen und Netzwerk-Grafikanzeigen.
- Anpassen direkt angeschlossener Zeichenkonsolenanzeigen.
- Anzeigen eines Auswahlbildschirms, mit dem der Benutzer Anmeldeanzeigen anderer Server im Netzwerk anzeigen kann.
- Kontrollierter Zugriff auf den Anmeldeserver.
- Ermöglichen des Zugriffs auf die Anmeldung im herkömmlichen Zeichenmodus.

Vom Anmeldeanager verwaltete Anzeigen können direkt mit dem Anmeldeserver oder mit einem X-Terminal bzw. einer Workstation im Netzwerk verbunden sein. Für lokale Anzeigen startet der Anmeldeserver automatisch einen X-Server und zeigt eine Anmeldeanzeige an. Für Netzwerkanzeigen (z. B. X-Terminals) unterstützt der Anmeldeserver das XDMCP-Protokoll (X Display Manager Protocol) 1.0, mit dem Anzeigen eine Anmeldeanzeige vom Anmeldeserver angefordert werden können.

Den Anmeldeserver starten

Der Anmeldeserver wird normalerweise beim Starten des Systems ausgeführt. Er kann aber auch in der Befehlszeile gestartet werden.

- Mit dem folgenden Befehl kann der Anmeldeserver so konfiguriert werden, daß er beim Systemstart ausgeführt wird: `/usr/dt/bin/dtconfig -e`.

Der Server wird dann bei jedem Neustart des Systems automatisch aufgerufen.

Weitere Informationen zum Desktop-Konfigurationsprogramm `dtconfig` finden Sie in **Anhang A**. Er enthält eine Kopie der Hilfeseite `dtconfig.1`.

- Mit dem folgenden Befehl starten Sie den Anmeldeserver in der Befehlszeile:
`/usr/dt/bin/dtlogin -daemon ; exit`.

Hinweis - Der Anmeldeserver kann zwar in der Befehlszeile aufgerufen werden, um eine bestimmte Konfiguration zu testen, er sollte aber normalerweise beim Systemstart ausgeführt werden.

Lokale und Netzwerkanzeigen verwalten

Abbildung 1-1 zeigt eine mögliche Konfiguration des Anmeldeservers.

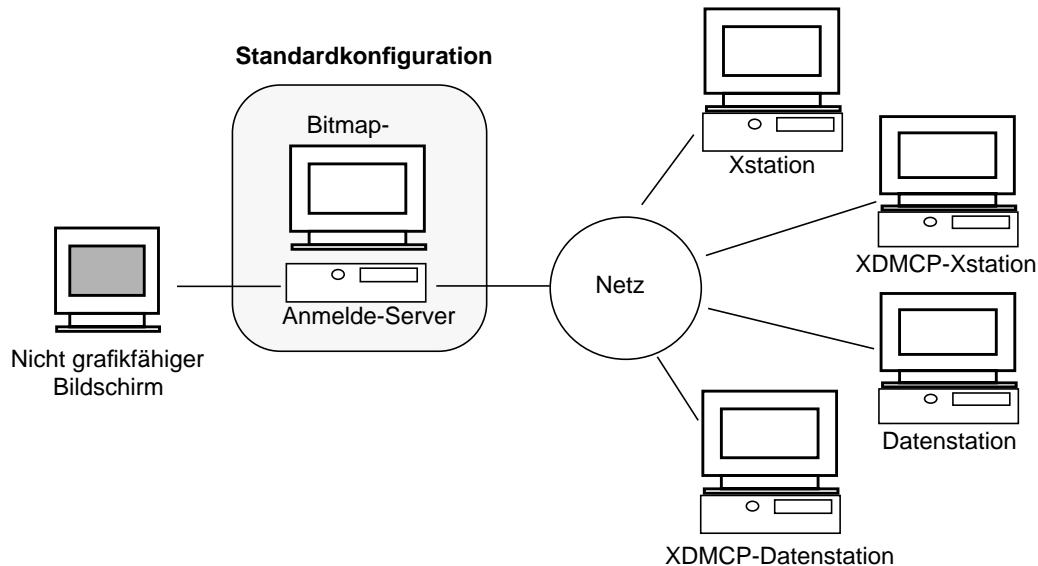


Abbildung 1-1 Mögliche Konfiguration des Anmeldeservers

Die Prozeß-ID des Anmeldeservers ermitteln

Standardmäßig wird die Prozeß-ID des Anmeldeservers in der Datei `/var/dt/Xpid` gespeichert.

Soll eine andere Datei verwendet werden, passen Sie die Ressource `Dtlogin.pidFile` in der Datei `Xconfig` entsprechend an. Wird eine Änderung vorgenommen, muß das angegebene Verzeichnis beim Starten des Anmeldeservers bereits existieren.

Um die Datei `Xconfig` zu ändern, kopieren Sie sie aus `/usr/dt/config` in das Verzeichnis `/etc/dt/config`. Nehmen Sie die Änderungen in der Datei `/etc/dt/config/Xconfig` vor, und weisen Sie den Anmeldeserver durch folgenden Befehl an, `Xconfig` erneut zu lesen:

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

Dadurch wird der Befehl `kill -HUP Anmeldeserver_Prozeß_ID` aufgerufen.

Soll die Prozeß-ID des Anmeldeservers beispielsweise in der Datei `/var/myserver/Dtpid` gespeichert werden, nehmen Sie folgende Zeile in die Datei `Xconfig` auf:

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

Nach dem Neustarten des Anmeldeservers wird seine Prozeß-ID in der Datei `/var/myserver/Dtpid` gespeichert. Das Verzeichnis `/var/myserver` muß bereits vorhanden sein.

Eine Anmeldeanzeige auf einem lokalen Bildschirm anzeigen

Beim Starten ermittelt der Anmeldeserver in der Datei `xservers`, ob ein X-Server gestartet werden muß sowie ob und wie Anmeldeanzeigen auf lokalen und Netzwerkbildschirmen angezeigt werden sollen.

Um die Datei `xservers` zu ändern, kopieren Sie sie aus `/usr/dt/config` in das Verzeichnis `/etc/dt/config`. Nehmen Sie die Änderungen in `/etc/dt/config/xservers` vor, und weisen Sie den Server durch folgenden Befehl an, `xservers` erneut zu lesen:

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

Dadurch wird der Befehl `kill -HUP Anmeldeserver_Prozeß_ID` ausgeführt.

Die Zeilen in der Datei `xservers` müssen im folgenden Format angegeben werden:

```
Anzeigename Anzeigeklasse  
Anzeigetyp X_Server_Befehl
```

Die Argumente haben folgende Bedeutung:

Anzeigename - Der für die Verbindung mit dem X-Server zu verwendende Verbindungsname (:0 im folgenden Beispiel). Der Wert * (Sternchen) wird zu *Hostname:0* erweitert. Der angegebene Wert muß mit der Verbindungsnummer im Argument *X_Server_Befehl* übereinstimmen.

Anzeigeklasse - Spezifische Ressourcen für die Anzeige (im folgenden Beispiel `Local`).

Anzeigetyp - Dieses Argument gibt an, ob eine lokale oder eine Netzwerkanzeige vorliegt und wie die Option 'Befehlszeilenanmeldung' in der Anmeldeanzeige verwaltet wird (`local@console` im folgenden Beispiel).

X_Server_Befehl - Die Befehlszeile, Verbindungsnummer und andere Optionen, die der Anmeldeserver zum Starten des X-Servers verwendet (im folgenden Beispiel `/usr/bin/X11/X: 0`). Die Verbindungsnummer muß mit dem in *Anzeigename* angegebenen Wert übereinstimmen.

Die Standardzeile in `xservers` sieht ungefähr folgendermaßen aus:

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

Den Anmeldeserver ohne lokale Anzeige ausführen

Besitzt das Anmeldeserversystem keine Grafikanzeige, können Sie den Anmeldeserver auch ohne lokale Anzeige starten, indem Sie die Zeile für die lokale

Anzeige in der Datei `Xservers` mit einem `#`-Zeichen auskommentieren. Ein Beispiel:

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

Der Server läuft dann nach dem Start als Hintergrundprozeß und wartet auf Anforderungen von Netzwerkanzeigen.

Auf die Befehlszeilenanmeldung in einer lokalen Anzeige zugreifen

Wählt der Benutzer in der Anmeldeanzeige die Option 'Befehlszeilenanmeldung', beendet der Anmeldeserver vorübergehend den X-Server und ermöglicht auf einem grafikfähigen Terminal den Zugriff auf die herkömmliche Anmeldung in der Befehlszeile. Nachdem sich der Benutzer an- und wieder abgemeldet hat bzw. nach einer definierten Zeitüberschreitung startet der Anmeldeserver den X-Server neu.

Hinweis - Auf Netzwerkanzeigen ist die Option 'Befehlszeilenanmeldung' nicht verfügbar.

Das Argument *Anzeigetyp* steuert die Funktionsweise der Option 'Befehlszeilenanmeldung'. Das Format von *Anzeigetyp* lautet:

- `local@Anzeige_Terminalgerät`
- `local`
- `foreign`

Wird `local@Anzeige_Terminalgerät` angegeben, sucht der Anmeldeserver den X-Server und `/dev/Anzeige_Terminalgerät` auf demselben physischen Gerät und setzt voraus, daß eine Befehlszeilenanmeldung (normalerweise `getty`) auf dem Gerät ausgeführt wird. Wählt der Benutzer 'Befehlszeilenanmeldung', wird der X-Server beendet und ermöglicht den Zugriff auf die Befehlszeilenanmeldung (`getty`), die auf `/dev/Anzeige_Terminalgerät` ausgeführt wird.

Die Option 'Befehlszeilenanmeldung' kann für eine Anzeige deaktiviert werden, indem der Wert `none` als *Anzeige_Terminalgerät* angegeben wird. Der Standardwert für *Anzeige_Terminalgerät* ist `console`. Wird der Wert `local` angegeben, erhält *Anzeige_Terminalgerät* automatisch den Standardwert `console`. Bei Angabe von `foreign` wird die Option 'Befehlszeilenanmeldung' deaktiviert.

Hinweis - Die Option 'Befehlszeilenanmeldung' ist auf einer lokalen Anzeige deaktiviert, wenn der Anmeldeserver von der Befehlszeile aus gestartet wird.

Eine zeichenorientierte Anzeige anpassen

Verfügt das Anmeldeserversystem über eine direkt angeschlossene Zeichenanzeige als Konsole, sollte *Anzeige_Terminalgerät* auf *none* gesetzt werden, damit die Option 'Befehlszeilenanmeldung' auf einer grafikfähigen Anmeldeanzeige deaktiviert wird.

Sie können aber auch *Anzeige_Terminalgerät* auf der Grafikanzeige in das Gerät für die Befehlszeilenanmeldung (*getty*) ändern, wenn dieses auf der Zeichenkonsole und der Grafikanzeige ausgeführt wird.

Befindet sich beispielsweise die Befehlszeilenanmeldung für die Grafikanzeige (*getty*) auf dem Gerät */dev/tty01*, ändern Sie den Wert von *Anzeigetyp* in *local@tty01*.

Eine Anmeldeanzeige auf einer Netzwerkanzeige anzeigen

Der Anmeldeserver kann Anforderungen von Netzwerkanzeigen zur Anzeige eines Anmeldefensters entgegennehmen. Die Netzwerkanzeige ist normalerweise ein X-Terminal, kann aber auch eine Workstation sein.

Um Anforderungen von Netzwerkanzeigen zu verwalten, unterstützt der Anmeldeserver das Protokoll XDMCP (X Display Manager Protocol) 1.0. Es ermöglicht dem Server, die Anforderungen anzunehmen oder abzulehnen. Bei den meisten X-Terminals ist XDMCP integriert.

Direkte XDMCP-Anforderungen von Netzwerkanzeigen

Wird ein X-Terminal konfiguriert, XDMCP direkt zu verwenden (Abfragemodus), wird dem X-Terminal der Hostname des Anmeldeserversystems mitgeteilt. Das X-Terminal nimmt dann beim Starten automatisch Kontakt mit dem Anmeldeserver auf, und dieser zeigt ein Anmeldefenster auf dem X-Terminal an. In der Dokumentation des X-Terminals wird beschrieben, wie das Terminal für den XDMCP-Direktmodus konfiguriert werden kann.

Die meisten X-Server unterstützen auch die Option *-query*. In diesem Modus verhält sich der X-Server wie ein X-Terminal und sendet eine Anforderung an den Anmeldeserver, damit dieser ein Anmeldefenster anzeigt. Wird beispielsweise der X-Server auf einer Grafikanzeige auf Workstation *bridget* gestartet, zeigt der Anmeldeserver *anita* eine Anmeldeanzeige auf dem X-Server an:

```
x -query anita
```

Indirekte XDMCP-Anforderungen von Netzwerkanzeigen

Wird ein X-Terminal konfiguriert, XDMCP indirekt zu verwenden, wird dem X-Terminal der Hostname des Anmeldeserversystems mitgeteilt. Das X-Terminal

nimmt dann beim Starten automatisch Kontakt mit dem Anmeldeserver auf, und dieser zeigt in einem Auswahlfenster eine Liste der anderen Anmeldeserver-Hosts im Netzwerk an. Der Benutzer kann aus der Liste den Host wählen, der ein Anmeldefenster auf dem X-Terminal anzeigen soll. In der Dokumentation des X-Terminals wird beschrieben, wie das Terminal für den indirekten XDMCP-Modus konfiguriert werden kann.

Die meisten X-Server unterstützen die Option `-indirect`. In diesem Modus nimmt der X-Server im indirekten XDMCP-Modus Kontakt mit dem Anmeldeserver auf.

Netzwerkanzeigen verwalten, die XDMCP nicht unterstützen

Ältere X-Terminals unterstützen möglicherweise XDMCP nicht. Damit der Anmeldeserver eine Anmeldeanzeige auf diesen Terminals anzeigen kann, muß der Name des X-Terminals in die Datei `xservers` aufgenommen werden.

Da sich die Anzeige in einem Netzwerk befindet, muß *Anzeigenname* den Hostnamen als Namenskomponente enthalten. Mit *Anzeigeklasse* können spezifische Ressourcen einer bestimmten Klasse von X-Terminals angegeben werden (die Klasse des X-Terminals können Sie der zugehörigen Dokumentation entnehmen). Der Wert `foreign` für *Anzeigetyp* weist den Anmeldeserver an, keinen eigenen X-Server zu starten, sondern eine Verbindung mit einem vorhandenen X-Server herzustellen. In diesem Fall wird *X_Server_Befehl* nicht angegeben.

Beispiel

Durch die folgenden Zeilen in der Datei `xservers` zeigt der Anmeldeserver eine Anmeldeanzeige auf den beiden X-Terminals `ruby` und `wolfie` an, die XDMCP nicht unterstützen:

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

Den Zugriff auf den Anmeldeserver steuern

Standardmäßig kann jeder Host im Netzwerk die Anzeige eines Anmeldefensters anfordern, der Zugriff auf den Anmeldeserver-Host hat. Der Zugriff auf den Anmeldeserver kann durch Ändern der Datei `xaccess` eingeschränkt werden.

Um die Datei `xaccess` zu ändern, kopieren Sie sie aus `/usr/dt/config` in das Verzeichnis `/etc/dt/config`. Nehmen Sie die gewünschten Änderungen in `/etc/dt/config/xaccess` vor, und weisen Sie anschließend den Anmeldeserver durch folgenden Befehl an, die Datei erneut zu lesen:

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

Dadurch wird der Befehl `kill -HUP Anmeldeserver_Prozeß_ID` ausgeführt.

Direkter XDMCP-Modus

Versucht ein Host, im direkten XDMCP-Modus eine Verbindung mit dem Anmeldeserver herzustellen, wird der Hostname mit den Einträgen in der Datei `Xaccess` verglichen. In dieser Datei sind die Hosts aufgeführt, die auf den Anmeldeserver zugreifen dürfen. Jeder Eintrag in `Xaccess` ist ein Hostname, der auch die Platzhalterzeichen `*` (Sternchen) und `?` (Fragezeichen) enthalten kann. Ein `*` entspricht beliebig vielen und ein `?` einem beliebigen Zeichen. Wird vor einem Eintrag ein `!` (Ausrufezeichen) angegeben, ist dem betreffenden Host der Zugriff nicht gestattet.

Ein Beispiel: Die Datei `Xaccess` enthält die folgenden drei Einträge:

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

Der erste Eintrag erlaubt dem Host `amazon.waterloo.com` den Zugriff auf den Anmeldeserver. Der zweite Eintrag gestattet allen Hosts den Zugriff, deren vollständiger Domain-Name mit `dept5.waterloo.com` endet. Der letzte Eintrag gibt an, daß allen anderen Hosts der Zugriff verweigert wird.

Indirekter XDMCP-Modus

Versucht ein Host, im indirekten XDMCP-Modus eine Verbindung mit dem Anmeldeserver herzustellen, wird der Hostname mit den Einträgen in der Datei `Xaccess` verglichen. In dieser Datei sind die Hosts aufgeführt, die auf den Anmeldeserver zugreifen dürfen. Die Einträge in `Xaccess` entsprechen den Einträgen für den direkten XDMCP-Modus (einschließlich Platzhalterzeichen), werden aber zusätzlich mit der Zeichenfolge `CHOOSE` gekennzeichnet. Ein Beispiel:

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

Wie im vorhergehenden Beispiel erlaubt der erste Eintrag dem Host `amazon.waterloo.com` den Zugriff auf den Anmeldeserver, der zweite allen Hosts, deren Domain-Name mit `dept5.waterloo.com` endet, und der letzte Eintrag verweigert allen anderen Hosts den Zugriff.

Nach der Zeichenfolge `CHOOSE` kann einer der folgenden Werte angegeben werden.

- `BROADCAST`
- *Liste der Hostnamen*

`BROADCAST` bewirkt, daß der Anmeldeserver eine Nachricht an alle Systeme seines Teilnetzwerks sendet, um eine Liste der verfügbaren Anmeldeserver-Hosts zu

erstellen. Wird zusätzlich eine Liste angegeben, wird diese als Liste der verfügbaren Anmeldeserver-Hosts verwendet. Ein Beispiel:

```
amazon.waterloo.com  CHOOSER shoal.waterloo.com alum.waterloo.com

*.dept5.waterloo.com  CHOOSER BROADCAST

!*  CHOOSER BROADCAST
```

Stellt `amazon.waterloo.com` eine Verbindung im indirekten XDMCP-Modus her, erhält das System eine Liste mit den beiden Hosts `shoal` und `alum`. Stellt `alice.dept5.waterloo.com` eine Verbindung her, enthält die Liste alle verfügbaren Anmeldeserver-Hosts im Teilnetzwerk des Anmeldeservers. Anforderungen anderer Hosts im indirekten XDMCP-Modus werden nicht akzeptiert.

Anstatt eine Liste der Hostnamen anzugeben, können auch wie im folgenden Beispiel ein oder mehrere Makros definiert werden, die eine Host-Liste angeben:

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

amazon.waterloo.com  CHOOSER %list1
```

Auf Fehler prüfen

Standardmäßig protokolliert der Anmeldeserver alle Fehler in der Datei `/var/dt/xerrors`. Soll eine andere Datei verwendet werden, ändern Sie die Ressource `Dtlogin.errorLogFile` in der Konfigurationsdatei `Xconfig`. Das im Pfad angegebene Verzeichnis muß beim Start des Anmeldeservers bereits existieren.

Sollen die Fehler beispielsweise in der Datei `/var/mylogs/Derrors` protokolliert werden, muß die Datei `Xconfig` folgende Zeile enthalten:

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

Wird der Anmeldeserver anschließend neu gestartet, schreibt er die Fehler in die Datei `/var/mylogs/Derrors`. Das Verzeichnis `/var/mylogs` muß jedoch bereits vorhanden sein.

Den Anmeldeserver beenden

- Soll der Anmeldeserver beim Booten des Systems nicht gestartet werden, geben Sie folgenden Befehl ein:

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

Der Anmeldeserver wird dann nicht automatisch ausgeführt, wenn das System neu gestartet wird.

- Sie können den Anmeldeserver auch beenden, indem Sie die Prozeß-ID folgendermaßen löschen:

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

Dadurch wird folgender Befehl ausgeführt: `kill Anmeldeserver_Prozeß_ID`.

Hinweis - Durch das Abbrechen des Anmeldeserver-Prozesses werden alle vom Server verwalteten Benutzersitzungen beendet.

Sie können den Anmeldeserver durch Löschen der Prozeß-ID beenden. Die Prozeß-ID wird in der Datei `/var/dt/Xpid` oder in der Datei gespeichert, die in der Ressource `Dtlogin.pidFile` in der Konfigurationsdatei `Xconfig` angegeben ist.

Sind Sie beim Beenden des Anmeldeservers beim Desktop angemeldet, wird die Desktop-Sitzung sofort geschlossen.

Die Anmeldeanzeige

Die Anmeldeanzeige stellt eine attraktive Alternative zu herkömmlichen Anmeldefenstern im Zeichenmodus dar und bietet ein Leistungsspektrum, das weit über die Möglichkeiten der Anmeldung im Zeichenmodus hinausgeht.

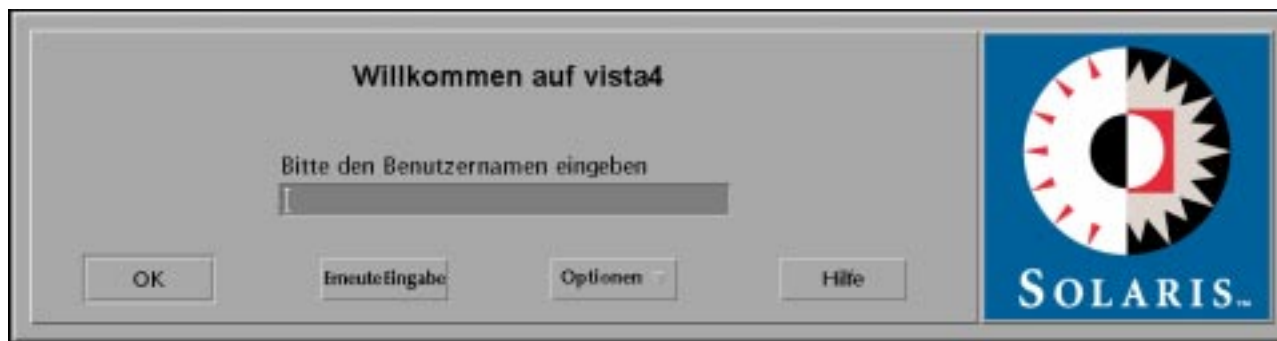


Abbildung 1-2 Desktop-Anmeldeanzeige

Wie bei der Anmeldung im Zeichenmodus muß der Benutzer einen Benutzernamen und ein Paßwort eingeben. Besteht Zugriffsberechtigung, startet der Anmeldeserver

eine Desktop-Sitzung für den Benutzer. Beendet der Benutzer die Sitzung, zeigt der Anmeldeserver ein neues Anmeldefenster an, und der Prozeß wird wiederholt.

Die Anmeldeanzeige kann folgendermaßen angepaßt werden:

- Das Erscheinungsbild der Anzeige ändern.
- Die X-Zugriffsberechtigung für den X-Server konfigurieren.
- Die Standardsprache ändern.
- Befehle vor dem Anzeigen des Anmeldefensters ausführen.
- Das Menü 'Sprachen' der Anmeldeanzeige ändern.
- Den Befehl zum Starten der Benutzersitzung angeben.
- Befehle vor dem Starten der Benutzersitzung ausführen.
- Befehle nach dem Beenden der Benutzersitzung ausführen.

Jede dieser Aktionen kann für alle Anzeigen gemeinsam oder für jede Anzeige einzeln durchgeführt werden.

Das Erscheinungsbild der Anzeige ändern

Das Erscheinungsbild der Anmeldeanzeige kann angepaßt werden, indem das Logo oder die Grafik, die Begrüßungsmeldungen und die Schriftarten geändert werden.

Um die Datei `Xresources` zu ändern, kopieren Sie sie aus `/usr/dt/config/Sprache` in das Verzeichnis `/etc/dt/config/Sprache`. Die vorgenommenen Änderungen werden beim nächsten Anzeigen des Anmeldefensters wirksam. Soll die Anmeldeanzeige sofort aktualisiert werden, wählen Sie 'Anmeldeanzeige zurücksetzen' im Menü 'Optionen'.

Folgende Attribute der Anmeldeanzeige können mit den Ressourcen in der Datei `Xresources` festgelegt werden:

`Dtlogin*logo*bitmapFile` - Bitmap- oder Pixmap-Datei für Logo

`Dtlogin*greeting*persLabelString` - Persönliche Begrüßungsmeldung

`Dtlogin*greeting*labelString` - Begrüßungsmeldung

`Dtlogin*greeting*fontList` - Schriftart für Begrüßungsmeldung

`Dtlogin*labelFont` - Schriftart für Schaltflächen und Beschriftungen

`Dtlogin*textFont` - Schriftart für Hilfe- und Fehlermeldungen

`Dtlogin*Sprache *languageName` - Alternativer Text für regionale Einstellungen
Sprache

▼ Das Logo ändern

- ◆ **Das Logo wird mit der Ressource** `Dtlogin*logo*bitmapFile` **in der Datei** `Xresources` **festgelegt.**

Es kann eine farbige Pixmap- oder eine Bitmap-Datei verwendet werden.

Im folgenden Beispiel wird die Bitmap-Datei `Mylogo` als Logo verwendet:

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

▼ Die Begrüßungsmeldung ändern

Standardmäßig zeigt der Anmeldeserver die Meldung `Willkommen auf Rechner Hostname` in der Anmeldeanzeige an. Dieser Text kann folgendermaßen geändert werden:

- ◆ **Ändern Sie den Wert der Ressource** `Dtlogin*greeting*labelString` **in der Datei** `Xresources`.

Die Ressource `labelString` kann den Wert `% LocalHost %` enthalten, der durch den Hostnamen des Anmeldeservers ersetzt wird, sowie den Wert `% DisplayName %`, für den der X-Server-Anzeigename eingesetzt wird.

Im folgenden Beispiel wird die Begrüßungsmeldung `Hier ist Hostname!` angezeigt:

```
Dtlogin*greeting*labelString: Hier ist %LocalHost%!
```

Nachdem der Benutzername eingegeben wurde, zeigt der Anmeldeserver die Standardmeldung `Willkommen: Benutzername` an. Dieser Text kann mit der Ressource `Dtlogin*greeting*persLabelString` in der Datei `Xresources` geändert werden. Der Wert von `persLabelString` kann die Variable `% s` enthalten, die durch den *Benutzernamen* ersetzt wird.

Im folgenden Beispiel wird die persönliche Begrüßungsmeldung in `Hallo Benutzername` geändert.

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

▼ Die Schriftarten ändern

Die in der Anmeldeanzeige verwendeten Schriftarten können mit den folgenden Ressourcen in der Datei `Xresources`: geändert werden.

Dtlogin*greeting*fontList - Schriftart für Begrüßungsmeldungen

Dtlogin*labelFont - Schriftart für Schaltflächen und Beschriftungen

Dtlogin*textFont - Schriftart für Hilfe- und Fehlermeldungen

Mit folgendem Befehl können die verfügbaren Schriftarten abgerufen werden:

```
xlsfonts [-Optionen][-  
fn Muster]
```

Im folgenden Beispiel wird für die Begrüßungsmeldung eine große Schrift verwendet (der Wert muß in eine Zeile eingegeben werden):

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

▼ Für jede Sprache einen anderen Text anzeigen

Soll im Menü 'Sprachen' der Anmeldeanzeige statt der Standardnamen landesspezifischer Text angezeigt werden, ändern Sie die Ressource

Dtlogin**Sprache**languageName in der Datei Xresources:

```
Dtlogin*En_US*languageName: American
```

Der Text American wird ab jetzt statt des Standardnamens En_US angezeigt.

Die Funktionsweise der Anmeldeanzeige ändern

Die Funktionsweise der Anmeldeanzeige kann durch Ändern der entsprechenden Ressourcen in der Datei Xconfig angepaßt werden.

Kopieren Sie dazu die Datei Xconfig aus /usr/dt/config in das Verzeichnis /etc/dt/config. Nehmen Sie die gewünschten Änderungen in /etc/dt/config/Xconfig vor, und veranlassen Sie anschließend den Anmeldeserver mit dem folgenden Befehl, die Datei Xconfig neu zu lesen:

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

Dadurch wird folgender Befehl ausgeführt: kill - HUP *Anmeldeserver_Prozeß_ID*.

Die Datei Xconfig enthält folgende Ressourcen:

Dtlogin*authorize - Xaccess - Dateispezifikation

Dtlogin*environment - X-Server-Umgebung

Dtlogin*language - Standardsprache

Dtlogin*languageList - Liste der Sprachen für das Menü 'Sprachen'

Dtlogin*resources - Xresources - Dateispezifikation

Dtlogin*setup - Xsetup - Dateispezifikation
Dtlogin*startup - Xstartup - Dateispezifikation
Dtlogin*session - Xsession - Dateispezifikation
Dtlogin*failsafeClient - Xfailsafe - Skriptspezifikation
Dtlogin*reset - Xreset - Skriptspezifikation
Dtlogin*userPath - PATH für Xsession und Xfailsafe
Dtlogin*systemPath - PATH für Xsetup , Xstartup und Xfailsafe
Dtlogin*systemShell - SHELL für Xsetup , Xstartup und Xfailsafe
Dtlogin.timeZone - Zeitzone für alle Skripts

Die Funktionsweise des Anmeldefensters für jede Anzeige einzeln ändern

In den folgenden Beispielen wird durch das Ändern einer Ressource in der Datei `Xconfig` die betreffende Funktion für alle Anzeigen angepaßt. Die mit einem * (Sternchen) gekennzeichneten Ressourcen können für jede Anzeige getrennt festgelegt werden. Dadurch können für einzelne Anzeigen angepaßte Anmeldefenster konfiguriert werden. Soll eine Ressource für eine bestimmte Anzeige festgelegt werden, geben Sie die Ressource als `Dtlogin* Anzeigename *Ressource` an. Folgendermaßen können Sie beispielsweise die benutzerbasierte Zugriffssteuerung für die Anzeige `expo:0` deaktivieren, jedoch für alle anderen Anzeigen aktiviert lassen:

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

Hinweis - Alle Sonderzeichen im Anzeigenamen (z. B. Doppelpunkte oder Punkte) werden durch Unterstriche (`_`) ersetzt.

Zugriffsrechte für den X-Server ändern

Standardmäßig steuert der Anmeldeserver die Zugriffsrechte für den X-Server für jeden Benutzer getrennt. Dazu werden Informationen in der Datei *Home-Verzeichnis*/`.Xauthority` gespeichert und geschützt. Nur Benutzer mit Leserechten für diese Datei können eine Verbindung mit dem X-Server herstellen. Normalerweise ist das die beste Methode, den Zugriff auf den X-Server zu steuern.

Eine Alternative zur benutzerbasierten ist die Zugriffssteuerung auf Host-Basis. Bei dieser Vorgehensweise kann jeder Benutzer auf einem Host mit Zugriffsberechtigung

eine Verbindung mit dem X-Server herstellen. Es gibt folgende Gründe für die Verwendung der Zugriffssteuerung auf Host-Basis:

- Ältere R2 und R3 X-Clients können mit der benutzerbasierten Zugriffssteuerung keine Verbindung mit dem X-Server herstellen.
- In ungesicherten Netzwerken können die Berechtigungsinformationen abgefangen werden, die zwischen dem X-Client und dem X-Server im Netzwerk ausgetauscht werden.

Mit der Ressource `Xconfig Dtlogin*authorize` wird festgelegt, daß der Anmeldeserver die benutzerbasierte Zugriffssteuerung verwendet. Soll die Zugriffssteuerung auf Host-Basis erfolgen, ändern Sie den Wert der Ressource in `False`. Ein Beispiel:

```
Dtlogin*authorize: False
```

Die X-Server-Umgebung ändern

Sollen dem X-Server beim Start durch den Anmeldeserver eine oder zwei Umgebungsvariablen und -werte zur Verfügung gestellt werden, geben Sie diese mit der Ressource `Dtlogin*environment` in der Datei `xconfig` an. Ein Beispiel:

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

Der lokale X-Server erhält den Zugriff auf die Variablen `VAR1` und `VAR2`. Die Variablen werden auch in die Skripts `Xsession` und `Xfailsafe` exportiert.

Die Standardsprache ändern

Wenn sich der Benutzer in der Anmeldeanzeige beim Desktop anmeldet, wird die Benutzersitzung mit den im Menü 'Sprachen' gewählten landesspezifischen Einstellungen ausgeführt. Wählt der Benutzer keine Sprache aus, wird die Standardeinstellung des Anmeldeservers verwendet. Die Standardsprache kann mit der Ressource `Dtlogin*language` in der Datei `xconfig` festgelegt werden. Ein Beispiel:

```
Dtlogin*language: Ja_JP
```

Der Dokumentation des Systems können Sie entnehmen, welche Sprachen verfügbar sind.

Den Inhalt des Menüs 'Sprachen' der Anmeldeanzeige ändern

Standardmäßig werden im Menü 'Sprachen' der Anmeldeanzeige alle im System installierten Sprachen angezeigt. Wählt der Benutzer eine Sprache aus der Liste, wird die Anmeldeanzeige in der gewählten Sprache erneut angezeigt. Meldet sich der Benutzer später an, wird die Desktop-Sitzung mit dieser Einstellung gestartet.

Sie können eine eigene Liste mit Sprachen erstellen, indem Sie die Ressource `Dtlogin*languageList` in der Datei `Xconfig` entsprechend ändern:

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

Der Anmeldeserver zeigt jetzt nur die Einträge `En_US` und `De_DE` im Menü 'Sprachen' an.

Befehle vor dem Anzeigen des Anmeldefensters ausführen

Der Anmeldeserver führt nach dem Starten des X-Servers, jedoch vor dem Anzeigen des Anmeldefensters das Skript `Xsetup` aus. `Xsetup` wird mit der Zugriffsberechtigung 'root' ausgeführt und enthält Befehle, die vor dem Anzeigen des Anmeldefensters aufgerufen werden müssen.

Um die Datei `Xsetup` zu ändern, kopieren Sie sie aus `/usr/dt/config` in das Verzeichnis `/etc/dt/config`. Beim nächsten Erscheinen der Anmeldeanzeige wird dann die geänderte `Xsetup`-Datei ausgeführt.

Befehle vor dem Starten der Benutzersitzung ausführen

Nachdem Benutzername/Passwort eingegeben und die Zugriffsberechtigung überprüft wurde, aber bevor die Benutzersitzung gestartet wird, führt der Anmeldeserver das Skript `Xstartup` aus. `Xstartup` wird mit der Zugriffsberechtigung 'root' ausgeführt und enthält Befehle, die vor dem Starten der Benutzersitzung ausgeführt werden müssen.

Um die Datei `Xstartup`, zu ändern, kopieren Sie sie aus `/usr/dt/config` in das Verzeichnis `/etc/dt/config`. Meldet sich der Benutzer das nächste Mal an, wird die geänderte `Xstartup`-Datei ausgeführt.

Eine Desktop-Sitzung starten

Standardmäßig startet der Anmeldeserver die Benutzersitzung, indem er das Skript `Xsession` ausführt. `Xsession` wird mit der Zugriffsberechtigung des Benutzers ausgeführt und enthält die Befehle, die zum Starten des Desktop benötigt werden.

Hinweis - Aktualisieren Sie das Skript `Xsession` nicht direkt.

Informationen zur Anpassung des Starts der Desktop-Sitzung des Benutzers finden Sie in Kapitel 2.

Eine Failsafe-Sitzung starten

Wählt der Benutzer im Untermenü 'Sitzungen' des Menüs 'Optionen' die Option 'Failsafe-Sitzungen', führt der Anmeldeserver das Skript `Xfailsafe` aus. `Xfailsafe` wird mit der Zugriffsberechtigung des Benutzers ausgeführt und enthält die Befehle, die zum Starten einer minimalen Fensterumgebung benötigt werden (normalerweise ein Terminalfenster und ein optionaler Fenstermanager).

Um die Datei `Xfailsafe` zu ändern, kopieren Sie sie aus `/usr/dt/config` in das Verzeichnis `/etc/dt/config`. Meldet sich der Benutzer das nächste Mal an, wird die geänderte Datei ausgeführt.

Befehle nach dem Beenden der Benutzersitzung ausführen

Nachdem der Benutzer die Desktop- oder Failsafe-Sitzung beendet hat, führt der Anmeldeserver das Skript `Xreset` aus. `Xreset` wird mit der Zugriffsberechtigung 'root' ausgeführt und enthält Befehle, die nach dem Beenden der Benutzersitzung ausgeführt werden müssen.

Um die Datei `Xreset` zu ändern, kopieren Sie sie aus `/usr/dt/config` in das Verzeichnis `/etc/dt/config`. Bei der nächsten Sitzung des Benutzers wird die geänderte `Xreset`-Datei verwendet.

Die Umgebung des Anmeldeservers

Die Umgebung des Anmeldeservers wird in die Skripts `Xsetup`, `Xstartup`, `Xsession`, `Xfailsafe` und `Xreset` exportiert. Tabelle 1-1 enthält eine Beschreibung der verschiedenen Werte. Es können auch weitere Variablen vom Anmeldeserver exportiert werden.

TABELLE 1-1 Umgebung des Anmeldeservers

Umgebungs- variable	Xsession				Beschreibung
	Xsetup	Xstartup	Xfailsafe	Xreset	
LANG	X	X	X	X	Standard- oder gewählte Sprache.
XAUTHORITY	X	X	X	X	Andere X-Berechtigungsdatei (optional).

TABELLE 1-1 Umgebung des Anmeldeservers (fortgesetzt)

Umgebungsvariable	Xsession				Beschreibung
	Xsetup	Xstartup	Xfailsafe	Xreset	
PATH	X	X	X	X	Wert der Ressource Dtlogin*userPath (Xsession, Xfailsafe) oder Dtlogin*systemPath (Xsetup, Xstartup, Xreset).
DISPLAY	X	X	X	X	Verbindungsnummer des X-Servers.
SHELL	X	X	X	X	In Datei /etc/passwd (Xsession, Xfailsafe) oder Ressource Dtlogin*systemShell (Xsetup, Xstartup, Xreset) angegebene Shell.
TZ	X	X	X	X	Wert der Ressource Dtlogin.timeZone oder vom System vorgegebene Zeitzone.
USER		X	X	X	Benutzername
HOME		X	X	X	In /etc/passwd angegebenes Home-Verzeichnis.
LOGNAME		X	X	X	Benutzername

Den Benutzer- oder Systempfad ändern

Der Anmeldeserver setzt die Umgebungsvariable PATH beim Ausführen der Skripts Xsession und Xfailsafe. Sie könne auch einen anderen Pfad für die Skripts angeben.

Den Benutzerpfad ändern

- ◆ **Der Benutzerpfad wird mit der Ressource Dtlogin*userPath in der Datei Xconfig festgelegt. Ein Beispiel:**

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

Den Systempfad ändern

- ◆ **Der Systempfad wird mit der Ressource `Dtlogin*systemPath` in der Datei `Xconfig` festgelegt. Ein Beispiel:**

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

▼ Die System-Shell ändern

Der Anmeldeserver setzt die Umgebungsvariable `SHELL` beim Ausführen der Skripts `Xsetup`, `Xstartup` und `Xfailsafe`. Der Standardwert ist `/bin/sh`. Soll eine andere Shell für diese Skripts verwendet werden, ändern Sie den Wert der Ressource `Dtlogin*systemShell` in der Datei `Xconfig`. Ein Beispiel:

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

▼ Die Zeitzone ändern

Der Anmeldeserver setzt die Umgebungsvariable `TZ` beim Ausführen der Skripts `Xsetup`, `Xstartup`, `Xsession`, `Xfailsafe` und `Xreset`. Der Standardwert wird aus dem System ermittelt und muß normalerweise nicht geändert werden. Soll eine andere Zeitzone verwendet werden, ändern Sie den Wert der Ressource `Dtlogin.timeZone` in der Datei `Xconfig`. Ein Beispiel:

```
Dtlogin.timeZone: CST6CDT
```

Den Anmeldemanager verwalten

Beim Aufrufen des Anmeldeservers wird ein `dtlogin`-Prozeß gestartet. Der Prozeß liest die Datei `Xconfig`, um die Anfangskonfiguration des Anmeldeservers zu ermitteln und die anderen Server-Konfigurationsdateien zu suchen. Der Anmeldeserver liest dann die Datei `Xservers`, um festzustellen, ob explizit zu verwaltende Anzeigen vorhanden sind, und anschließend die Datei `Xaccess`, um den Zugriff zu steuern.

Stellt der Anmeldeserver in der Datei `Xservers` fest, daß eine lokale Anzeige verwaltet werden muß, startet er nach Maßgabe der Informationen in der Datei einen X-Server und zeigt auf dieser Anzeige ein Anmeldefenster an.

Stellt der Anmeldeserver in der Datei `Xservers` fest, daß eine Netzwerkanzeige verwaltet werden muß, wird unterstellt, daß bereits ein X-Server mit dem angegebenen Anzeigenamen läuft und dort ein Anmeldefenster angezeigt.

Anschließend wartet der Anmeldeserver auf XDMCP-Anforderungen aus dem Netzwerk.

Für jede verwaltete Anzeige erstellt der Anmeldeserver zuerst einen neuen `dtlogin`-Prozeß. Verwaltet der Server n Anzeigen, sind $n+1$ `dtlogin`-Prozesse vorhanden. Der Anmeldeserver führt dann das Skript `Xsetup` aus, lädt die Datei `Xresources` und startet `dtgreet`, um das Anmeldefenster anzuzeigen. Nachdem der Benutzername und das Paßwort eingegeben und die Zugriffsberechtigung geprüft wurde, wird zuerst das Skript `Xstartup` und anschließend das Skript `Xsession` oder `Xfailsafe` ausgeführt. Nachdem der Benutzer die Sitzung beendet hat, wird noch das Skript `Xreset` vom Anmeldeserver gestartet.

Erhält der Anmeldeserver eine indirekte XDMCP-Anforderung, führt er `dtchooser` aus, um eine Liste der Anmeldeserver-Hosts auf der Anzeige auszugeben. Wählt der Benutzer einen Host aus der Liste, verwaltet dieser Anmeldeserver die Anzeige.

Bei den Konfigurationsdateien `Xaccess`, `Xconfig`, `Xfailsafe`, `Xreset`, *Sprache*/`Xresources`, `Xservers`, `Xsetup` und `Xstartup` sucht der Anmeldeserver standardmäßig zuerst im Verzeichnis `/etc/dt/config` und dann in `/usr/dt/config`. Er verwendet dann die zuerst gefundene Datei.

Dateien des Anmeldemanagers

Die Dateien des Anmeldemanagers befinden sich in folgenden Standardverzeichnissen:

- `/usr/dt/bin/dtlogin` - Der Anmeldeserver und Anzeigemanager.
- `/usr/dt/bin/dtgreet` - Zeigt ein Anmeldefenster für eine Anzeige an.
- `/usr/dt/bin/dtchooser` - Zeigt ein Auswahlfenster für eine Anzeige an.
- `/usr/dt/bin/Xsession` - Startet eine Desktop-Sitzung.
- `/usr/dt/config/Xfailsafe` - Startet eine Failsafe-Sitzung.
- `/usr/dt/config/Xconfig` - Konfigurationsdatei des Anmeldeservers.
- `/usr/dt/config/Xservers` - Anzeigebeschreibungsdatei des Anmeldeservers.
- `/usr/dt/config/Xaccess` - Zugriffsbeschreibungsdatei des Anmeldeservers.
- `/usr/dt/config/Sprache /Xresources` - Anzeige-Layoutressourcen.

/usr/dt/config/Xsetup - Setup-Datei für die Anzeige.
/usr/dt/config/Xstartup - Startdatei zur Sitzungsvorbereitung.
/usr/dt/config/Xreset - Rücksetzungsdatei nach der Sitzung.
/var/dt/Xpid - Prozeß-ID des Anmeldeservers.
/var/dt/Xerrors - Fehlerprotokolldatei des Anmeldeservers.

Session Manager konfigurieren

Der Session Manager sorgt für den Start des Desktop und übernimmt das automatische Speichern und Wiederherstellen von aktiven Anwendungen und von Einstellungen für Farben, Schriftarten, Mausverhalten, Lautstärke und Tastaturklicken.

Der Session Manager bietet aus der Sicht des Benutzers die folgenden Möglichkeiten:

- Die Start Sitzung für alle Desktop-Benutzer anpassen.
- Die Umgebung und Ressourcen für alle Desktop-Benutzer anpassen.
- Die Startmeldung der Sitzung ändern.
- Die Parameter für Dämonen und Sitzungsstart—Hilfsprogramme ändern.
- Das Desktop-Farbschema für alle Benutzer ändern.
- „ Was ist eine Sitzung?“ auf Seite 23
- „Sitzungen starten“ auf Seite 25
- „Zu Beginn einer Sitzung“ auf Seite 25
- „Zusätzliche Anpassungen beim Start einer Sitzung “ auf Seite 32
- „Dateien und Verzeichnisse des Session Managers“ auf Seite 37

Was ist eine Sitzung?

Der Begriff Sitzung steht für die Gesamtheit der Anwendungen, Einstellungen und Ressourcen auf dem Desktop eines bestimmten Benutzers. Die Sitzungsverwaltung wiederum umfaßt eine Reihe von Konventionen und Protokollen, auf deren Basis der Session Manager eine Benutzersitzung speichern und wiederherstellen kann. Damit wird erreicht, daß der Benutzer beim Anmelden dieselben aktiven Anwendungen

sowie dieselben Einstellungen und Ressourcen vorfindet wie bei der letzten Abmeldung. Wenn sich der Benutzer zum ersten Mal beim Desktop anmeldet, wird eine standardisierte Start Sitzung geladen. Danach unterscheidet der Session Manager zwischen aktueller und Home-Sitzung.

Die Start Sitzung

Wenn sich ein Benutzer zum ersten Mal beim Desktop anmeldet, generiert der Session Manager unter Verwendung von Systemstandardwerten eine Start Sitzung für den Benutzer. Normalerweise werden der Dateimanager und ein Hilfeband gestartet, der eine Einführung in die Funktionen des Desktop gibt.

Die aktuelle Sitzung

Unter der aktuellen Sitzung versteht man die jeweils aktive Sitzung eines Benutzers. Dabei spielt es keine Rolle, ob es sich um eine zuvor gespeicherte und nun wiederhergestellte Home-Sitzung oder um die Standard-Start Sitzung des Systems handelt. Sobald der Benutzer die Sitzung beendet, speichert der Session Manager die aktuelle Konfiguration nach Maßgabe der Starteinstellungen des Style-Managers. Bei der nächsten Anmeldung des Benutzers wird die zuvor gespeicherte aktuelle Sitzung erneut gestartet, das heißt, der Desktop befindet sich wieder in demselben Zustand wie zum Zeitpunkt der letzten Abmeldung.

Die Home—Sitzung

Der Desktop kann auch bei jeder Anmeldung in demselben Status wiederhergestellt werden, unabhängig vom Status bei der letzten Abmeldung. Dazu speichert der Benutzer die aktuelle Sitzung und veranlaßt den Session Manager, genau diese Sitzung auf Basis der Starteinstellungen des Style-Managers nach jeder Anmeldung zu starten.

Bildschirmspezifische Sitzungen

Sitzungen lassen sich auch bildschirmspezifisch definieren. Dazu kopiert der Benutzer das Verzeichnis *HomeVerzeichnis/.dt/sessions* in das Verzeichnis *HomeVerzeichnis/.dt/anzeige*, wobei *anzeige* für den realen, nicht qualifizierten Hostnamen handelt (z.B. ist `pablo:0` gültig, `pablo.gato.com:0` oder `unix:0` dagegen nicht). Meldet sich der Benutzer beim Bildschirm `pablo:0` an, startet der Session Manager diese bildschirmspezifische Sitzung.

Sitzungen starten

Der Session Manager wird mit `/usr/dt/bin/Xsession` gestartet. Wenn die Anmeldung über den Login Manager erfolgt, wird `Xsession` standardmäßig gestartet.

Alternativ dazu kann sich der Benutzer auch im herkömmlichen, zeichenorientierten Modus anmelden (`getty`) und den Session Manager manuell über ein Hilfsprogramm wie `xinit` aufrufen, das wiederum einen X-Server startet. Ein Beispiel: `xinit /usr/dt/bin/Xsession .`

Zu Beginn einer Sitzung

Nach dem Aufruf des Session Managers wird eine Sitzung in folgenden Schritten gestartet:

1. **Das Skript *HomeVerzeichnis/.dtprofile* wird ausgeführt.**
2. **Die Skripts *Xsession.d* werden ausgeführt.**
3. **Eine Begrüßungsmeldung wird angezeigt.**
4. **Die Desktop-Suchpfade werden eingerichtet.**
5. **Verfügbare Anwendungen werden erfaßt.**
6. **Optional werden die Skripts *HomeVerzeichnis/.profile* oder *HomeVerzeichnis/.login* ausgeführt.**
7. **Der nachrichtendämon ToolTalk[®] wird gestartet.**
8. **Sitzungsressourcen werden geladen.**
9. **Der Farbserver wird gestartet.**
10. **Der Workspace Manager wird gestartet.**
11. **Die Sitzungsanwendungen werden gestartet.**

Die folgenden Abschnitte beschreiben diese einzelnen Schritte.

Das Skript `.dtprofile` ausführen

Beim Start der Anwendung führt das Skript `Xsession` das Skript `HomeVerzeichnis/.dtprofile` des Benutzers aus. Dabei handelt es sich um ein `/bin/sh-` oder `/bin/ksh-` Skript, das Benutzern die Einstellung von Umgebungsvariablen für ihre Sitzungen gestattet. Details hierzu finden Sie unter „Zusätzliche Anpassungen beim Start einer Sitzung“ auf Seite 32.

Sollte das Skript `HomeVerzeichnis/.dtprofile` nicht vorhanden sein, beispielsweise weil sich der Benutzer zum ersten Mal beim Desktop anmeldet, kopiert `Xsession` die Standard-Desktopdatei `sys.dtprofile` auf `HomeVerzeichnis/.dtprofile`.

Die Standards für den Desktop sind definiert in `/usr/dt/config/sys.dtprofile`. Wenn das Skript `sys.dtprofile` angepaßt werden soll, kopieren Sie `sys.dtprofile` aus `/usr/dt/config` nach `/etc/dt/config`, und bearbeiten Sie die neue Datei.

Das Skript `Xsession.d` ausführen

Nach Ausführung des Skripts `HomeVerzeichnis/.dtprofile` führt das Skript `Xsession` die Skripts in `Xsession.d` aus. Diese übernehmen die Einrichtung zusätzlicher Umgebungsvariablen und starten optionale Dämonen für die Benutzersitzung. Standardmäßig enthält `Xsession.d` die folgenden Skripts:

- 0010.`dtpaths`—Dokumentiert die einstellbaren Desktop-Suchpfade.
- 0020.`dtims`—Startet einen optionalen Eingabemethoden-Server.
- 0030.`dttmpdir`—Legt ein benutzerspezifisches temporäres Verzeichnis für jede Sitzung an.
- 0040.`xmbind`—Richtet `$XMBINDDIR` entsprechend dem Desktopstandard ein.

In `Xsession.d` befinden sich eventuell weitere, herstellerepezifische Skripts.

`Xsession` führt zuerst alle Dateien im Verzeichnis `/etc/dt/config/Xsession.d` aus und dann die Dateien in `/usr/dt/config/Xsession.d`.

Die Standardskripts der Gruppe `Xsession.d` für den Desktop befinden sich im Verzeichnis `/usr/dt/config/Xsession.d`. Wenn eines dieser Skripts angepaßt werden soll, kopieren Sie es von `/usr/dt/config/Xsession.d` nach `/etc/dt/config/Xsession.d` und bearbeiten die neue Datei. Dazu müssen Sie allerdings das Ausführungsrecht besitzen.

Wenn Sie möchten, daß `Xsession` eines Ihrer eigenen Skripts ebenfalls automatisch ausführen soll, müssen Sie es nach `/etc/dt/config/Xsession.d` kopieren.

Hinweis - Immer wenn Sie eines der Skripts in `Xsession.d` verändern oder ein neues erstellen, sollten Sie auf eine geringe Ausführungsdauer der Vordergrundbefehle achten, um die für den Sitzungsstart benötigte Zeit kurz zu halten. Wenn ein Vordergrundbefehl nicht existiert, wird der Sitzungsstart angehalten. Alle Befehle in einem `Xsession.d`—Skript, die für die gesamte Dauer der Sitzung aktiv bleiben sollen, sollten im Hintergrund ablaufen.

Begrüßungsmeldung anzeigen

Nach Ausführung der Skripts *HomeVerzeichnis*/`.dtprofile` und `Xsession.d` zeigt `Xsession` eine Begrüßungsmeldung über den gesamten Bildschirm an. Die Meldung kann in ihrem Text verändert oder ganz deaktiviert werden. Die Anzeige steuert der Client `dthello`.

Wenn Sie den Meldungstext wechseln möchten, ändern Sie die `dthello`—Optionen über die Variable `dtstart_hello[0]` .

Zum Ändern von `dtstart_hello[0]` erstellen Sie ein Skript `/etc/dt/config/Xsession.d`, das den neuen Wert setzt. Wenn die Meldung des Tages für alle Benutzer angezeigt werden soll, erzeugen Sie ein ausführbares `sh`- oder `ksh`-Skript, beispielsweise `/etc/dt/config/Xsession.d/myvars` , und setzen `dtstart_hello[0]` folgendermaßen:

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

Auf ähnliche Weise können Benutzer die Begrüßungsmeldung für ihre Sitzungen ändern. Dazu muß `dtstart_hello[0]` in *HomeVerzeichnis*/`.dtprofile` gesetzt werden.

Zum Deaktivieren der Begrüßungsmeldung setzen Sie `dtstart_hello[0]=" "`.

Weitere Informationen zu `dthello` finden Sie in der Handbuchseite `dthello`.

Desktop-Suchpfade einrichten

Die Desktop-Suchpfade werden bei der Anmeldung von `dtsearchpath` erzeugt. Von `dtsearchpath` werden zwei Kategorien von Umgebungsvariablen verwendet:

Eingabevariablen—Systemweite und persönliche Umgebungsvariablen, deren Werte vom Systemverwalter oder vom Endanwender gesetzt werden.

Ausgabevariablen—Variablen, die von `dtsearchpath` erzeugt und verwaltet werden. Der Wert jeder Variablen entspricht dem Suchpfad für die Desktopsitzung.

Die Befehlszeilenooptionen für `dtsearchpath` werden über die Variable `dtstart_searchpath` geändert. Wenn dies für alle Benutzer

`dtstart_searchpath` erfolgen soll, erstellen Sie ein ausführbares `sh-` oder `ksh-` Skript, beispielsweise `/etc/dt/config/Xsession.d/myvars`, und setzen `dtstart_searchpath` folgendermaßen:

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

Auf ähnliche Weise können Benutzer die `dtsearchpath` —Optionen nur für ihre eigenen Sitzungen ändern. Dazu muß `dtstart_searchpath` in *HomeVerzeichnis* `/.dtprofile` gesetzt werden.

Weitere Informationen zu `dtsearchpath` finden Sie in Kapitel 9. Details zu den Optionen von `dtsearchpath` enthält die Handbuchseite `dtsearchpath`.

Verfügbare Anwendungen erfassen

Der nächste Schritt nach Einrichtung der Desktop-Suchpfade besteht in der Erfassung der verfügbaren Anwendungen mit Hilfe von `dtappgather`. Die Befehlszeilenooptionen von `dtappgather` werden über die Variable `dtstart_appgather` geändert. Wenn dies für alle Benutzer erfolgen soll, erzeugen Sie ein ausführbares `sh-` oder `ksh-` Skript, beispielsweise `/etc/dt/config/Xsession.d/myvars`, und setzen `dtstart_appgather` folgendermaßen:

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

Auf ähnliche Weise können Benutzer die Optionen von `dtappgather` nur für ihre eigenen Sitzungen ändern. Dazu wird `dtstart_appgather` in *HomeVerzeichnis* `/.dtprofile` gesetzt.

Näheres zu den Optionen von `dtappgather` finden Sie in der Handbuchseite `dtappgather(4)`.

Optionale Ausführung der Skripts `.profile` und `.login`

`Xsession` ist in der Lage, die konventionellen Skripts *HomeVerzeichnis*/`.profile` oder *HomeVerzeichnis*/`.login` eines Benutzers auszuführen. Per Voreinstellung ist dieses Merkmal jedoch deaktiviert. Soll `Xsession` die Skripts `.profile` oder `.login` ausführen, setzen Sie die Variable `DTSOURCEPROFILE` auf `true`.

Wenn die Einstellung von `DTSOURCEPROFILE` für alle Benutzer geändert werden soll, erzeugen Sie ein Skript `/etc/dt/config/Xsession.d`, das den neuen Wert zuweist. Damit `DTSOURCEPROFILE` für alle Benutzer auf `true` gesetzt wird, erzeugen Sie ein ausführbares `sh-` oder `ksh-` Skript, beispielsweise

`/etc/dt/config/Xsession.d/myvars`, und setzen `DTSOURCEPROFILE` folgendermaßen:

```
DTSOURCEPROFILE=true
```

Auf ähnliche Weise können Benutzer den Wert von `DTSOURCEPROFILE` für ihre eigene Sitzung ändern. Dazu wird `DTSOURCEPROFILE` in *HomeVerzeichnis*/`.dtprofile` auf `true` gesetzt.

Den ToolTalk—Nachrichtendämon starten

Der ToolTalk—Nachrichtendämon, `ttsession`, ermöglicht die Kommunikation von Anwendungen untereinander, ohne daß diese in direkter Verbindung stehen. Hierzu erzeugen und senden die Anwendungen ToolTalk—Nachrichten. `ttsession` übernimmt den Datenaustausch im Netz und leitet die Nachrichten weiter.

Die Befehlszeilenoptionen für `ttsession` werden über die Variable `dtstart_ttsession` geändert. Wenn dies für alle Benutzer geschehen soll, erstellen Sie ein ausführbares `sh`- oder `ksh`- Skript, beispielsweise `/etc/dt/config/Xsession.d/myvars`, und setzen `dtstart_ttsession` folgendermaßen:

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

Auf ähnliche Weise können Benutzer die Optionen von `ttsession` für ihre eigene Sitzung ändern. Dazu muß `dtstart_ttsession` in *HomeVerzeichnis*/`.dtprofile` gesetzt werden.

Näheres zu den Optionen von `ttsession` enthält die Handbuchseite `ttsession`. Details zum Modul `ttsession` finden Sie unter *Common Desktop Environment: ToolTalk Messaging Overview*.

Client des Session Managers starten

Zu diesem Zeitpunkt startet `Xsession` das Modul `/usr/dt/bin/dtsession`, das den Startvorgang der Sitzung fortführt.

Sitzungsressourcen laden

Der Session Manager macht die Desktop-Ressourcen mit Hilfe der X—Servereigenschaft `RESOURCE_MANAGER` allen Anwendungen verfügbar. Das Laden von `RESOURCE_MANAGER` umfaßt folgende Schritte:

- Die Standardressourcen des Systems werden geladen.

- Alle vom Systemverwalter festgelegten systemweiten Ressourcen werden hinzugefügt.
- Alle vom Benutzer festgelegten Ressourcen werden hinzugefügt.

Die Desktop-Standardressourcen sind in

`/usr/dt/config/sprache/sys.resources` definiert. Über die Eigenschaft `RESOURCE_MANAGER` werden diese Ressourcen allen Benutzersitzungen verfügbar gemacht. Es hat keinen Sinn, die Datei zu verändern, weil sie bei nachfolgenden Desktop-Installationen überschrieben wird.

Die Anzahl der Standard-Systemressourcen läßt sich erhöhen, indem eine Datei `/etc/dt/config/sprache/sys.resources` angelegt wird. In dieser Datei können für alle Desktop-Benutzer Standardressourcen undefiniert oder weitere Ressourcen hinzugefügt werden. In dieser Datei sollten nur neue oder aktualisierte Ressourcenspezifikationen plaziert werden, weil die Datei beim Start der Sitzung mit den Standard-Desktopressourcen kombiniert wird. Über die Eigenschaft `RESOURCE_MANAGER` werden sämtliche in dieser Datei angegebenen Ressourcen allen Benutzersitzungen verfügbar gemacht. Die Definitionen in dieser Datei haben Vorrang vor denen der Standard-Ressourcendatei für den Desktop.

Benutzer können die Anzahl der Standardressourcen für den Desktop und der systemweit definierten Ressourcen über die Datei `HomeVerzeichnis/.Xdefaults` erhöhen. Über die Eigenschaft `RESOURCE_MANAGER` werden die hier definierten Ressourcen der Sitzung dieses Benutzers verfügbar gemacht. Die Definitionen in dieser Datei haben Vorrang vor denen der Standard-Ressourcendatei des Desktop oder der Ressourcendatei des Systemverwalters.

Hinweis - Das Dienstprogramm X Toolkit Intrinsic lädt die Ressourcen für eine Anwendung entweder aus `RESOURCE_MANAGER` oder aus `HomeVerzeichnis/.Xdefaults`, aber nicht aus beiden Quellen. Das bedeutet normalerweise, daß die Datei `HomeVerzeichnis/.Xdefaults` des Benutzers ignoriert wird. Wie bereits erwähnt, kombiniert der Session Manager die Datei `HomeVerzeichnis/.Xdefaults` jedoch beim Start der Sitzung mit `RESOURCE_MANAGER`. Änderungen des Benutzers an `HomeVerzeichnis/.Xdefaults` wirken sich erst dann auf neue Anwendungen aus, wenn der Benutzer die Aktion 'Ressourcen neu laden' ausführt. Dadurch wird der Session Manager veranlaßt, `RESOURCE_MANAGER` mit den Standardressourcen, den systemweit definierten Ressourcen und vom Benutzer festgelegten Ressourcen zu laden. Erst dadurch werden Änderungen an systemweiten oder persönlichen Ressourcendateien für Anwendungen verfügbar.

Weitere Informationen finden Sie an folgenden Stellen:

-
- Handbuchseite `dtresourcesfile(4)`

Farbserver starten

Der Session Manager fungiert für den Desktop als Farbserver. Zur Konfiguration dienen die folgenden `dtsession`-Ressourcen.

`foregroundColor`—Steuert, ob einem Pixel die Vordergrundfarbe zugeordnet ist.

`dynamicColor`—Gibt an, ob schreibgeschützte Farben zugeordnet sind.

`shadowPixmap`—Gibt an, ob Farben für den oberen oder unteren Schatten zugeordnet sind.

`colorUse`—Begrenzt die Farbzuordnung.

`writeXrdbColors`—Gibt an, ob die `*background` — und `*foreground`—Ressourcen in die Ressourcendatenbank eingetragen werden.

Wenn die Farbserver-Ressourcen für alle Benutzer geändert werden sollen, legen Sie eine Datei `/etc/dt/config/sprache/sys.resources` an und tragen dort die Ressourcen ein.

Auf ähnliche Weise können Benutzer Farbserver-Ressourcen für ihre eigenen Sitzungen festlegen. Dies geschieht in der Datei `HomeVerzeichnis/.Xdefaults`.

Weitere Informationen zur Einstellung der Farbserver-Ressourcen finden Sie unter „Farben verwalten“ auf Seite 327.

Workspace Manager starten

Der Session Manager ist verantwortlich für den Start des Workspace Managers. Per Voreinstellung wird `/usr/dt/bin/dtwm` aufgerufen. Mit der Ressource `wmStartupCommand` kann ein alternativer Fenstermanager festgelegt werden.

Wenn ein alternativer Fenstermanager für alle Benutzer festgelegt werden soll, erzeugen Sie eine Datei `/etc/dt/config/sprache/sys.resources` und geben dort mit der Ressource `Dtsession*wmStartupCommand` den vollständigen Pfadnamen und die Optionen für diesen Fenstermanager an.

Auf ähnliche Weise können Benutzer einen alternativen Fenstermanager für ihre eigenen Sitzungen auswählen. Dazu muß in `HomeVerzeichnis/.Xdefaults` die Ressource `Dtsession*wmStartupCommand` definiert werden.

Weitere Informationen hierzu finden Sie in Kapitel 16.

Anwendungen einer Sitzung starten

Zu Beginn einer Sitzung startet der Session Manager wieder alle Anwendungen, die als Teil der Sitzung gespeichert wurden. Die Anwendungen, die standardmäßig bei einer Start Sitzung geladen werden, sind definiert in

`/usr/dt/config/ sprache/sys.session`. Diese Datei sollte nicht bearbeitet werden, weil sie bei nachfolgenden Desktop-Installationen überschrieben wird.

Weitere Informationen finden Sie auf der Handbuchseite `dtsessionfile(4)`.

Wenn ein Systemverwalter andere Anwendungen für eine Start Sitzung vorsehen möchte, kann er die Datei `/usr/dt/config/sprache/sys.session` auf die Datei `/etc/dt/config/sprache/sys.session` kopieren und letztere Datei bearbeiten. Im Gegensatz zu den Ressourcendateien ersetzt diese Datei die Desktop-Standarddatei vollständig, so daß an einer Kopie der System-Standarddatei die nötigen Änderungen vorgenommen werden können.

Zusätzliche Anpassungen beim Start einer Sitzung

Dieser Abschnitt umfaßt die folgenden Themen:

- Umgebungsvariablen einstellen
- Ressourcen einstellen
- Bildschirmspezifische Sitzungen
- Skripts bei der Anmeldung ausführen
- Gespeicherte Sitzungen wiederherstellen

▼ Umgebungsvariablen einstellen

- ◆ **Zur Einstellung von systemweiten (globalen) Umgebungsvariablen legen Sie im Verzeichnis `/etc/dt/config/Xsession.d` eine Datei an, in der die Variable gesetzt und exportiert wird.**

Wenn Sie beispielsweise ein ausführbares `ksh`—Skript `/etc/dt/config/Xsession.d/myvars` mit folgendem Inhalt erzeugen:

```
export MYVARIABLE="wert"
```

dann wird die Variable `MYVARIABLE` bei der nächsten Anmeldung in den jeweiligen Benutzerumgebungen gesetzt.

- ◆ **Die Einstellung persönlicher Umgebungsvariablen erfolgt in *HomeVerzeichnis*/`.dtprofile`.**

Ein Beispiel:

```
export MYVARIABLE="wert"
```


Die Variable `MYVARIABLE` wird bei der nächsten Anmeldung in den jeweiligen Benutzerumgebungen gesetzt.

Hinweis - Der Session Manager liest die Dateien `.profile` und `.login` nicht automatisch. Allerdings kann er so konfiguriert werden, daß er auf diese Dateien zugreift. Siehe dazu .

▼ Ressourcen einstellen

- ◆ Zur Einstellung systemweiter Ressourcen müssen diese in die Datei `/etc/dt/config/sprache/sys.resources` eingefügt werden. (Eventuell müssen Sie diese Datei noch anlegen.)

Hinweis - `.dtprofile` unterstützt nur die Syntax `/bin/sh` oder `/bin/ksh`.

Wenn Sie beispielsweise in die Datei `/etc/dt/config/C/sys.resources` folgendes einfügen:

```
AnApplication*resource: wert
```

dann wird bei der nächsten Anmeldung die Ressource `AnApplication*resource` in der Eigenschaft `RESOURCE_MANAGER` jedes Benutzers gesetzt.

- ◆ Persönliche Ressourcen müssen in die Datei **HomeVerzeichnis** `/.Xdefaults` eingetragen werden.

▼ Bildschirmspezifische Ressourcen einstellen

Bildschirmspezifische Ressourcen können entweder für alle Desktop-Benutzer eines Systems oder für einzelne Sitzungen eingestellt werden. Somit können die verwendeten Ressourcen vom Bildschirm abhängig gemacht werden, über den sich der Benutzer beim Desktop anmeldet.

- ◆ Zur Einstellung bildschirmspezifischer Ressourcen für alle Desktop-Benutzer des Systems erstellen Sie eine Datei `/etc/dt/config/sprache/sys.resources`, welche die Festlegungen enthält.

- ◆ Persönliche bildschirmspezifische Ressourcen werden in der Datei **HomeVerzeichnis** `/.Xdefaults` angegeben.

Ressourcen lassen sich situationsbezogen zuordnen, indem man Sie in bedingte `cpp`-Anweisungen einschließt. Ein Makro `DISPLAY_Bildschirmname` wird abhängig vom Wert der Variablen `$DISPLAY` definiert. Dazu werden alle

Zeichen . (Punkt) und : (Doppelpunkt) durch das Zeichen _ (Unterstrich) ersetzt und alle Bildschirmspezifikationen entfernt. Anschließend erhält das Ergebnis das Präfix DISPLAY_ .

Beispielsweise liefert \$DISPLAY mit dem Wert :0 das Ergebnis DISPLAY_0, und \$DISPLAY mit dem Wert blanco.gato.com:0.0 ergibt DISPLAY_blanco_gato_com_0 . Das Ergebnis kann in einer Sitzungsressourcendatei als Teil einer Fallunterscheidung mit cpp eingesetzt werden. Betrachten Sie als Beispiel folgendes Konstrukt in /etc/dt/config/C/sys.resources :

```
Myapp*resource: wert

#ifdef DISPLAY_blanco_gato_com_0
    Myapp*resource: sepzialwert1
#endif

#ifdef DISPLAY_pablo_gato_com_0
    Myapp*resource: spezialwert2
#endif
```

Damit wird die Ressource MyApp*resource in RESOURCE_MANAGER auf spezialwert1 gesetzt, wenn sich der Benutzer bei Bildschirm blanco.gato.com:0 anmeldet, und entsprechend auf spezialwert2, wenn die Anmeldung über Bildschirm pablo.gato.com:0 erfolgt. Die Anmeldung von einem anderen Bildschirm aus ergibt die Einstellung wert.

▼ Anwendungen für die Startsituation ändern

Sie können festlegen, welche Anwendungen als Teil der Startsituation eines Benutzers aufgerufen werden.

1. **Kopieren Sie die Datei** /usr/dt/config/*sprache*/sys.session **auf** /etc/dt/config/ *sprache*/sys.session.
2. **Bearbeiten Sie die neue Datei** sys.session.
Die Einträge in sys.session haben folgende Form:

```
dtsmcmd -cmd befehl_und_optionen
```

Wenn eine weitere Anwendung als Teil der Startsituation geladen werden soll, fügen Sie in sys.session einen weiteren Eintrag mit vollständigem Pfad ein.

Wenn beispielsweise `/usr/bin/X11/xclock` ausgeführt werden soll, fügen Sie einen Eintrag `xclock` in die Datei `/etc/dt/config/C/sys.session` ein:

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

▼ Bildschirmspezifische Sitzungen einrichten

Benutzer haben die Möglichkeit, die Konfiguration einer Sitzung an einen bestimmten Bildschirm anzupassen.

- ◆ **Kopieren Sie das Verzeichnis *HomeVerzeichnis*/.dt/sessions in *HomeVerzeichnis*/.dt/*anzeige*, wobei *anzeige* der reale, nicht qualifizierte Hostname ist (z.B. ist `pablo:0` gültig, `pablo.gato.com:0` oder `unix:0` dagegen nicht).**

Als Beispiel soll eine bildschirmspezifische Sitzung für `pablo.gato.com:0` definiert werden:

```
cp -r HomeVerzeichnis/.dt/sessions  
Homeverzeichnis/.dt/pablo:0
```

Wenn sich der Benutzer das nächste Mal beim Bildschirm `pablo.gato.com:0` anmeldet, startet der Session Manager diese bildschirmspezifische Sitzung.

Zusätzliche Befehle beim Start der Sitzung und bei der Abmeldung ausführen

Als Benutzer können Sie veranlassen, daß bei der Anmeldung zusätzliche Befehle ausgeführt werden. Besonders hilfreich ist dies für X—Einstellungen, die nicht vom Session Manager gespeichert werden. Beispielsweise läßt sich mit `xsetroot` der Zeiger auf Root (Arbeitsbereich) anpassen. Eine weitere Möglichkeit ist der Start von Anwendungen, die vom Session Manager nicht gespeichert und wiederhergestellt werden können.

▼ So führen Sie beim Start der Sitzung zusätzliche Befehle aus

- ◆ **Legen Sie eine Datei *HomeVerzeichnis*/.dt/sessions/sessionetc an, welche die Befehle enthält.**

Grundsätzlich ist diese Datei ein Skript, für das das Ausführungsrecht vorliegen muß. Alle in `sessionetc` gestarteten Prozesse sollten im Hintergrund ablaufen.

Hinweis - Starten Sie mit `sessionetc` keine Clients, die vom Session Manager automatisch wiederhergestellt werden, damit nicht mehrere Instanzen der Anwendung laufen. Unter Umständen erkennen Sie dies nicht sofort, wenn die Fenster übereinander liegen.

▼ So führen Sie beim Abmelden zusätzliche Befehle aus

Das Gegenstück zur Datei `sessionetc` ist `sessionexit`. Mit `sessionexit` können beim Beenden der Sitzung bestimmte Operationen ausgeführt werden, die der Session Manager nicht automatisch übernimmt.

- ◆ **Legen Sie eine Datei *HomeVerzeichnis*/.dt/sessions/sessionexit an, welche die Befehle enthält.**

Ebenso wie `sessionetc` ist auch diese Datei ein Skript mit Ausführungsberechtigung.

▼ Gespeicherte Sitzungen wiederherstellen

Jedesmal, wenn der Session Manager eine Sitzung speichert, werden die relevanten Informationen im Verzeichnis *HomeVerzeichnis*/.dt/sessions oder bei bildschirmpezifischen Sitzungen im Verzeichnis *HomeVerzeichnis*/.dt/ abgelegt. In diesen Verzeichnis erzeugt der Session Manager ein Unterverzeichnis `current` oder `home`, wo jeweils die Konfiguration der aktuellen bzw. der Home-Sitzung gespeichert wird. Vorher wird allerdings ein Backup der vorherigen Sitzung gleichen Namens angelegt und unter `current.old` oder `home.old` gespeichert.

- 1. Wählen Sie im Anmeldebildschirm 'Failsafe-Sitzung' oder 'Befehlszeilenanmeldung'.**
- 2. Kopieren Sie das Backup-Sitzungsverzeichnis in das Verzeichnis der aktiven Sitzung. Das Backup einer Home-Sitzung stellen Sie beispielsweise folgendermaßen wieder her:**

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

Bei bildschirmspezifischen Sitzungen gehen Sie genauso vor.

▼ Probleme beim Start einer Sitzung

- ◆ Untersuchen Sie den Inhalt der Datei *HomeVerzeichnis*/.dt/startlog.

In dieser Datei protokolliert der Session Manager den Ablauf eines Sitzungsstarts für jeden Benutzer.

Dateien und Verzeichnisse des Session Managers

- /usr/dt/bin/Xsession
- /usr/dt/config/Xsession.d/*
- /usr/dt/bin/dtsession
- /usr/dt/bin/dtsession_res
- *HomeVerzeichnis*/.dt/sessions/current
- *HomeVerzeichnis*/.dt/sessions/home
- *HomeVerzeichnis*/.dt/anzeige/current
- *HomeVerzeichnis*/.dt/anzeige/home

Fehlerbeseitigung bei Anmelde- und Sitzungsstartproblemen

Dieses Kapitel beschreibt die Solaris CDE-Startdateien, mögliche Probleme beim Start von Solaris CDE und enthält Informationen zur Beseitigung dieser Probleme.

- „Anmeldedateien“ auf Seite 39
- „Fehlerprotokollverzeichnisse“ auf Seite 40
- „Benutzer-Startdateien“ auf Seite 41
- „Solaris CDE-Startbeispiele“ auf Seite 41

Anmeldedateien

Sobald Solaris CDE Login Manager den Benutzer überprüft hat, wird das folgende Skript aufgerufen, um den Desktop zu starten:

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

Die erste von `Xsession` aufgerufene benutzerspezifische Datei ist *Homeverzeichnis/.dtprofile*.

Wenn sich ein neuer Benutzer erstmals im Solaris CDE anmeldet, wird `.dtprofile`-Datei in das Home-Verzeichnis des Benutzers kopiert. Standardmäßig führt diese Datei keine Operationen aus. Sie enthält jedoch Informationen zur Bearbeitung. Der Benutzer kann die Datei bearbeiten, um benutzerspezifische Umgebungsvariablen einzufügen.

Hinweis - Wie das Skript `Xsession`, das die Datei `.dtprofile` aufruft, verwendet auch diese Datei die `ksh` -Syntax.

Eine mögliche Bearbeitung der Datei `.dtprofile` besteht darin, die Ausführung der letzten Zeile zu ermöglichen:

```
DTSOURCEPROFILE=true
```

Dieser Zeile ermöglicht den Zugriff auf *Homeverzeichnis* / `.login` (für `csch`-Benutzer) oder *Homeverzeichnis* / `.profile` (für Benutzer anderer Shells) im Rahmen des Startprozesses.

Fehlerprotokollverzeichnisse

Wählen Sie “Failsafe-Sitzung” im Menü “Optionen” des Anmeldebildschirms von Solaris CDE, um den normalen Xsession-Start zu umgehen und die Fehlerprotokolle anzuzeigen sowie mögliche Probleme in Benutzerdateien zu beheben. Tabelle 3-1 zeigt die Fehlerprotokolle und deren Positionen.

TABELLE 3-1 Verzeichnisse der Fehlerprotokolle

Verzeichnis	Fehlerprotokoll
<code>/var/dt/Xerrors</code>	Solaris CDE-Systemfehler vor der Benutzeranmeldung
<i>Homeverzeichnis</i> / <code>.dt/startlog</code>	Solaris CDE-Startfehler während der Ausführung von Xsession, <code>.dtprofile</code> , <code>.login</code> oder <code>.profile</code>
<i>Homeverzeichnis</i> / <code>.dt/errorlog</code>	Solaris CDE-Fehler nach dem Xsession-Start
<i>Homeverzeichnis</i> / <code>.dt/sessionlogs</code>	Verzeichnis der Sitzungsprotokolle für Session Manager- und Fenstermanager-Fehler

Einige Fehler werden auch im Fenster der System-Konsole angezeigt. Wenn die System-Konsole nicht läuft, wird statt dessen die Protokolldatei `wscn` der Fallback-Konsole mit angehängtem Uhrzeit-/Anzeige-code verwendet:

```
/usr/tmp/wscnAAAa004EE:0.0
```

Benutzer-Startdateien

Damit auf die Datei `.login` (für `csh`-Benutzer) oder `.profile` (für `sh`- oder `ksh`-Benutzer) zugegriffen werden kann, muß die Ausführung der letzten Zeile in der Datei `.dtprofile` folgendermaßen ermöglicht werden:

```
DTSOURCEPROFILE=true
```

In den meisten Fällen sollte dies ausreichen. Da jedoch bestimmte Befehle nicht mit dem Solaris CDE Login Manager verwendet werden können, muß gegebenenfalls die Datei `.login` bzw. `.profile` bearbeitet werden, um alle entsprechenden Befehle zu entfernen. Wenn mit einem der Befehle ein Problem auftritt, liegt dies meistens an einer Datei, die eine Terminal-Tastatureingabe erwartet: über einen Befehl wie `stty` oder `tset` bzw. einen Befehl, der eine Eingabe anfordert.



Warnung -



Wenn `.dtprofile` auf eine der Dateien `.login` oder `.profile` zugreift, die ihrerseits einen Befehl enthält, der zum Absturz der Shell führt, kann das Desktop nicht gestartet werden. Es wird also kein Desktop geöffnet. Statt dessen wird wieder der Anmeldebildschirm von Solaris CDE angezeigt. Startfehler von `.login` oder `.profile` werden normalerweise in *HomeVerzeichnis* `/.dt/startlog` protokolliert. Korrigieren Sie die entsprechenden Befehle in der Datei `.login` oder `.profile` mit einer Failsafe-Anmeldesitzung oder einer Befehlszeilenanmeldung.

Weitere Informationen zu Problemen und möglichen Lösungen finden Sie in der Datei `.dtprofile`. Normalerweise beziehen sich die problematischen Befehle auf Terminal-Informationen und die Terminal-Steuerung.

Solaris CDE-Startbeispiele

Dieser Abschnitt enthält Beispiele zur möglichen Bearbeitung der Benutzer-Startdateien:

- `.login` (für `csh`-Benutzer)
- `.profile` (für `sh`- oder `ksh`-Benutzer)
- `.xdefaults`

Der Solaris CDE-Startprozeß definiert die Shell-Variablen DT im Skript `.login` oder `.profile`. Die Prüfung kann also im Solaris CDE-Startprozeß durchgeführt werden. Die Ausführung terminal-bezogener Befehle wie `tty` und `stty` kann somit verhindert werden. Beachten Sie die folgenden Skriptbeispiele:

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

    stty erase '^h'

endif

.profile (sh oderksh )
if [ ! ``$DT`` ]; then

    stty erase '^h'

fi
```

Die IF-Abfrage zu DT unterdrückt die Warnungen der C-Shell, wenn DT nicht definiert ist. DT ist nicht definiert, wenn sich der Benutzer an einer konventionellen textbasierten Konsoleneingabeaufforderung anmeldet. Weitere Informationen zur Einrichtung der Punkt-Dateien finden Sie in der Datei *HomeVerzeichnis/.dtprofile*.

Hinweis - Die `tty`-Einstellungen für Solaris CDE werden in `.Xdefaults` vorgenommen.

Auf die Datei `.Xdefaults` des Benutzers wird während des Solaris CDE-Starts zugegriffen, um die benutzerspezifischen Ressourcen zu ermitteln. `ttyModes` stellt beispielsweise sicher, daß die bevorzugten `tty`-Einstellungen des Benutzers in Terminalemulationsfenstern wie `dterm` und `xterm` verwendet werden. Die folgende Zeile enthält eine typische `ttyModes`-Einstellung in der Datei `.Xdefaults`:

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

Hinweis - Die bevorzugten Solaris CDE-Ressourcen unterscheiden sich von den Standardeinstellungen.

Die Position der Symbole wird durch den Workspace Manager des Desktop aufgrund von Standardwerten festgelegt. Die folgende Zeile in der Datei `.Xdefaults` zeigt die Standardpositionierung der Symbole:

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

Anwendungen hinzufügen und verwalten

Beim Anwendungsmanager handelt es sich um den Desktop-Container für Anwendungen, die dem Benutzer zur Verfügung stehen.

- „Struktur des Anwendungsmanagers ” auf Seite 43
- „Anwendungen zum Anwendungsmanager hinzufügen” auf Seite 47
- „Allgemeine Gruppen von Anwendungen erstellen und verwalten” auf Seite 51
- „Suchpfad für Anwendungen ändern ” auf Seite 52
- „Allgemeine Anwendungsmanagerverwaltung ” auf Seite 54
- „Texteditor und den Terminal-Emulator ändern” auf Seite 55

Struktur des Anwendungsmanagers

Die oberste Ebene des Anwendungsmanagers enthält im allgemeinen Verzeichnisse. Jedes dieser Verzeichnisse und dessen Inhalt wird als *Gruppe von Anwendungen* bezeichnet.



Abbildung 4-1 Gruppen von Anwendungen im Anwendungsmanager

Die Gruppen von Anwendungen und deren Inhalt werden aus vielen verschiedenen Positionen, sowohl lokal als auch im Netz, zusammengestellt.

Verzeichnisposition des Anwendungsmanagers

Im Dateisystem handelt es sich beim Anwendungsmanager um das Verzeichnis `/var/dt/appconfig/appmanager/anmeldung-hostname-anzeige`. Dieses Verzeichnis wird bei der Anmeldung eines Benutzers dynamisch erstellt.

Meldet sich beispielsweise der Benutzer `ronv` am Bildschirm `wxyz:0` an, wird für den Anwendungsmanager das Verzeichnis `/var/dt/appconfig/appmanager/ronv-wxyz-0` erstellt.

Wie der Anwendungsmanager Anwendungen sucht und zusammenstellt

Der Anwendungsmanager wird aufgebaut, indem lokale und ferne Gruppen von Anwendungen zusammengestellt werden. Die Gruppen von Anwendungen werden aus Verzeichnissen zusammengestellt, die auf dem Anwendungssuchpfad liegen.

Der Standardanwendungssuchpfad besteht aus folgenden Positionen:

TABELLE 4-1 Verzeichnisse im Suchpfad für Standardanwendungen

Bereich	Position
Integriert	<code>/usr/dt/appconfig/appmanager/<i>sprache</i></code>
Systemweit	<code>/etc/dt/appconfig/appmanager/ <i>sprache</i></code>
Persönlich	<i>Home-Verzeichnis</i> / <code>.dt/appmanager</code>

Um die oberste Ebene des Anwendungsmanagers zu erstellen, werden zum Zeitpunkt der Anmeldung Verbindungen von den Gruppen von Anwendungen (Verzeichnisse), die sich in Verzeichnissen im Anwendungssuchpfad befinden, zum Anwendungsmanagerverzeichnis

`/var/dt/appconfig/appmanager/anmeldung-hostname-anzeige` hergestellt. Dies erfolgt durch das Desktop-Dienstprogramm `dtappgather`, das vom Anmeldemanager automatisch ausgeführt wird, nachdem sich der Benutzer erfolgreich angemeldet hat.

Beispielsweise stellt das Desktop folgende integrierte Gruppe von Anwendungen zur Verfügung:

```
/usr/dt/appconfig/appmanager/  
sprache/Desktop_Werkzeuge
```

Zum Zeitpunkt der Anmeldung wird eine symbolische Verbindung hergestellt zu:

```
/var/dt/appconfig/appmanager/  
  
anmeldung-hostname-anzeige/Desktop_Werkzeuge
```

Der Anwendungssuchpfad kann ferne Verzeichnisse einschließen. Dadurch können Gruppen von Anwendungen aus Systemen im gesamten Netz zusammengestellt werden. Weitere Informationen könnendem Abschnitt „Anwendungs-Server dem Anwendungssuchpfad hinzufügen“ auf Seite 53. entnommen werden.

Vorrangsregeln beim Zusammenstellen von Anwendungen

Sind Anwendungen im Suchpfad mehrfach vorhanden, haben für den einzelnen Benutzer definierte Gruppen von Anwendungen Vorrang vor für das System definierten Gruppen, und für das System definierte Gruppen haben Vorrang vor integrierten Gruppen. Sind, beispielsweise, sowohl

`/usr/dt/appconfig/appmanager/C/Desktop_Werkzeuge` als auch `/etc/dt/appconfig/appmanager/C/Desktop_Werkzeuge` vorhanden , wird die Gruppe von Anwendungen unter `/etc` verwendet.

Gruppen von Anwendungen, die mit dem Standard-Desktop zur Verfügung gestellt werden

Das nicht angepaßte Desktop stellt vier Gruppen von Anwendungen zur Verfügung:

- Desktop_Anwendungen
- Desktop_Werkzeuge
- Informationen
- Systemadministrator

Beispiel für die Zusammenstellung von Gruppen von Anwendungen

Abbildung 4-2 zeigt ein Anwendungsmanagerfenster mit verschiedenen Gruppen von Anwendungen. Tabelle 4-2 zeigt die Verzeichnisse, aus denen die Gruppen von Anwendungen zusammengestellt werden.

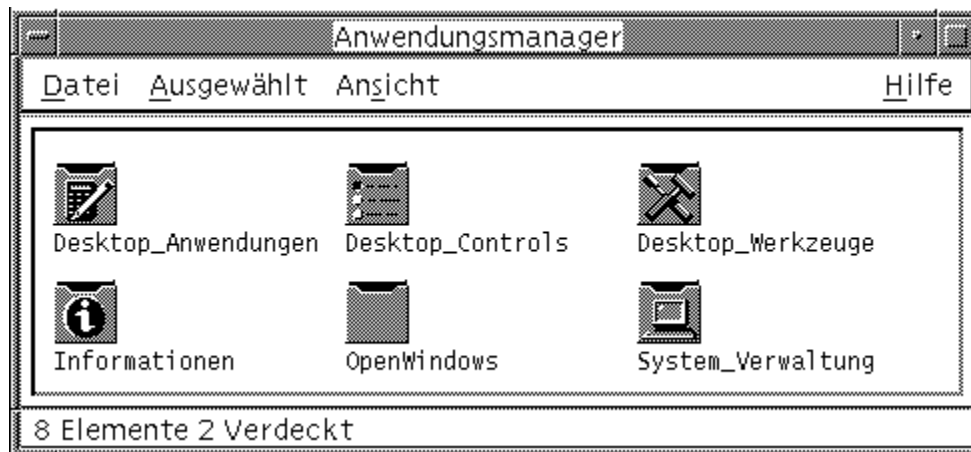


Abbildung 4-2 Ein typisches Anwendungsmanagerfenster

TABELLE 4-2 Quelle der Gruppen von Anwendungen in

TABELLE P-2 Quelle der Gruppen von Anwendungen in Abbildung 4-2 (fortgesetzt)

Abbildung 4-2

Name	Verzeichnis
CAD_Anwendungen	/net/ApServA/etc/dt/appconfig/appmanager/C/ CAD_App
Zeichen_Anwendungen	/etc/dt/appconfig/appmanager/C/DrawingApp
Desktop_Anwendungen	/usr/dt/appconfig/appmanager/C/Desktop_Apps
Desktop_Werkzeuge	/usr/dt/appconfig/appmanager/C/Desktop_Tools
Informationen	/usr/dt/appconfig/appmanager/C/Information
Systemadministrator	/etc/dt/appconfig/appmanager/C/System_Admin
Tabellenkalkulation	/users/anna/.dt/appmanager/MySpreadSheet
Medien_Werkzeuge	/etc/dt/appconfig/appmanager/C/ Media_Tools

Wurden die Gruppen von Anwendungen Informationen oder Systemadministrator angepaßt, werden sie stattdessen aus /etc/dt/appconfig/appmanager/C erstellt.

Die Gruppe CAD_Anwendungen wird bei der Zusammenstellung erfaßt, weil ein System mit dem Namen ApServA zum Anwendungssuchpfad hinzugefügt wurde (Abschnitt „Anwendungs-Server dem Anwendungssuchpfad hinzufügen“ auf Seite 53). Bei der Tabellenkalkulation handelt es sich um eine für den einzelnen Benutzer definierte Gruppe von Anwendungen, die nur dem Benutzer anna zur Verfügung steht.

Anwendungen zum Anwendungsmanager hinzufügen

Wurde eine Anwendung zum Anwendungsmanager hinzugefügt, erscheint ein Symbol in einer Gruppe von Anwendungen, über das diese Anwendung gestartet wird.

Viele Anwendungen stellen eine Gruppe von Anwendungen zur Verfügung. Bei der Gruppe von Anwendungen handelt es sich um ein Verzeichnis auf der obersten Ebene des Anwendungsmanagers, das das Anwendungssymbol sowie weitere der Anwendung zugehörige Dateien enthält.

Einige Anwendungen verfügen möglicherweise nicht über eine eigene Gruppe von Anwendungen. Statt dessen befindet sich das Symbol, mit dem diese Anwendung gestartet wird, in einer allgemeinen Gruppe von Anwendungen. Beispielsweise kann der Benutzer eine leere Gruppe von Anwendungen mit dem Namen "Spiele" erstellen, die als Container für alle Spiele, die am System installiert werden, verwendet werden kann.

Verschiedene Möglichkeiten, Anwendungen zum Anwendungsmanager hinzuzufügen

Es gibt zwei verschiedene Möglichkeiten, eine Anwendung dem Anwendungsmanager hinzuzufügen:

- Registrierung der Anwendung
- Hinzufügen eines Anwendungssymbols ohne Registrierung der Anwendung

Registrierung der Anwendung

Die Registrierung der Anwendung stellt die vollständige Integration der Anwendung zur Verfügung.

Eine registrierte Anwendung:

- hat ihre eigenen Gruppen von Anwendungen
- ist eine Anwendung, bei der die Desktop-Konfigurationsdateien an einer Position zusammengestellt sind. Diese Gruppe von Desktop-Konfigurationsdateien wird *Registrierungspaket* genannt.
- verfügt möglicherweise über einen registrierten Hilfeband

Es gibt zwei Möglichkeiten, eine Anwendung zu registrieren:

- Wird eine desktop-fähige Anwendung installiert, wird automatisch eine Registrierung durchgeführt. Weitere Informationen können dem Abschnitt „Desktop-fähige Anwendungen zum Anwendungsmanager hinzufügen“ auf Seite 49 entnommen werden.
- Eine bereits vorhandene Anwendung kann durch Erstellen eines Registrierungspakets registriert werden. Weitere Informationen können dem Abschnitt „Eine bereits vorhandene oder eine nicht Desktop-fähige Anwendung registrieren“ auf Seite 50 entnommen werden.

Die Verwendung eines Registrierungs pakets vereinfacht die Verwaltung einer Anwendung auf dem Desktop. Das Registrierungs paket darf nicht an derselben Stelle im Dateisystem erstellt werden, an der sich die Desktop-Konfigurationdateien befinden.

Anwendungen ohne ein Registrierungs paket hinzufügen

Soll der Anwendungsmanager nur ein Symbol zum Starten der Anwendung enthalten, wird diese Art und Weise, eine Anwendung hinzuzufügen, bevorzugt.

Eine Anwendung, die ohne ein Registrierungs paket hinzugefügt wird,

- hat möglicherweise eine eigene Gruppe von Anwendungen, aber normalerweise wird das Symbol in eine bereits vorhandene Gruppe von Anwendungen gestellt
- ist eine Anwendung, deren Desktop-Konfigurationsdateien sich direkt in den Suchpfaden des Desktops befinden.

Weitere Informationen können dem Abschnitt „Anwendungssymbole einer bereits vorhandenen Gruppe von Anwendungen hinzufügen“ auf Seite 50 entnommen werden.

▼ Desktop-fähige Anwendungen zum Anwendungsmanager hinzufügen

Bei einer Desktop-fähigen Anwendung handelt es sich um eine Anwendung, die bei der Installation automatisch im Anwendungsmanager registriert wird. Die Dateigruppen der Anwendung enthalten auch das für das Desktop erforderliche Registrierungs paket.

1. **Die Anwendung mit den mitgelieferten Instruktionen installieren.**
2. **Nachdem die Installation vollständig ausgeführt ist, ‘Anwendungen erneut laden’ in der Gruppe von Anwendungen ‘Desktop-Werkzeuge’ doppelt anklicken.**
3. **Überprüfen, ob die Installation vollständig ausgeführt ist.**
 - a. **Den Anwendungsmanager öffnen und überprüfen, ob die neue Gruppe von Anwendungen vorhanden ist.**
 - b. **Um die Anwendung zu öffnen, die Gruppe von Anwendungen öffnen und das Symbol der Anwendung doppelt anklicken.**

▼ Eine bereits vorhandene oder eine nicht Desktop-fähige Anwendung registrieren

Hierbei handelt es sich um die bevorzugte Art und Weise, eine Anwendung auf dem Desktop vollständig zu integrieren.

Das Desktop stellt ein Werkzeug, `dtappintegrate`, zur Verfügung, das Verbindungen zwischen den Registrierungsdateien und den Verzeichnissen im Suchpfad des Desktops herstellt.

Die Registrierung auf dem Desktop wird in "näher erläutert.

▼ Anwendungssymbole einer bereits vorhandenen Gruppe von Anwendungen hinzufügen

In diesem Abschnitt wird erläutert, wie ein Anwendungssymbol zu einer bereits vorhandenen Gruppe von Anwendungen hinzugefügt wird.

Beispielsweise stellt das Desktop eine Gruppe von Anwendungen mit dem Namen 'Systemadministrator' zur Verfügung, die für verschiedene Anwendungen und Prozeduren der Systemverwaltung reserviert ist. Bei einer Prozedur, die von Benutzern häufig ausgeführt wird, besteht möglicherweise der Wunsch, diese Prozedur durch Doppelklicken auf einem Symbol in der Gruppe von Anwendungen 'Systemadministrator' auszuführen.

1. Mit 'Aktion erstellen' kann eine Aktionsdefinition für die Anwendung erstellt werden.

Weitere Informationen zu Aktionen erstellen können Kapitel 11 entnommen werden.

2. Eine ausführbare Datei mit demselben Namen wie dem Aktionsnamen in dem Verzeichnis für die Gruppe von Anwendungen erstellen. Der Inhalt der Datei ist irrelevant.

Wurde, beispielsweise, eine Aktion mit dem Namen "Bereinigung" erstellt, die ein Systemverwaltungswerkzeug ausführt, muß die ausführbare Datei /
`etc/dt/appconfig/appmanager/ sprache/System_Admin/Bereinigung`
erstellt werden.

Allgemeine Gruppen von Anwendungen erstellen und verwalten

Bei einer allgemeinen Anwendung handelt es sich um eine Gruppe von Anwendungen (Verzeichnis), die nicht einem bestimmten Anwendungsprodukt zugeordnet ist. Beispielsweise handelt es sich bei der integrierten Gruppe von Anwendungen 'Desktop_Werkzeuge' um eine allgemeine Gruppe, die Symbole für eine große Anzahl von Anwendungen enthält, zwischen denen zwar Beziehungen bestehen, die aber nicht Bestandteil eines einzigen Produkts sind.

Es können zusätzliche allgemeine Gruppen von Anwendungen erstellt werden. Beispielsweise könnte der Benutzer eine Gruppe mit dem Namen Spiele erstellen, in die er alle am System verfügbare Spiele stellt.

Eine allgemeine Gruppe von Anwendungen kann sowohl für das System als auch für den einzelnen Benutzer definiert werden.

▼ Eine für das System definierte allgemeine Gruppe von Anwendungen erstellen

1. Als Benutzer 'root' anmelden.
2. Ein Verzeichnis in `/etc/dt/appconfig/appmanager/` **sprache** erstellen.
Der Name des Verzeichnisses wird zum Namen der Gruppe von Anwendungen.
3. Auf 'Anwendungen erneut laden' in der Gruppe von Anwendungen 'Desktop-Werkzeuge' doppelklicken.

▼ Eine für den einzelnen Benutzer definierte allgemeine Gruppe von Anwendungen erstellen

1. Ein Verzeichnis in **Home-Verzeichnis** `/.dt/appmanager` erstellen.
Der Name des Verzeichnisses wird zum Namen der Gruppe von Anwendungen.
2. Auf 'Anwendungen erneut laden' in der Gruppe von Anwendungen 'Desktop_Werkzeuge' doppelklicken.

▼ Eine integrierte Gruppe von Anwendungen anpassen

1. Als Benutzer 'root' anmelden.

2. Befindet sich die Gruppe von Anwendungen in /usr/dt/appconfig/appmanager/ *sprache*, die Gruppe von Anwendungen in /etc/dt/appconfig/appmanager/ *sprache* kopieren.

Der folgende Befehl kopiert, beispielsweise, die Gruppe von Anwendungen 'Desktop_Werkzeuge':

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

Die neue Kopie der Gruppe von Anwendungen hat Vorrang vor der integrierten Version.

3. Die Kopie der Gruppe von Anwendungen ändern. So können, beispielsweise, neue Aktionsdateien (ausführbare Dateien mit demselben Namen wie Aktionen) hinzugefügt werden.

4. Damit die Änderungen sichtbar werden, muß der Benutzer sich ab- und wieder anmelden.

Suchpfad für Anwendungen ändern

Der wichtigste Grund für die Änderung des Anwendungssuchpfads ist das Hinzufügen eines Anwendungs-Servers. Wird ein Anwendungs-Server zum Suchpfad hinzugefügt, stellt der Anwendungsmanager alle für das System definierte Gruppen von Anwendungen des Servers zusammen.

Weitere Informationen zum Anwendungssuchpfad können dem Abschnitt „Anwendungssuchpfad“ auf Seite 149 entnommen werden.

Standardsuchpfad

Der Standardanwendungssuchpfad enthält die folgenden Verzeichnisse:

TABELLE 4-3

Bereich	Suchpfadverzeichnis
persönlich	<i>Home-Verzeichnis</i> / .dt/appmanager
systemweit	/etc/dt/appconfig/appmanager/ <i>sprache</i>
integriert	/usr/dt/appconfig/appmanager/ <i>sprache</i>

Anwendungs-Server dem Anwendungssuchpfad hinzufügen

Zusätzlich zu der Änderung des Anwendungssuchpfads ist es möglicherweise nötig, zusätzliche Konfiguration-Aufgaben auszuführen, um die Kommunikation mit dem Anwendungs-Server zu ermöglichen. Weitere Informationen können dem Abschnitt „Anwendungs-Services verwalten“ auf Seite 129 entnommen werden.

Einen für das System definierten Anwendungssuchpfad setzen

1. **Als Benutzer 'root' anmelden.**
2. **Ist die Datei /etc/dt/config/Xsession.d/0010.dtpaths nicht vorhanden, wird sie durch Kopieren von /usr/dt/config/Xsession.d/0010.dtpaths erstellt.**
3. **/etc/dt/Xsession.d/0010.paths zum Bearbeiten öffnen. Eine Zeile mit der Variablen DTSPSYSAPPHOSTS bearbeiten oder hinzufügen:**

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

Die folgende Zeile fügt, beispielsweise, das System ApServA zum Anwendungssuchpfad hinzu:

```
DTSPSYSAPPHOSTS=ApServA:
```

4. Alle Benutzer am System müssen darüber informiert werden, daß sie sich ab- und wieder anmelden müssen, damit die Änderung sichtbar wird.

Einen für den einzelnen Benutzer definierten Anwendungssuchpfad setzen

1. **Home-Verzeichnis**/.dtprofile zum Bearbeiten öffnen.

2. Eine Zeile mit der Variablen DTSPUSERAPPHOSTS bearbeiten oder hinzufügen:

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

Die folgende Zeile fügt, beispielsweise, die Systeme ApServB und ApServC zum Anwendungssuchpfad hinzu:

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

3. Ab- und wieder anmelden.

Allgemeine Anwendungsmanagerverwaltung

Zur allgemeinen Anwendungsmanagerverwaltung gehören:

- das Entfernen einer Anwendung
- das erneute Lesen der Datenbank für Anwendungen während einer Session

▼ Anwendung entfernen

Wurde eine Anwendung mit dem Werkzeug dtappintegrate registriert, kann dieser Prozeß mit dtappintegrate auch wieder rückgängig gemacht werden. Ist eine Anwendung nicht registriert, wird ihre Gruppe von Anwendungen aus dem Anwendungsmanager entfernt und ihre Aktionen, Datentypen, Symbole und Hilfe sind nicht länger verfügbar.

1. Als Benutzer 'root' anmelden.

2. Folgenden Befehl ausführen:

```
dtappintegrate -s app_root -u
```

▼ Anwendungsmanager während einer Session aktualisieren

Werden Anwendungen zum Anwendungsmanager hinzugefügt und sollen diese Änderungen sofort sichtbar werden, muß der Anwendungsmanager erneut generiert werden.

- ◆ **Die Gruppe von Anwendungen 'Desktop_Werkzeuge' öffnen und auf 'Anwendungen erneut laden' doppelt klicken.**

Die Aktion 'Anwendungen erneut laden' kann zum Aktualisieren des Anwendungsmanagers verwendet werden, wenn Anwendungen zu einem Anwendungs-Server hinzugefügt werden. Aber 'Anwendungen erneut laden' stellt nicht fest, daß Anwendungen vom Anwendungs-Server entfernt wurden oder von einer Position zu einer anderen verschoben wurden. Diese Änderungen werden erst sichtbar, wenn der Benutzer sich ab- und wieder anmeldet.

Texteditor und den Terminal-Emulator ändern

Sowohl die Anwendung 'Texteditor' als auch die Anwendung 'Terminal-Emulator' können durch Auswahl eines Bedienelements im Bedienfeld oder durch Doppelklicken eines Symbols im Anwendungsmanager gestartet werden.

Diese Anwendungen können auch durch andere Desktop-Aktivitäten gestartet werden.

- Die Anwendung 'Texteditor' wird geöffnet wenn der Benutzer eine Textdatei im Dateimanager auswählt und im Menü 'Ausgewählt Öffnen' auswählt. Der Standard-Texteditor ist `dtpad`.
- Ein Terminal-Emulator wird ausgeführt, wenn ein Benutzer im Menü 'Datei' des Dateimanagers 'Terminal-Fenster Öffnen' auswählt, oder wenn eine Aktion ein Terminal-Emulator-Fenster öffnet. Der Standard-Terminal-Emulator ist `dtterm`.

Das Desktop kann so konfiguriert werden, daß in diesen Situationen ein anderer Texteditor oder ein anderer Terminal-Emulator verwendet wird.

▼ Den standardmäßigen Texteditor oder Terminal-Emulator ändern

1. Soll die Änderung für das gesamte System gelten, als Benutzer 'root' anmelden.

2. Eine Aktion für den neuen Texteditor oder Terminal-Emulator erstellen.

- Hierfür kann die Anwendung 'Aktion erstellen' verwendet werden. Abbildung 3-3 zeigt ein Fenster 'Aktion erstellen', in dem eine Anwendung mit dem Namen TextPad angegeben ist. Weitere Informationen zu 'Aktion erstellen' können Kapitel 11 entnommen werden.



Abbildung 4-3 Fenster 'Aktion erstellen'

- Oder aber, der Benutzer erstellt die Aktionsdefinition manuell, zum Beispiel:

TABELLE 4-4

```

ACTION TextPad
{
    LABEL    TextPad

    TYPE    COMMAND

    WINDOW_TYPE    NO_STDIO

    EXEC_STRING    /usr/TP/bin/TextPad %(File)Arg_1%

    DESCRIPTION    Dieses Symbol doppelklicken, um \           die Anwendung
    TextPad zu starten.
}

```

Weitere Informationen zum manuellen Erstellen von Aktionsdefinitionen können Kapitel 12 entnommen werden.

3. **Die Konfigurationsdatei mit der neuen Aktion in das entsprechende Verzeichnis stellen:**
 - systemweit /etc/dt/appconfig/types/ *sprache*
 - persönlich : *Home-Verzeichnis* / .dt/types
4. **Ist die Datei zutreffende Datei user-prefs.dt nicht bereits vorhanden, wird sie durch Kopieren von /usr/dt/appconfig/types/ *sprache*/user-prefs.dt in folgende Verzeichnisse erstellt:**
 - systemweit: Verzeichnis /etc/dt/appconfig/types/ *sprache*
 - persönlich: Verzeichnis *Home-Verzeichnis* / .dt/types
5. **Die Texteditor- oder Terminal-Aktion in der systemweiten oder persönlichen Datei user-prefs.dt bearbeiten. Die Zeile MAP_ACTION so ändern, daß die Aktion an die neue Aktion adressiert wird.**

TABELLE 4-5

Beispielsweise kann die Zeile

MAP_ACTION Dtpad

wie folgt geändert werden:

MAP_ACTION TxtPd

- 6. Die Datei `user-prefs.dt` sichern.**
- 7. Auf 'Aktionen erneut laden' in der Gruppe von Anwendungen 'Desktop_Werkzeuge' doppelklicken, um die Datenbank für Aktionen erneut zu laden.**

Anwendungen registrieren

In diesem Kapitel wird beschrieben, wie ein Registrierungspaket für eine Anwendung und wie die Anwendung am Desktop registriert wird.

- „Übersicht zur Anwendungsregistrierung ” auf Seite 60
- „Allgemeine Schritte zur Registrierung einer Anwendung” auf Seite 63
- „Schritt 1: Schriftarten- und Farbbressourcen ändern” auf Seite 64
- „Schritt 2: Desktop-Anwendungsstammverzeichnis erstellen” auf Seite 66
- „Schritt 3: Registrierungspaketverzeichnisse erstellen” auf Seite 66
- „Schritt 4: Aktionen und Datentypen für die Anwendung erstellen” auf Seite 69
- „Schritt 5: Hilfdaten in das Registrierungspaket einfügen” auf Seite 72
- „Schritt 6: Symbole für die Anwendung erstellen” auf Seite 73
- „Schritt 7: Die Anwendungsgruppe erstellen ” auf Seite 74
- „Schritt 8: Anwendungen mit dtappintegrate registrieren” auf Seite 81
- „Beispiel zur Erstellung eines Registrierungspakets ” auf Seite 84

Wenn eine Anwendung vollständig am Desktop registriert ist, besitzt sie folgende Komponenten:

- Eine Anwendungsgruppe in der obersten Ebene des Anwendungsmanagers.
- Eine Aktion, die die Anwendung startet. Diese Aktion wird durch ein Symbol in der Anwendungsgruppe repräsentiert.
- Optionale Datentypen für die Datendateien.

Bei der Anwendungsregistrierung handelt es sich um eine Operation, die die Anwendung nicht beeinflusst:

- Die ausführbare Datei der Anwendung wird nicht geändert. Daher können auch bereits vorhandene Anwendungen im System registriert werden.

- Es ist nicht erforderlich, daß eine der gelieferten Dateien der Anwendung (z. B. die ausführbare Datei `app-defaults`) an eine andere Position im Dateisystem verschoben wird.
- Die Anwendungsregistrierung kann leicht rückgängig gemacht werden. Das Werkzeug `dtappintegrate`, mit dem Anwendungen registriert werden, stellt eine Befehlszeilenoption zur Verfügung, mit der der Prozeß rückgängig gemacht werden kann.

In folgenden Fällen wird ein Registrierungspaket erstellt:

- Ein Systemadministrator will eine existierende Anwendung am Desktop registrieren.
- Ein Programmierer will ein Installationspaket für eine desktop-fähige Anwendung erstellen.

Übersicht zur Anwendungsregistrierung

Dieser Abschnitt erläutert folgende Themen:

- Zweck der Anwendungsregistrierung
- Funktionen, die der Anwendung durch die Anwendungsregistrierung zur Verfügung gestellt werden

Hinweis - Ein detailliertes Beispiel zur Registrierung einer vorhandenen Anwendung finden Sie im Abschnitt „Beispiel zur Erstellung eines Registrierungspakets“ auf Seite 84.

Durch die Anwendungsregistrierung bereitgestellte Funktionen

Die Anwendungsregistrierung bietet folgende Möglichkeiten für den Benutzer:

- Anwendung suchen.

Bei der Installation wird die Anwendung im Anwendungsmanager registriert und erhält eine eigene Anwendungsgruppe.



Abbildung 5-1 Anwendungsgruppe in der obersten Ebene des Anwendungsmanagers

- Anwendung starten.

Die Anwendungsgruppe der Anwendung enthält ein Symbol, auf das der Benutzer doppelklicken kann, um die Anwendung zu starten.

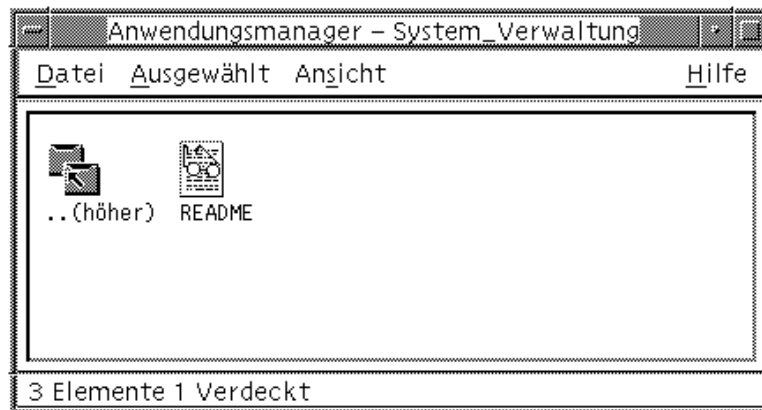


Abbildung 5-2 Eine Anwendungsgruppe mit einem Symbol zum Starten der Anwendung

- Datendateien identifizieren und bearbeiten. Die Datendateien der Anwendung besitzen ein eindeutiges Symbol im Dateimanager.

Der Benutzer kann mit den Symbolen von Datendateien folgende Operationen durchführen:

- Anwendung starten
- Datendateien drucken

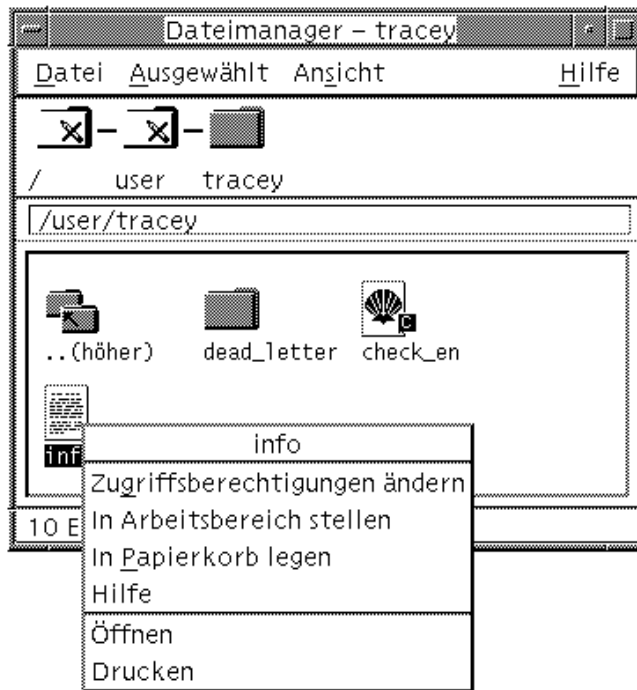


Abbildung 5-3 Menü einer Datendatei mit den Optionen “Öffnen” und “Drucken”

- Operationen wie E-Mail, Komprimieren, Anzeigen oder Abspielen (Audiodaten)

Zweck der Anwendungsregistrierung

Eine registrierte Desktop-Anwendung verfügt über verschiedene Konfigurationsdateien, die das Desktop verwendet, um die Benutzerschnittstelle der Anwendung zur Verfügung zu stellen:

- Definitionsdateien für Aktionen und Datentypen
- Symboldateien (Pixmap oder Bitmap)
- Ein Verzeichnis mit den Dateien, die die Anwendungsgruppe bilden
- Optional Desktop-Hilfdateien und Bedienfelddefinitionsdateien

Damit diese Dateien vom Desktop erkannt und verwendet werden können, müssen sie sich in Verzeichnissen befinden, die in den Suchpfaden des Desktops angegeben sind.

Die Verwaltung einer Anwendung wird erschwert, wenn deren Konfigurationsdateien auf viele verschiedene Verzeichnisse verteilt sind. Daher kann eine Anwendung alle Desktop-Konfigurationsdateien in einem Verzeichnis speichern. Diese Dateigruppierung wird als *Registrierungspaket* bezeichnet.

Ist die Anwendung desktop-fähig, stellt sie ein Registrierungspaket als Bestandteil des Installationspakets zur Verfügung. Erstellt der Systemadministrator die Konfigurationsdateien selbst, kann er auch das Registrierungspaket erstellen.

Die Konfigurationsdateien im Registrierungspaket sind für das Desktop nicht verfügbar, weil sie sich nicht in den Verzeichnissen im Suchpfad befinden. Das Kopieren dieser Dateien an die richtige Position wird als *Registrierung* oder *Integration* der Anwendung bezeichnet.

Das Desktop stellt das Werkzeug `dtappintegrate` zur Verfügung, das die Registrierung durchführt, indem es symbolisch verknüpfte Repräsentationen der Dateien in den richtigen Verzeichnissen im Suchpfad erstellt.

Viele desktop-fähige Anwendungen führen `dtappintegrate` während des Installationsprozesses automatisch aus. Integriert ein Systemadministrator eine bereits vorhandene Anwendung, kann er das Dienstprogramm ausführen, nachdem das Registrierungspaket erstellt wurde.

Sobald eine Anwendung am Desktop eines Systems registriert ist, steht die Anwendung allen Benutzern des Systems zur Verfügung. Ist das System als Desktop-Anwendungs-Server konfiguriert, steht die Anwendung außerdem allen anderen Systemen im Netzwerk zur Verfügung.

Das Werkzeug `dtappintegrate` verfügt über eine Befehlszeilenoption, die den Prozeß rückgängig macht, indem sie die Verknüpfungen löst. Es ist daher kein Problem, die Anwendung aus dem Anwendungsmanager zu entfernen, um sie einem anderen Anwendungs-Server hinzuzufügen oder sie zu aktualisieren.

Allgemeine Schritte zur Registrierung einer Anwendung

Hinweis - Ein ausführliches Beispiel, bei dem anhand dieser Schritte ein Anwendungspaket erstellt wird, kann dem Abschnitt „Beispiel zur Erstellung eines Registrierungspakets“ auf Seite 84 entnommen werden.

- 1. Ändern Sie alle Anwendungsressourcen, die Schriftarten und Farben einstellen. Andernfalls arbeiten die dynamischen Schriftarten und Farben des Desktops nicht richtig.**

Weitere Informationen können dem Abschnitt „Schritt 1: Schriftarten- und Farbressourcen ändern“ auf Seite 64 entnommen werden.

- 2. Erstellen Sie ein Anwendungsstammverzeichnis.**

Weitere Informationen können dem Abschnitt „Schritt 2: Desktop-Anwendungsstammverzeichnis erstellen“ auf Seite 66 entnommen werden.

3. Erstellen Sie die Verzeichnisstruktur unter dem Stammverzeichnis der Anwendung.

Weitere Informationen können dem Abschnitt „Schritt 3: Registrierungspaketverzeichnisse erstellen“ auf Seite 66 entnommen werden.

4. Erstellen Sie die Aktionen und Datentypen für die Anwendung.

Weitere Informationen können dem Abschnitt „Schritt 4: Aktionen und Datentypen für die Anwendung erstellen“ auf Seite 69 entnommen werden.

5. Kopieren Sie die Hilfedateien in ein geeignetes Verzeichnis.

Weitere Informationen können dem Abschnitt „Schritt 5: Hilfedateien in das Registrierungspaket einfügen“ auf Seite 72 entnommen werden.

6. Erstellen Sie Symbole für die Anwendung.

Weitere Informationen können dem Abschnitt „Schritt 6: Symbole für die Anwendung erstellen“ auf Seite 73 entnommen werden.

7. Erstellen Sie die Anwendungsgruppe für die Anwendung.

Weitere Informationen können dem Abschnitt „Schritt 7: Die Anwendungsgruppe erstellen“ auf Seite 74 entnommen werden.

8. Registrieren Sie die Anwendung mit `dtappintegrate`.

Weitere Informationen können dem Abschnitt „Schritt 8: Anwendungen mit `dtappintegrate` registrieren“ auf Seite 81 entnommen werden.

Schritt 1: Schriftarten- und Farbressourcen ändern

Hinweis - Ein Beispiel zur Änderung der Ressourcen für eine Anwendung finden Sie in Schritt 1 im Abschnitt „Beispiel zur Erstellung eines Registrierungspakets“ auf Seite 84.

Das Desktop stellt Mechanismen für das Setzen und Bearbeiten von Schnittstellenschriftarten und Fensterfarben zur Verfügung. Damit eine Anwendung diese Mechanismen ohne Probleme verwenden kann, muß möglicherweise die Datei `app-defaults` der Anwendung geändert werden.

Schriftartenressourcen ändern

Hinweis - Dieser Abschnitt bezieht sich auf Anwendungen, die mit OSF/Motif 1.2TM (oder neueren Versionen) erstellt wurden. Der Style-Manager kann keine Schnittstellenschriftarten für Anwendungen setzen, die mit älteren Versionen von OSF/Motif geschrieben wurden.

Der Desktop-Style-Manager setzt Schnittstellenschriftarten für Anwendungen, die mit OSF/Motif 1.2 (oder neueren Versionen) erstellt wurden, sofern die Anwendung keine anwendungsspezifische Schnittstellenschriftarten verwendet.

Der Style-Manager stellt zwei Schriftarten zur Verfügung:

Systemschriftart — wird vom System für Bezeichnungen, Menüs und Tasten verwendet.

Benutzerschriftart — wird für bearbeitbare Bereiche wie Textfelder verwendet.

Jede Schriftart steht in sieben Größen, die im Dialogfenster Schriftarten mit 1 bis 7 bezeichnet sind, zur Verfügung. Die Schriftarten des Style-Managers werden durch die Style-Manager-Ressourcen, die in `/usr/dt/app-defaults/sprache /Dtstyle` gesetzt werden, den tatsächlichen Schriftarten des Systems zugeordnet.

Soll die Anwendung Schriftarten des Style-Managers verwenden, müssen Anwendungsressourcen, die Schnittstellenschriftarten angeben, entfernt werden. Das Desktop legt die Ressourcen für die Anwendung automatisch fest:

`FontList` — Systemschriftart

`XmText*FontList` — Benutzerschriftart

`XmTextField*FontList` — Benutzerschriftart

Farbressourcen ändern

Der Style-Manager kann Anwendungsfarben dynamisch ändern. Die Anwendung muß ein OSF/Motif 1.1- oder 1.2-Client sein. Clients, die mit anderen Toolkits geschrieben wurden, können Farben nicht dynamisch ändern. In diesen Fällen muß der Client neu gestartet werden.

Der einfachste Weg zur Verwendung der dynamischen Farben des Desktops ist das Löschen aller Farbressourcen der Anwendung für die Hintergrund- und die Vordergrundfarbe.

Schritt 2: Desktop-Anwendungsstammverzeichnis erstellen

Hinweis - Ein Beispiel zur Erstellung des Desktop-Anwendungsstammverzeichnisses für eine Anwendung kann Schritt 2 des Abschnitts „Beispiel zur Erstellung eines Registrierungspakets“ auf Seite 84 entnommen werden.

Die Dateien des Registrierungspakets werden im Anwendungsstammverzeichnis *app_root* gruppiert. Das Verzeichnis *app_root* für die Desktop-Konfigurationsdateien kann mit dem Installationsverzeichnis *app_root* der Anwendung identisch sein oder sich an einer anderen Position befinden.

Ein Beispiel: Eine Anwendung ist im Verzeichnis `/usr/BTE` installiert. Dieses Verzeichnis kann gleichzeitig als *app_root* für die Desktop-Konfigurationsdateien verwendet werden. Wenn Sie jedoch eine Anwendung integrieren, die nicht desktop-fähig ist, sollten Sie ein eigenes Desktop-Verzeichnis *app_root* erstellen. Dieses Vorgehen verhindert das Überschreiben der von Ihnen erstellten Konfigurationsdateien, wenn die Anwendung aktualisiert wird.

Ein Systemadministrator kann also beispielsweise das Verzeichnis `/etc/desktop_approots/BTE` als Desktop-Verzeichnis *app_root* erstellen.

Schritt 3: Registrierungspaketverzeichnisse erstellen

Hinweis - Ein Beispiel zum Erstellen von Registrierungspaketverzeichnissen für eine Anwendung finden Sie in Schritt 3 des Abschnitts „Beispiel zur Erstellung eines Registrierungspakets“ auf Seite 84.

Das Registrierungspaket ist die Gruppe der Desktop-Konfigurationsdateien, die vom Desktop zur Bereitstellung der grafischen Benutzeroberfläche für die Anwendung verwendet werden.

Inhalt des Registrierungspakets

Zu den Desktop-Konfigurationsdateien gehören:

- Definitionsdateien für Aktionen und Datentypen
- Symboldateien
- Ein Anwendungsgruppenverzeichnis und dessen Inhalt
- Optional: Hilfedateien und eine Bedienfeldkonfigurationsdatei

Das Registrierungspaket wird unter einem Verzeichnis der obersten Ebene zusammengestellt, dem Anwendungsstammverzeichnis oder *app_root*.

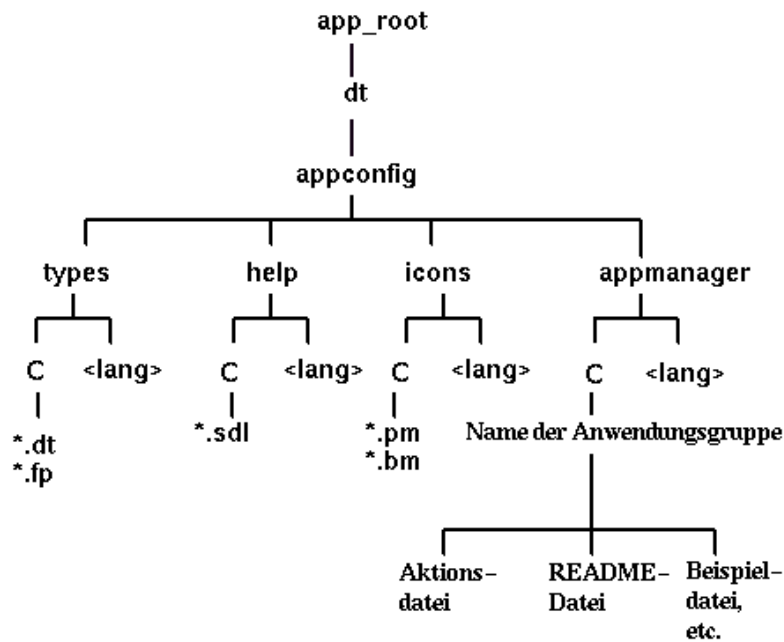


Abbildung 5-4 Ein Registrierungs paket unter einem Anwendungsstammverzeichnis

Die wichtigsten Kategorien der Konfigurationsdateien unter dem Verzeichnis `app_root/dt/appconfig` werden in Tabelle 5-1 beschrieben.

TABELLE 5-1 Wichtige Konfigurationsfeldkategorien

Unterverzeichnis	Inhalt
types	Definitionsdateien für Aktionen und Datentypen
help	Desktop-Hilfdateien
icons	Bitmap- und Pixmap-Dateien werden von den Aktionen und Datentypen der Anwendung verwendet
appmanager	Verzeichnis und Inhalte, die die Anwendungsgruppe bilden

Jede dieser Kategorien verfügt über Unterverzeichnisse für sprachabhängige Dateien. Standardsprachdateien befinden sich im Verzeichnis `C`.

Registrierungspaket erstellen

- ◆ Erstellen Sie die folgenden Verzeichnisse. Wenn Sie sprachabhängige Konfigurationsdateien bereitstellen, erstellen Sie ein separates Verzeichnis für jede Sprache. Stellen Sie die Dateien nur für eine Sprache bereit, sollten Sie diese in das Verzeichnis `C` kopieren.

- `app_root/dt/appconfig/types/ sprache`
- `app_root/dt/appconfig/help/ sprache`
- `app_root/dt/appconfig/icons/ sprache`
- `app_root/dt/appconfig/appmanager/ sprache/ Anwendungsgruppenname`. Dabei repräsentiert `Anwendungsgruppenname` den Namen der Anwendungsgruppe.

Ein Beispiel: Abbildung 5-5 zeigt den Anwendungsmanager mit einer Gruppe, für die `Anwendungsgruppenname` den Wert "Medienwerkzeuge" enthält.



Abbildung 5-5 Anwendungsgruppe auf der obersten Ebene des Anwendungsmanagers

Das Werkzeug `dtappintegrate` kann nur für die Desktop-Konfigurationsdateien in den Verzeichnissen `types`, `help`, `icons` und `appmanager` verwendet werden. Die ausführbare Binärdatei der Anwendung, `app-defaults`, und die Nachrichtenkatalogdateien werden separat verwaltet.

Schritt 4: Aktionen und Datentypen für die Anwendung erstellen

Hinweis - Ein Beispiel für die Erstellung der Aktionen und Datentypen für eine Anwendung kann Schritt 4 des Abschnitts „Beispiel zur Erstellung eines Registrierungs Pakets “ auf Seite 84 entnommen werden.

Aktionen und Datentypen stellen eine Benutzerschnittstelle für die Anwendung zur Verfügung.

- Aktionen stellen eine Benutzerschnittstelle für den Befehl zur Verfügung, der die Anwendung startet.
- Datentypen stellen ein angepaßtes Erscheinungsbild und Verhalten für die Datendateien der Anwendung zur Verfügung.

Für eine Anwendung erforderliche Aktionen und Datentypen

Für typische Anwendungen sind folgende Definitionen für Aktionen und Datentypen erforderlich:

- Eine Aktion, die die Anwendung öffnet.
- Ein Datentyp für die Datendateien der Anwendung. Wird ein Datentyp erstellt, müssen auch folgende Komponenten erstellt werden:
 - Eine “Öffnen”-Aktion für die Datendateien der Anwendung
 - Eine “Drucken”-Aktion für die Datendateien der Anwendung
- Ein Datentyp für die Anwendungsgruppe (siehe „Eindeutige Symbole für Anwendungsgruppen konfigurieren“ auf Seite 76).

Eine Einführung zur Verwendung von Aktionen und Datentypen auf dem Desktop enthält .

Position der Konfigurationsdateien mit den Definitionen für Aktionen und Datentypen

Aktionen und Datentypen werden in Konfigurationsdateien definiert. Bei der Benennung der Dateien muß das Suffix `.dt` verwendet werden. Per Konvention erhält die Datei den Namen `aktion_name .dt` oder `anwendung_name .dt`.

Dateien mit Aktionen und Datentypen sollten unter dem Anwendungsstammverzeichnis im Verzeichnis `app_root/dt/appconfig/types/ sprache` gespeichert werden. Der Standardwert für `sprache` ist `C`.

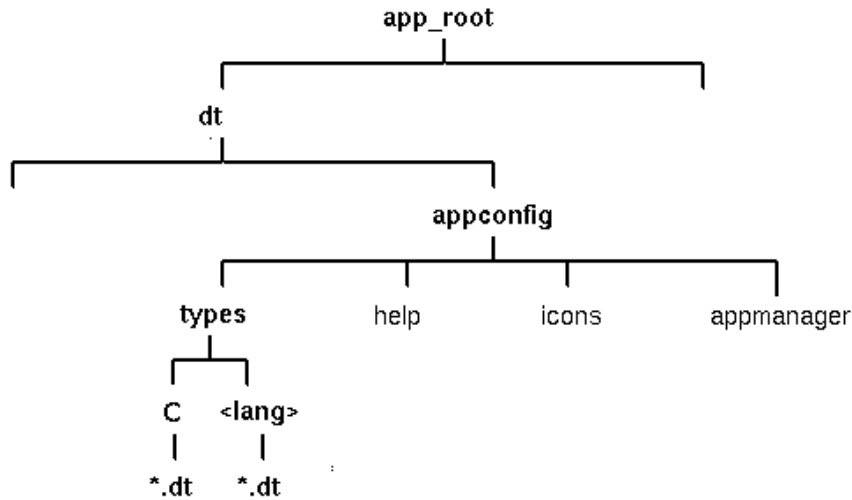


Abbildung 5-6 Definitionen für Aktionen und Datentypen

Möglichkeiten zum Erstellen von Aktionen und Datentypen

Definitionen für Aktionen und Datentypen für eine Anwendung können auf zwei verschiedene Weisen erstellt werden:

- Verwenden Sie das Werkzeug “Aktion erstellen”.
 - “Aktion erstellen” stellt eine benutzerfreundliche Schnittstelle mit Textfeldern zur Verfügung, die der Benutzer ausfüllen muß. Das Werkzeug unterliegt jedoch verschiedenen Beschränkungen.
- Erstellen Sie die Definitionen manuell.
 - Hierfür ist es erforderlich, daß der Benutzer die Syntax für die Erstellung der Definitionen erlernt, er erhält dafür Zugriff auf sämtliche Funktionen.

Aktionen und Datentypen mit “Aktion erstellen” erstellen

Bei dieser Prozedur wird das Dienstprogramm “Aktion erstellen” verwendet, um eine Aktion und Datentypen für die Anwendung zu erstellen.

Weitere Informationen zu “Aktion erstellen” können der zugehörigen Online-Hilfe oder entnommen werden.

1. **Öffnen Sie die Anwendungsgruppe Desktop_Apps, und klicken Sie doppelt auf “Aktion erstellen”.**
2. **Verwenden Sie “Aktion erstellen”, um die Definitionen für Aktionen und Datentypen für die Anwendung und deren Datentyp zu erstellen.**

Die mit "Aktion erstellen" erstellte Konfigurationsdatei wird in das Verzeichnis *Home-Verzeichnis* / `.dt/type/aktion_name.dt` geschrieben. Die Aktionsdatei (die ausführbare Datei mit dem Namen der Aktion) wird im Home-Verzeichnis erstellt.

3. **Testen Sie die Aktion mit der im Home-Verzeichnis erstellten Aktionsdatei.**
4. **Kopieren Sie die Definitionsdatei der Aktion (*Home-Verzeichnis* / `.dt/type/aktion_name .dt`) in das Verzeichnis `app_root /dt/appconfig/types/sprache .`**
5. **Nachdem das Verzeichnis für die Anwendungsgruppe erstellt wurde (siehe „Schritt 7: Die Anwendungsgruppe erstellen “ auf Seite 74), kopieren Sie die Aktionsdatei *Home-Verzeichnis* / `aktion_name` in das Verzeichnis `app_root/dt/appconfig/appmanager/ sprache/anwendungsgruppenname .`**

Aktionen und Datentypen manuell erstellen

- ◆ **Erstellen Sie eine Konfigurationsdatei mit den Definitionen für Aktionen und Datentypen für die Anwendung.**

Definitionsdateien für Aktionen und Datentypen müssen der Namenskonvention `name.dt` entsprechen.

Sie können alle Definitionen für Aktionen und Datentypen in eine Datei einfügen oder mehrere Dateien verwenden. Jede Datei sollte einen Dateinamen erhalten, dem Systemadministratoren die zugehörige Anwendung entnehmen können.

Die Namen für Aktionen und Datentypen müssen aus einem Wort bestehen (keine Leerzeichen). Unterstriche dürfen verwendet werden. Per Konvention wird der erste Buchstabe des Aktions- oder Datentypnamens als Großbuchstabe geschrieben. Es darf kein bereits vorhandener Aktions- oder Dateiname verwendet werden. Es sollte ein Name verwendet werden, dem fortgeschrittene Benutzer und Systemadministratoren die zugehörige Anwendung entnehmen können.

Soll das Anwendungssymbol mit einem anderen Namen als dem der Aktion beschriftet werden, geben Sie das Feld LABEL in der Definition der Aktion an.

Weitere Informationen zur Erstellung von Aktionen und Datentypen können folgenden Kapiteln entnommen werden:

-
-
- Kapitel 12
- Kapitel 13

Schritt 5: Hilfedateien in das Registrierungspaket einfügen

Hinweis - Ein Beispiel zum Einfügen von Hilfedateien in ein Registrierungspaket finden Sie in Schritt 5 des Abschnitts „Beispiel zur Erstellung eines Registrierungspakets“ auf Seite 84.

Verfügt die Anwendung über einen Desktop-Hilfebund (einen Hilfebund, der mit dem Help Developer's Kit des Desktop erstellt wurde), sollte die Stammdatei für den Hilfebund (*.sdl) in das Verzeichnis `app_root/appconfig/help/ sprache` kopiert werden.

Grafiken für Hilfedateien werden normalerweise in das Unterverzeichnis `graphics` kopiert. Dieses Unterverzeichnis muß sich relativ zur Stammdatei für den Hilfebund (*.sdl) in demselben Unterverzeichnis wie zum Zeitpunkt der Erstellung des Hilfebands befinden.

Besitzt die Anwendung keinen Hilfebund, können Sie gegebenenfalls mit Help Developer's Kit einen eigenen Band erstellen.

Es gibt zwei Integrationsstufen für einen Hilfebund:

- **Volle Integration.**

Ist die Desktop-Hilfe vollständig integriert, kann auf den Hilfebund von der Anwendung aus zugegriffen werden, zum Beispiel über "Kontexthilfe" und das Menü "Hilfe". Für die vollständige Integration ist die Änderung der ausführbaren Dateien der Anwendung erforderlich.

- **Partielle Integration.**

Auf eine partiell integrierte Desktop-Hilfe kann von der obersten Ebene des Hilfemanagers zugegriffen werden. In den Fenstern der Anwendung kann dagegen nicht auf die Hilfe zugegriffen werden. Sie können aber eine Aktion definieren, die den Zugriff auf die Hilfe in der Anwendungsgruppe ermöglicht. Die folgende Beispielaktion zeigt den Hilfebund in der Stammdatei `MyApp.sdl` an:

```
ACTION OpenMyAppHelp
{
    LABEL           MyAppHelp
    ARG_COUNT       0
    TYPE            COMMAND
    WINDOW_TYPE     NO_STDIO
    EXEC_STRING     /usr/dt/bin/dthelpview -helpVolume MyApp
    DESCRIPTION     Zeigt Hilfe für die Anwendung MyApp an.
```


}

Schritt 6: Symbole für die Anwendung erstellen

Hinweis - Ein Beispiel zur Erstellung der Symboldateien für eine Anwendung finden Sie in Schritt 6 des Abschnitts „Beispiel zur Erstellung eines Registrierungs Pakets “ auf Seite 84.

Das Desktop stellt Standardsymbole für Aktionen, Datentypen und Anwendungsgruppen zur Verfügung. Sie können aber auch eigene Symbole für die Anwendung erstellen.

Symbole befinden sich im Verzeichnis *app_root* /dt/appconfig/icons/*sprache*.

Für das Desktop erforderliche Symbole

Die Anwendung verwendet folgende Symbole auf dem Desktop:

- **Aktionssymbol.** Der Benutzer klickt doppelt auf dieses Symbol, um die Anwendung (Aktionen) zu starten. Es wird im Feld `ICON` der Aktion angegeben, die die Anwendung startet.

Drei Größen sind möglich: klein, mittel und groß.

- **Datentypsymbol.** Dieses Symbol repräsentiert die Datendateien der Anwendung im Dateimanager. Es wird im Feld `ICON` der Datentypdefinition angegeben.

Unterstützt die Anwendung mehrere Datentypen, sollte für jeden Datentyp ein anderes Symbol bereitgestellt werden.

Zwei Größen sind möglich: klein und mittel.

- **Anwendungsgruppensymbol.** Dieses Symbol repräsentiert das Verzeichnis auf der obersten Ebene des Anwendungsmanagers. Es wird im Feld `ICON` der Datentypdefinition für die Anwendungsgruppe angegeben. (Siehe „Schritt 7: Die Anwendungsgruppe erstellen “ auf Seite 74.)

Zwei Größen sind möglich: klein und mittel.

Sie sollten gegebenenfalls Pixmap- und Bitmap-Versionen jedes Symbols bereitstellen, um Farb- (acht Bit und mehr) und Monochrombildschirme (weniger als acht Bit) zu unterstützen.

TABELLE 5-2 Namenskonventionen für Symboldateien

Größe	Maße in Pixel	Bitmap-Name	Pixmap-Name
Klein	16 x 16	<i>Basisname</i> . t . bm	<i>Basisname</i> . t . pm
Mittel	32 x 32	<i>Basisname</i> . m . bm	<i>Basisname</i> . m . pm
Groß	48 x 48	<i>Basisname</i> . l . bm	<i>Basisname</i> . l . pm

Werden keine Bitmap-Dateien zur Verfügung gestellt, ordnet das Desktop den Farbwerten Schwarz- und Weißwerte zu. Dabei kommt es möglicherweise zu Abweichungen vom gewünschten Erscheinungsbild.

Weitere Informationen zu Symbolen können dem Abschnitt „Symbolbilddateien“ auf Seite 253.

Schritt 7: Die Anwendungsgruppe erstellen

Hinweis - Ein Beispiel zur Erstellung einer Anwendungsgruppe finden Sie in Schritt 7 des Abschnitts „Beispiel zur Erstellung eines Registrierungs pakets “ auf Seite 84.

Nachdem die Definitionen für Aktionen und Datentypen für die Anwendung erstellt wurden, müssen die Konfigurationsdateien für die Bildschirmanzeige, also die Anwendungsgruppe und deren Inhalt erstellt werden.

Die Anwendungsgruppe ist ein Verzeichnis in der obersten Ebene des Anwendungsmanagers (siehe Abbildung 5-1).

Anwendungsgruppen werden in drei Schritten erstellt:

- Erstellen Sie das Anwendungsgruppenverzeichnis im Registrierungs paket.
- Optional: Konfigurieren Sie ein eindeutiges Symbol für die Anwendung. Dies setzt die Erstellung einer Datentypdefinition für das Anwendungsgruppenverzeichnis voraus.
- Erstellen Sie den Inhalt der Anwendungsgruppe.

Das Anwendungsgruppenverzeichnis erstellen

Erstellen Sie die Verzeichnisse im Registrierungs paket unter `appmanager` (siehe Abbildung 5-7), um eine Anwendungsgruppe zu erstellen.

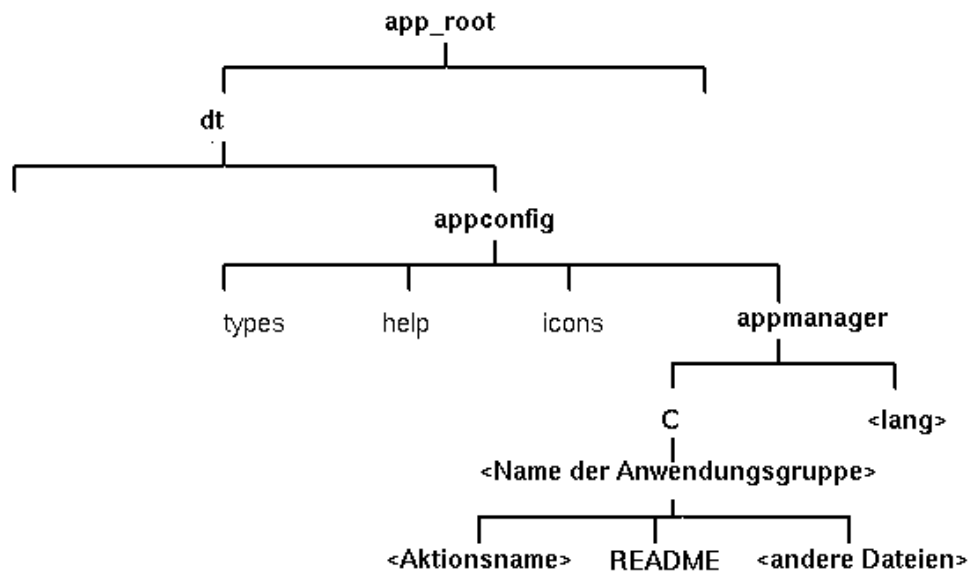


Abbildung 5-7 Das Verzeichnis appmanager

Name der Anwendungsgruppe

<anwendungsgruppenname> in Abbildung 5-7 ist der Name der Anwendungsgruppe.

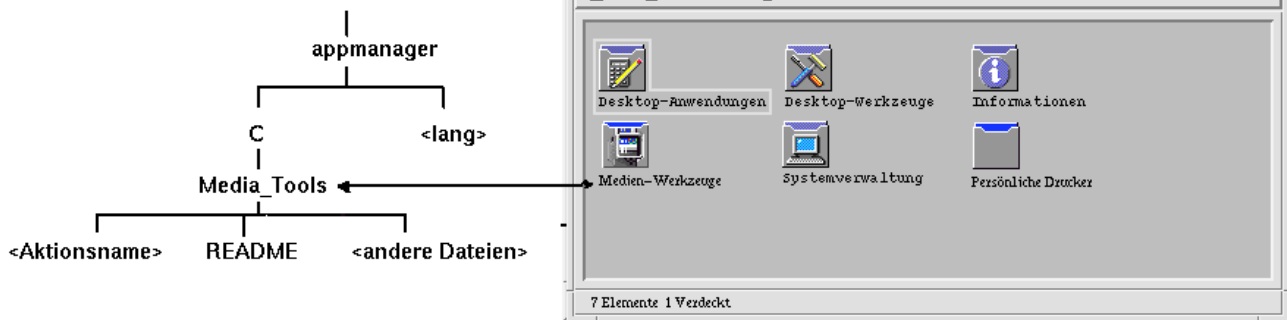


Abbildung 5-8 Der Anwendungsgruppenname (<anwendungsgruppenname>)

Bei dem Namen kann es sich um jeden zulässigen Dateinamen bzw. Verzeichnisnamen handeln. Der Name sollte die Anwendung beschreiben.

Eindeutige Symbole für Anwendungsgruppen konfigurieren

Das Desktop stellt ein Standardsymbol für Anwendungsgruppen zur Verfügung. Sie können jedoch ein eigenes Symbol erstellen.

Wollen Sie ein eindeutiges Symbol für die Anwendungsgruppe erstellen, müssen Sie folgende Komponenten erstellen:

- Ein Datentyp für das Verzeichnis auf der obersten Ebene des Anwendungsmanagers.
- Die Aktionen “Öffnen” und “Drucken” für den Datentyp.

Ein Beispiel: Sie wollen eine Anwendungsgruppe mit dem Namen “Medienwerkzeuge” erstellen. Mit der folgenden Datentypdefinition in der Datei *app_root/dt/appconfig/types/sprache/name.dt* wird der Anwendungsgruppe ein eindeutiges Symbol zugewiesen.

```
DATA_ATTRIBUTES MedienwerkzeugeAppgroup
{
    ACTIONS      OpenInPlace,OpenNewView
    ICON         MediaTools
    DESCRIPTION  Doppelklick zum Öffnen der Anwendungsgruppe\
```

```

}

DATA_CRITERIA MedienwerkzeugeAppgroupCriteria1
{
    DATA_ATTRIBUTES_NAME MedienwerkzeugeAppgroup
    MODE d
    PATH_PATTERN */appmanager/*/Medienwerkzeuge
}

```

Die Attribute der Definition geben das zu verwendende Symbol an. Der Abschnitt mit den Kriterien gibt an, daß der Datentyp in einem Verzeichnis Medienwerkzeuge definiert werden soll, das ein Unterverzeichnis des Verzeichnisses appmanager ist.

Abbildung 5-9 illustriert das Verhältnis von Anwendungsgruppenname und Datentypdefinition. Das Feld PATH_PATTERN der Datentypdefinition stellt eine Verknüpfung zwischen einem eindeutigen Symbol und der Anwendungsgruppe her.

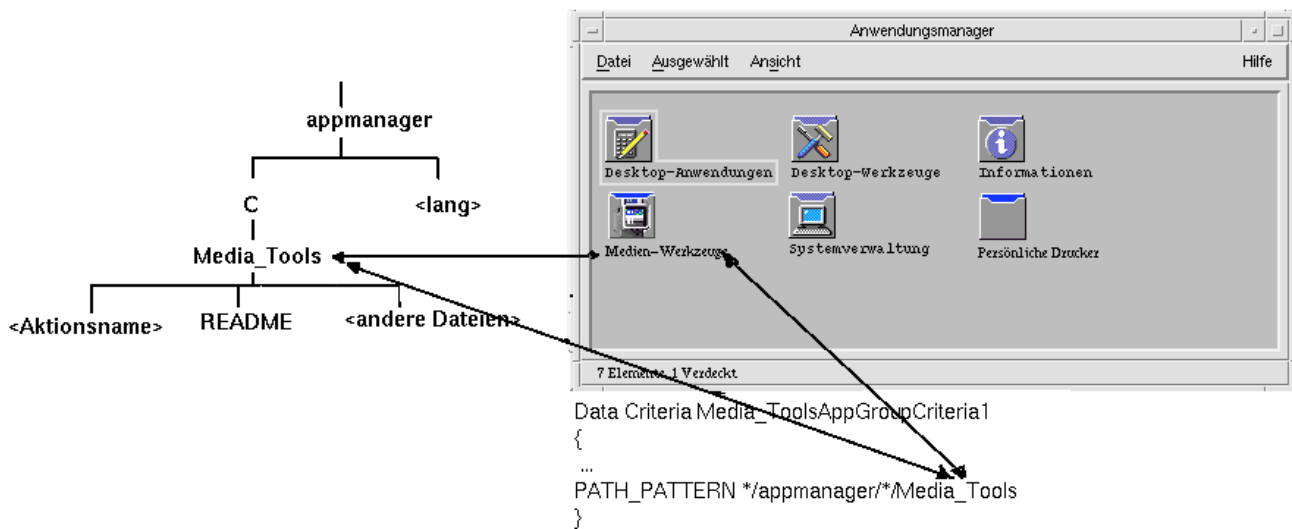


Abbildung 5-9 Einer Anwendungsgruppe ein eindeutiges Symbol zuweisen

Sie sollten außerdem die Aktionen "Öffnen" und "Drucken" für den Datentyp der Anwendungsgruppe erstellen:

```

ACTION Open
{
    ARG_TYPE    MedienwerkzeugeAppGroup
    TYPE        MAP
    MAP_ACTION  OpenAppGroup
}

```

```

ACTION Print
{
    ARG_TYPE    MedienwerkzeugeAppGroup
    TYPE        MAP
    MAP_ACTION  PrintAppGroup
}

```

Die Aktionen `OpenAppGroup` und `PrintAppGroup` sind integrierte Aktionen, deren Definition sich in `/usr/dt/appconfig/types/sprache/dtappman.dt` befindet.

Inhalt der Anwendungsgruppe erstellen

Das wichtigste Element in einer Anwendungsgruppe ist ein Symbol, mit dem die Anwendung gestartet wird (ein Aktionssymbol). Enthält die Anwendungsgruppe mehrere Anwendungen, gibt es normalerweise ein Symbol für jede Anwendung.

Zusätzlich zu einem oder mehreren Aktionssymbolen kann die Anwendungsgruppe folgendes enthalten:

- README-Dateien
- Beispiel-Datendateien
- Schablonen
- Ein Symbol zum Anzeigen der Hilfe
- Eine Hilfeseite
- Ein spezialisiertes Bedienfeld

Die Anwendungsgruppe kann Unterverzeichnisse enthalten.

Die Aktionsdatei (Anwendungssymbol) erstellen

Die Anwendungsgruppe sollte ein Symbol enthalten, das die Anwendung startet. Enthält die Gruppe mehrere Anwendungen, sollte für jede Anwendung ein Symbol

vorhanden sein. Diese Symbole werden *Anwendungssymbole* oder *Aktionssymbole* genannt, da sie die zugrundeliegende Aktion repräsentieren.

Eine Aktionsdatei ist eine ausführbare Datei mit dem Namen der auszuführenden Aktion:

```
app_root/dt/appconfig/appmanager/  
Anwendungsgruppenname/aktion_name
```

Diese Datei wird *Aktionsdatei* genannt, da sie eine visuelle Repräsentation der zugrundeliegenden Aktion erstellt.

Haben Sie beispielsweise eine Aktion namens "BesterTextEditor" für die Anwendung BesterTextEditor erstellt, wird eine ausführbare Datei mit dem Namen "BesterTextEditor" benötigt. Im Dateimanager und im Anwendungsmanager verwendet die Aktionsdatei die in der Aktionsdefinition angegebene Symboldatei.

Abbildung 5-10 illustriert die Beziehung zwischen der Aktionsdefinition, der Aktionsdatei und dem sichtbaren Eintrag im Fenster des Anwendungsmanagers.

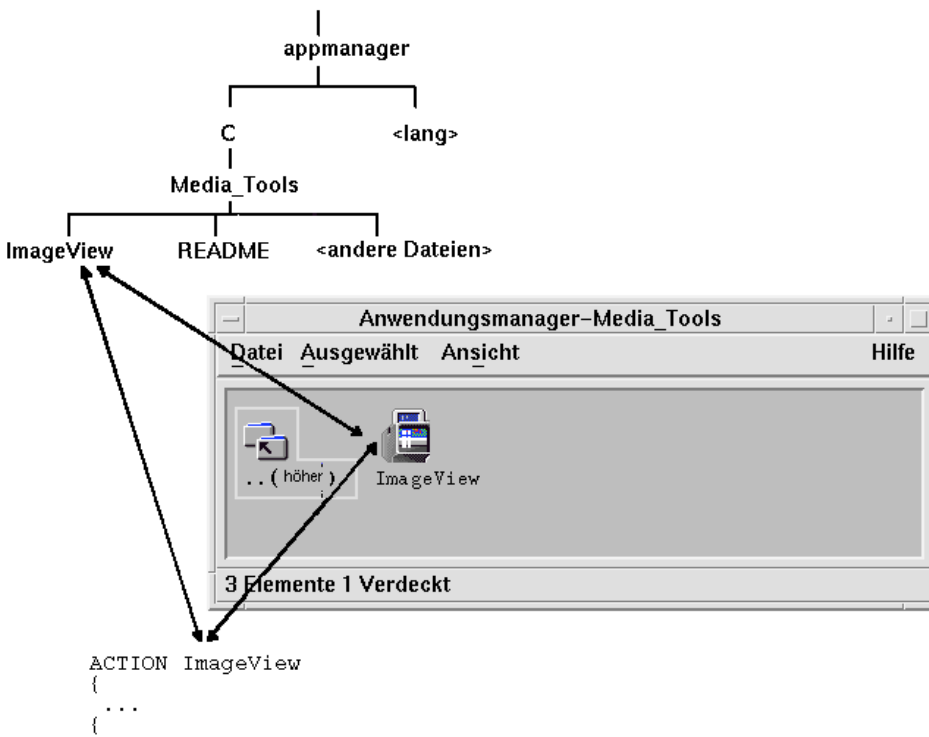


Abbildung 5-10 Das Anwendungssymbol ist eine Datei in der Anwendungsgruppe

Readme-Dateien

Das Desktop stellt den Datentyp README für die README-Dateien von Anwendungen zur Verfügung. Verwenden Sie die folgenden Namenskonventionen:

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

Ein spezialisiertes Bedienelement im Bedienfeld erstellen

In den meisten Fällen ist es nicht notwendig, eine Definition für das Bedienelement im Bedienfeld bereitzustellen. Der Benutzer kann die Anwendung dem Bedienfeld hinzufügen, indem er das Aktionssymbol auf dem Bedienelement "Symbol installieren" eines untergeordneten Bedienfeldes ablegt.

Sie können eine Konfigurationsdatei für ein Bedienfeld mit einer Elementdefinition für die Anwendung erstellen, damit Benutzer ein Bedienelement installieren können, das andere Eigenschaften als das Aktionssymbol aufweist - beispielsweise ein Bedienelement zur Überwachung einer Datei, dessen Erscheinungsbild sich ändert, wenn die betreffende Datei geändert wird.

Die Konfigurationsdateien für Bedienfelder werden im Verzeichnis `app_root/dt/appconfig/types/ sprache` gespeichert. Die Namenskonvention lautet `name.fp`.

Wenn Sie eine Konfigurationsdatei mit einem Bedienelement bereitstellen, kann der Benutzer das Bedienelement in ein untergeordnetes Bedienfeld einfügen, indem er die `*.fp`-Datei auf dem Bedienelement "Symbol installieren" des untergeordneten Bedienfeldes ablegt.

Beispielsweise kann die folgende Definition in eine Bedienfeldkonfigurationsdatei in der Anwendungsgruppe eingefügt werden. Übergibt der Benutzer diese Datei dem Bedienelement "Symbol installieren" eines untergeordneten Bedienfeldes, wird im Bedienfeld ein Bedienelement erstellt, das eine Instanz der Anwendung 'BesterTextEditor' ausführt. Wurde 'BesterTextEditor' bereits gestartet, wird das Fenster im aktuellen Arbeitsbereich in den Vordergrund gebracht.

```
CONTROL BestTextEditorControl
{
    TYPE        icon
    ICON        BTEFPanel
    PUSH_RECALL    True
}
```



```
CLIENT_NAME      BTEd
PUSH_ACTION      BTEditor
DROP_ACTION      BTEditor
HELP_STRING      Startet die Anwendung BesterTextEditor.
}
```

Weitere Informationen zum Erstellen von Bedienfeldkonfigurationsdateien befinden sich in:

- Kapitel 15
- Hilfeseite `dtfpfile(4)`

Schritt 8: Anwendungen mit `dtappintegrate` registrieren

Hinweis - Ein Beispiel zur Registrierung einer Anwendung finden Sie in Schritt 8 des Abschnitts „Beispiel zur Erstellung eines Registrierungspakets “ auf Seite 84.

Sobald ein Registrierungspaket in einem Anwendungsstammverzeichnis erstellt wurde, kann die Anwendungsregistrierung durchgeführt werden.

Bei der Anwendungsregistrierung werden Verknüpfungen zwischen dem Registrierungspaket und den Verzeichnissen im Suchpfad des Desktop eingerichtet (siehe „Wie `dtappintegrate` Anwendungen integriert“ auf Seite 82).

Anwendungen mit `dtappintegrate` registrieren

Ist die Anwendung desktop-fähig, wird `dtappintegrate` als letzter Schritt des Installationsprozesses automatisch ausgeführt. Erfolgt die Ausführung nicht automatisch, oder haben Sie Konfigurationsdateien zur Integration einer nicht desktop-fähigen Anwendung erstellt, können Sie `dtappintegrate` manuell ausführen.

1. **Melden Sie sich als Benutzer “root” an.**
2. **Führen Sie den folgenden Befehl aus:**

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

app_root ist das Desktop-Anwendungsstammverzeichnis. Weitere Informationen finden Sie auf der Hilfeseite `dtappintegrate(1)`.

3. Öffnen Sie die Anwendungsgruppe “Desktop-Werkzeuge”, und klicken Sie doppelt auf “Anwendungen erneut laden”.
4. Prüfen Sie, ob die Anwendung richtig registriert wurde:
 - a. Zeigen Sie die oberste Ebene des Anwendungsmanagers an. Die neue Anwendungsgruppe sollte im Anwendungsmanager sichtbar sein.
 - b. Öffnen Sie die Anwendungsgruppe, und klicken Sie doppelt auf das Aktionssymbol.

Syntax und Optionen für dtappintegrate

```
dtappintegrate -s
app_root [-t zielpfad ] [-l
sprache ] [-u]
```

- s <i>app_root</i>	Erforderlicher Parameter, der das Anwendungsstammverzeichnis angibt, in dem die Anwendung installiert wurde.
-t <i>zielpfad</i>	Optionaler Parameter mit dem Systemverzeichnis als Standardwert: <code>/etc/dt/appconfig</code> . Bestimmt die Position, zu der die Desktop-Konfigurationsdateien eine Verknüpfung herstellen. Sie müssen eines der Verzeichnisse im Anwendungssuchpfad verwenden.
-l <i>sprache</i>	Optionaler Parameter, der standardmäßig alle Sprachen berücksichtigt. Gibt die zu integrierenden sprachabhängigen Konfigurationsdateien an.
- u	Optionaler Parameter, der die Integration einer Anwendung rückgängig macht und alle während der Integration eingerichteten Verknüpfungen entfernt.

Wie dtappintegrate Anwendungen integriert

dtappintegrate richtet Verknüpfungen zwischen den installierten Dateien und den Verzeichnissen ein, in denen das Desktop die Konfigurationsdateien sucht.

Aktionen und Datentypen

dtappintegrate erstellt symbolische Verknüpfungen zwischen den Definitionsdateien für Aktionen und Datentypen im Registrierungspaket und dem für das System definierten Verzeichnis im Suchpfad der Aktionsdatenbankhilfe. Hierzu werden Verknüpfungen eingerichtet zwischen

```
app_root/dt/appconfig/types/  
sprache/*.dt
```

und

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

Hilfeinformationsdateien

dtappintegrate erstellt symbolische Verknüpfungen zwischen den Hilfedateien im Registrierungs paket und dem für das System definierten Verzeichnis im Hilfesuchpfad. Hierzu werden Verknüpfungen eingerichtet zwischen

```
app_root/dt/appconfig/help/  
sprache/hilfedatei.sdl
```

und

```
/etc/dt/appconfig/help/sprache.sdl
```

Symboldateien

dtappintegrate erstellt symbolische Verknüpfungen zwischen den Symboldateien im Registrierungs paket und dem für das System definierten Verzeichnis im Symbolsuchpfad. Hierzu werden Verknüpfungen erstellt zwischen

```
app_root/dt/appconfig/icons/  
sprache/symboldateien
```

und

```
/etc/dt/appconfig/icons/sprache
```

Anwendungsgruppe

Um die Anwendungsgruppe für die Anwendung in die oberste Ebene des Anwendungsmanagers einzufügen, erstellt dtappintegrate eine Verknüpfung zwischen dem Anwendungsgruppenverzeichnis im Registrierungs paket und dem für das System definierten Verzeichnis im Anwendungssuchpfad. Dies erfolgt durch Erstellung einer Verknüpfung zwischen

```
app_root/dt/appconfig/appmanager/  
sprache/anwendungsgruppenname
```

und

```
/etc/dt/appconfig/appmanager/sprache/  
anwendungsgruppenname
```

Beispiel zur Erstellung eines Registrierungs pakets

Mit den folgenden Schritten wird ein Registrierungs paket für eine bereits vorhandene, nicht desktop-fähige Anwendung mit dem Namen "BesterTextEditor" erstellt.

Erforderliche Informationen zu "BesterTextEditor"

Im Beispiel werden die folgenden Daten zur Anwendung BesterTextEditor vorausgesetzt:

- Die Anwendung wurde im Verzeichnis `/usr/BTE` installiert.
- Die Sitzungssprache des Benutzers entspricht dem Standardwert `C`.
- Die Befehlszeile zum Starten von BesterTextEditor lautet:

```
BTEd {dateiname}
```

dateiname ist der Name der Datendatei, die im neuen Fenster geöffnet werden soll. BesterTextEditor erstellt ein eigenes Fenster, wird also nicht in einem Terminalemulationsfenster ausgeführt.

- BesterTextEditor erstellt und verwendet zwei Typen von Datendateien:
 - Dokumentationsdateien. Für diese Dateien gilt die Namenskonvention `*.bte`. BesterTextEditor stellt eine Befehlszeilenschnittstelle zum Drucken der `.bte`-Datendateien bereit. Die Syntax dieses Befehls lautet:

```
BTEPrint [-d ziel] [-s]  
dateiname
```

Es gilt:

`-d ziel` - gibt den Drucker an.

`-s` - der Ausdruck erfolgt, ohne daß zuvor das Dialogfenster "Drucken" angezeigt wird.

dateiname - gibt die zu druckende Datei an.

- Schablonendateien. Für diese Dateien gilt die Namenskonvention `*.tpl`. Schablonendateien können nicht gedruckt werden.
- Die bereits vorhandenen, nicht desktop-fähigen `app-defaults`-Dateien für "BesterTextEditor" enthalten Ressourcen für Schnittstellenschriftarten sowie Vordergrund- und Hintergrundfarben.

- Ein Online-Hilfeband für 'BesterTextEditor' wurde mit dem Help Developer's Kit des Desktop erstellt. Für die Erstellung des Online-Hilfebands wurden folgende Quellendateien verwendet:

.../BTEHelp.htg

.../graphics/BTE1.xwd

.../graphics/BTE2.xwd

Die Datei .../BTEHelp.sdl wurde generiert.

Schritte zum Registrieren von "BesterTextEditor"

Mit den folgenden Schritten wird BesterTextEditor registriert.

1. Ändern Sie die Schriftarten- und Farbressourcen.

In der Datei app-defaults von BesterTextEditor müssen Ressourcen mit folgenden Einstellungen entfernt werden:

- Schriftarten für Text
- Farben für Vorder- und Hintergrund

2. Erstellen Sie das Anwendungsstammverzeichnis.

Erstellen Sie das folgende Verzeichnis:

```
/desktop_approots/BTE
```

Wird eine bereits vorhandene Anwendung integriert, sollte das Anwendungsstammverzeichnis nicht im Installationsverzeichnis eingerichtet werden, da andernfalls die erstellten Konfigurationsdateien bei einer Aktualisierung der Anwendung möglicherweise gelöscht werden.

3. Erstellen Sie die Registrierungspaketverzeichnisse.

Erstellen Sie die folgenden Verzeichnisse:

```
/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/BesterTextEditor
```

4. Erstellen Sie Aktionen und Datentypen für die Anwendung.

a. Erstellen Sie die Konfigurationsdatei für die Definitionen für Aktionen und Datentypen:

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

b. Die Aktionsdefinition für die Ausführung von "BesterTextEditor":

```
ACTION BTEditor
{
    WINDOW_TYPE    NO_STUDIO

    ICON          BTERun

    DESCRIPTION    Klicken Sie doppelt auf dieses Symbol, oder legen \
                  Sie eine BTE-Datendatei ab, um \
                  BesterTextEditor auszuführen.

    EXEC_STRING    /usr/BTE/BTEd %Arg_1%
}

```

c. Erstellen Sie den Datentyp für *.bte-Dateien:

```
DATA_ATTRIBUTES BTEDataFile
{
    DESCRIPTION    BesterTextEditor-Datendatei.

    ICON          BTEData

    ACTIONS        Open,Print
}

DATA_CRITERIA BTEDataFileCriterial
{
    DATA_ATTRIBUTES_NAME    BTEDataFile

    NAME_PATTERN              *.bte

    MODE                      f
}

```

d. Erstellen Sie den Datentyp für *.tpl-Dateien:

```
DATA_ATTRIBUTES BTETemplateFile
{
    DESCRIPTION    BesterTextEditor-Schablonendatei.
    ICON          BTETempl
    ACTIONS       Open
}

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

e. Erstellen Sie die Aktion "Öffnen" für *.bte-Dateien:

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

f. Erstellen Sie die Aktion "Drucken" für *.bte -Dateien:

Nachstehend finden Sie einfache "Drucken"-Aktionen, die Datendateien drucken. Diese Aktionen setzen voraus, daß die Umgebungsvariable LPDEST einen Wert enthält. Die Druckoption `-s` wird ignoriert. (Ist LPDEST nicht gesetzt, kann die Aktion fehlschlagen.)

```
ACTION Print
{
    ARG_TYPE    BTEDataFile
```

```

        TYPE      MAP
        MAP_ACTION BTEPrintData
    }

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

```

Nachstehend finden Sie eine andere Version der Aktion BTEPrintData und das zugehörige Skript. Diese werden verwendet, wenn LPDEST nicht gesetzt ist oder das Dialogfenster "Drucken" nicht angezeigt werden soll.

```

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

```

Das Skript /usr/BTE/bin/BTEenvprint hat folgenden Inhalt:

```

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

```

g. Erstellen Sie die Aktion "Öffnen" für *.tpl-Dateien:

```

ACTION Open
{
    ARG_TYPE    BTETemplateFile
}

```



```

TYPE      MAP

MAP_ACTION BTEditor

}

```

h. Erstellen Sie die Aktion “Drucken” für *.tpl -Dateien:

```

ACTION Print

{

  ARG_TYPES  BTETemplateFile

  TYPE      MAP

  MAP_ACTION NoPrint

}

```

NoPrint ist eine integrierte Aktion, die ein Dialogfenster mit dem Hinweis anzeigt, daß die Datei nicht gedruckt werden kann.

5. Fügen Sie die Hilfedateien in das Registrierungspaket ein.

a. Kopieren Sie die Hilfedateien in die folgenden Verzeichnisse:

```

/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. Erstellen Sie die folgende Datei:

```

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

```

Fügen Sie die folgende Aktionsdefinition in die Datei ein:

```

ACTION BTEHelp

{

  WINDOW_TYPE    NO_STDIO

  EXEC_STRING     /usr/dt/bin/dthelpview -helpVolume \

                  BTEHelp.sdl

  DESCRIPTION     Öffnet den Hilfeband BesterTextEditor.
}

```

}

6. Erstellen Sie die Symbole für die Anwendung.

Verwenden Sie den Symboleditor, um die Symbole zu erstellen. Tabelle 5-3 enthält Hinweise zu den möglichen Größen.

TABELLE 5-3 Richtlinien zu Symbolgrößen

Name	Größe
<i>basisname.t.pm</i>	16 x 16
<i>basisname.m.pm</i>	32 x 32
<i>basisname.l.pm</i>	48 x 48

Erstellen Sie die folgenden Symboldateien im Verzeichnis

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

- Symbole zur Repräsentation der Aktion, die die Anwendung startet:
`BTERun.t.pm`, `BTERun.m.pm`, `BTERun.l.pm`
- Symbole zur Repräsentation der *.bte-Dateien: `BTEData.t.pm`,
`BTEData.m.pm`,
- Symbole zur Repräsentation der *.tpl-Dateien: `BTETempl.t.pm`,
`BTETempl.m.pm`
- Symbole zur Repräsentation der Anwendungsgruppe (vgl. Schritt 7):
`BTEApp.t.pm`, `BTEApp.m.pm`

7. Erstellen Sie die Anwendungsgruppe.

a. Erstellen Sie das Verzeichnis, wenn dies noch nicht geschehen ist.

`/desktop_approots/BTE/dt/appconfig/appmanager/C/BesterTextEditor`

b. Dieser Schritt ist optional. Er stellt ein eindeutiges Symbol für die Anwendungsgruppe zur Verfügung, indem ein Datentyp und zugehörige

Aktionen für die Anwendungsgruppe erstellt werden. Wird dieser Schritt nicht ausgeführt, verwendet die Anwendungsgruppe das Standardsymbol. Fügen Sie die folgenden Definitionen von Datentypen und Aktionen in die Datei /desktop_approots/BTE/dt/appconfig/types/C/BTE.dt ein. Der Datentyp gibt das von der Anwendungsgruppe BesterTextEditor zu verwendende Symbol an. Die Aktionen "Öffnen" und "Drucken" entsprechen in ihrer Funktionalität den Aktionen der Anwendungsgruppe.

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

DATA_CRITERIA BesterTextEditorAppGroupCriterial
{
    DATA_ATTRIBUTES_NAME BesterTextEditorAppGroup
    MODE                 d
    PATH_PATTERN          */appmanager/*/BesterTextEditor
}

ACTION Open
{
    ARG_TYPE      BesterTextEditorAppGroup
    TYPE          MAP
    MAP_ACTION    OpenAppGroup
}

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

- c. **Erstellen Sie in der Anwendungsgruppe ein Symbol zum Starten der Anwendung. Erstellen Sie zu diesem Zweck die Datei:**

```
/desktop_approots/BTE/dt/appconfig/appmanager/C \
```

```
/BesterTextEditor/BTEditor
```

als ausführbare Datei.

- d. **Erstellen Sie in der Anwendungsgruppe die Aktionsdatei, mit der der Hilfeband geöffnet wird. Erstellen Sie zu diesem Zweck die Datei:**

```
/desktop_approots/BTE/dt/appconfig/appmanager/C \
```

```
/BesterTextEditor/BTEHelp
```

als ausführbare Datei.

- e. **Fügen Sie weitere Dateien in die Anwendungsgruppe ein, beispielsweise README-Dateien, Beispieldaten und Schablonendateien.**

8. **Registrieren Sie die Anwendung.**

Führen Sie folgende Schritte in einem Terminalemulationsfenster durch:

- a. **Melden Sie sich als Benutzer "root" an.**

- b. **Führen Sie folgenden Befehl aus:**

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

- c. **Öffnen Sie die Anwendungsgruppe "Desktop-Werkzeuge", und klicken Sie doppelt auf "Anwendungen erneut laden".**

Hinweise zur Konfiguration

Dieses Kapitel enthält verschiedene Themen zur Einrichtung und Verwaltung des Systems.

- „Verzeichnisstruktur des Solaris CDE“ auf Seite 93
- „Zentrale Konfigurationsdateien“ auf Seite 95
- „Login Server starten“ auf Seite 96
- „Ein installiertes CDE von einer anderen Workstation- oder Netzwerkserver-Installationsposition einhängen“ auf Seite 99
- „Den Desktop für mehrere Bildschirme konfigurieren“ auf Seite 100
- „Desktops im Netzwerkbetrieb“ auf Seite 102
- „XTerminals verwenden“ auf Seite 103
-
- „Workstations als XTerminals einsetzen“ auf Seite 105
- „Spezielle CDE-Konfigurationen“ auf Seite 107

Verzeichnisstruktur des Solaris CDE

Dieser Abschnitt beschreibt die Hauptverzeichnisse der Desktop-Umgebung.

/usr/dt

Dieses Verzeichnis ist das Installationsverzeichnis des Solaris CDE. Es kann außerdem als Einhängpunkt für einen entfernten Dateiserver dienen. Tabelle 6-1 beschreibt die Unterverzeichnisse von /usr/dt.

TABELLE 6-1 Unterverzeichnisse von /usr/dt

Unterverzeichnis	Inhalt
/bin	SCDE -Anwendungen und -Hilfsprogramme
/lib	gemeinsam genutzte SCDE-Laufzeitbibliotheken
/config	Standard-Systemkonfigurationsdateien
/man	[Optional] Handbuchseiten
/app-defaults	Standard-Anwendungsressourcen
/appconfig	Standard-Anwendungssymbole, -Typen und -Aktionen
/examples	[Optional] CDE Code- und Programmbeispiele
/include	[Optional] Include-Dateien für Entwickler
/palettes	Farbpaletten
/share	CDE AnswerBook-Dokumentation und Standard-Hintergrund-Ablagen

/etc/dt

Dieses Verzeichnis enthält benutzerspezifisch angepaßte Konfigurationsdateien für die jeweilige Workstation. Die Dateien bieten für die Anpassung der Umgebung die folgenden Möglichkeiten:

- Optionen für die X-Server—Konfiguration einstellen.
- Mehrere Bildschirme verwenden.
- Workstation-Aktionsdateien, Datentypen, Symbole und Schriftarten anpassen.

/var/dt

In diesem Verzeichnis werden temporäre Dateien abgelegt, die von Solaris CDE—Anwendungen benötigt werden, beispielsweise von Login Manager und Anwendungsmanager.

Home-Verzeichnis

In diesem Verzeichnis befinden sich benutzerspezifische Dateien, die mit der individuellen Desktop-Einrichtung in Zusammenhang stehen. Dazu gehören Anwendungen, Farbtabelle, Fehlerprotokolle sowie Änderungen am Menü 'Arbeitsbereich' und am vorderen Steuerbereich.

Zentrale Konfigurationsdateien

An vielen Konfigurationsvorgängen der Desktop-Umgebung sind mehrere Dateien beteiligt. Einige Beispiele:

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

Xconfig

Xconfig ist die Hauptkonfigurationsdatei, die von `dtlogin` verwendet wird. Darin sind Anmelderessourcen und die Verzeichnispositionen zusätzlicher Dateien festgelegt, die von `dtlogin` benötigt werden. Das folgende ist ein Auszug aus einer standardmäßigen Datei Xconfig:

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

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

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

Informationen zu dieser Datei finden Sie in Anhang A." Weitere erläuternde Kommentare enthält die Datei selbst.

Xservers

dtlogin verwendet ebenso wie XDM (X Display Manager), auf dem das Modul basiert, die Datei `Xservers` zur Startkonfiguration des lokalen Xserver. Per Voreinstellung hat die letzte Zeile in dieser Datei die Form:

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

- `:0` definiert den Xserver—Bildschirm als `<localhost:0>`.
- `local` bedeutet, daß der neue X11-Server lokal gestartet werden soll.

Hinweis - Ersetzen Sie `local` durch `foreign`, wenn der Anschluß an einen laufenden X11-Server erfolgen soll.

- `console` bedeutet, daß die bei der Option 'Befehlszeilenanmeldung' angegebene Escape-Sequenz auf `/dev/console` verweist.
- `/usr/openwin/bin/Xsun` ist der Pfad zum X11-Server.

Hinweis - In den Umgebungen Solaris CDE und OpenWindows[™] läuft derselbe X11-Server.

Weitere Informationen zu dieser Datei finden Sie auf der Handbuchseite `dtlogin(1)`. Die Datei selbst enthält ebenfalls erläuternde Kommentare.

Login Server starten

Der Login Server wird normalerweise automatisch beim Hochfahren des Systems gestartet. Alternativ dazu kann er auch von einer Befehlszeile aus gestartet werden, wenn Sie sich als Root-Benutzer angemeldet haben.

Wenn der Login Server beim Hochfahren des Systems gestartet werden soll, gehen Sie folgendermaßen vor:

- ◆ **Geben Sie den folgenden Befehl ein, und drücken Sie die Eingabetaste:**

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

Damit wird eine Datei `S99dtlogin` in das Verzeichnis `/etc/rc2.d` eingefügt. Beim nächsten Hochfahren des Systems wird der Login Server nun automatisch gestartet.

Wenn der automatische Start deaktiviert werden soll, gehen Sie folgendermaßen vor:

- ◆ **Geben Sie den folgenden Befehl ein, und drücken Sie die Eingabetaste:**


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

So starten Sie den Login Server von einer Befehlszeile aus:

- ◆ **Geben Sie den folgenden Befehl ein, und drücken Sie die Eingabetaste:**

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

Hinweis - Der Start des Login Servers sollte vorzugsweise nur zum Test einer Konfiguration erfolgen. In der Praxis ist der automatische Start beim Hochfahren des Systems sinnvoller.

So brechen Sie die Ausführung von Xserver und des gesamten Solaris CDE—Desktop ab:

- ◆ **Geben Sie den folgenden Befehl ein, und drücken Sie die Eingabetaste:**

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

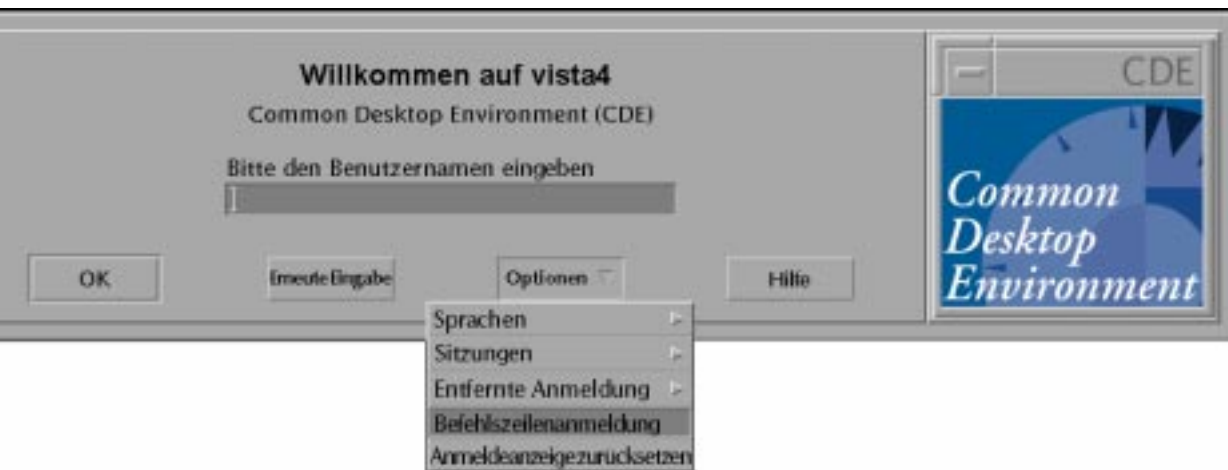
Weitere Informationen über das Desktop-Konfigurationsprogramm

dtconfig enthält **Anhang A**. Dort finden Sie eine Kopie der Handbuchseite

dtconfig.1.

So verlassen Sie den Solaris CDE—Anmeldebildschirm mit der Option 'Befehlszeilenanmeldung':

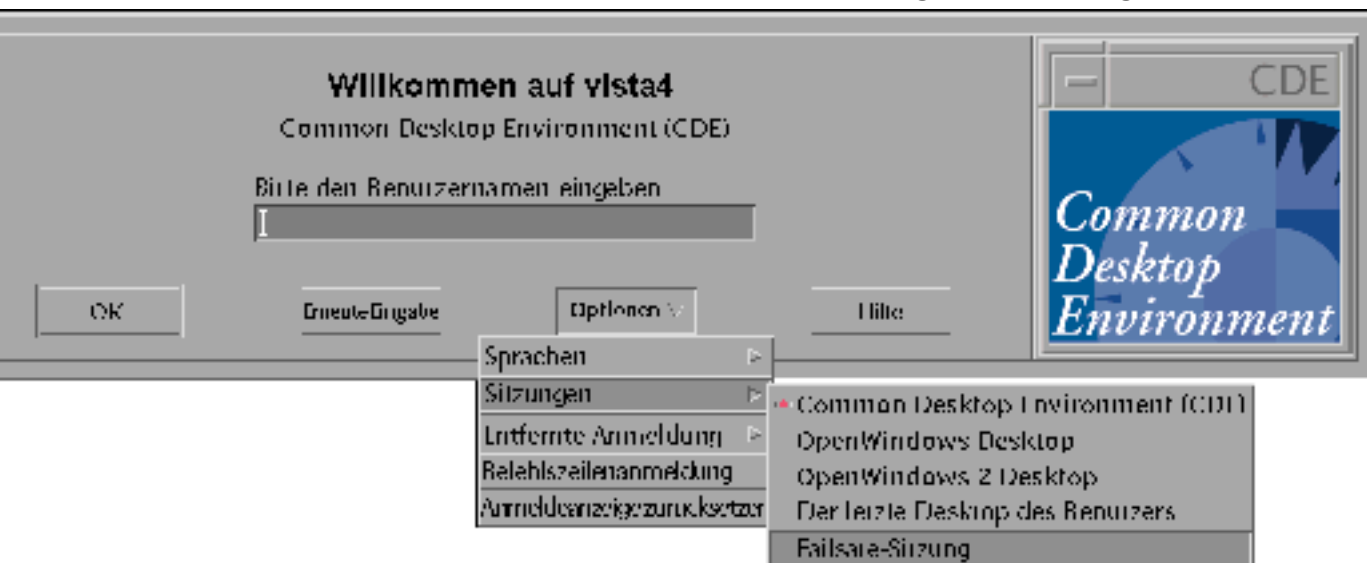
- ◆ **Wählen Sie im Solaris CDE-Anmeldebildschirm die Option 'Befehlszeilenanmeldung'. Sobald der Bildschirm gelöscht ist, aktivieren Sie die Anmeldeeingabeaufforderung mit der Eingabetaste.**



Hinweis - Der Desktop-Anmeldedämon startet automatisch den Anmeldebildschirm, wenn die Befehlszeilenanmeldung erfolgt ist.

So starten Sie ein einzelnes Fenster mit Terminal-Emulation:

- ◆ Wählen Sie im Anmeldebildschirm des Solaris CDE aus dem Menü 'Optionen' das Untermenü 'Session' und dann den Eintrag 'Failsafe-Sitzung'.



Wenn der Xserver weiter laufen soll, wählen Sie aus dem Untermenü 'Sitzungen' den Eintrag 'Failsafe-Sitzung'. Damit starten Sie ein einzelnes `xterm`-Fenster. Die Option 'Failsafe-Sitzung' ist immer verfügbar, auch dann wenn die Option 'Befehlszeilenanmeldung' nicht verfügbar ist.

Ein installiertes CDE von einer anderen Workstation- oder Netzwerkserver-Installationsposition einhängen

Das Einhängen eines installierten CDE kann von einer Installationsposition auf einer anderen Workstation oder einem anderen Netzwerkserver aus erfolgen. Dies bietet sich an, wenn lokal nicht genügend Speicherplatz zur Verfügung steht.

Hinweis - Weil sich die `/usr/dt`-Verzeichnisstruktur der Solaris-Versionen und der Client-Workstations unterscheidet (die SPARC-Verzeichnisstruktur ist bei Intel-Versionen anders aufgebaut), muß auf der Client-Workstation das passende NFS-Server-Abbild von `/usr/dt` eingehängt werden. Beispielsweise muß auf einem SPARC-System mit Solaris 2.5 die Struktur `/usr/dt` von einem anderen Solaris 2.5-System eingehängt werden, auf dem das CDE in `/usr/dt` installiert ist.

▼ Ein installiertes CDE einhängen

1. Hängen Sie ein `/usr/dt`-Verzeichnis von einer bereits installierten Workstation oder einem Netzwerkserver beim Verzeichnis `/usr/dt` auf Ihrer Workstation ein.

2. Geben Sie ein: `/usr/dt/bin/dtconfig -inetd`

3. Aktivieren Sie die Solaris-Desktop-Anmeldung:

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

4. Starten Sie die Workstation erneut.

Weitere Informationen zum Desktop-Konfigurationsprogramm `dtconfig` enthält Anhang A. Dort finden Sie eine Kopie der Handbuchseite `dtconfig.1`.

▼ Ein CDE-Verzeichnis aushängen

1. **Deaktivieren Sie die Solaris-Desktop-Anmeldung mit folgender Eingabe:**

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

2. **Geben Sie ein:** `/usr/dt/bin/dtconfig -inetd.ow`

3. **Hängen Sie `/usr/dt` aus.**

4. **Starten Sie die Workstation neu.**

Weitere Informationen über das Desktop-Konfigurationsprogramm `dtconfig` enthält Anhang A. Dort finden Sie eine Kopie der Handbuchseite `dtconfig.1`.

Den Desktop für mehrere Bildschirme konfigurieren

Bei der Standardanmeldung wird der Desktop auf einem einzigen Bildschirm gestartet. Durch Änderungen in der Datei `xconfig` kann erreicht werden, daß der Desktop auf mehreren Bildschirmen startet. Dazu müssen Sie als Root-Benutzer angemeldet sein.

Hinweis - Zur schnelleren Bearbeitung der Anmeldeinformationen starten Sie mit der Option 'Failsafe-Sitzung' nur ein einziges `xterm`-Fenster anstelle des gesamten Desktop.

▼ Den Desktop auf mehreren Bildschirmen starten

1. **Legen Sie mit dem folgenden Befehl eine Kopie der Datei `Xserver` an:**

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

Hinweis - `/etc/dt/config/Xservers` überschreibt die Datei `/usr/dt/config/Xservers`.

2. Bearbeiten Sie die Datei `/etc/dt/config/Xservers`, um zwei Frame-Puffer (Bildschirme) einzurichten:

a. Suchen Sie nach dieser Zeile:

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

b. Fügen Sie am Ende dieser Zeile zwei `-dev`-Optionen ein. Im folgenden Beispiel wird vorausgesetzt, daß diese mit `/dev/fb0` und `/dev/fb1` benannt sind. Fügen Sie das folgende hinzu (mit führendem Leerzeichen):

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

3. Wählen Sie 'Anmeldeanzeige zurücksetzen' aus dem Menü 'Optionen'.



Hinweis - Wenn das Solaris CDE auf mehreren Bildschirmen ausgeführt wird, erscheint der vordere Steuerbereich nur auf dem ersten Bildschirm. Zum Start weiterer Anwendungen verwenden Sie das Menü 'Programme'. Dieses Menü wird mit der Maustaste 3 geöffnet, während sich der Mauszeiger über einem freien Bereich des Desktop befindet. Wenn Sie mit einer 2-Tasten-Maus arbeiten, drücken Sie Maustaste 2.

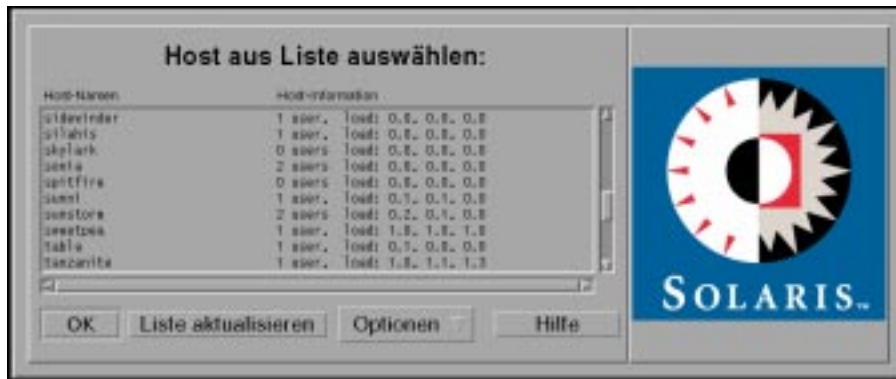
Desktops im Netzwerkbetrieb

Der Solaris CDE Login Manager ist netzwerkfähig. Per Voreinstellung reagieren die Anmeldebildschirme automatisch auf Anforderungen der Solaris CDE—Auswahl.

So öffnen Sie die Liste der entfernten Hosts:

- ◆ Wählen Sie 'Host aus Liste auswählen' aus dem Menü 'Entfernte Anmeldung' im Solaris CDE—Anmeldebildschirm.

Wenn Sie alternativ dazu die Auswahl unabhängig vom Anmeldebildschirm ausführen wollen, finden Sie Hinweise im Abschnitt „Workstations als XTerminals einsetzen“ auf Seite 105. Der folgende Bildschirm zeigt eine typische Auswahlliste mit verfügbaren Servern.



Sie können einen im Leerlauf befindlichen Server aus der Liste auswählen, um dessen Ressourcen zur Anmeldung beim Solaris CDE zu nutzen.

Jede Workstation im Netz, auf der der Solaris CDE—Login Manager läuft, kann sowohl einen lokalen Desktop-Benutzer als auch mehrere entfernte Benutzer (einschließlich XTerminal—Benutzer) unterstützen, indem die Workstation als ein Solaris CDE—Desktop—Server eingesetzt wird. (Siehe Abbildung 6-1.)

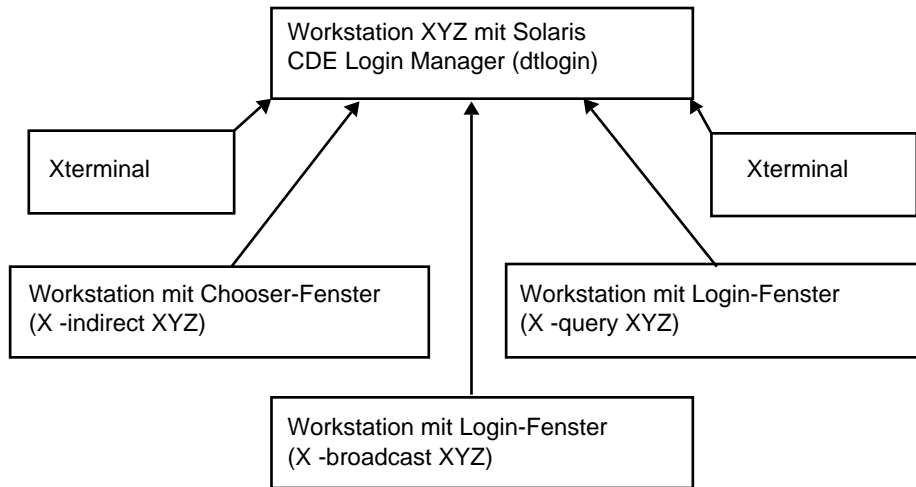


Abbildung 6-1 Desktops im Netzwerk

Hinweis - Abbildung 6-1 kann eine Umgebung mit kombinierter Hardware enthalten.

XTerminals verwenden

Jedes XTerminal, welches das XDM-Protokoll unterstützt, kann sich mit der Auswahl beim Solaris CDE anmelden. Hinweise zur Ausführung der Auswahl finden Sie unter „Workstations als XTerminals einsetzen“ auf Seite 105. Die SPARC Xterminal™-Softwareversion 2.0 (oder höher) arbeitet sehr gut mit dem Solaris CDE zusammen. Klicken Sie im XTerminal-Fenster 'Steuerungen' auf 'Eigenschaften', und wählen Sie 'Session' als Kategorie, um den Bildschirm zur Sitzungskonfiguration anzuzeigen, wie in Abbildung 6-2 gezeigt.

Abbildung 6-2 Bildschirm zur Sitzungskonfiguration

Wählen Sie keinen lokalen Fenstermanager sowie eine indirekte XDM-Verbindung zu dem entfernten System aus, auf dem das Solaris CDE läuft.

Schriftartenpfad und Sprachtreiber für die Anmeldung

Den bevorzugten Sprachtreiber wählen Sie während der Anmeldung beim CDE aus. Der folgende Anmeldebildschirm zeigt die Auswahl von 'Sprachen' im Menü 'Optionen'.



Auf Workstations werden Schriftartenpfade automatisch so gesetzt, daß sowohl die Schriftarten (und Aliase) der bevorzugten Sprache als auch des Standard-Systemsprachtreibers 'C' eingebunden werden. Bei XTerminals werden diese Schriftarten automatisch von einem X11-Schriftarten-Server bereitgestellt.

Workstations als XTerminals einsetzen

Wenn Sie mit älteren Workstations arbeiten oder mit Workstations, die nur 16 MByte oder weniger Arbeitsspeicher besitzen, dann können Sie diese als XTerminals einsetzen, wenn darauf die Software Solaris 2.4 oder höher installiert ist (Alternativ eignet sich auch Solaris 2.3 mit Xserver—Zusatzmodul).

Hinweis - Wenn die Schriftarten bei Ausführung der im folgenden beschriebenen Arbeitsgänge auf dem Bildschirm der Workstation nicht korrekt angezeigt werden, dann ist eventuell das Solaris CDE auf der Workstation nicht installiert. Mit dem Dienstprogramm `pkgadd` können Sie das benötigte Schriftarten-Package `SUNWdftt()` separat aus der Solaris CDE—Version 1.0.x hinzufügen.

▼ Mit Hilfe der Auswahl eine Host—CDE—Anmeldung bestimmen

1. Beenden Sie sämtliche aktiven Fenstersysteme.
2. Starten Sie die Auswahl. Geben Sie an der Konsolen-Befehlszeile die folgenden Befehle ein (mit `csH`):

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

```
/usr/openwin/bin/X -indirect CDE_login_host
```

Der Xserver startet und zeigt das Auswahlfenster vom Host an.

3. Wählen Sie den gewünschten Anmelde-Host aus dem Fenster.

▼ Einen bestimmten Host—CDE—Anmeldebildschirm verwenden

- ◆ Geben Sie den folgenden Befehl ein, und drücken Sie die Eingabetaste:

```
/usr/openwin/bin/X -once -query CDE_login_host
```

Xserver startet und zeigt den vom Host übernommenen Anmeldebildschirm an. Die Option `-once` bewirkt das Verlassen des Servers nach einem Anmelde- und Abmeldezyklus (Sitzung). Wenn die Option `-once` weggelassen wird, erscheint der Anmeldebildschirm nach der Abmeldung beim Solaris CDE erneut.

▼ Den ersten verfügbaren Host-Anmeldebildschirm verwenden

- ◆ Geben Sie den folgenden Befehl ein, und drücken Sie die Eingabetaste:

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

Damit wird der Xserver gestartet. Auf dem lokalen Teilnetz wird eine Anforderung für einen XDM-Anmeldeservice rundgesendet (XDM = X Display Manager). Wenn es im Teilnetz Systeme gibt, auf denen der Solaris CDE Login Manager läuft (oder ein anderes XDM-basiertes Anmeldefenster aktiv ist), dann plazierte der erste antwortende Host sein Anmeldefenster auf dem Desktop.

Spezielle CDE-Konfigurationen

Dieser Abschnitt behandelt spezielle Aspekte der Konfiguration.

Den Druck von E-Mail-Nachrichten anpassen

Der CDE-Mailer druckt Mail-Nachrichten mit der Aktion 'Drucken', die für den Datentyp `DTMAIL_FILE` definiert ist. Das vom OpenWindows—Mail—Programm definierte Druckerskript wird nicht verwendet. Wenn der Ablauf eines Druckvorgangs geändert werden soll, muß diese 'Drucken'-Aktion modifiziert werden.

Die 'Drucken'-Aktion umdefinieren

1. Erzeugen Sie mit einem Editor Ihrer Wahl die folgende Datei:

Home-Verzeichnis/.dt/types/dtmail.dt

2. Fügen Sie die folgenden Zeilen in die Datei ein:

```
#  
  
# Vordefinierte DRUCKEN-Aktion für Mailboxen außer Kraft setzen  
  
#  
  
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%;'

}

```

Hinweis - dtmailpr ist ein Druckfilter, der Anlagen entfernt und lediglich normalen Text übrig läßt. dtlp ist die voreingestellte CDE-Schnittstelle zu lp. %(File)Arg_1% ist die Datei, die gedruckt wird.

3. Ergänzen Sie EXEC_STRING mit den gewünschten Druckbefehlen.
4. Starten Sie den Mailer neu.

Neues Datumsformat für Kalender

Das erweiterbare Datenformat der Version 4 ist ein neues Format, das vom CDE-Kalender unterstützt wird. Der Kalender-Manager von OpenWindows kann dieses Format nicht verarbeiten. Wenn Sie häufiger zwischen den Plattformen OpenWindows und CDE wechseln müssen, sollten Sie Ihren Kalender nicht in das Datenformat der Version 4 konvertieren. Sie können allerdings mit dem Skript sdtcm_convert alle überflüssigen Elemente aus dem Kalender entfernen.

Hinweis - Das für Ihren Kalender gültige Format können Sie mit 'Info über Kalender' aus dem Menü 'Hilfe' ermitteln.

In anderen Fällen können Sie Ihren Kalender mit dem Skript `sdtcm_convert` in das Datenformat der Version 4 konvertieren.

Näheres zu dieser Datei finden Sie auf der Handbuchseite `sdtcm_convert(1)`.

Hinweis - Wenn Sie nur die CDE-Packages für Endanwender installiert haben, sind die Handbuchseiten nicht verfügbar und müssen separat mit dem Dienstprogramm `pkgadd` installiert werden. Weitere Informationen finden Sie im *Solaris Common Desktop Environment: Installationshandbuch*.

Das AnswerBook—Package über das Netz hinzufügen

Bei der Installation des Solaris CDE gibt es eine Option zur Installation des AnswerBook CDE—Package. Wenn Sie hier 'Ja' wählen, wird das AnswerBook—Package im Verzeichnis `/usr/dt/share/answerbooks/ sprache` plaziert (*sprache* steht hier für die länderspezifische Bezeichnung der lokalisierten AnswerBook—Fassung).

Das AnswerBook—Package belegt 109 MByte auf der Festplatte, weshalb Sie sich eventuell gegen eine Installation entscheiden. Sie haben dann zwei Alternativen:

- Sie können das Package auf `/usr/dt/share/answerbooks/ sprache` einhängen (oder von `/net/. . .` aus verknüpfen).
- Sie können die Umgebungsvariable `AB_CARDCATALOG` in die Datei *Home-Verzeichnis*/`.dtprofile` einfügen. Wenn auf Ihrem Netz beispielsweise eine exportierte Version vorliegt, fügen Sie die folgende Zeile in die Datei *Home-Verzeichnis*/`.dtprofile` ein:

```
export AB_CARDCATALOG=/net/hostname/usr/dt/share/answerbooks \  
/sprache/ab_cardcatalog
```

Das CDE—Environment außerhalb des CDE-Desktop einrichten

Umgebungsvariablen für die Solaris CDE—Anwendungen können auch außerhalb des CDE—Desktop gesetzt werden. Sie könnten sich beispielsweise bei einer entfernten Workstation anmelden und eine CDE—Anwendung auf Ihrer persönlichen Workstation aktivieren. Mit dem CDE-Dienstprogramm `dtsearchpath` können Sie dazu verschiedene CDE-Shell-Umgebungsvariablen einstellen.

Bei der Bourne— und der Korn—Shell geben Sie dazu folgenden Befehl ein:

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

Bei der C-Shell verwenden Sie folgenden Befehl:

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

Im Anschluß daran können Sie DISPLAY wieder auf Ihre lokale Workstation zurücksetzen und die CDE-Anwendung entfernt ausführen. Die Ergebnisse werden an der lokalen Workstation angezeigt.

Hinweis - In diesem Beispiel ist die Bourne— und Korn—Shell—Syntax für dtsearchpath Bestandteil der CDE—Beispielimplementierung, die auf allen Plattformen verfügbar ist, auf denen das CDE läuft. Die C-Shell-Option (-c) ist allerdings nur auf einer Sun-Plattform verfügbar.

Desktop—Umgebungsdatei

Der CDETM-Desktop bietet eine Laufzeitumgebung für die OpenWindows DeskSet —Anwendungen. Diese Funktionalität erfordert keine besondere Einrichtung des CDE. Einige Aspekte der Umgebungsconfiguration werden von der folgenden Datei definiert:

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

Wenn für eine bestimmte Workstation Umgebungsmerkmale hinzugefügt oder gelöscht werden sollen, kann diese Datei entweder direkt oder als Kopie bearbeitet werden. Eine Kopie sollte an folgender Position erzeugt werden:

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

Ein Beispiel für eine Hinzufügung wäre der Start der früheren virtuellen OpenWindows—Tastatur in Verbindung mit OpenWindows—DeskSet—Anwendungen. Da die meisten Anwendungen auf dem CDE—Desktop (und auch die Mehrheit der Sun-Anwender) dieses Dienstprogramm nicht benötigen, wurde es von der Standard-Startsequenz ausgenommen, um den Desktop-Startprozeß zu beschleunigen.

Die Kommentare in der Datei 0015.sun.env bieten weitere Informationen zum Start der optionalen, virtuellen OpenWindows —Tastatur (vkbd).

Disketten und CDs im Apple Macintosh Application Environment einsetzen

Wenn Sie das Apple Macintosh Application Environment (MAE) Version 1.0 installiert haben und über den OpenWindows-Dateimanager auf Wechselmedien zugreifen wollen (Disketten oder CD-ROMs), müssen Sie die Datei /etc/rmmount.conf anpassen. Damit stellen Sie sicher, daß MAE korrekt mit dem Dateimanager und den Wechselmedien zusammenarbeitet.

Die Datei /etc/rmmount.conf ändern

1. **Melden Sie sich als Root-Benutzer an. Geben Sie dazu erst `su` ein und dann Ihr Paßwort.**
2. **Wechseln Sie in das Verzeichnis /etc. Geben Sie den folgenden Befehl ein, und drücken Sie die Eingabetaste:**

```
# cd /etc
```

3. **Öffnen Sie die Datei /etc/rmmount.conf in einem Editor Ihrer Wahl.**
4. **Plazieren Sie die folgende Zeile unter #Actions am Ende der Liste:**

```
action floppy action_macfs.so
```

Ein Beispiel:

```
# more rmmount.conf

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

#

# Removable Media Mounter configuration file.

#

# File system identification

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

# Actions

action cdrom action_filemgr.so

action floppy action_filemgr.so
```

```
action floppy action_macfs.so
```

5. Speichern Sie die Datei, und verlassen Sie den Editor.

Wenn das MAE installiert ist, wird die Zeile `macfs action` automatisch in die Datei `rmmount.conf` eingefügt. Andernfalls wird die Zeile gelöscht.

Nach dieser Modifikation können Sie Wechselmedien im Dateimanager von OpenWindows oder CDE verwenden und in der MAE weiterhin mit Macintosh-Disketten arbeiten. Der Zugriff auf unformatierte, nicht lesbare oder DOS-Disketten ist nicht mehr möglich. Die Behandlung solcher Disketten wird an den OpenWindows—Dateimanager übergeben.

Wenn MAE auf nicht lesbare, unformatierte oder DOS-Disketten zugreifen können soll, muß die Zeile `action floppy action action_macfs.so` vor der Zeile `action floppy action_filemgr.so` eingefügt werden.

Desktop in einem Netzwerk konfigurieren

Das Desktop ist speziell für den Betrieb in einer stark vernetzten Umgebung ausgerichtet. Die Architektur des Desktops ermöglicht Systemadministratoren Computerressourcen im gesamten Netzwerk zu verteilen. Zu diesen Ressourcen gehören:

- Anwendungen.
- Datendateien für Anwendungen.
- Desktop-Session-Services (Desktop-Anwendungen wie zum Beispiel Anmeldemanager und Dateimanager).
- Hilfe-Services. Hilfedatendateien können auf einen zentralen Hilfe-Server gestellt werden.
- „Übersicht über das Desktop im Netzwerk“ auf Seite 114
- „Allgemeine Schritte für die Konfiguration eines Desktops im Netzwerk“ auf Seite 119
- „Netzwerk des Basisbetriebssystems für das Desktop konfigurieren“ auf Seite 119
- „Clients und Server für das Desktop konfigurieren“ auf Seite 123
- „Anwendungs-Services verwalten“ auf Seite 129

Übersicht über das Desktop im Netzwerk

Das Betriebssystem stellt eine Reihe von Services im Netzwerk, einschließlich verteilter Dateisysteme und der Ausführung auf fernen Systemen zur Verfügung. X-Server stellen zusätzliche Netzwerkfähigkeiten, wie den Zugriff auf ferne Bildschirme und Sicherheitsservices zur Verfügung.

Das Desktop legt eine Benutzerschnittstelle auf diese Netzwerkfunktionen. Ziel dieser Schnittstelle und der ihr zugrundeliegenden Architektur ist

- die einfachere Handhabung von Netzwerksystemen. Benutzer können Anwendungen ausführen und auf Datendateien zugreifen, ohne sich damit beschäftigen zu müssen, wo sich im Netzwerk die Anwendungen und Daten befinden.
- die einfachere Verwaltung von Netzwerksystemen. Das Desktop stellt Werkzeuge zur Anwendungsintegration sowie Netzwerksuchpfade zur Verfügung, die es Systemen leichter machen, ferne Daten und Anwendungen zu finden. Zusätzlich erleichtert der Dateinamenzuordnungsprozeß des Desktops die Verwaltung komplexer Netzwerke mit mehreren Servern.
- Flexibilität. Die Verwaltungsfunktionen des Desktops wurden zwar für bestimmte häufig vorkommende Netzwerksituationen entwickelt, aber das Desktop kann für viele andere angepaßte Netzwerkkonfigurationen eingerichtet werden.

Typen von Desktop Services im Netzwerk

Bei der Arbeit im Netzwerk kann ein Benutzer, der an einem bestimmten Bildschirm sitzt, auf verschiedene Computer-Services zugreifen, die anderen Systemen verfügbar gemacht werden. Dazu gehören:

- die Desktop-Session und ihre Anwendungen— zum Beispiel Arbeitsbereichsmanager und Dateimanager
- andere Anwendungen
- Datendateien

In der Netzwerkterminologie wird der Begriff *Server* verwendet, um ein System zu beschreiben, das einem oder mehreren Systemen Computerservices zur Verfügung stellt. Empfängt ein System Services von einem Server, wird es als ein *Client* dieses Servers bezeichnet.

In einem komplexen Netzwerk verwendet ein System möglicherweise Services, die sich auf einer Reihe von verschiedenen Systemen im gesamten Netzwerk befinden. Überdies kann ein System als ein bestimmter Servertyp fungieren (zum Beispiel als

Session-Server) und gleichzeitig ein Client (zum Beispiel eines Anwendungs-Servers) sein.

Typische Netzwerksituationen

Aus der Perspektive des Desktops enthält eine typische Netzwerkkonfiguration möglicherweise eine Kombination aus den folgenden wichtigen Komponenten:

TABELLE 7-1

Bildschirme	an denen der X-Server ausgeführt wird
Anmelde-/Session-Server	an denen die Desktop-Anwendungen (Anmeldemanager, Arbeitsbereichsmanager, usw.) ausgeführt werden
Anwendungs-Server	an denen andere Anwendungen ausgeführt werden
Datei-Server	in dem Anwendungen verwendete Daten gespeichert werden

Zu den am häufigsten vorkommenden Netzwerkkonfigurationen gehören Systeme, die auf einen Anwendungs-Server zugreifen. Abbildung 7-1 zeigt eine Datenstation, die einen Anwendungs-Server verwendet. Der X-Server und die Desktop-Session werden auf der Datenstation ausgeführt.

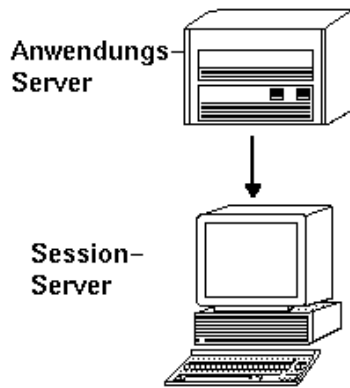


Abbildung 7-1 Anwendungs-Server stellen Services für die Desktop-Sessions zur Verfügung

Netzwerke verwenden Datei-Server häufig, um große Datenmengen zu steuern. Diese Daten werden von Anwendungen verwendet, die auf einem Anwendungs-Server ausgeführt werden oder von den Desktop-Anwendungen (zum Beispiel benötigt der Dateimanager Zugriff auf Datendateien, um diese im Dateimanagerfenster anzuzeigen).

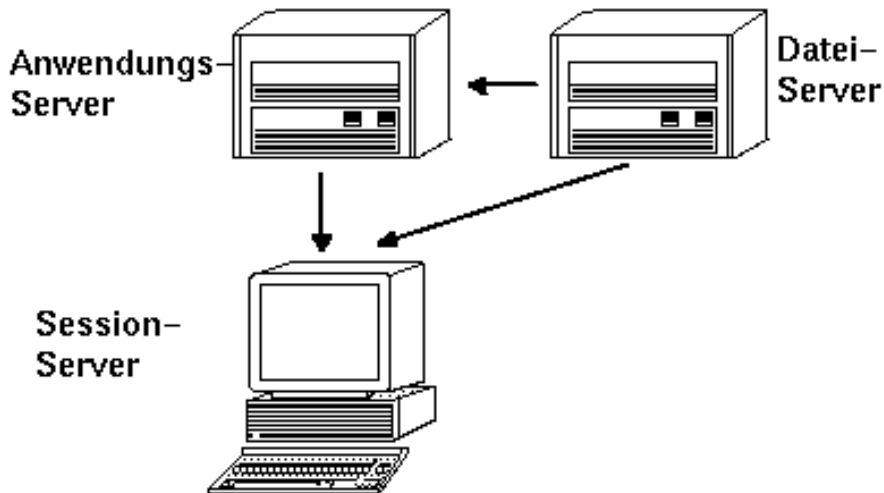


Abbildung 7-2 Datei-Server stellen Daten für Anwendungs-Server und Session-Server zur Verfügung

X-Stations führen den X-Server aus und erhalten Desktop-Session-Services von einem anderen System.

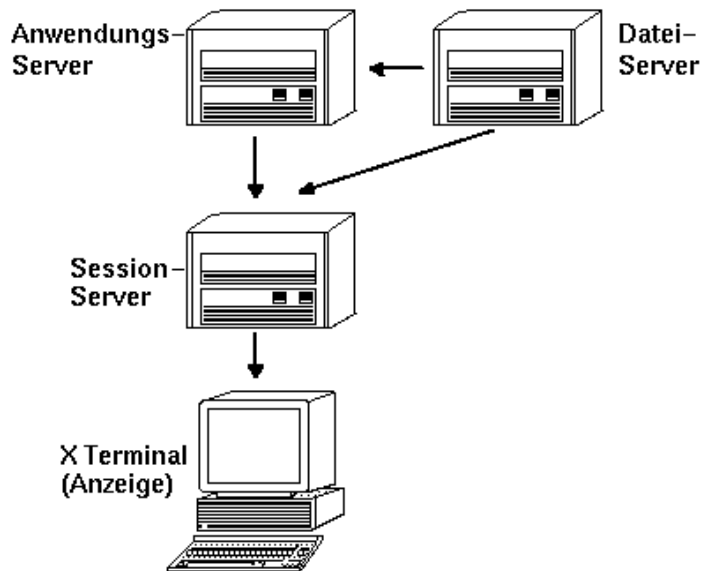


Abbildung 7-3 X-Stations erhalten Session-Services von einem Session-Server

Weitere Netzwerksituationen

Das Desktop ist flexibel und kann komplexere Netzwerkkonfigurationen unterstützen. Dazu gehört auch, daß verschiedene Services, zusätzlich zu Datei-Servern, für Anwendungs-Server verfügbar gemacht werden.

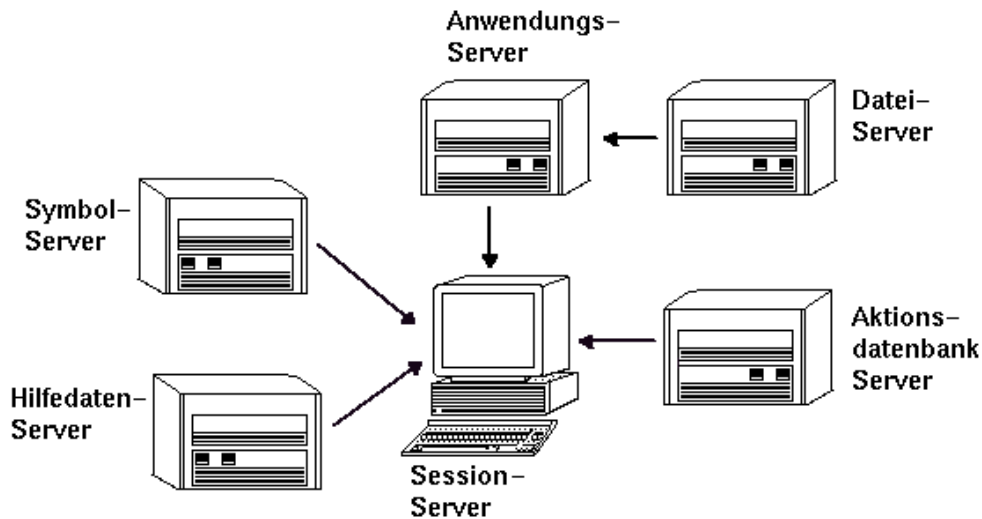


Abbildung 7-4 Services, die für einen Desktop-Anwendungs-Server erforderlich sind, können verteilt sein

Zusammenfassung—Servertypen

Bildschirm - Das System, auf dem X-Server ausgeführt wird.

Anmelde- und Session-Server - Das System, das die Desktop-Session (Anmeldemanager, Session-Manager, Festermanager, Dateimanager usw.) ausführt.

Anwendungs-Server - Ein System, auf dem eine Anwendung ausgeführt wird. Wird auch als *Ausführungs-Host* bezeichnet.

Datei-Server - Ein System, auf dem Datendateien für Anwendungen gespeichert sind.

Hilfe-Server - Ein System, auf dem Hilfedateien gespeichert sind.

(Aktions-)Datenbank-Server - Ein System, auf dem Dateien mit Definitionen für Aktionen und Datentypen gespeichert sind.

Symbol-Server - Ein System, auf dem Symboldateien gespeichert sind.

Das Netzwerk enthält möglicherweise noch zusätzliche Server, wie beispielsweise einen Kennwort-Server, einen Post-Server, einen Video-Server, usw.

Allgemeine Schritte für die Konfiguration eines Desktops im Netzwerk

Für die Konfiguration eines Desktops im Netzwerk sind drei grundlegende Schritte erforderlich:

1. Die Netzwerk-Services des Basisbetriebssystems konfigurieren.

Hierbei handelt es sich um die Netzwerk-Services des Betriebssystems, auf die das Desktop angewiesen ist. Weitere Informationen hierzu können dem Abschnitt „Netzwerk des Basisbetriebssystems für das Desktop konfigurieren“ auf Seite 119 entnommen werden.

2. Netzwerk-Software und -Services für das Desktop installieren und konfigurieren.

Hierbei handelt es sich um Services, die für das Desktop, unabhängig vom Typ des Clients oder Server-Systems, das eingerichtet wird, erforderlich sind. Weitere Informationen hierzu können dem Abschnitt „Clients und Server für das Desktop konfigurieren“ auf Seite 123 entnommen werden.

3. Den genauen Typ von Server oder Client konfigurieren.

Beispielweise erfordert die Konfiguration eines Anwendungs-Servers andere Schritte als die Konfiguration eines Datei-Servers. Weitere Informationen hierzu können dem Abschnitt „Anwendungs-Services verwalten“ auf Seite 129 entnommen werden.

Netzwerk des Basisbetriebssystems für das Desktop konfigurieren

Für das Desktop ist folgende Basisnetzwerkkonfiguration erforderlich:

- Benutzer müssen über ein Anmeldeabrechnungskonto am Session-Server und auf jedem System, das Desktop-Services für den Session-Server zur Verfügung stellt, verfügen. Dem Benutzer muß an allen Client- und Server-Systemen dieselbe Benutzer- und Gruppen-ID zugeordnet sein.

- Systeme müssen auf ferne Dateisysteme mit Daten, die die Session und andere Anwendungen verwenden, Zugriff haben.
- Der Druck-Spooler `lp` muß für den Zugriff auf ferne Drucker konfiguriert sein.
- `sendmail` muß für E-mail-Services konfiguriert sein.
- X-Berechtigung muß eingerichtet sein.

Anmeldeabrechnungen für Benutzer einrichten

Dieser Abschnitt beschreibt, welche Anmeldeabrechnungen für das Desktop im Netzwerk erforderlich sind.

Anmeldeabrechnungen einrichten

Benutzer müssen ein Anmeldeabrechnungskonto für folgende Server und Systeme haben:

- alle Systeme, die Services für das Desktop zur Verfügung stellen, einschließlich Anwendungs-Server, Datei-Server und Systeme, die Drucker im Netzwerk zur Verfügung stellen.
- alle Session-Server, auf die der Benutzer zugreift. Session-Server werden normalerweise mit Xstations verwendet.

Konsistente Benutzer- und Gruppen-IDs

UNIX-Benutzer werden durch einen Anmeldenamen und eine numerische Benutzer-ID (UID) identifiziert. In einem Desktop-Netzwerk sollten Anmelde- name und UID des Benutzers an allen Client- und Server-Systemen dieselbe sein.

UNIX-Benutzer werden auch einer oder mehreren Anmeldegruppen zugeordnet. Jede Gruppe verfügt über einen Gruppennamen und eine numerische Gruppen-ID (GID). In einem Desktop-Netzwerk sollten alle Systeme konsistente Gruppennamen und Gruppen-IDs verwenden.

Weitere Informationen können den Hilfeseiten `id(1)` or `id(1M)` entnommen werden.

Zugriff auf verteilte Dateisysteme konfigurieren

Das Desktop verwendet NFS damit Dateien von verschiedenen Systemen gemeinsam benutzt werden können. Alle Dateisysteme im Netzwerk, die gemeinsam benutzte Dateien enthalten, müssen identifiziert werden, und es muß sichergestellt werden, daß sie korrekt an allen betreffenden Systemen angehängt sind.

In der Regel muß der folgende Zugriff auf ferne Dateien zur Verfügung gestellt werden:

- Das Home-Verzeichnis des Benutzers muß von allen Client- und Server-Systemen des Desktop gemeinsam benutzt werden. Dies ist notwendig, weil:
 - das Home-Verzeichnis Datendateien enthält, auf die von Anwendungen auf fernen Systemen zugegriffen wird. Beispielsweise verwenden Anwendungen, die mit Datendateien arbeiten, häufig das Home-Verzeichnis als die Standarddateiposition.
 - Das Home-Verzeichnis ist das Standardidentifikationsverzeichnis `dtspcd`. Weitere Informationen zu `dtspcd` können dem Abschnitt „Den Steuerungsdaemonprozeß für Unterprozesse konfigurieren“ auf Seite 127 entnommen werden.
- Möchten Benutzer auf Datendateien zugreifen, die sich nicht in ihrem Home-Verzeichnis befinden, müssen diese Datendateien von allen Client- und Server-Systemen des Desktop, die mit den Datendateien arbeiten, gemeinsam benutzt werden.
- Die Desktop-Installations- und Konfigurationsverzeichnisse (`/usr/dt` und `/etc/dt`) müssen von allen Client- und Server-Systemen des Desktop gemeinsam benutzt werden, so daß alle Anwendungen des Benutzers auf dieselben Desktop-Konfigurationsdateien zugreifen.

Home-Verzeichnis im Netzwerk einrichten

Ein Desktop-Netzwerk arbeitet am effektivsten, wenn Benutzer über ein einziges Home-Verzeichnis verfügen, das alle Client- und Server-Systeme im Netzwerk gemeinsam benutzen.

Ein Home-Verzeichnis im Netzwerk erlaubt einem Benutzer, verschiedene Systeme im Netzwerk zu verwenden, ohne daß die von ihm durchgeführten Anpassungen und Konfigurationen verlorengehen. Dies kommt daher, weil die Anpassungen des Benutzers und Informationen, die für die Wiederherstellung einer vorherigen Session notwendig sind, in Unterverzeichnissen des Home-Verzeichnisses gespeichert sind.

Ein gemeinsames Home-Verzeichnis ist ebenfalls erforderlich für:

- den standardmäßigen X-Identifikationsmechanismus. Weitere Informationen hierzu können dem Abschnitt „X-Berechtigung konfigurieren“ auf Seite 122 entnommen werden.
- den Desktop-Steuerungsdaemonprozeß für Unterprozesse, der beim Start ferner Anwendungen beteiligt ist, und in das Home-Verzeichnis des Benutzers schreiben können muß.

Dateinamenkonsistenz

Das Netzwerk sollte so konfiguriert sein, daß Benutzer von allen Systemen aus mit demselben Namen auf ihre Datendateien zugreifen können. Dies wird *Dateinamenkonsistenz* genannt und wird normalerweise erreicht, indem die entsprechenden symbolischen Verbindungen erstellt werden. Beispielsweise kann jedes System so konfiguriert werden, daß das Home-Verzeichnis jedes Benutzers als `/users/anmeldung_name` zur Verfügung steht, indem eine symbolische Verbindung zur tatsächlichen Mount-Position des Verzeichnisses hergestellt wird.

Zugriff auf ferne Drucker konfigurieren

Das Desktop verwendet den Druck-Spooler `lp` für den Zugriff auf lokale oder ferne Drucker. Weitere Informationen zur Konfiguration des Druck-Spoolers `lp` kann der Hilfeseite `lpadmin(1M)` entnommen werden.

Bevor der Benutzer versucht, über die grafische Schnittstelle des Desktops zu drucken, sollte er testen, ob mit dem Befehl `lp` an allen Druckern problemlos gedruckt werden kann.

Verwenden Sie in jedem Fall konsistente Druckereinheitennamen. Ist beispielsweise ein bestimmter Drucker dem System, mit dem er direkt verbunden ist, als `Postscript1` bekannt, sollten alle weiteren Systeme, die auf den Drucker fern zugreifen, den Namen `Postscript1` verwenden.

Elektronische Post konfigurieren

Das Desktop-Postprogramm verwendet `sendmail`, um Post zwischen Systemen zu versenden. Weitere Informationen zur Konfigurierung von E-mail-Konnektivität können der Hilfeseite `sendmail(1M)` entnommen werden.

Bevor der Benutzer versucht, Post über das Desktop zu senden oder zu empfangen, sollte er testen, ob er Post problemlos mit dem Befehl `mailx` senden und empfangen kann.

X-Berechtigung konfigurieren

Das Desktop verwendet den standardmäßigen X-Mechanismus, um ferne Anwendungen (X-Clients) zum Zugriff auf einen lokalen Bildschirm zu berechtigen. Der einfachste Weg, dies zu erreichen, ist, ein Home-Verzeichnis im Netzwerk für jeden Benutzer zur Verfügung zu stellen. Hierdurch wird sichergestellt, daß folgende Erfordernisse erfüllt sind:

- Der Benutzer muß über Schreib- und Leseberechtigung für die Datei *Home-Verzeichnis* / `.xauthority` verfügen.

- Die Datei `.xauthority` auf einem Anwendungs-Server muß den Eintrag für den Bildschirm enthalten, auf dem die Anwendung ausgeführt wird.

Weitere Informationen können den Hilfeseiten `X(1)` oder `xauth(1)` entnommen werden.

Clients und Server für das Desktop konfigurieren

In diesem Abschnitt werden Desktop-spezifische Erfordernisse der Netzwerkkonfiguration behandelt. Das heißt, diese Funktionen werden durch das Desktop und nicht durch das Basisbetriebssystem zur Verfügung gestellt.

Der Abschnitt besteht aus zwei Teilen:

- der Konfiguration von Anmelde- und Session-Services.
- der Konfiguration von Services, die für Anwendungen und deren Daten erforderlich sind. Hierzu gehören: Anwendungs-, Datenbank-, Symbol-, Datei- und Hilfe-Server und deren Clients.

Anmelde- und Session-Services konfigurieren

Bei einem Anmelde-/Session-Server handelt es sich um ein System, das Desktop-Services (Anmeldemanager, Session-Manager, Dateimanager, Fenstersteuerung usw.) einem Bildschirm und X-Server zur Verfügung stellt.

In der Regel stellt ein Session-Server XStations Services zur Verfügung. Aber es kann auch eine Netzwerkkonfiguration eingerichtet werden, die Session-Services auf einem oder mehreren Servern konzentriert, auf die sowohl von XStations als auch von Datenstationen zugegriffen wird.

Beim Anmeldemanager handelt es sich um die Desktop-Komponente, die dafür zuständig ist, daß Anmelde-Services anderen Bildschirmen zur Verfügung gestellt werden. Sobald der Benutzer sich angemeldet hat, wird der Session-Manager für den Benutzer gestartet.

Weitere Informationen zur Konfiguration von Anmelde-/und Session-Servern, sowie XStations können dem Abschnitt „Eine Anmeldeanzeige auf einer Netzwerkanzeige anzeigen“ auf Seite 6 entnommen werden.

Weitere anwendungsbezogene Services konfigurieren

Dieser Abschnitt behandelt Erfordernisse des Desktops im Netzwerk für folgende Server:

- Anwendungs-Server
- Datenbank-Server
- Symbol-Server
- Hilfe-Server

Clients und Server für das Desktop konfigurieren

1. Die notwendigen Netzwerkkonfigurationen für das Betriebssystem zur Verfügung stellen, die für das Desktop erforderlich sind.

Weitere Informationen hierzu können dem Abschnitt „Netzwerk des Basisbetriebssystems für das Desktop konfigurieren“ auf Seite 119 entnommen werden.

2. Das Desktop oder die Mindestdateigruppe installieren :

Es müssen installiert werden:

- die gesamten Dateigruppen von ‘Common Desktop Environment runtime’
- *oder*, folgende Dateigruppen: CDE-MIN und CDE-TT

Hinweis - Installation und Dateigruppen können bei verschiedenen Lieferanten unterschiedlich zusammengesetzt sein.

3. Das System für den Dämon des Servers für die ToolTalk-Dateinamendatenbank `rpc.ttdbserver` konfigurieren.

Dies sollte bei der Installation des Desktops automatisch geschehen. Weitere Informationen können dem Abschnitt „ToolTalk-Datenbank-Server konfigurieren“ auf Seite 128 entnommen werden.

4. Den Steuerungsdämonprozeß für Unterprozesse (`dtspcd`) installieren und konfigurieren.

Dies sollte bei der Installation des Desktops automatisch geschehen. Weitere Informationen können dem Abschnitt „Den Steuerungsdämonprozeß für Unterprozesse konfigurieren“ auf Seite 127 entnommen werden

5. Alle erforderlichen fernen Daten anhängen.

Daten werden als "fern" bezeichnet, wenn sie sich auf einem anderen System als dem System befinden, auf dem die Anwendung, die die Daten verwendet, ausgeführt wird.

Beispiel:

- Verwendet eine Anwendung Daten auf einem Datei-Server, müssen diese Dateien angehängt werden.
- Befinden sich Dateimanagersymbole auf einem Symbol-Server, muß der Session-Server diese Dateien anhängen.
- Verwendet das Netzwerk einen Hilfe-Server für Desktop-Hilfedateien, müssen der Session-Server und alle Anwendungs-Server die Hilfedaten anhängen.

Weitere Informationen zu Einhängpunkten können dem nächsten Abschnitt, „Den Einhängpunkt für ferne Dateisysteme konfigurieren“ auf Seite 125“ entnommen werden.

Den Einhängpunkt für ferne Dateisysteme konfigurieren

Wenn das Desktop Dateinamen von einem System zum anderen übermittelt, müssen diese Dateinamen in Namen umgesetzt, bzw. *zugeordnet* werden, die für das Zielsystem verständlich sind. Diese Zuordnung ist notwendig, weil eine Datei möglicherweise an den verschiedenen Systemen an verschiedenen Positionen eingehängt ist, und deshalb verschiedene Namen verwendet werden müssen, um auf die Datei zugreifen zu können. Beispielsweise wird auf die Datei `/projects/big` auf `sysA` von `sysB` aus mit dem Namen `/net/sysA/projects/big` zugegriffen.

Erfordernisse für die Dateinamenzuordnung

Um die Dateinamenzuordnung problemlos durchführen zu können, muß einer der folgenden Punkte zutreffen:

- Der Befehl `mount` wird verwendet, um Dateisysteme statisch einzuhängen. Diese Art des statischen Einhängens wird in der Regel in einer Datei wie beispielsweise `/etc/checklist`, `/etc/mnttab` oder `/etc/filesystems` verwendet.

Damit die Dateinamenzuordnung zwischen Systemen problemlos funktioniert, müssen beim Einhängen von Dateisystemen konsistente Host-Namen verwendet werden. Verfügt ein Host über mehrere Namen, (zum Beispiel Alias-Namen, oder der Host hat mehr als eine LAN-Adresse mit verschiedenen Namen), muß bei allen Einhängvorgängen derselbe Name und dasselbe Format verwendet werden.

- *oder*, der Automounter wird verwendet, um Dateisysteme am Standardeinhängepunkt `/net` einzuhängen.
- *oder*, der Automounter wird verwendet, um Dateisysteme an einer anderen Stelle als `/net` einzuhängen, und die Umgebungsvariable `DTMOUNTPOINT` wird gesetzt, um den Einhängpunkt anzugeben. Weitere Informationen hierzu können

dem nächsten Abschnitt, „Einen Wert für DTMOUNTPOINT setzen ” auf Seite 126” entnommen werden.

Weitere Informationen zum automounter können der Hilfeseite automount(1M) entnommen werden.

Einen Wert für DTMOUNTPOINT setzen

Treffen die beiden folgenden Bedingungen zu, muß die Umgebungsvariable DTMOUNTPOINT gesetzt werden:

- Der Automounter wird verwendet, um Dateisysteme einzuhängen
- *und*, ferne Dateisysteme werden an einer anderen Position als /net eingehängt.
- Zu den Prozessen, für die DTMOUNTPOINT gesetzt werden muß, gehören:
 - die Desktop-Prozesse des Benutzers, die automatisch gestartet werden, wenn der Benutzer sich anmeldet, wie zum Beispiel der Arbeitsbereichsmanager (dtwm) und der Dateimanager (dtfile)
 - Systemprozesse, wie zum Beispiel `rpc.ttdbserver` und `dtspcd`, die durch Mechanismen wie `inetd` gestartet werden.
 - Anwendungen, die vom Desktop auf lokalen oder fernen Systemen gestartet werden
 - Anwendungen, die vom Benutzer über eine Shell-Befehlszeile gestartet werden.

Um DTMOUNTPOINT für diese Prozesse zu setzen, muß der Benutzer

1. die Datei `/etc/inetd.conf` folgendermaßen bearbeiten:

a. den Eintrag `dtspcd` suchen und folgendes hinzufügen:

`-mount_point` *einhängepunkt*

b. den Eintrag `rpc.ttdbserver` suchen und folgendes hinzufügen:

`-m` *einhängepunkt*

Wird, beispielsweise, der Automounter mit dem Einhängepunkt `/nfs` verwendet, sehen die Einträge in `/etc/inetd.conf` folgendermaßen aus:

```
dtspc 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. Die Systemprozedur ausführen, mit der `/etc/inetd.conf` erneut gelesen wird. Weitere Informationen können der Hilfeseite `inetd(1M)` entnommen werden.

3. DTMOUNTPOINT so setzen, daß der Wert von Benutzeranmeldungen übernommen wird.

Dies geschieht, indem die Variable in `/etc/dt/config/Xsession.d` gesetzt wird. Weitere Informationen zum Setzen von Umgebungsvariablen können dem Abschnitt entnommen werden.

Den Steuerungsdaemonprozeß für Unterprozesse konfigurieren

Der Unterprozeßsteuerungs-Service (SPC) des Desktops stellt Client-/Server-Befehlsausführungen zur Verfügung.

Der Steuerungsdaemonprozeß für Unterprozesse des Desktops (`dtspcd`) wird vom Desktop verwendet, um ferne Anwendungen zu starten. Es handelt sich um einen `inet`-Dämon, der Anforderungen von fernen Clients zur Befehlsausführung akzeptiert. Weitere Informationen zur Konfiguration von `inet`-Dämonen können der Hilfeseite `inetd.conf(1M)` entnommen werden.

Die Aktionsaufrufbibliothek des Desktop verwendet den SPC-Service, um ferne Aktionen aufzurufen.

Dtspcd konfigurieren

- ◆ **Überprüfen, ob `dtspc` sowohl in `/etc/services` als auch in `/etc/inetd.conf` korrekt registriert ist. Weitere Informationen hierzu können dem Abschnitt Hilfeseite `dtspcd(1M)`.**

SPC-Sicherheit

Die Identifikationsüberprüfung für den Unterprozeßsteuerungsservice basiert auf der Dateisystemidentifikationsüberprüfung. `dtspcd` muß auf ein *Identifikationsüberprüfungsverzeichnis* Zugriff haben, das bei allen SPC-Client-Systemen ebenfalls eingehängt ist.

Standardmäßig handelt es sich bei dem Identitätsüberprüfungsverzeichnis für `dtspcd` um das Home-Verzeichnis des Benutzers. Aber `dtspcd` kann auch so konfiguriert werden, daß ein anderes Verzeichnis gewählt wird, indem die Option `-auth_dir` in dem Verzeichnis `/etc/inetd.conf` gesetzt wird. Weitere Informationen hierzu können der Hilfeseite `dtspcd(1M)` entnommen werden.

Da die SPC-Identitätsüberprüfung auf der Identitätsüberprüfung des Dateisystems basiert, ist der SPC-Service nur so sicher, wie das verteilte Dateisystem. Wird das Desktop in einem Netzwerk verwendet, in dem der Benutzer dem verteilten Dateisystem nicht traut, kann er `dtspcd` inaktivieren. Hierzu muß der Eintrag `dtspc` in `/etc/services` auf Kommentar gesetzt werden

Umgebungsvariablen für eine Ausführung auf einem fernen System konfigurieren

Verwendet das Desktop eine Aktion, um eine Anwendung auf einem fernen System zu starten, werden die Umgebungsvariablen des Benutzers in das ferne System kopiert und in die Umgebung der Anwendung gestellt.

Standardmäßig werden einige der Umgebungsvariablen geändert, bevor sie in das ferne System kopiert werden. Sowohl die Aktionsaufrufkomponente als auch der Steuerungsdaemonprozeß für Unterprozesse des Desktops können so konfiguriert werden, daß eine zusätzliche Verarbeitung der Umgebungsvariablen erfolgt, bevor die Variablen in die Umgebung der Anwendung gestellt werden.

Weitere Informationen zur Standardkonfiguration und den Möglichkeiten, diese zu ändern, können den Hilfeseiten `dtactionfile(4)` und `dtspcdenv(4)` entnommen werden.

ToolTalk-Datenbank-Server konfigurieren

Eine Komponente von ToolTalk ist der ToolTalk-Datenbank-Server, `/usr/dt/bin/rpc.ttdbserver`.

Der ToolTalk-Datenbank-Server wird vom ToolTalk-Meldungsservice und für die Dateinamenzuordnung verwendet. Dieser Server wird normalerweise bei der Installation des Desktops in `/etc/inetd.conf` registriert und eine zusätzliche Konfiguration ist nicht notwendig.

Weitere Informationen zum ToolTalk-Datenbank-Server und dessen Konfigurationsoptionen können der Hilfeseite `rpc.ttdbserver(1M)` entnommen werden.

ToolTalk-Meldungs-Server konfigurieren

Bei dem ToolTalk-Meldungs-Server handelt es sich um `ttsession`. Standardmäßig muß bei diesem Server keine Konfiguration vorgenommen werden. Er wird während der Anmeldung durch die Prozedur `Xsession` gestartet.

Weitere Informationen zum ToolTalk-Meldungs-Server und dessen Konfigurationsoptionen können der Hilfeseite `ttsession` entnommen werden.

Kalender-Dämon konfigurieren

Eine Komponente der Anwendung 'Kalender' ist der Kalender-Dämon `rpc.cmsd`. Dieser Dämon wird normalerweise bei der Installation des Desktops in `/etc/inetd.conf` registriert und eine zusätzliche Konfiguration ist nicht notwendig.

Weitere Informationen zum Kalender-Dämon und dessen Konfigurationsoptionen können der Hilfeseite `rpc.cmsd(1)` entnommen werden.

Anwendungs-Services verwalten

Dieser Abschnitt behandelt spezifische Konfigurationserfordernisse für:

- Anwendungs-Server und deren Clients
 - Desktop-Server, die spezielle Services zur Verfügung stellen, wie z.B. Datenbank-Server, Symbol-Server und Hilfe-Server
- Außerdem werden Netzwerkerfordernisse für zwei spezielle Konfigurationen für Anwendungen im Netzwerk behandelt:
- Hosts für Ausführungen auf einem fernen System
 - Anwendungen, die über Dateisystemeinhängungen ausgeführt werden

Suchpfadumgebungsvariablen

Das Desktop verwendet eine Gruppe von Umgebungsvariablen, um den Suchpfad anzugeben, über den Desktop-Konfigurationsdateien für Anwendungen, wie z. B. die Datenbank für Aktionen und Datentypen, Hilfedateien oder Symboldateien, gesucht werden.

Weitere Informationen zur Verwendung der Suchpfadumgebungsvariablen können Kapitel 9 oder der Hilfeseite `dtenvvar(5)` entnommen werden.

Anwendungs-Server und dessen Clients konfigurieren

In der Standardkonfiguration des Anwendungs-Servers enthält der Anwendungs-Server alle zu einer Anwendung gehörenden binären Dateien und Konfigurationsdateien, wie:

- die ausführbare(n) Datei(en) der Anwendung
- Standardanwendungskonfigurationsdateien wie `app-defaults`, Nachrichtenkataloge sowie gemeinsam benutzte Bibliotheken für diese Anwendung
- Desktop- Konfigurationsdateien:
 - Definitionsdateien für Aktionen und Datentyp
 - Symbolabbilddateien

- Desktop-Hilfedatendateien

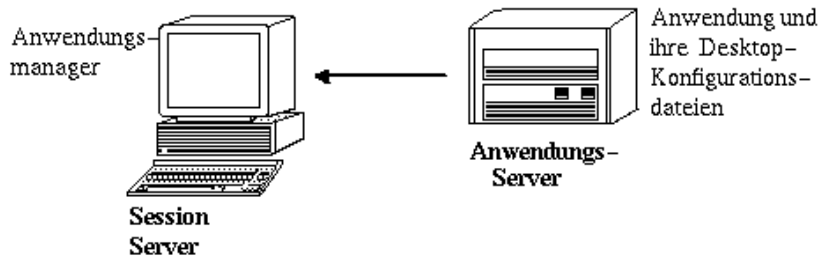


Abbildung 7-5 Standardkonfiguration des Anwendungs-Servers

Anwendungs-Server konfigurieren

1. Die für das Desktop erforderlichen Netzwerkkonfigurationen des Betriebssystems einrichten.

Weitere Informationen hierzu können dem Abschnitt „Netzwerk des Basisbetriebssystems für das Desktop konfigurieren“ auf Seite 119 entnommen werden.

2. Die für Server erforderliche allgemeine Desktop-Konfiguration einrichten.

Weitere Informationen hierzu können dem Abschnitt „Clients und Server für das Desktop konfigurieren“ auf Seite 124 entnommen werden.

3. Die Anwendung(en) installieren.

4. Wird eine Anwendung nicht automatisch registriert, muß die Registrierung durchgeführt werden.

Weitere Informationen hierzu können dem Abschnitt " entnommen werden.

Client eines Anwendungs-Servers konfigurieren

1. Die für das Desktop erforderlichen Netzwerkkonfigurationen des Betriebssystems einrichten.

Weitere Informationen hierzu können dem Abschnitt „Netzwerk des Basisbetriebssystems für das Desktop konfigurieren“ auf Seite 119 entnommen werden.

2. Die für Clients erforderliche allgemeine Desktop-Konfiguration einrichten.

Weitere Information hierzu können dem Abschnitt „Clients und Server für das Desktop konfigurieren“ auf Seite 124 entnommen werden.

3. Den Anwendungs-Server dem Anwendungssuchpfad entweder für das gesamte System oder für den einzelnen Benutzer hinzufügen:

TABELLE 7-2

systemweit	Die Variable DTSPSYSAPPHOSTS in <code>/etc/dt/config/Xsession.d/0010.dtpaths</code> setzen .
persönlich	Die Variable DTSPUSERAPPHOSTS in <i>Home-Verzeichnis</i> / <code>.dtprofile</code> setzen.

Beispielsweise fügt die folgende Zeile in `/etc/dt/config/Xsession.d/0010.dtpaths` ein System mit den Host-Namen `SysAAA` und `SysBBB` zum Anwendungssuchpfad hinzu:

```
DTSPSYSAPPHOSTS=SysAAA: ,SysBBB:
```

Weitere Informationen zur Einrichtung des Anwendungssuchpfads können folgenden Abschnitten entnommen werden:

- „Anwendungssuchpfad“ auf Seite 149
- „Wert eines Suchpfads setzen“ auf Seite 147

Datenbank-, Symbol- und Hilfe-Services

Normalerweise werden die zu einer Anwendung gehörenden Definitionen für Aktionen und Datentyp, Symbole und Hilfedatendateien auf dasselbe System wie die Anwendung installiert.

Die typische Konfiguration von Hilfedatendateien sieht folgendermaßen aus:

- Die Hilfedateien für den Dateimanager befinden sich normalerweise auf dem Session-Server. Das Desktop findet sie, weil der Hilfesuchpfad automatisch die richtigen Positionen auf dem Session-Server durchsucht.
- Die Hilfedateien für andere Anwendungen befinden sich normalerweise auf demselben Anwendungs-Server wie die Anwendung. Der Session-Server findet sie, weil eine Änderung des Anwendungssuchpfads automatisch den Hilfesuchpfad ändert.

Möglicherweise gibt es Situationen, in denen der Benutzer Datenbank- (Aktionen und Datentypen), Hilfe- oder Symboldaten an eine anderen Position im Netzwerk stellen möchte. Verwendet das Netzwerk beispielsweise mehrere Session-Server, soll möglicherweise ein Hilfe-Server erstellt werden, auf dem alle

Hilfedateien für Desktop-Anwendungen (Dateimanager, Umgebungsmanager, usw) gespeichert werden. Dadurch wird Plattenspeicherplatz gespart, da die Hilfedateien nicht auf jedem einzelnen Session-Server vorhanden sein müssen.

Datenbank-, Hilfe- oder Symbol-Server erstellen

1. Die für das Desktop erforderlichen Netzwerkkonfigurationen des Betriebssystems einrichten.

Weitere Informationen hierzu können dem Abschnitt „Netzwerk des Basisbetriebssystems für das Desktop konfigurieren“ auf Seite 119 entnommen werden.

2. Die für Clients erforderliche allgemeine Desktop-Konfiguration einrichten.

Weitere Informationen hierzu können dem Abschnitt „Clients und Server für das Desktop konfigurieren“ auf Seite 124 entnommen werden.

3. Die Datenbank-, Hilfe- oder Symboldateien installieren.

Es ist dem Benutzer freigestellt, wo er die Dateien auf das System stellen möchte. Es empfiehlt sich jedoch, die folgenden Positionen zu verwenden, da diese Verzeichnisse automatisch durchsucht werden, wenn ein System als Anwendungs-Server verwendet wird.

- Datenbankdateien: `/etc/dt/appconfig/types/ sprache`
- Hilfedateien: `/etc/dt/appconfig/help/ sprache`
- Symboldateien: `/etc/dt/appconfig/icons/ sprache`

Wird ein Datenbank-Server eingerichtet, müssen die Aktionen geschrieben werden, um anzugeben, wo die zugehörigen Befehle (EXEC_STRING s) ausgeführt werden. Weitere Informationen hierzu können dem Abschnitt „Rechner für die Ausführung auf einem fernen System angeben“ auf Seite 133 entnommen werden.

Verbindung zwischen Session-Server und Datenbank-, Symbol- oder Hilfe-Server konfigurieren

1. Die für das Desktop erforderlichen Netzwerkkonfigurationen des Betriebssystems einrichten.

Weitere Informationen hierzu können dem Abschnitt „Netzwerk des Basisbetriebssystems für das Desktop konfigurieren“ auf Seite 119 entnommen werden.

2. Die für Clients erforderliche allgemeine Desktop-Konfiguration einrichten.

Weitere Informationen hierzu können dem Abschnitt „Clients und Server für das Desktop konfigurieren ” auf Seite 124 entnommen werden.

3. Den Datenbank-, Symbol- oder Hilfe-Server im entsprechenden Suchpfad hinzufügen.

- Wurden die Datendateien an die Positionen gestellt, die in Schritt 3 des Abschnitts „Datenbank-, Hilfe- oder Symbol-Server erstellen” auf Seite 132” angegeben wurden, kann der Anwendungssuchpfad geändert werden.
- Wurden die Datendateien an andere Positionen gestellt, muß der spezifische Suchpfad geändert werden.

Wurden beispielsweise die Hilfedateien in das Verzeichnis `/etc/dt/help on system SysCCC` gestellt, muß die folgende Zeile zu `/`

`etc/dt/config/Xsession.d/0010.dtpaths` hinzugefügt werden:

```
DTSPSYSHELP=/net/SysCCC/etc/dt/help
```

Weitere Informationen zur Einrichtung von Suchpfaden können folgenden Abschnitten entnommen werden:

- „Datenbanksuchpfad (Aktionen/Datentypen) ” auf Seite 153
- „ Symbolsuchpfad” auf Seite 156
- „ Hilfesuchpfad” auf Seite 159
- „Wert eines Suchpfads setzen” auf Seite 147

Spezielle Konfigurationen für Anwendungen im Netzwerk

In diesem Abschnitt wird beschrieben, wie Systeme konfiguriert werden, damit Anwendungen:

- an einem anderen System, als an dem System, das die Aktion enthält, ausgeführt werden, d.h. auf einem Rechner für die Ausführung auf einem fernen System.
- lokal über Dateisystemeinhängungen ausgeführt werden können.

Rechner für die Ausführung auf einem fernen System angeben

In der typischen Anwendungs-Serverkonfiguration befindet sich die Aktionsdefinition auf demselben System wie die ausführbare Datei der Anwendung. Aber es können auch Aktionen geschrieben werden, mit denen Befehle an anderen Systemen ausgeführt werden können. In einer solchen Konfiguration wird das System, das die Anwendung enthält, als der *ausführende Rechner* bezeichnet.

Die Aktionsdefinition kann sich sowohl auf dem Session-Server als auch auf einem System, das dem Session-Server Aktions- und Datentyp-Services zur Verfügung stellt und das als *Datenbank-Server* oder Datenbankrechner bezeichnet wird, befinden.

In Aktionsdefinitionen wird durch das Feld EXEC_HOST angegeben, wo die Befehle (EXEC_STRINGs) ausgeführt werden sollen. Beispielsweise wird in der folgenden Aktionsdefinition angegeben, daß ein Client xload auf einem System mit dem Rechnernamen SysDDD ausgeführt wird:

TABELLE 7-3

ACTION XloadSysDDD	
TYPE	COMMAND
EXEC_HOST	SysDDD
EXEC_STRING	/usr/bin/X11/xload -label SysDDD
}	

In Aktionsdefinitionen wird durch das Feld EXEC_HOST mehr als ein Rechnername angegeben, dann versucht das Desktop, EXEC_STRING nacheinander auf jedem der angegebenen Rechner auszuführen, bis es einen Rechner findet, der die Aktion ausführen kann. Beispielsweise wird im folgenden Feld EXEC_HOST angegeben, daß die Aktion zunächst versuchen soll, EXEC_STRING auf SysDDD, und falls dies fehlschlägt, auf SysEEE auszuführen:

TABELLE 7-4

EXEC_HOST	SysDDD,SyseEE
-----------	---------------

Ist das Feld EXEC_HOST für eine Aktion nicht gesetzt, wird der Standardwert %DatabaseHost% gesetzt. Der Wert %DatabaseHost% wird dem Datenbanksuchpfad entnommen.

Angenommen, der Datenbanksuchpfad wurde geändert, indem folgende Zeile zu /etc/dt/config/Xsession.d/0010.dtpaths hinzugefügt wurde:

```
DTSPSYSDATABASEHOSTS=SysAAA: , /net/SysBBB/etc/dt/appconfig/types/C
```

sysAAA wird mit der qualifizierten Syntax für die Angabe des Rechners, sysAAA: angegeben. Eine Aktion, die dieses Element des Suchpfads verwendet, setzt den Datenbankrechner auf sysAAA. Verwendet eine Aktion den Abschnitt /net/sysBBB... des Suchpfads, wird als Datenbankrechner das lokale System verwendet, da die Syntax das Qualifikationsmerkmal nicht enthält.

Rechner für die Ausführung auf einem fernen System konfigurieren

1. Die für das Desktop erforderlichen Netzwerkkonfigurationen des Betriebssystems einrichten.

Weitere Informationen hierzu können dem Abschnitt „Netzwerk des Basisbetriebssystems für das Desktop konfigurieren“ auf Seite 119 entnommen werden.

2. Die für Server erforderliche allgemeine Desktop-Konfiguration einrichten.

Weitere Informationen hierzu können dem Abschnitt „Clients und Server für das Desktop konfigurieren“ auf Seite 124 entnommen werden.

3. Sicherstellen, daß die Anwendungen für die Ausführung auf dem lokalen System richtig installiert und konfiguriert sind.

System, das die Aktionsdefinitionen enthält, konfigurieren

1. Die für das Desktop erforderlichen Netzwerkkonfigurationen des Betriebssystems einrichten.

Weitere Informationen hierzu können dem Abschnitt „Netzwerk des Basisbetriebssystems für das Desktop konfigurieren“ auf Seite 119 entnommen werden.

2. Die für Server erforderliche allgemeine Desktop-Konfiguration einrichten.

Weitere Informationen hierzu können dem Abschnitt „Clients und Server für das Desktop konfigurieren“ auf Seite 124 entnommen werden.

3. Die Aktionsdefinitionen und Anwendungsgruppen erstellen und installieren.

Weitere Informationen hierzu können dem Abschnitt und „Allgemeine Gruppen von Anwendungen erstellen und verwalten“ auf Seite 51 entnommen werden.

Session-Server konfigurieren

1. Die für das Desktop erforderlichen Netzwerkkonfigurationen des Betriebssystems einrichten.

Weitere Informationen hierzu können dem Abschnitt „Netzwerk des Basisbetriebssystems für das Desktop konfigurieren“ auf Seite 119 entnommen werden.

2. Die für Clients erforderliche allgemeine Desktop-Konfiguration einrichten.

Weitere Informationen hierzu können dem Abschnitt „Clients und Server für das Desktop konfigurieren“ auf Seite 124 entnommen werden.

3. Den Datenbankrechner dem Suchpfad für Aktionen hinzufügen.

Weitere Informationen hierzu können dem Abschnitt „Datenbanksuchpfad (Aktionen/Datentypen)“ auf Seite 153 entnommen werden.

4. Den ausführenden Rechner dem Anwendungssuchpfad hinzufügen.

Weitere Informationen hierzu können dem Abschnitt „Anwendungssuchpfad“ auf Seite 149 entnommen werden.

Anwendungen lokal ausführen

Die Server-Standardanwendungsconfiguration führt Anwendungen auf dem Anwendungs-Server aus. Manchmal ist es wünschenswert, daß eine Anwendung auf einem fernen System installiert ist, aber auf dem Session-Server lokal ausgeführt wird.

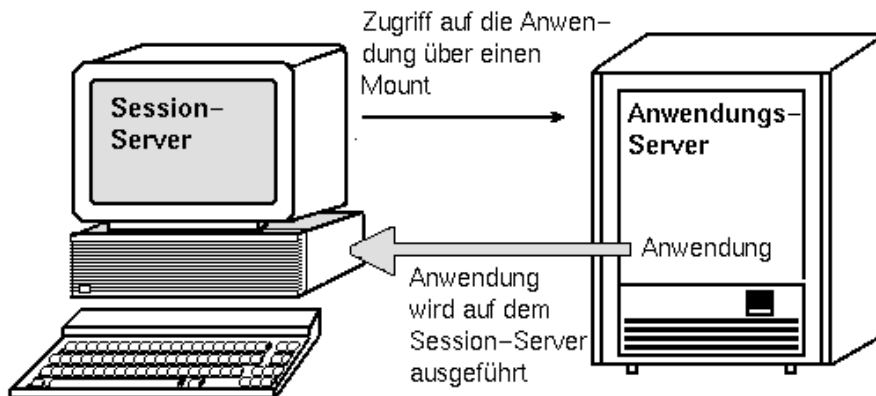


Abbildung 7-6 Ausführung über Einhängepunkte

Anwendungs-Server konfigurieren

Eine spezielle Konfiguration ist nicht erforderlich.

Session-Server konfigurieren

- ◆ **Den Anwendungssuchpfad ändern. Den lokalen, absoluten Pfad zu der Anwendung verwenden.**

Beispielsweise könnte die folgende Variablendefinition verwendet werden, um eine Anwendung zu finden, die auf `sysAAA` registriert ist:

```
DTSPSYSAPPHOSTS=/net/SysAAA/etc/dt/appconfig/appmanager/C
```

Der Session-Server muß auf die Konfigurationsdateien der Anwendung, wie `app-defaults`, Nachrichtenkataloge und gemeinsam benutzte Bibliotheken zugreifen können.

Verwaltung und Drucken über das Desktop konfigurieren

Für den Desktop-Benutzer gibt es verschiedene Möglichkeiten, Dateien zu drucken. Es wird zwischen zwei Kategorien unterschieden: Drucken über das Desktop und Drucken über eine Anwendung.

Zum Drucken über das Desktop gehören:

- Das Auswählen einer Datei im Dateimanager und das Anklicken von 'Drucken' im Menü 'Ausgewählt' oder im (temporär eingeblendeten) Menü des Symbols.
- Das Ziehen einer Datei vom Dateimanager zum Druckersymbol im Bedienfeld oder der Bedientafel 'Persönliche Drucker'.
- das Ziehen einer Datei vom Dateimanager zu einem Drucker im Hauptfenster des Druckmanagers

Um über eine Anwendung zu drucken, den Befehl 'Drucken' auf den über ein Menü oder ein anderes Bedienelement innerhalb des Anwendungsfensters zugegriffen wird, verwenden.

- „Drucker hinzufügen und löschen ” auf Seite 139
- „Druckersymbolbilder” auf Seite 141
- „Standarddrucker konfigurieren” auf Seite 142
- „Druckerkonzepte” auf Seite 143

Drucker hinzufügen und löschen

In diesem Abschnitt werden die Prozeduren erläutert, mit denen Drucker dem Desktop hinzugefügt oder vom Desktop gelöscht werden.

▼ Drucker dem Desktop hinzufügen

1. Einen Drucker der Konfiguration des Systems hinzufügen.

Dabei den Anweisungen in der Dokumentation zur Systemverwaltung folgen.

2. Folgenden Befehl ausführen:

```
env LANG=Wesprache /usr/dt/bin/dtprintinfo -populate
```

3. Den Druckmanager erneut starten oder 'Aktionen erneut laden' in der Gruppe von Anwendungen 'Desktop_Werkzeuge' im Anwendungsmanager doppelklicken. Überprüfen, ob der Drucker angezeigt wird.

4. Über die elektronische Post die Benutzer benachrichtigen, daß sie ebenfalls den Druckmanager erneut starten oder 'Aktionen erneut laden' ausführen sollten.

Bei jedem Aufruf liest der Druckmanager die Konfigurationsliste für Systemdrucker erneut. Wird ein neuer Drucker festgestellt, wird automatisch eine neue Desktop-Druckeraktion und ein Symbol für diesen Drucker erstellt. Der Benutzer muß keine weiteren Schritte unternehmen, damit der Drucker auf dem desktop angezeigt wird.

▼ Drucker vom Desktop löschen

1. Den Drucker aus der Konfiguration des Systems entfernen.

Dabei den Anweisungen in der Dokumentation zur Systemverwaltung folgen.

2. Den Druckmanager erneut starten oder 'Aktionen erneut laden' in der Gruppe von Anwendungen Desktop_Werkzeuge im Anwendungsmanager doppelklicken. Überprüfen, ob der Drucker nicht mehr angezeigt wird.

3. Über die elektronische Post die Benutzer benachrichtigen, daß sie ebenfalls den Druckmanager erneut starten oder 'Aktionen erneut laden' ausführen sollten.

Bei jedem Aufruf liest der Druckmanager die Konfigurationsliste für Systemdrucker. Wird festgestellt, daß ein Drucker von der Liste entfernt wurde, wird automatisch die Druckeraktion und das Symbol für diesen Drucker aus dem Druckmanager und dem Dateimanager entfernt. Der Benutzer muß keine weiteren Schritte unternehmen, damit der Drucker vom Desktop gelöscht wird.

Hinweis - Der Druckmanager kann Drucker nicht aus dem Bedienfeld entfernen. Deshalb sollten, wenn ein Drucker aus der Konfiguration entfernt wird, alle Benutzer über die elektronische Post benachrichtigt werden, daß sie Symbole des gelöschten Druckers im Bedienfeld löschen sollten.

Intervall, in dem Jobs aktualisiert werden, ändern

Um zu ändern, wie häufig die im Druckmanager angezeigten Informationen aktualisiert werden, muß das Intervall für die Aktualisierung von Jobs geändert werden. Standardmäßig fragt der Druckmanager Drucker alle dreißig Sekunden nach dem Status ihrer Druckjobs ab. Dieses Zeitintervall kann mit der Schiebeleiste für das Aktualisierungsintervall im Dialogfenster 'Optionen festlegen' geändert werden. (Dieses Dialogfenster wird angezeigt, indem 'Optionen festlegen' im Menü 'Ansicht' ausgewählt wird.)

Druckersymbolbilder

Wird ein Drucker hinzugefügt, wird ihm automatisch das Standarddruckersymbol zugeordnet. Soll dem Drucker ein anderes Symbol verfügbar gemacht werden, die Symboldateien in `/etc/dt/appconfig/icons/ sprache` oder in ein anderes Verzeichnis im Symbolsuchpfad stellen. Die Benutzer können dann dieses Symbol auswählen, um das Standardsymbol für den Drucker zu ersetzen.

Es muß ein vollständiger Satz (groß, mittel und klein) der Symbole vorhanden sein, sonst werden sie im Anwendungsauswahlnenü für Symbole des Druckmanagers nicht angezeigt.

Weiter Informationen zum können dem Abschnitt „ Symbolsuchpfad ” auf Seite 156 entnommen werden.

Symboldateinamen und -größen

Für Symboldateinamen ist folgendes Format erforderlich:

dateiname ohne pfad.größe.
typ

wobei

größe die Größe angibt. Möglich sind: `l` (groß), `m` (mittel), `t` (klein). Weitere Informationen zu der Größe von Symbolen können dem Abschnitt „Symbolgrößenkonventionen ” auf Seite 255 entnommen werden.

typ angibt, ob es sich um eine Pixmap- (`pm`) oder um eine Bitmap-Datei (`bm`) handelt.

Symboldateinamen für mittelgroße und kleine Pixmap-Symbole für einen Farbdrucker könnten beispielsweise als `ColorPrinter.m.pm` und `ColorPrinter.t.pm` bezeichnet werden..

Weitere Informationen zum Erstellen von Symbolen können Kapitel 14” entnommen werden.

▼ Symbol, Druckerbezeichnung oder Beschreibung eines Druckers global ändern

Globale Druckermerkmale sollten sofort nach Hinzufügen des Druckers geändert werden, bevor Benutzer sie mit dem Druckmanager ändern. Sobald ein Benutzer die Druckermerkmale mit dem Druckmanager geändert hat, werden nachträgliche globale Änderungen nicht mehr sichtbar.

Die Datei `/etc/dt/appconfig/types/sprache/druckerschlange_name.dt` bearbeiten und die gewünschten Änderungen für das Symbol, die Druckerbeschreibung und die Beschreibung vornehmen:

1. **Im Feld `ICON` für *dateiname ohne pfad* den neuen Dateinamen ohne Pfad für das Symbol eingeben.**
2. **Im Feld `LABEL` für *labelname* die neue Bezeichnung für den Drucker eingeben.**
3. **Den Text im Feld `DESCRIPTION` aktualisieren.**

Hier empfiehlt es sich, die Position des Druckers, den Druckertyp und den zuständigen Hardwarebeauftragten anzugeben. Soll mehr Text als eine Zeile geschrieben werden, am Ende der Zeile einen umgekehrten Schrägstrich `\` setzen. Beispiel:

```
DESCRIPTION Dies ist ein PostScript-Drucker in Gebäude 1\Zimmer 123.  
Bei Problemen bitte 555-5555 anrufen.
```

Standarddrucker konfigurieren

Auf den Standarddrucker wird zugegriffen, wenn der Benutzer

- ein Objekt an das Druckersymbol im Bedienfeld übergibt
- ein Objekt im Dateimanager auswählt und 'Drucken' im Menü 'Ausgewählt' oder im (temporär eingeblendeten) Menü des Symbols anklickt.
- über Anwendungen druckt, die den Standarddrucker verwenden.

▼ Standarddrucker ändern

Der Standarddrucker wird für alle Benutzer geändert, indem

1. **die Datei `/etc/dt/config/Xsession.d/0010.dtpaths` geöffnet wird.**

Ist `/etc/dt/config/Xsession.d/0010.dtpaths` nicht vorhanden, muß sie aus `/usr/dt/config/Xsession.d/0010.dtpaths` kopiert werden.

2. **in der Zeile `LPDEST=drucker` für `drucker` der neue Standarddrucker angegeben wird.**

Ist die Zeile nicht vorhanden, eine Zeile `LPDEST=drucker` hinzufügen, in der für `drucker` der Name des Druckers der als Standarddrucker verwendet werden soll, angegeben wird.

3. **die Benutzer sich ab- und wieder anmelden.**

Der Standarddrucker wird für einen einzelnen Benutzer geändert, indem

- ◆ **ein anderer Drucker aus der Bedientafel 'Persönliche Drucker' in das Bedienfeld kopiert wird.**

Ein anderer Drucker wird als Standarddrucker angegeben, indem

1. **das Home-Verzeichnis angeklickt und die Datei `.dtprofile` geöffnet wird.**

2. **eine Zeile bearbeitet oder hinzugefügt wird, die einen Wert für die Umgebungsvariable `LPDEST` setzt:**

```
LPDEST=drucker_gerät;
```

```
export LPDEST Wird csh verwendet, lautet die Syntax: setenv LPDEST drucker_gerät
```

Beispielsweise setzt die folgende Zeile als Standarddrucker den Drucker mit dem Einheitenamen `laser3d`.

```
LPDEST=laser3d; export LPDEST Wird csh verwendet, lautet die Syntax: setenv LPDEST laser3d
```

Druckerkonzepte

Wird eine Druckanforderung eingeleitet, indem eine Datei einem Druckerbedienelement übergeben wird, läuft im System folgende Prozedur ab:

1. Das System durchsucht die Datentypdatenbank nach einer Definition für das übergebene Objekt.
2. Ist eine eindeutige Druckaktion für den Datentyp vorhanden (in dem Feld ARG_TYPE in der Druckaktion angegeben), wird diese verwendet. Ansonsten wird die Standarddruckaktion (dtp) verwendet. Handelt es sich bei der Datei, beispielsweise, um eine PostScript -Datei , verwendet das System die Druckaktion für PostScript-Dateien. (Diese Aktion ist in /usr/dt/appconfig/types/ *sprache* /dt.dt definiert.) Wurde das Werkzeug 'Aktion erstellen' für diesen Datentyp verwendet, handelt es sich bei dem Druckbefehl um die eindeutige Druckaktion, die verwendet wird, um Dateien mit diesem Datentyp zu drucken.
3. Die Datei wird über das normale UNIX-Drucksubsystem lp an den Drucker geschickt.

Desktop-Suchpfade

Das Desktop verwendet Suchpfade, um die Position von Anwendungen und deren zugehörigen Desktop-Dateien festzustellen.

- „Desktop-Suchpfade und ihre Umgebungsvariablen ” auf Seite 146
- „Wert eines Suchpfads setzen” auf Seite 147
- „Anwendungssuchpfad” auf Seite 149
- „Datenbanksuchpfad (Aktionen/Datentypen) ” auf Seite 153
- „ Symbolsuchpfad” auf Seite 156
- „ Hilfesuchpfad” auf Seite 159
- „Positionen der Suchpfade” auf Seite 161

Das Desktop stellt vier Suchpfade zur Verfügung, die in Tabelle 7-1 beschrieben sind.

TABELLE 9-1 Desktop-Suchpfade

Suchpfad	Beschreibung
Anwendungen	Wird verwendet, um die Position von Anwendungen festzustellen. Der Anwendungsmanager verwendet den Anwendungssuchpfad, um bei der Anmeldung eines Benutzers die oberste Ebene dynamisch zu erstellen.
Datenbank	Wird verwendet, um zusätzliche Positionen für Definitionsdateien für Aktionen und Datentypen (*.dt -Dateien) und Bedienfelddateien (*.fp-Dateien) anzugeben.

TABELLE P-1 Desktop-Suchpfade (fortgesetzt)

Suchpfad	Beschreibung
Symbole	Wird verwendet, um zusätzliche Positionen für Symbole anzugeben.
Hilfedaten	Wird verwendet, um zusätzliche Positionen für Desktop-Hilfedaten anzugeben.

Die Suchpfade können sowohl lokale als auch ferne Verzeichnisse enthalten. Auf diese Weise spielen Suchpfade eine wichtige Rolle in der Netzwerkarchitektur des Desktops. Beispielsweise findet ein System Anwendungen auf einem Anwendungs-Server, weil dieser Anwendungs-Server im Anwendungssuchpfad aufgelistet ist.

Enthält ein Suchpfad eine ferne Position, muß der ferne Zugriff auf diese Position konfiguriert werden. Weitere Informationen hierzu können „Zugriff auf verteilte Dateisysteme konfigurieren “ auf Seite 120 entnommen werden.

Desktop-Suchpfade und ihre Umgebungsvariablen

Die Desktop-Suchpfade werden bei der Anmeldung durch das Desktop-Dienstprogramm `dtsearchpath` erstellt. Dieses Dienstprogramm verwendet eine Kombination aus Umgebungsvariablen und integrierten Positionen, um die Suchpfade zu erstellen.

Die Umgebungsvariablen, die `dtsearchpath` liest, werden als *Eingabevariablen* bezeichnet. Diese Variablen werden vom Systemadministrator oder vom Endbenutzer gesetzt. Die Eingabevariablen verwenden die Benennungskonvention `DTSP*`.

Wird `dtsearchpath` zum Zeitpunkt der Anmeldung ausgeführt, assembliert es die diesen Variablen zugeordneten Werte, fügt integrierte Positionen hinzu, und erstellt Werte für *Ausgabevariablen* . Für jeden Suchpfad gibt es eine Ausgabevariable.

TABELLE 9-2 Umgebungsvariablen des Desktop-Suchpfads

Suchpfad für:	Ausgabeumgebungsvariable	Systemweite Eingabevariable	Persönliche Eingabevariable
Anwendungen	DTAPPSEARCHPATH	DTSPSYSAPPHOSTS	DTSPUSERAPPHOSTS
Datenbank ¹	DTDATABASESEARCHPATH	DTSPSYSDATABASEHOSTS	DTSPUSERDATABASEHOSTS
Symbole	XMICONSEARCHPATH, XMICONBMSEARCHPATH	DTSPSYSICON	DTSPUSERICON
Hilfedaten	DTHELPSEARCHPATH	DTSPSYSHELP	DTSPUSERHELP

1. Aktionen, Datentyp-Definitionen und Definitionen des vorderen Steuerbereichs

Komponenten verwenden die Werte der Ausgabevariablen. Zum Beispiel verwendet der Anwendungsmanager den Wert des Anwendungssuchpfads (DTAPPSEARCHPATH), um die Position von Gruppen von Anwendungen festzustellen.

Wert eines Suchpfads setzen

Suchpfade können entweder für das gesamte System oder für einzelne Benutzer geändert werden. Änderungen werden durchgeführt, indem Werte für die für das System definierten oder persönlichen Eingabevariablen gesetzt werden. Sämtliche Änderungen werden den integrierten Suchpfaden hinzugefügt.

▼ Aktuellen Wert für einen Suchpfad anzeigen (Ausgabevariable)

◆ **Mit dem Befehl `dtsearchpath` die aktuellen Werte für die Suchpfade anzeigen:**

- Um den Wert für den aktuellen Benutzer (der sich gerade anmeldet) zu erhalten, folgenden Befehl eingeben:

```
dtsearchpath -v
```

- Um den Wert für einen anderen Benutzer zu erhalten, folgenden Befehl eingeben:

```
dtsearchpath -u benutzer
```

Zu den Suchpfadwerten gehören diese Variablen:

%H - Wird in `DTHELPPSEARCHPATH` verwendet. Es handelt sich um den Namen des Hilfebandes.

%B - Wird in `XMICONSEARCHPATH` verwendet. Es handelt sich um den Dateinamen ohne Pfad einer Symboldatei.

%M - Die Größe der Symboldatei (`.l`, `.m`, `.s`, `.t`), die in `XMICONSEARCHPATH` verwendet wird.

%L - Wert der Umgebungsvariablen `LANG`.

▼ Persönliche Änderungen an einem Suchpfad vornehmen

1. Das Verzeichnis *Home-Verzeichnis* `/.dtprofile` zum Bearbeiten öffnen.

2. Eine Zeile hinzufügen oder Bearbeiten, in der ein Wert für die persönliche Eingabevariable definiert wird.

Beispielsweise fügt die folgende Zeile eine Position zum persönlichen Anwendungssuchpfad des Benutzers hinzu:

```
DTSPUSERAPPHOSTS=/projects1/editors
```

3. Damit die Änderungen angezeigt werden, muß der Benutzer sich ab- und wieder anmelden.

▼ Systemweite Änderungen an einem Suchpfad vornehmen

1. Als Benutzer 'root' anmelden.

2. Ist die Datei `/etc/dt/config/Xsession.d/0010.dtpaths` nicht vorhanden, muß sie durch Kopieren von

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

 erstellt werden.

3. `/etc/dt/Xsession.d/0010.paths` zum Bearbeiten öffnen. Eine Zeile in der ein Wert für die systemweite Eingabevariable definiert ist, hinzufügen oder bearbeiten.

Beispielsweise fügt die folgende Zeile eine Position zum systemweiten Hilfesuchpfad hinzu:

```
DTSPSYSHELP=/applications/helpdata
```

4. Alle Benutzer im System darüber informieren, daß sie sich ab- und wieder anmelden müssen, damit die Änderungen angezeigt werden.

Anwendungssuchpfad

Bei dem Anwendungssuchpfad handelt es sich um den primären Suchpfad, den das Desktop verwendet, um die Position von Anwendungen im lokalen System und auf Anwendungs-Servern im gesamten Netzwerk festzustellen.

Werden Positionen zum Anwendungssuchpfad hinzugefügt, werden die übrigen Suchpfade (Datenbank-, Symbol- und Hilfesuchpfad) automatisch aktualisiert, um die entsprechenden Positionen für diese Daten anzugeben. Auf diese Weise stellt der Anwendungssuchpfad eine einfache Verwaltung für Anwendungen und deren Desktop-Konfigurationsdateien zur Verfügung. Weitere Informationen hierzu können dem Abschnitt „Wie der Anwendungssuchpfad sich auf die Datenbank-, Symbol- und Hilfesuchpfade auswirkt“ auf Seite 152 entnommen werden.

Standardanwendungssuchpfad

Der Standardanwendungssuchpfad enthält persönliche, systemweite und integrierte Positionen. Der Standardwert für *sprache* ist C.

TABELLE 9-3

Persönliche Position	Home-Verzeichnis / .dt/appmanager
Systemweite Position	/etc/dt/appconfig/appmanager/ <i>sprache</i>
Integrierte Position	/usr/dt/appconfig/appmanager/ <i>sprache</i>

Umgebungsvariablen des Anwendungssuchpfads

Der Anwendungssuchpfad wird aus integrierten Positionen und den folgenden Eingabevariablen assembliert:

TABELLE 9-4

DTSPSYSAPPHOSTS	Eingabevariable für den systemweiten Anwendungssuchpfad
DTSPUSERAPPHOSTS	Eingabevariable für den persönlichen Anwendungssuchpfad

Der assemblierte Suchpfad wird durch die Ausgabevariable DTAPPSEARCHPATH angegeben.

Syntax für die Eingabevariablen des Anwendungssuchpfads

Die Syntax für die Variablen DTSPSYSAPPHOSTS und DTSPUSERAPPHOSTS lautet:

VARIABLE=position
[, *position* . . .]

wobei *position* folgende Syntax haben kann:

TABELLE 9-5

<i>/pfad</i>	Gibt ein Verzeichnis im lokalen System (Session-Server) an. Diese Syntax wird verwendet, um ein lokales Verzeichnis hinzuzufügen.
<i>hostname</i> :	Gibt das systemweite Verzeichnis <code>/etc/dt/appconfig/appmanager/ <i>sprache</i></code> auf dem System <i>hostname</i> an. Diese Syntax wird verwendet, um einen Anwendungs-Server hinzuzufügen.
<i>hostname: /pfad</i>	Gibt ein Verzeichnis auf einem fernen System <i>hostname</i> an.
<code>localhost</code> :	Gibt die lokale systemweite Position an. Dieses Schlüsselwort wird verwendet, um den Rang der lokalen systemweiten Position zu ändern. Weitere Informationen hierzu können dem Abschnitt „Rangfolge der systemweiten lokalen Positionen ändern“ auf Seite 151 entnommen werden.

Wie der Wert des Anwendungssuchpfads assembliert wird

Der Wert des Anwendungssuchpfads (DTAPPSEARCHPATH) wird erstellt, indem folgende Positionen, die ihrem Rang nach aufgelistet sind, assembliert werden:

- Positionen, die über die Variable DTSPUSERAPPHOSTS angegeben sind
- die standardmäßige persönliche Position: *Home-Verzeichnis*/`.dt/appmanager`
- die Standardposition: `/etc/dt/appconfig/appmanager/ sprache`
- Positionen, die über die Variable DTSPSYSAPPHOSTS angegeben sind
- `/usr/dt/appconfig/appmanager/ sprache`

Die Syntax:

hostname:

wird erweitert, um das Verzeichnis `/etc/dt/appconfig/appmanager` auf dem System *hostname* anzugeben.

Rangfolge der systemweiten lokalen Positionen ändern

Standardmäßig hat die lokale systemweite Position

(`/etc/dt/appconfig/appmanager/sprache`) Vorrang vor fernen Positionen. Auf

diese Weise haben lokale Gruppen von Anwendungen Vorrang vor fernen Gruppen von Anwendungen mit demselben Namen. Verfügen beispielsweise sowohl lokale als auch ferne Systeme über Gruppen von Anwendungen mit dem Namen Drucker (/etc/dt/appconfig/appmanager/ *sprache*/Printers), wird die lokale Gruppe verwendet.

Die Eingabevariablen des Anwendungssuchpfads stellen eine Syntax, die den Vorrang der lokalen systemweiten Gruppen von Anwendungen angibt, zur Verfügung:

```
localhost:
```

Angenommen, ein System soll auf die Anwendungs-Server *SysA*, *SysB* und *SysC* zugreifen und die systemweiten Gruppen von Anwendungen auf *SysB* sollen Vorrang vor jeder lokalen Gruppe mit demselben Namen haben.

Dies wird erreicht, indem der folgende Wert für DTSPSYSAPPHOSTS gesetzt wird:

```
DTSPSYSAPPHOSTS=SysB: ,localhost: ,SysA: ,SysC:
```

Wie der Anwendungssuchpfad sich auf die Datenbank-, Symbol- und Hilfesuchpfade auswirkt

Bei Hinzufügungen zum Anwendungssuchpfad werden die entsprechenden Positionen automatisch auch zu den Datenbank-, Symbol- und Hilfesuchpfaden hinzugefügt. Auf diese Weise kann ein Anwendungs-Server einem Suchpfad hinzugefügt werden, indem lediglich die Eingabevariable für den Anwendungssuchpfad gesetzt wird.

Wird beispielsweise DTSPSYSAPPHOSTS wie folgt gesetzt:

```
DTSPSYSAPPHOSTS=servera:
```

sind davon folgende Suchpfade betroffen:

Suchpfad Dem Suchpfad hinzugefügtes Verzeichnis

TABELLE 9-6

Anwendungen	servera:/etc/dt/appconfig/ appmanager/ sprache
Datenbank	servera:/etc/dt/appconfig/types/ sprache
Symbol	servera:/etc/dt/appconfig/icons/ sprache
Hilfe	servera:/etc/dt/appconfig/help/ sprache

So ähnlich sind, wenn DTSPSYSAPPHOSTS wie folgt gesetzt wird:

DTSPSYSAPPHOSTS=/projects1/apps

folgende Suchpfade betroffen: Suchpfad Dem Suchpfad hinzugefügtes Verzeichnis

TABELLE 9-7

Anwendungen	/projects1/apps/appmanager/ sprache
Datenbank	/projects1/apps/types/ sprache
Symbol	/ projects1/apps/icons/ sprache
Hilfe	/ projects1/apps/help/ sprache

Datenbanksuchpfad (Aktionen/ Datentypen)

Der Datenbanksuchpfad weist das Desktop an, angegebene Positionen nach Dateien zu durchsuchen, die folgendes enthalten:

- Definitionen für Aktionen und Datentypen (*.dt -Dateien)
- Bedienfelddefinitionen (*.fp-Dateien).

Möglicherweise muß der Benutzer den Datenbanksuchpfad ändern, wenn er einen Datenbank-Server erstellt oder eine lokale Position für Datenbankdateien hinzufügt.

Standarddatenbanksuchpfad

Der Standarddatenbanksuchpfad enthält persönliche, systemweite und integrierte Positionen. Der Standardwert für *sprache* ist C.

TABELLE 9-8

Persönliche Position	Home-Verzeichnis /.dt/types
Systemweite Position	/etc/dt/appconfig/types/ <i>sprache</i>
Integrierte Position	/usr/dt/appconfig/types/ <i>sprache</i>

Wie der Anwendungssuchpfad sich auf den Datenbanksuchpfad auswirkt

Wird eine Position zum Anwendungssuchpfad hinzugefügt, wird das entsprechende Datenbankunterverzeichnis automatisch dem Datenbanksuchpfad hinzugefügt. (Weitere Informationen hierzu können dem Abschnitt „Wie der Anwendungssuchpfad sich auf die Datenbank-, Symbol- und Hilfesuchpfade auswirkt“ auf Seite 152 entnommen werden.)

Wird beispielsweise der Anwendungs-Server `hosta`: dem Anwendungssuchpfad hinzugefügt, wird das Verzeichnis `hosta:/etc/dt/appconfig/types/ sprache` automatisch dem Datenbanksuchpfad hinzugefügt.

Umgebungsvariablen des Datenbanksuchpfads

Der Datenbanksuchpfad wird aus den integrierten Positionen und folgenden Eingabevariablen assembliert:

TABELLE 9-9

DTSPSYSDATABASEHOSTS	Eingabevariable für den systemweiten Datenbanksuchpfad
DTSPUSERDATABASEHOSTS	Eingabevariable für den persönlichen Datenbanksuchpfad

Diese Eingabevariablen verwenden, um Positionen außerhalb des Anwendungssuchpfads anzugeben.

Der assemblierte Datenbanksuchpfad wird durch die Ausgabevariable DTDATABASESEARCHPATH angegeben.

Syntax für die Eingabevariablen des Datenbanksuchpfads

Die Syntax für die Variablen DTSPSYSDATABASEHOSTS und DTSPUSERDATABASEHOSTS lautet:

VARIABLE=position
[, *position* . . .]

wobei *position* folgende Syntax haben kann:

TABELLE 9-10

<i>/pfad</i>	Gibt ein Verzeichnis im lokalen System (Session-Server) an. Diese Syntax wird verwendet, um ein lokales Verzeichnis hinzuzufügen.
<i>hostname</i> :	Gibt das systemweite Verzeichnis <i>/etc/dt/appconfig/types/ sprache</i> auf dem System <i>hostname</i> an.
<i>hostname</i> : <i>/pfad</i>	Gibt ein Verzeichnis auf dem fernen System <i>hostname</i> an.

Wie der Datenbanksuchpfad assembliert wird

Der Wert des Datenbanksuchpfads (DTDATABASESEARCHPATH) wird erstellt, indem folgende Positionen, die ihrem Rang nach aufgelistet sind, assembliert werden:

- Positionen, die über die Variable DTSPUSERDATABASEHOSTS angegeben sind
- Positionen, die von der Variablen DTSPUSERAPPHOSTS abgeleitet werden.
- die standardmäßige persönliche Position: *Home-Verzeichnis*/.dt/types
- die Standardposition: /etc/dt/appconfig/types/ *sprache*
- Positionen, die über die Variable DTSPSYSDATABASEHOSTS angegeben werden
- Positionen, die von der VariablenDTSPSYSAPPHOSTS abgeleitet werden
- /usr/dt/appconfig/types/ *sprache*

Die Syntax:

hostname:

wird erweitert, um das Verzeichnis /etc/dt/appconfig/types auf dem System *hostname* anzugeben.

Symbolsuchpfad

Der Symbolsuchpfad weist das Desktop an, angegebene Positionen nach Dateien zu durchsuchen, die Bitmap- und Pixmap-Abbilddateien für das Desktop enthalten.

Standardsymbolsuchpfad

Der Standardsymbolsuchpfad enthält persönliche, systemweite und integrierte Positionen. Der Standardwert für *sprache* ist C.

TABELLE 9-11

Persönliche Position	Home-Verzeichnis /.dt/icons
Systemweite Position	/etc/dt/appconfig/icons/ <i>sprache</i>
Integrierte Position	/usr/dt/appconfig/icons/ <i>sprache</i>

Wie der Anwendungssuchpfad sich auf den Symbolsuchpfad auswirkt

Wird eine Position zum Anwendungssuchpfad hinzugefügt, wird das entsprechende Symbolunterverzeichnis automatisch dem Symbolsuchpfad hinzugefügt. (Weitere Informationen hierzu können dem Abschnitt „Wie der Anwendungssuchpfad sich auf die Datenbank-, Symbol- und Hilfesuchpfade auswirkt“ auf Seite 152 entnommen werden.)

Wird beispielsweise der Anwendungs-Server `hosta:` dem Anwendungssuchpfad hinzugefügt, wird das Verzeichnis `hosta:/etc/dt/appconfig/icons/ sprache` automatisch dem Symbolsuchpfad hinzugefügt.

Umgebungsvariablen des Symbolsuchpfads

Der Symbolsuchpfad wird aus den integrierten Positionen und folgenden Eingabevariablen assembliert:

TABELLE 9-12

DTSPSYSICON	Eingabevariable für den systemweiten Symbolsuchpfad
DTSPUSERICON	Eingabevariable für den persönlichen Symbolsuchpfad

Diese Eingabevariablen verwenden, um Positionen außerhalb des Anwendungssuchpfads anzugeben.

Der assemblierte Symbolsuchpfad wird durch zwei Ausgabevariablen angegeben:

TABELLE 9-13

XMICONSEARCHPATH	Wird von Farbbildschirmen verwendet.
XMICONBMSEARCHPATH	Wird von Monochrombildschirmen verwendet.

Syntax für die Eingabevariablen des Symbolsuchpfads

Die Syntax für die Variablen DTSPSYSICON und DTSPUSERICON lautet:

VARIABLE=position
[, *position . . .*]

wobei *position* folgende Syntax haben kann:

TABELLE 9-14

<i>/pfad</i>	Gibt ein Verzeichnis im lokalen System (Session-Server) an. Diese Syntax wird verwendet, um ein lokales Verzeichnis hinzuzufügen.
--------------	---

Soll eine Position in einem anderen System angegeben werden, muß der Netzwerkdateiname, z. B. */nfs/servera/projects/icons* verwendet werden.

Wie der Symbolsuchpfad assembliert wird

Der Wert des Symbolsuchpfads (XMICONSEARCHPATH und XMICONBMSEARCHPATH) wird erstellt, indem folgende Positionen, die ihrem Rang nach aufgelistet sind, assembliert werden:

- Positionen, die über die Variable DTSPUSERICON angegeben sind

- Positionen, die von der Variablen `DTSPUSERAPPHOSTS` abgeleitet werden
- die standardmäßige persönliche Position: *Home-Verzeichnis*/`.dt/icons`
- die Standardposition: `/etc/dt/appconfig/icons/ sprache`
- Positionen, die über die Variable `DTSPSYSICON` angegeben sind
- Positionen, die von der Variablen `DTSPSYSAPPHOSTS` abgeleitet werden
- `/usr/dt/appconfig/icons/ sprache`

Farb- und Monochromsuchpfade unterscheiden sich lediglich in der Rangfolge, die Pixmap- und Bitmap-Dateien zugeordnet wird. Die Variable `XMICONSEARCHPATH` listet Pixmap-Dateien vor Bitmap-Dateien, die Variable `XMICONBMSEARCPATH` listet Bitmap-Dateien vor Pixmap-Dateien.

Hilfesuchpfad

Der Hilfesuchpfad weist das Desktop an, angegebene Positionen nach Dateien mit Hilfeinformationen zu durchsuchen, die auf dem System registriert werden.

Standardhilfesuchpfad

Der Standardhilfesuchpfad enthält persönliche, systemweite und integrierte Positionen. Der Standardwert für *sprache* ist `C`.

TABELLE 9–15

Persönliche Position	<i>Home-Verzeichnis</i> / <code>.dt/help</code>
Systemweite Position	<code>/etc/dt/appconfig/.dt/help/ sprache</code>
Integrierte Position	<code>/usr/dt/appconfig/help/ sprache</code>

Wie der Anwendungssuchpfad sich auf den Hilfesuchpfad auswirkt

Wird eine Position zum Anwendungssuchpfad hinzugefügt, wird das entsprechende Hilfeunterverzeichnis automatisch dem Hilfesuchpfad hinzugefügt (Weitere Informationen hierzu können dem Abschnitt „Wie der Anwendungssuchpfad sich auf die Datenbank-, Symbol- und Hilfesuchpfade auswirkt“ auf Seite 152 entnommen werden.)

Wird beispielsweise der Anwendungs-Server `hosta:` dem Anwendungssuchpfad hinzugefügt, wird das Verzeichnis `hosta:/etc/dt/appconfig/help/ sprache` automatisch dem Hilfesuchpfad hinzugefügt.

Umgebungsvariablen des Hilfesuchpfads

Der Hilfesuchpfad wird aus den integrierten Positionen und folgenden Eingabevariablen assembliert:

`DTSPSYSHELP` — Eingabevariable für den systemweiten Hilfesuchpfad

`DTSPUSERHELP`— Eingabevariable für den persönlichen Hilfesuchpfad

Diese Eingabevariablen verwenden, um Positionen außerhalb des Anwendungssuchpfads anzugeben.

Der assemblierte Datenbanksuchpfad wird durch die Ausgabevariable `DTHELPSEARCHPATH` angegeben.

Syntax für die Eingabevariablen des Hilfesuchpfads

Die Syntax für die Variablen `DTSPSYSHELP` und `DTSPUSERHELP` lautet:

```
VARIABLE=position  
[ , position . . . ]
```

wobei *position* folgende Syntax haben kann:

<i>/pfad</i>	Gibt ein Verzeichnis im lokalen System (Session-Server) an. Diese Syntax wird verwendet, um ein lokales Verzeichnis hinzuzufügen.
--------------	---

Soll eine Position in einem anderen System angegeben werden, muß der Netzwerkdateiname, z. B. `/nfs/servera/projects/help` verwendet werden.

Wie der Hilfesuchpfad assembliert wird

Der Wert des Hilfesuchpfads (DTHELPPSEARCHPATH) wird erstellt, indem folgende Positionen, die ihrem Rang nach aufgelistet sind, assembliert werden:

- Positionen, die über die Variable `DTSPUSERHELP` angegeben werden
- Positionen, die von der Variablen `DTSPUSERAPPHOSTS` abgeleitet werden
- die standardmäßige persönliche Position: *Home-Verzeichnis*/`.dt/help`
- die Standardposition: `/etc/dt/appconfig/help/ sprache`
- Positionen, die über die Variable `DTSPSYSHELP` angegeben werden
- Positionen, die von der Variablen `DTSPSYSAPPHOSTS` abgeleitet werden
- `/usr/dt/appconfig/help/ sprache`

Positionen der Suchpfade

Die Ausgabevariablen enthalten sowohl Einträge für sprachenspezifische Positionen wie auch für Standardpositionen (C).

Beispielsweise lautet der Standardanwendungssuchpfad:

TABELLE 9-17

Home-Verzeichnis/.dt/appmanager

/etc/dt/appconfig/appmanager/ sprache

/etc/dt/appconfig/appmanager/C

/usr/dt/appconfig/appmanager/ sprache

/usr/dt/appconfig/appmanager/C

wobei *sprache* den Wert der Umgebungsvariablen LANG darstellt.

Für jeden Bereich (systemweit oder integriert) hat die sprachenspezifische Position Vorrang vor der Standardposition.

Einführung in Aktionen und Datentypen

Bei *Aktionen* und *Datentypen* handelt es sich um wichtige Komponenten bei der Integrierung von Anwendungen auf dem Desktop. Sie stellen eine Möglichkeit zur Verfügung, eine Benutzerschnittstelle zum Starten von Anwendungen und zum Bearbeiten von deren Datendateien zu benutzen.

- „Einführung in Aktionen“ auf Seite 164
- „Einführung in Datentypen“ auf Seite 172

In diesem Kapitel wird das Konzept von Aktionen und Datentypen erläutert. Es wird beschrieben,

- warum es günstig ist, Aktionen und Datentypen für Anwendungen zu erstellen.
- welche Beziehung zwischen Aktionen und Datentypen besteht.
- in welcher Beziehung Aktionen und Datentypen zum Drucken über das Desktop stehen.

Weiterführende Informationen

Die Verfahren und Regeln für das Erstellen von Aktionen und Datentypen werden in diesem Handbuch in drei Kapiteln behandelt:

- In Kapitel 11 wird erläutert, wie Aktionen und Datentypen mit der Desktop-Anwendung 'Aktion erstellen' erstellt werden.

Mit 'Aktion erstellen' können Aktionen und Datentypen für die meisten Anwendungen erstellt werden, ohne daß der Benutzer die Syntaxregeln für deren Definitionen lernen muß.

- In Kapitel 12 und Kapitel 13 wird erläutert, wie Aktionen und Datentypen manuell erstellt werden, indem Konfigurationsdateien erstellt und bearbeitet werden.

Wenn der Benutzer erweiterte Funktionen verwenden möchte, die durch 'Aktion erstellen' nicht unterstützt werden, ist es notwendig, daß er Aktionen und Datentypen manuell erstellt.

Einführung in Aktionen

Bei Aktionen handelt es sich um Instruktionen, mit denen Desktop-Aufgaben wie die Ausführung von Anwendungen und das Öffnen von Datendateien automatisiert werden. Aktionen funktionieren ähnlich wie Anwendungsmakros oder Programmierfunktionen. Jede Aktion verfügt über einen Namen, der für die Ausführung der Aktion verwendet wird.

Sobald eine Aktion definiert wird, kann sie dazu verwendet werden, die Desktop-Benutzerschnittstelle anzupassen, so daß Aufgaben einfacher ausgeführt werden können. Das Desktop verfügt über Benutzerschnittstellenkomponenten wie zum Beispiel Symbole, Bedienelemente im Bedienfeld und Menüelemente für Aktionen zur Verfügung zu stellen.

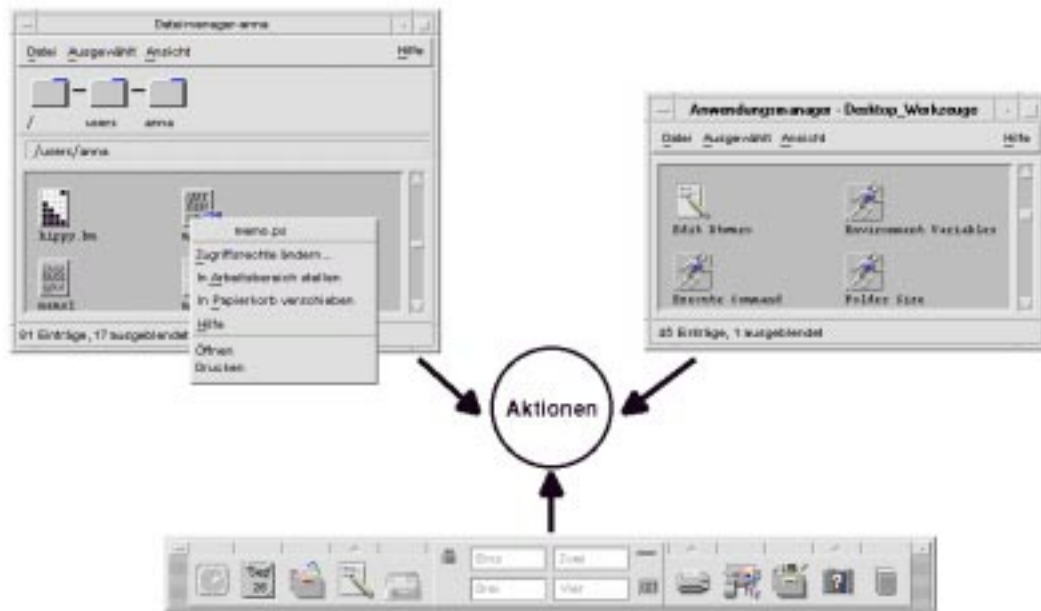


Abbildung 10-1 Verwendungsmöglichkeiten für Aktionen

Beispielsweise enthält die Gruppe von Anwendungen 'Desktop_Werkzeuge' im Anwendungsmanager Symbole, über die verschiedene Dienstprogramme gestartet werden.



Abbildung 10-2 Symbole für Aktionen in der Gruppe von Anwendungen Desktop_Werkzeuge

Jedes dieser Symbole führt eine Aktion aus, wenn auf diesem Symbol doppelt geklickt wird. Das folgende Beispiel zeigt einen Teil der Definition der Aktion, die ausgeführt wird, wenn der Benutzer auf dem Symbol mit der Bezeichnung Xwd Display doppelt klickt. Die Aktion ist in der Konfigurationsdatei

`/usr/dt/appconfig/types/sprache /xclients.dt` definiert.

TABELLE 10-1

ACTION Xwd	
{	
LABEL	Xwd Display
TYPE	COMMAND
EXEC_STRING	<code>/usr/bin/X11/xwd -noclick -in \</code>
	<code>%(File)Arg_1"Xwd File To Display:"%</code>

TABELLE P-1 (fortgesetzt)

ACTION Xwud

} ...

Der Befehl im Feld EXEC_STRING der Aktion wird ausgeführt, wenn der Benutzer dieses Symbol doppelklickt.

Das Bedienfeld verwendet ebenfalls Aktionen. Das folgende Beispiel zeigt einen Teil der Definition des Bedienelements mit der Bezeichnung 'Terminal' in der Bedientafel 'Persönliche Anwendungen'. Das Bedienelement ist in der Konfigurationsdatei /usr/dt/appconfig/types/*sprache*/dtwm.fp definiert.

TABELLE 10-2

CONTROL Term

```
{  
ICON          Fpterm  
LABEL         Terminal  
PUSH_ACTION   Dtterm  
              ...  
}
```

Das Feld PUSH_ACTION gibt an, welche Aktion ausgeführt werden soll, wenn der Benutzer das Bedienelement anklickt. In diesem Fall handelt es sich um eine Aktion mit dem Namen Dtterm.

Daneben werden Aktionen häufig in Menüs verwendet. Datendateien verwenden normalerweise Aktionen in ihrem Menü 'Ausgewählt' im Dateimanager. Zum Beispiel verfügen XWD-Dateien (Dateien, deren Namen die Erweiterung .xwd oder .wd haben) über eine Aktion 'Öffnen', die das Anzeigebild anzeigt, indem die Aktion 'Xwud' ausgeführt wird.

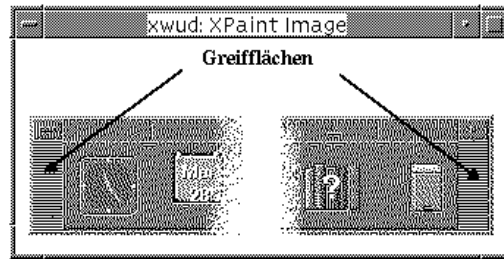
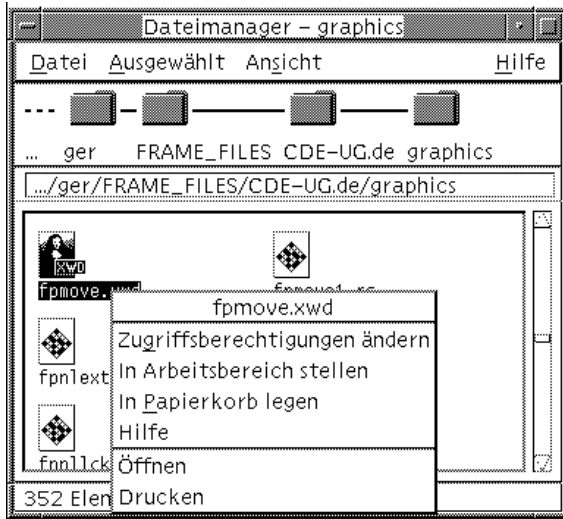


Abbildung 10-3 Aktion 'Öffnen' für Dateien des Datentyps XWD

Die Aktionen im Menü 'Ausgewählt' werden in der Datentypdefinition für XWD-Dateien angegeben. Die Definition befindet sich in der Konfigurationsdatei `/usr/dt/appconfig/types/sprache /xclients.dt`.

TABELLE 10-3

DATA_ATTRIBUTES XWD

```
{
  ACTIONS          Open,Print
  ICON             Dtxwd
  ...
}
```

Der Datentyp XWD und die zugehörigen Aktionen 'Öffnen' und 'Drucken', werden im Abschnitt erläutert.

Wie Aktionen Symbole für Anwendungen erstellen

Als Beispiel soll das Symbol Xwd Display in der Gruppe von Anwendungen Desktop_Werkzeuge betrachtet werden. Wird auf diesem Symbol doppelt geklickt, wird der X-Client `xwud` ausgeführt. Dennoch stellt dieses Symbol nicht direkt die ausführbare Datei `/usr/bin/X11/xwud` dar.

Das Symbol Xwd Display wird in der Gruppe von Anwendungen angezeigt, weil in diesem Verzeichnis eine Datei mit dem Namen `xwud` steht (siehe Abbildung 10-4). Diese Datei stellt eine zugrundeliegende AKTION mit demselben Namen, nämlich Xwd dar. In der Aktionsdefinition ist der Aktionsname der Name, der nach dem Schlüsselwort ACTION angegeben ist:

TABELLE 10-4

ACTION Xwd

```
{
```

LABEL	Xwd Display
TYPE	COMMAND
WINDOW_TYPE	NO_STDIO
EXEC_STRING	<code>/usr/bin/X11/xwud</code> <code>-noclick -in \</code>
	<code>%(File)Arg_1"Xwd</code> <code>Datei anzeigen:"%</code>
DESCRIPTION	Die Xwd-Anzeige (Xwd) 'Aktion Xwd-Anzeige' zeigt eine xwd-Datei, die mit der Aktion 'Xws Capture' (Xwd) erstellt wurde. Es wird der Befehl 'xwud' verwendet.

```
}
```

Die Datei wird als *Aktionsdatei* bezeichnet, weil sie eine Aktion darstellt. Eine Datei ist eine Aktionsdatei, wenn es sich um eine ausführbare Datei mit demselben Namen wie eine Aktion handelt. Das zugehörige Symbol im Anwendungsmanager (oder im Dateimanager) wird *Aktionssymbol* oder *Anwendungssymbol* genannt, weil durch Doppelklicken auf diesem Symbol eine Anwendung gestartet wird.

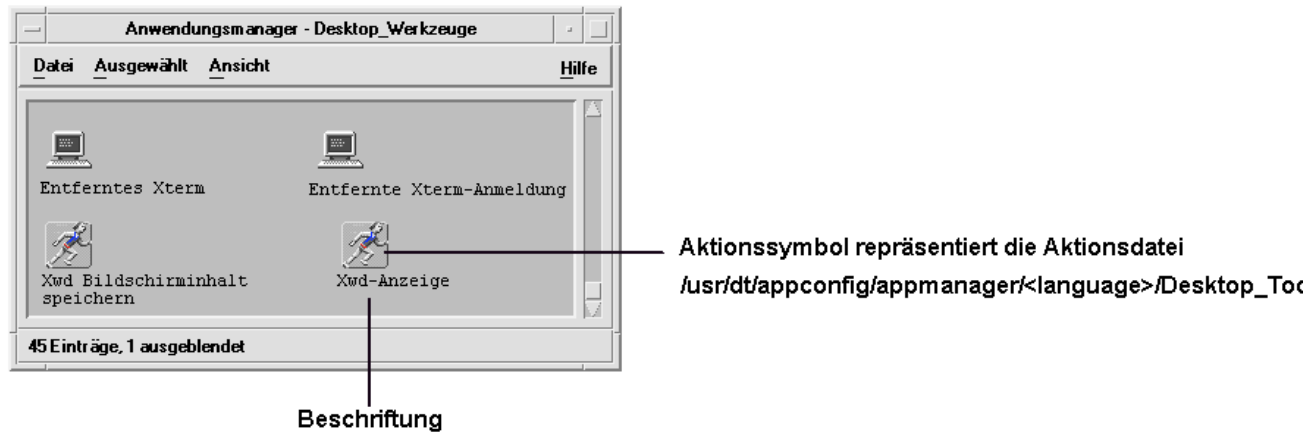


Abbildung 10-4 Anwendungs-(Aktions-)symbol, das eine Aktionsdatei darstellt

Stellt der Anwendungsmanager eine ausführbare Datei fest, durchsucht er die Datenbank für Aktionen, um festzustellen, ob Aktionen vorhanden sind, deren Namen dem Dateinamen entsprechen. Wird eine Entsprechung gefunden, weiss der Anwendungsmanager, daß es sich bei der Datei um eine Aktionsdatei handelt.

Der Inhalt der Aktionsdatei ist irrelevant. Aktionsdateien enthalten normalerweise Kommentare mit der Beschreibung ihrer Desktop-Funktionen.

Hinweis - Bei einer *Aktionsdatei* handelt es sich nicht um eine *Definitionsdatei für Aktionen*. Die *Aktionsdatei* ist eine Datei mit demselben Namen wie die Aktion. Sie wird verwendet, um das *Anwendungssymbol* im Dateimanager oder im Anwendungsmanager zu erstellen. Die *Definitionsdatei für Aktionen* ist die Datei mit dem Namen *name* .dt, die die Definition der Aktion enthält.

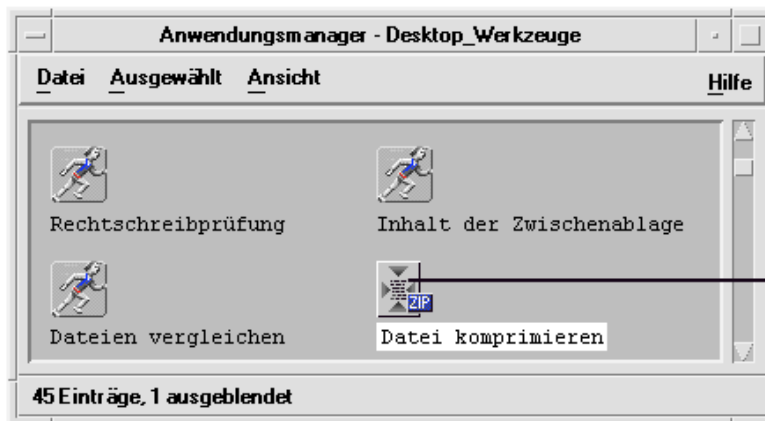
Sobald das Desktop festgestellt hat, daß es sich bei einer Datei um eine Aktionsdatei handelt, wird die zugrundeliegende Aktionsdefinition verwendet, um das Erscheinungsbild und das Verhalten der Aktionsdatei zu definieren.

- Im Feld EXEC_STRING wird das Verhalten des Anwendungssymbols angegeben. Im Fall des Symbols Xwd Display wird im Feld EXEC_STRING angegeben, daß das Aktionssymbol den X-Client xwud mit bestimmten Befehlszeilenargumenten ausführt.
- Im Feld LABEL wird die Bezeichnung für das Anwendungssymbol angegeben.
- Im Feld DESCRIPTION wird der Text angegeben, der angezeigt wird, wenn der Benutzer die Kontexthilfe aufruft.
- Das Anwendungssymbol Xwud verwendet das Standardsymbolbild für Aktionen, da die Aktionsdefinition kein Feld ICON enthält, in dem ein anderes Abbild definiert ist.

Im Gegensatz dazu verwendet das Symbol mit der Bezeichnung Compress File ein anderes Symbolbild, da die zugrundeliegende Aktionsdefinition ein Feld `ICON` enthält:

TABELLE 10-5

Beispiel:	
ACTION Compress	
LABEL	Datei komprimieren
ICON	Dtcmprs
	...
	}



Symbolbild wird durch
SYMBOL-Feld festgelegt

Abbildung 10-5 Symbolbild, das durch das Feld `ICON` in der Aktionsdefinition angegeben wird

Die Aktion `Xwud` wird als *Befehlsaktion* bezeichnet, da ihre Definition den auszuführenden Befehl (`EXEC_STRING`) enthält. Das Feld `TYPE` in der Aktionsdefinition definiert den Aktionstyp.

Zunächst wird das Symbol `Xwd Display` in der Gruppe von Anwendungen `Desktop_Werkzeuge` angezeigt. Der Benutzer kann jedoch zusätzliche Exemplare des Aktionssymbols in jedem Verzeichnis erstellen, für das er Schreibberechtigung hat. Solange die Aktionsdefinition für `Xwud` Bestandteil der Datenbank ist, wird jede

ausführbare Datei mit dem Namen `xwud`, die der Benutzer erstellt, eine Aktionsdatei sein, die diese Aktion darstellt, und das zugehörige Symbol im Dateimanager oder Anwendungsmanager kann verwendet werden, um die Aktion auszuführen.

Wie Aktionen Datendateien als Argumente verwenden

Ein *Argument* ist das Objekt, normalerweise eine Datei, an dem der Befehl ausgeführt wird. Aktionen können so definiert werden, daß sie Dateiangenome akzeptieren.

Beispielsweise gibt das Feld `EXEC_STRING` der Aktion `Xwud` an, daß ein Dateiangenome erforderlich ist:

TABELLE 10-6

<code>EXEC_STRING</code>	<code>/usr/bin/X11/xwud -noclick -in \ %(File)Arg_1"Xwd Datei anzeigen:"%</code>
--------------------------	--

Der Term `Arg` steht für das Wort *Argument*. Die Syntax `Arg_1` bedeutet das erste Argument, und `(File)` bedeutet, daß die Aktion dieses Argument als eine Datei behandelt.

Die einfachste Weise, in der ein Benutzer ein Dateiangenome zur Verfügung stellen kann, besteht darin, eine Datendatei dem Anwendungssymbol zu übergeben. Das Desktop stellt den Pfad der übergebenen Datei fest und setzt diesen in der Befehlszeile anstelle des Texts zwischen den Symbolen `% (File)Arg_1"Xwd Datei anzeigen:"%`. Dann wird folgender Befehl ausgeführt:

```
/usr/bin/X11/xwud -noclick -in datei_pfad
```

Wenn der Benutzer auf dem Anwendungssymbol doppelt klickt, stellt das Desktop im Feld `EXEC_STRING` fest, daß ein Dateiangenome erforderlich ist, und zeigt ein Dialogfenster an, in dem der Benutzer aufgefordert wird, einen Dateinamen oder Pfad einzugeben. Im Fall der Aktion `Xwud` lautet die Aufforderung:

```
Xwd Datei anzeigen:
```

Der Dateiname oder Pfad, den der Benutzer eingibt, wird als Dateiangenome verwendet.

Zusätzliche Verwendung für Aktionen

Zusätzlich zum Starten von Anwendungen werden Aktionen im gesamten Desktop verwendet, um Funktionalität in folgenden Bereichen herzustellen:

- im Bedienfeld

Die Definition für ein Bedienelement im Bedienfeld enthält Felder, in denen die Aktion angegeben wird, die ausgeführt wird, wenn der Benutzer das Bedienelement anklickt oder dem Bedienelement eine Datei übergibt. Weitere Informationen hierzu können dem Abschnitt entnommen werden

- in Menüs

Die Syntax von Fenstermenü- und Arbeitsbereichsmenüdefinitionen ermöglicht dem Benutzer, die Aktion anzugeben, die über ein Menüelement ausgeführt wird. Weitere Informationen hierzu können dem Abschnitt und der Hilfeseite dtwmrc(4) entnommen werden.

- bei der Kommunikation zwischen Anwendungen

Eine Anwendung kann so eingerichtet werden, daß sie mit einem speziellen Typ von Aktion, nämlich ToolTalk message (TT_MSG), Informationen sendet und empfängt TT_MSG-Aktionen werden in der Entwicklerumgebungsdokumentation für das Desktop beschrieben.

Einführung in Datentypen

Erstellt der Benutzer eine neue Datendatei, kann das Erscheinungsbild und das Verhalten des Symbols der Datei im Dateimanager je nach dem Typ der Datendatei sehr unterschiedlich sein. Diese Fähigkeit, ein angepaßtes Erscheinungsbild und angepaßtes Verhalten für Dateien und Verzeichnisse zu erstellen, wird über den Datentypmechanismus des Desktop zur Verfügung gestellt.

Was ist ein Datentyp?

Bei einem Datentyp handelt es sich um ein Konstrukt, das innerhalb der Desktop-Datenbank definiert ist. Das folgende Beispiel zeigt die Definition des Datentyps XWD. Die Definition befindet sich in der Konfigurationsdatei

```
/usr/dt/appconfig/types/sprache /xclients.dt.
```

```
DATA_ATTRIBUTES XWD
{
ACTIONS          Open,Print
ICON              Dtxwd
```

```

NAME_TEMPLATE %s.xwd
MIME_TYPE      application/octet-stream
SUNV3_TYPE     xwd-file
DESCRIPTION    Diese Datei enthält ein Grafikbild im Format XWD. Diese Dateien werden
                erstellt, wenn mit der Aktion 'XwdCapture' Bilder von Fensterausschnitten
                gemacht werden. Der Datentyp wird XWD genannt. XWD-Dateinamen enden
                mit '.xwd' oder '.wd'. Jede Datentypdefinition besteht aus zwei Teilen:

```

TABELLE 10-7

```

}

```

```

DATA_CRITERIA XWD1
{
DATA_ATTRIBUTES_NAME      XWD
MODE                      f
NAME_PATTERN              *.xwd
}

DATA_CRITERIA XWD2
{
DATA_ATTRIBUTES_NAME      XWD
MODE                      f
NAME_PATTERN              *.wd
}

```

TABELLE 10-8

DATA_ATTRIBUTES	beschreibt das Erscheinungsbild und das Verhalten des Datentyps.
DATA_CRITERIA,	gibt an, nach welchen Regeln (Name oder Inhalt) eine Datei diesem Datentyp zugerechnet wird.

Das Feld DATA_ATTRIBUTES_NAME verbindet die Kriterien mit den Attributen.

Einem DATA_ATTRIBUTE können mehrere DATA_CRITERIA zugeordnet sein. Beispielsweise sind für den Datentyp XWD zwei verschiedene Namenskriterien (NAME_PATTERN) definiert, nämlich Namen, die in .xwd oder in .wd enden.

Wie Datentypen Datendateien mit Aktionen verbinden

Als Beispiel soll der Datentyp XWD betrachtet werden. Der Benutzer erstellt eine Datei mit dem Datentyp XWD, indem er der Datei eine der zwei Dateinamenerweiterungen .xwd oder .wd gibt. Das Desktop verwendet den Dateinamen als das *Kriterium*, nach dem es eine Datei einem Datentyp zuordnet.

Der Datentyp XWD stellt für jede Datei dieses Datentyps folgendes zur Verfügung:

- Ein eindeutiges Symbol, das Benutzern hilft, die Datendateien zu erkennen.
- Kontexthilfe mit Informationen über den Datentyp.
- Ein angepaßtes Menü 'Ausgewählt' im Dateimanager, das die Aktionen 'Öffnen' und 'Drucken' enthält. Die Aktion 'Öffnen' für XWD-Dateien führt die Aktion Xwud aus.

Aktionen über das Menü 'Ausgewählt' ausführen

Das Menü 'Ausgewählt' im Dateimanager ist nur aktiv, wenn eine Datei oder ein Verzeichnis ausgewählt ist. Die Befehle, die unten im Menü 'Ausgewählt' angezeigt werden, sind abhängig vom Datentyp. Wird beispielsweise eine XWD-Datei ausgewählt, enthält das Menü 'Ausgewählt' die Elemente 'Öffnen' und 'Drucken'.

Das Feld ACTIONS in der Datentypdefinition gibt an, welche Befehle im Menü 'Ausgewählt' des Datentyps unten angefügt werden.

DATA_ATTRIBUTES XWD
{
ACTIONS Open,Print
...
}

Der Inhalt des Menüs 'Ausgewählt' ist abhängig vom Datentyp. Aber viele verschiedene Datentypen stellen eine Aktion 'Öffnen' zur Verfügung. Wird eine Datei dieses bestimmten Datentyps im Dateimanager ausgewählt und das Menü 'Ausgewählt' angezeigt, wird im Menü der Befehl 'Öffnen' angezeigt.

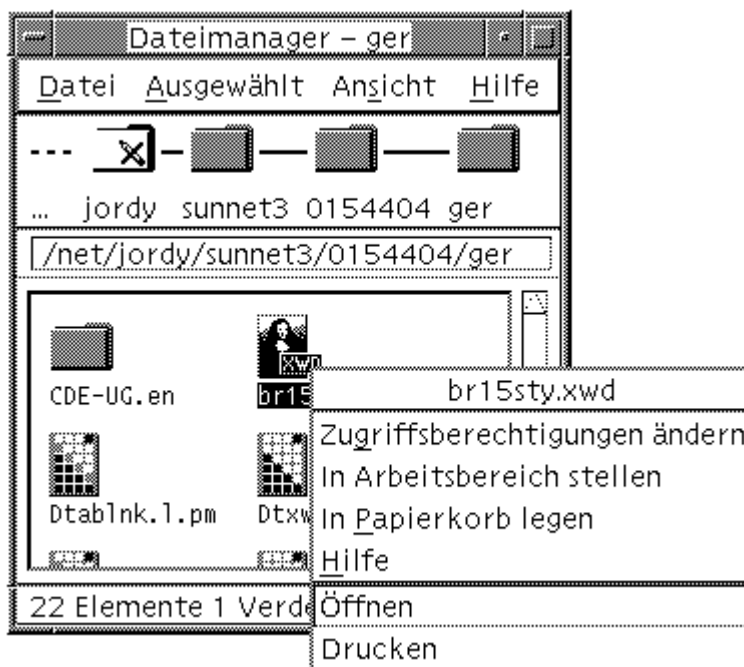


Abbildung 10-6 Das Menü 'Ausgewählt' für eine XWD-Datei

Die Aktion 'Öffnen' führt normalerweise die der Datendatei zugehörige Anwendung aus. Beispielsweise wird durch das Öffnen einer XWD-Datei die Aktion Xwud ausgeführt, die wiederum den X-Client xwud ausführt, um das Anzeigeabbild anzuzeigen. Das heißt, bei dem Datentyp XWD ist die Aktion 'Öffnen' synonym mit

der Aktion Xwud. Ebenso wird durch das Öffnen einer Datei vom Datentyp TEXTFILE der Texteditor und durch das Öffnen einer Bitmap- oder Pixmap-Datei der Symboleditor ausgeführt.

Die Möglichkeit, verschiedene Aktionen 'Öffnen' zu erstellen, die unterschiedlich verwendet werden, basiert auf zwei Merkmalen von Aktionsdefinitionen:

- **Aktionszuordnung**

Die Aktionszuordnung ermöglicht dem Benutzer, eine Aktion zu erstellen, die eine andere Aktion ausführt, anstatt einen Befehl direkt auszuführen. Beispielsweise kann eine Aktion 'Öffnen' erstellt werden, die die Aktion Xwud zuordnet (ausführt).

- **Datentypeinschränkungen für eine Aktion**

Aktionsdefinitionen können ein Feld ARG_TYPE enthalten, das die Aktion auf bestimmte Datentypen begrenzt. Beispielsweise kann angegeben werden, daß die Aktion 'Öffnen', die die Aktion Xwud zuordnet, nur auf Dateien des Datentyps XWD angewendet wird.

Das folgende Beispiel zeigt die Definition der Aktion, die die Aktion 'Öffnen' für den Datentyp XWD der Aktion Xwud zuordnet. Sie befindet sich in der Datenbankkonfigurationsdatei `/usr/dt/appconfig/types/C/xclients.dt` :

TABELLE 10-10

ACTION Open	
{	
LABEL	Öffnen
ARG_TYPE	XWD
TYPE	MAP
MAP_ACTION	Xwud
}	

In dem Feld TYPE wird angegeben, daß es sich um eine Zuordnungsaktion (map) handelt. Das Feld MAP_ACTION gibt an, daß diese Aktion die Aktion Xwud ausführt. Das Feld ARG_TYPE gibt an, daß diese Aktion nur auf Dateien vom Datentyp XWD angewendet wird.

Hierzu zum Vergleich die Definition der Aktion 'Öffnen' in der Datenbankdatei
/usr/dt/appconfig/types/C/dt.dt:

TABELLE 10-11

ACTION Open	
{	
LABEL	Öffnen
ARG_TYPE	BM
TYPE	MAP
MAP_ACTION	Dticon
}	

Diese Definition bezieht sich auf Dateien des Datentyps (ARG_TYPE) BM (Bitmap-Dateien). Die Definition ordnet die Aktion 'Öffnen' der Aktion Dticon zu, die den Symbol-Editor ausführt.

Verhalten des Datentyps bei Doppelklicken definieren

Das Verhalten des Datentyps bei Doppelklicken wird durch den ersten Eintrag im Feld `ACTIONS` bestimmt. Beispielsweise wird beim Datentyp XWD durch Doppelklicken die Aktion 'Öffnen' ausgeführt, die ihrerseits die Aktion Xwud ausführt.

Eine Datendatei an ein Aktionssymbol übergeben

Übergibt der Benutzer eine Datendatei an ein Aktionssymbol, führt das System die Aktion aus und verwendet diese Datendatei als das Argument für die Aktion. (Weitere Informationen können dem Abschnitt „Wie Aktionen Datendateien als Argumente verwenden“ auf Seite 171 entnommen werden.)

Wird beispielsweise eine XWD-Datendatei an ein Symbol Xwd Display übergeben, wird die Aktion Xwud ausgeführt und das Datendateiargument verwendet. Dadurch wird der X-Client `xwud` mit dieser Datendatei ausgeführt.

Drucken über das Desktop für einen Datentyp einrichten

Über das Desktop bestehen folgende Möglichkeiten, eine Datendatei zu drucken:

- über den Befehl 'Drucken' (falls verfügbar) im Menü 'Ausgewählt' des Dateimanagers.
- durch das Übergeben einer Datendatei an einen Desktop-Druckerübergabebereich (das Druckersymbol im Bedienfeld oder ein Druckersymbol im Druckmanager).

Zusätzlich zum Drucken über das Desktop stellen viele Anwendungen noch eine Möglichkeit zur Verfügung, von innerhalb der Anwendung zu drucken.

Beim Drucken über Desktop werden Aktionen mit dem Namen 'Drucken' verwendet. Bei 'Drucken' handelt es sich, wie bei 'Öffnen', um einen Aktionsnamen, der für viele verschiedene Datentypen verwendet wird. Aus diesem Grund werden bei den Aktionen 'Drucken' die Aktionszuordnung und das Feld `ARG_TYPE` verwendet, um das Drucken für jeden einzelnen Datentyp anzupassen.

Als Beispiel soll die Aktion 'Drucken' für den DatentypXWD betrachtet werden. Die Definition befindet sich im

Verzeichnis `/usr/dt/appconfig/types/ sprache/xclients.dt`:

TABELLE 10-12

ACTION Print	
{	
LABEL	Drucken
ARG_TYPE	XWD
TYPE	MAP
MAP_ACTION	NoPrint
}	

Diese Aktion 'Drucken', die sich speziell auf XWD-Dateien bezieht, ist einer Aktion 'NoPrint' zugeordnet. Bei der Aktion NoPrint handelt es sich um eine spezielle Aktion, die in `/usr/dt/appconfig/types/ sprache/dt.dt` definiert ist. Die Aktion 'NoPrint' zeigt ein Dialogfenster an, das dem Benutzer mitteilt, daß dieser Datentyp nicht gedruckt werden kann.

Hierzu zum Vergleich die Aktion 'Drucken' für PCL-Dateien:

TABELLE 10-13

ACTION Print	
LABEL	Drucken
ARG_TYPE	PCL
TYPE	MAP
MAP_ACTION	PrintRaw
}	

Die Aktion 'PrintRaw' , die in der Konfigurationsdatei `/usr/dt/appconfig/types/ sprache/print.dt` definiert ist, enthält die Befehlszeile zum Drucken von PCL-Dateien.

TABELLE 10-14

ACTION PrintRaw		
{		
TYPE		COMMAND
WINDOW_TYPE	NO_STDIO	
EXEC_STRING	/usr/dt/bin/dt1p -w %(File)Arg_1%	
}		

Aktionen und Datentypen mit 'Aktion erstellen' erstellen

Bei 'Aktion erstellen' handelt es sich um ein Werkzeug zum Erstellen

- einer Aktion, mit der eine Anwendung gestartet wird.
- eines oder mehrerer Datentypen für die Datendateien einer Anwendung.
- von Aktionen, mit denen die Datendateien einer Anwendung geöffnet und gedruckt werden.

'Aktion erstellen' kann auch dazu verwendet werden, einfache Aktionen zu erstellen, mit denen Betriebssystembefehle und Shell-Prozeduren ausgeführt werden.

- „Funktionsweise von 'Aktion erstellen' ” auf Seite 181
- „Einschränkungen von 'Aktion erstellen' ” auf Seite 182
- „Aktionen und Datentypen für eine Anwendung mit 'Aktion erstellen' erstellen” auf Seite 183

Weiterführende Informationen können der Hilfeseite dtcreate(1X) entnommen werden.

Funktionsweise von 'Aktion erstellen'

'Aktion erstellen' besteht aus einem Hauptfenster und einer Reihe von Dialogfenstern, über die eine Aktion und die zugehörigen Datentypen erstellt werden.

Das Werkzeug 'Aktion erstellen'

- erstellt eine Aktionsdefinition, die einen Befehl ausführt.

- erstellt eine Datei *Home-Verzeichnis* / `.dt/types/aktion_name.dt`. Diese Datei speichert die Definitionen für Aktion und Datentyp, die für die Anwendung erstellt wurden.
- erstellt eine *Aktionsdatei* im Home-Verzeichnis des Benutzers. Bei der Aktionsdatei handelt es sich um eine ausführbare Datei mit demselben Namen wie die Aktion.
Die Darstellung der Aktionsdatei im Dateimanager wird als *Anwendungssymbol* bezeichnet, da man durch Doppelklicken auf diesem Symbol die Anwendung starten kann.
Wahlweise kann der Benutzer das Aktionssymbol als Übergabebereich einrichten, indem er beim Erstellen der Aktion für die Übergabe geeignete Datentypen angibt.
- erstellt einen oder mehrere Datentypen für die Datendateien einer Anwendung (optional).
- erstellt eine Aktion 'Öffnen' für jeden Datentyp.
- erstellt eine Aktion 'Drucken' für jeden Datentyp (optional).
- lädt die Datenbank für Aktionen und Datentypen erneut. Auf diese Weise werden Aktionen und Datentypen sofort wirksam.

Einschränkungen von 'Aktion erstellen'

'Aktion erstellen' dient dazu, Aktionen und Datentypen für die Ausführung von Anwendungen zu erstellen. Aber Aktionen und Datentypen sind sehr flexibel und enthalten zusätzliche Funktionen, auf die nur zugegriffen werden kann, wenn die Definitionen manuell erstellt werden.

Weitere Informationen hierzu können folgenden Kapiteln entnommen werden:

- Kapitel 12
- Kapitel 13

Einschränkungen bei Aktionen

'Aktion erstellen' kann nicht verwendet werden, um die Aktion für eine Anwendung zu erstellen, wenn eine der folgenden Bedingungen zutrifft:

- Für die Befehlszeile ist ein Parameter (ein Argument, das 'Aktion erstellen' nicht der Aktionsdatei entnehmen kann) notwendig.

Beispielsweise kann 'Aktion erstellen' nicht verwendet werden, um eine Aktion für folgenden Befehl zu schreiben:

```
lp -dgerät dateiname
```

da der Benutzer bei jeder Ausführung des Befehls einen Wert für *gerät* angeben muß.

- Das Anwendungssymbol muß eine andere Bezeichnung tragen als der Aktionsname.

Beispielsweise kann 'Aktion erstellen' nicht verwendet werden, um eine Version einer bereits vorhandenen Aktion in der lokalen Sprache zu erstellen.

- Für die Aktion ist eine der erweiterten Funktionen der Aktionsdatenbank erforderlich.

Beispiele für diese erweiterten Funktionen sind Aktionen, die

- Befehle auf fernen Systemen, die nicht über die Aktionsdefinition verfügen, starten
- andere Aktionen aufrufen
- mit einer anderen Berechtigung ausgeführt werden müssen (zum Beispiel als Superuser)
- mit der Zuordnungsfunktion (map) ausgeführt werden
- je nach Anzahl der Dateiarumente, die die Aktion zur Verfügung stellt, ein sehr unterschiedliches Verhalten zeigen

Einschränkungen bei Datentypen

'Aktion erstellen' kann nicht verwendet werden, um den Datentyp für eine Anwendung zu erstellen, wenn eine der folgenden Bedingungen zutrifft:

- Für den Datentyp müssen neben 'Öffnen' und 'Drucken' noch weitere Aktionen definiert sein.
- Bei der Aktion 'Öffnen' für den Datentyp handelt es sich nicht um den Befehl der Aktion.

Beispielsweise kann 'Aktion erstellen' nicht verwendet werden, um den Datentyp zu erstellen, mit dem ein eindeutiges Symbol für das Verzeichnis, das die Gruppe von Anwendungen der Anwendung darstellt, zur Verfügung gestellt wird.

Aktionen und Datentypen für eine Anwendung mit 'Aktion erstellen' erstellen

Bevor 'Aktion erstellen' ausgeführt wird, muß der Benutzer einige Informationen über die Anwendung zur Verfügung haben.

- Die Befehlszeile, mit der die Anwendung gestartet wird.

Der Benutzer muß wissen, ob die Befehlszeile ein erforderliches Dateiarargument, ein wahlfreies Dateiarargument oder kein Dateiarargument enthält.

Ist für die Anwendung ein Parameter erforderlich, kann die Aktion nicht mit 'Aktion erstellen' erstellt werden.

- Die Typen von Datendateien, die die Anwendung akzeptieren kann.

Einige Anwendungen können nur einen Datentyp akzeptieren. Andere (zum Beispiel ein ASCII-Editor oder ein Grafikeditor) können verschiedene Datentypen akzeptieren.

- Wie die Anwendung ihre Datendateien identifiziert.

Hierbei kann es sich um eine Benennungskonvention (beispielsweise Dateinamen mit der Erweiterung .doc) handeln und/oder vom Inhalt der Datei abhängig sein. Verwendet die Anwendung keine Dateinamenskönvention, kann für das Aktionssymbol dennoch eine Konvention eingerichtet werden.

- Wahlfrei: Die Befehlszeile für den Druck der Dateien

▼ Eine Aktion für eine Anwendung erstellen

1. Auf 'Aktion erstellen' in der Gruppe Desktop-Anwendungen doppelt klicken.

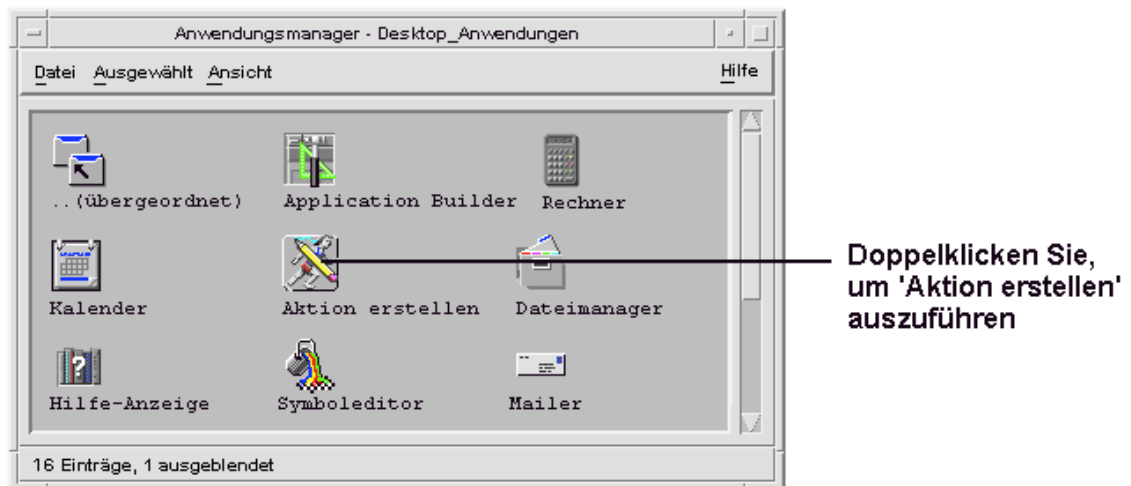


Abbildung 11-1 Das Symbol 'Aktion erstellen' im Anwendungsmanager

Hierdurch wird das Hauptfenster 'Aktion erstellen' angezeigt.

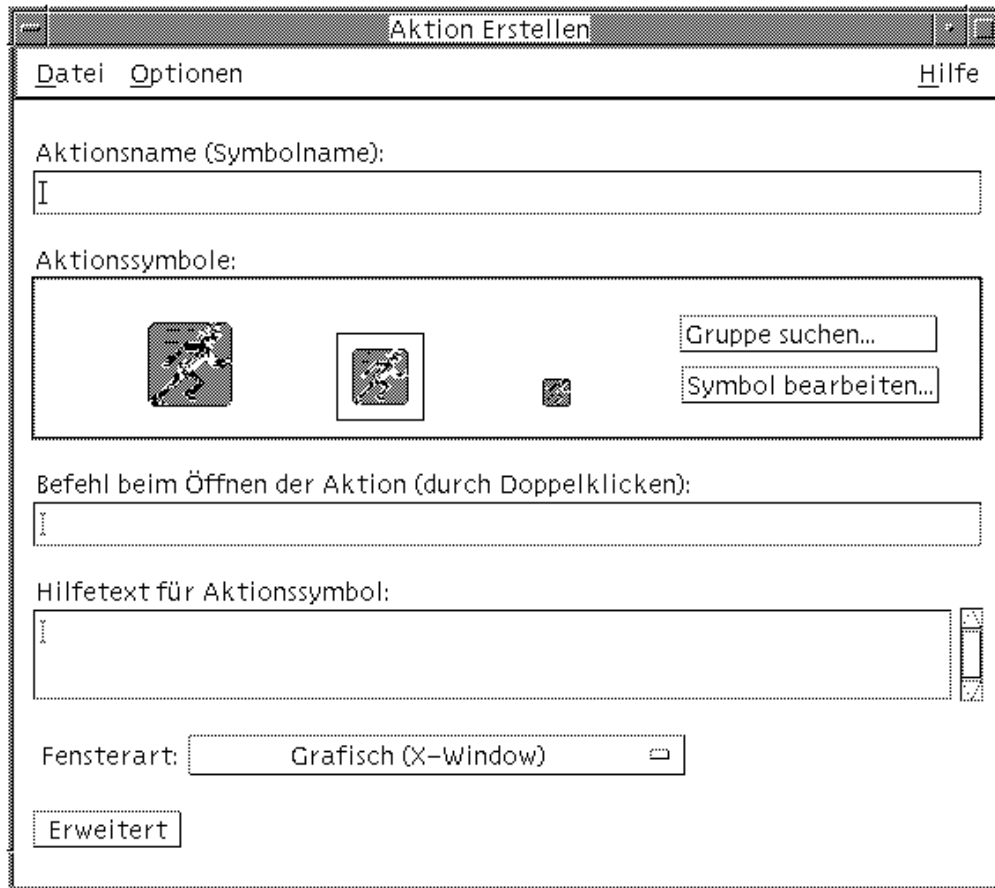


Abbildung 11-2 Hauptfenster 'Aktion erstellen'

2. Den Namen für das Aktionssymbol in das Textfeld 'Aktionsname' eingeben.
3. Die Bedienelemente für 'Aktionssymbole' verwenden, um das Symbol für die Anwendung anzugeben. Zu Beginn wird das Standardsymbol angezeigt.
 - Soll ein anderes, bereits vorhandenes Symbol ausgewählt werden, muß der Benutzer 'Gruppe suchen' anklicken, damit das Dialogfenster 'Gruppe suchen' angezeigt wird. Weitere Informationen hierzu können dem Abschnitt „Symbol über das Dialogfenster 'Gruppe suchen' angeben“ auf Seite 196 entnommen werden.

- Sollen neue Symbole erstellt werden, 'Symbol bearbeiten' auswählen, um den Symboleditor auszuführen.

4. In dem Textfeld 'Befehl beim Öffnen der Aktion (durch Doppelklicken)' den Befehl zum Starten der Anwendung eingeben.

Für ein Dateiarargument die Syntax `$ n` verwenden. Beispiel:

TABELLE 11-1

emacs
bitmap \$1
diff \$1 \$2
lp -oraw \$1

Enthält die Befehlszeile ein Dateiarargument (`$ n`), ist das Aktionssymbol ein Übergabebereich für Dateien.

Die Befehlszeilen werden nicht an eine Shell übergeben, es sei denn, die Verwendung einer Shell wird explizit angegeben. Zum Beispiel verwenden folgende Zeilen Shell-Verarbeitung:

TABELLE 11-2

<code>/bin/sh -c 'ps lp'</code>
<code>/bin/sh -c 'spell \$1 more'</code>

5. Den Kontexthilfetext für das Aktionssymbol im Textfeld 'Hilfetext für Aktionssymbol' eingeben.

Der Text wird im Textfeld automatisch umgebrochen. Diese Zeilenumbrüche werden jedoch online nicht beibehalten. Soll ein fester Zeilenumbruch angegeben werden, muß `\n` verwendet werden.

6. Die für die Aktion erforderliche Fensterunterstützung im Menü 'Optionen' 'Fensterart' auswählen.

TABELLE 11-3

Grafisch (X-Window)	Die Anwendung erstellt ihr eigenes Fenster.
Terminal (automatisches Schließen)	Die Anwendung wird in einem Terminal-Emulator-Fenster ausgeführt, das automatisch geschlossen wird, wenn der Benutzer die Anwendung verläßt
Terminal (manuelles Schließen)	Die Anwendung wird in einem Terminal-Emulator-Fenster ausgeführt, das geöffnet bleibt, bis der Benutzer es explizit schließt.
Keine Ausgabe	Die Anwendung zeigt keine Ausgabe am Bildschirm an.

7. Folgendermaßen vorgehen:

- Verfügt die Anwendung über Datendateien und möchte der Benutzer einen oder mehrere Datentypen für diese erstellen, kann er weitere Informationen hierzu dem Abschnitt, „Einen oder mehrere Datentypen für eine Anwendung erstellen“ auf Seite 188“ entnehmen.

- Soll kein Datentyp erstellt werden, die Aktion sichern, indem 'Speichern' im Menü 'Datei' ausgewählt wird. Anschließend die neue Aktion durch Doppelklicken auf ihrem Symbol im Home-Verzeichnis des Benutzers testen.

▼ Einen oder mehrere Datentypen für eine Anwendung erstellen

1. Die Aktion für die Anwendung unter Verwendung der Prozedur im vorherigen Abschnitt, „Eine Aktion für eine Anwendung erstellen“ auf Seite 184“ definieren.
2. Die Taste 'Erweitert' im Fenster 'Aktion erstellen' anklicken, um das Fenster zu erweitern.

The screenshot shows a dialog box titled 'Aktion erstellen' with the 'Basis' tab selected. It contains the following elements:

- A text field with the prompt: "Beim Öffnen einer Aktion Benutzer nach folgendem fragen:"
- A list box with the prompt: "Dateitypen, die diese Aktion verwenden:"
- Three buttons to the right of the list box: "Hinzufügen...", "Löschen", and "Bearbeiten..."
- At the bottom, a label "Für Übergabe geeignete Datentypen:" followed by two radio buttons: "Alle Dateitypen" (which is selected) and "Nur obige Liste".

Abbildung 11-3 Erweiterte Funktionen im Hauptfenster 'Aktion erstellen'

3. Soll das Anwendungssymbol nach einem Dateiarargument fragen, nachdem das Symbol doppelgeklickt wurde, den Text der Eingabeaufforderung in dem Textfeld 'Beim Öffnen einer Aktion Benutzer nach folgendem fragen:' eingeben.

Folgende Richtlinien gelten für dieses Textfeld:

- Dieses Feld muß verwendet werden, falls die Befehlszeile der Anwendung ein *erforderliches* Dateiarargument enthält.
- Enthält die Befehlszeile kein Dateiarargument, bleibt das Feld leer.
- Ist das Dateiarargument in der Befehlszeile der Anwendung optional, wird, falls ein Text für die Eingabeaufforderung eingegeben wurde, beim Doppelklicken

auf dem Symbol nach dem Dateiarargument gefragt. Wurde kein Text für die Eingabeaufforderung eingegeben, wird die Aktion ausgeführt und eine leere Zeichenfolge als Dateiarargument verwendet.

4. Die Dateitypen angeben, die die Aktion als Argumente akzeptiert:

- Akzeptiert die Aktion jeden Datentyp, 'Alle Datentypen' auswählen.
- Akzeptiert die Aktion nur den/die Datentyp(en), der/die für die Anwendung erstellt wurde(n), 'Nur obige Liste' auswählen.

Zu Beginn ist die Liste 'Datentypen, die diese Aktion verwenden' leer. Erstellt der Benutzer Datentypen für die Anwendung, werden sie zu der Liste hinzugefügt.

5. Die Taste 'Hinzufügen' neben dem Fenster mit der Liste 'Datentypen, die diese Aktion verwenden' anklicken, um das Dialogfenster 'Datentyp hinzufügen' anzuzeigen.

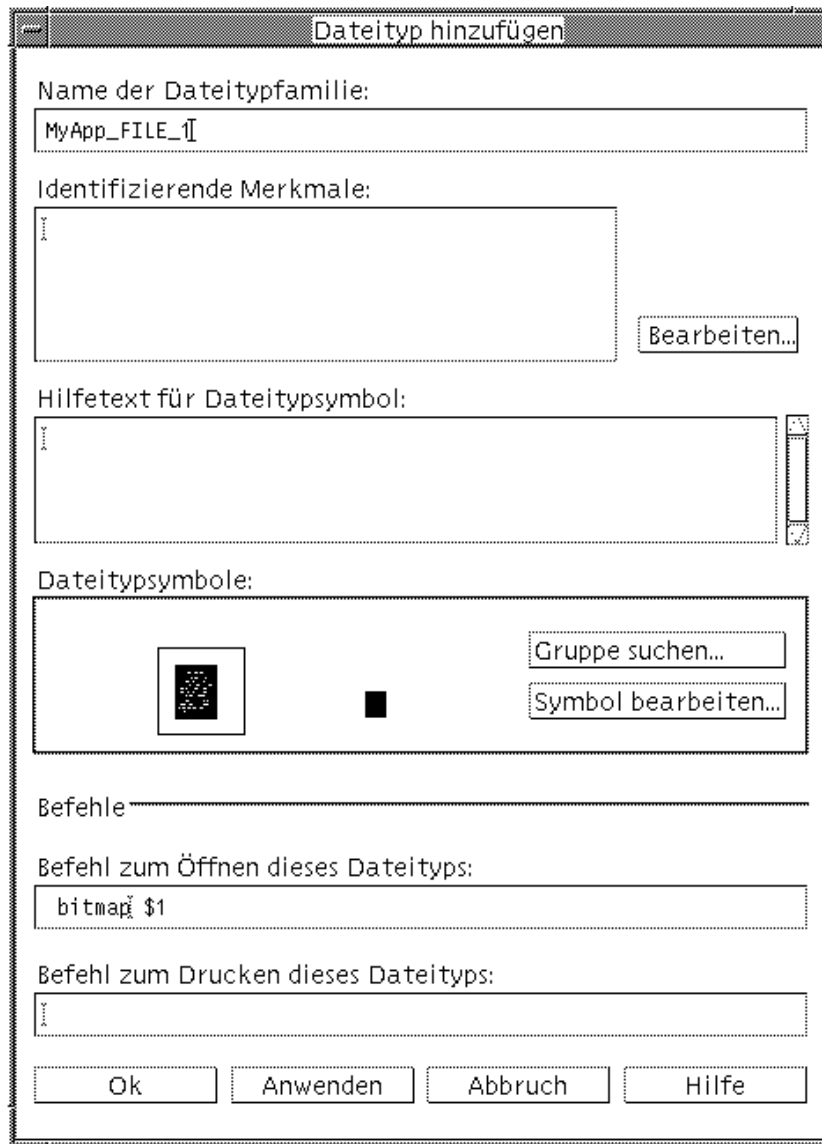


Abbildung 11-4 Dialogfenster 'Datentyp hinzufügen' in 'Aktion erstellen'

6. Optional: Soll der Standarddatentypname nicht verwendet werden, einen neuen Namen für den Datentyp im Textfeld 'Name der Dateitypfamilie' eingeben.

In dem Namen dürfen keine Leerzeichen vorkommen. Der Datentypname ist für Benutzer der Anwendung nicht sichtbar. Er wird in der Datenbank für Aktionen und Datentypen verwendet, um die Definition für den Datentyp zu identifizieren.

7. Die Taste 'Bearbeiten' neben dem Fenster 'Identifizierende Merkmale' anklicken, um das Dialogfenster 'Identifizierende Merkmale' anzuzeigen.

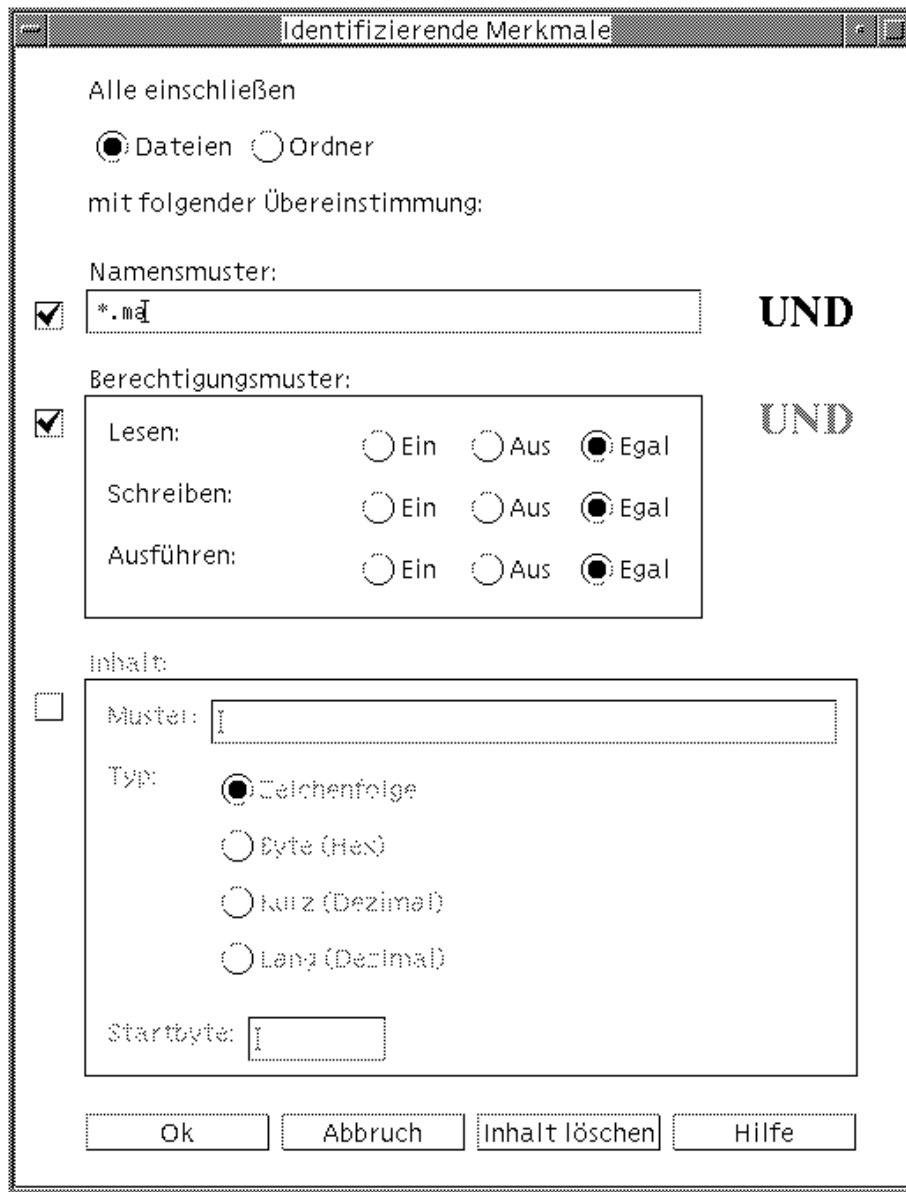


Abbildung 11-5 Dialogfenster 'Identifizierende Merkmale' in 'Aktion erstellen'

Datentypen werden anhand ihrer Merkmale voneinander unterschieden. Der Benutzer kann eines oder mehr der folgenden Kriterien auswählen:

TABELLE 11-4

Dateien oder Ordner	Der Datentyp gilt nur für Dateien oder nur für Ordner
Namensmuster	Datentypzuordnung auf der Grundlage des Dateinamens.
Berechtigungsmuster	Lese-, Schreib- und Ausführungsberechtigungen
Inhalt	Inhalt eines angegebenen Teils einer Datei

8. Auswählen, ob der Datentyp eine Datei oder einen Ordner darstellt.

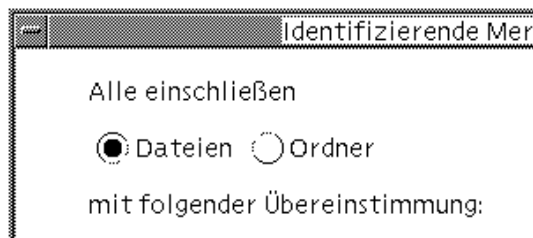


Abbildung 11-6 Das Merkmal für Datei oder Verzeichnis für einen Datentyp angeben.

9. Ist die Zuordnung eines Datentyps vom Namen abhängig, das Markierungsfeld 'Namensmuster' auswählen und den Namen im Textfeld eingeben.

Namensmuster:

UND

Abbildung 11-7 Das Merkmal Dateiname für einen Datentyp angeben.

Hierbei können * und ? als Platzhalterzeichen verwendet werden:

TABELLE 11-5

*	Entspricht einer beliebigen Folge von Zeichen
?	Entspricht einem beliebigen einzelnen Zeichen

10. Ist die Zuordnung eines Datentyps von den Berechtigungen abhängig, das Markierungsfeld 'Berechtigungs muster' auswählen und die Berechtigungen für den Datentyp auswählen..

Berechtigungs muster:

Ein Aus Egal
 Schreiben: Ein Aus Egal
 Ausführen: Ein Aus Egal"/> **UND**

Abbildung 11-8 Das Merkmal Zugriffsrechte für einen Datentyp angeben

TABELLE 11-6

Ein	Für die Datei muß die angegebene Berechtigung vorliegen
Aus	Für die Datei muß die angegebene Berechtigung fehlen
Egal	Die angegebene Berechtigung ist unwichtig

11. Ist die Zuordnung eines Datentyps vom Inhalt abhängig, das Markierungsfeld 'Inhalt' auswählen und die erforderlichen Informationen, Suchmuster und Typ des Inhalts, eingeben. Wahlweise kann der Benutzer die Byteposition angeben, an der die Suche starten soll.

Inhalt:

Muster:

Typ:

Zeichenfolge

Byte (Hex)

Kurz (Dezimal)

Lang (Dezimal)

Startbyte:

Abbildung 11-9 Das Merkmal Inhalt für einen Datentyp angeben

Hinweis - Die Datentypzuordnung auf der Grundlage des Inhalts beeinträchtigt möglicherweise die Leistung des Systems.

12. 'OK' anklicken, um das Dialogfenster 'Merkmale identifizieren' zu schließen.

Die Merkmale werden im Feld 'Merkmale identifizieren' mit folgender Codierung angezeigt:

TABELLE 11-7

d	Verzeichnis
r	Für die Datei liegt Lesezugriff vor
w	Für die Datei liegt Schreibzugriff vor
x	Für die Datei liegt Ausführungszugriff vor
!	Logischer Operator NICHT
&	Logischer Operator UND

13. Den Hilfetext für die Datendateien in das Textfeld 'Hilfetext' eingeben.

14. Die Bedienelemente für 'Aktionssymbole' verwenden, um das Symbol für die Anwendung anzugeben. Zu Beginn wird das Standardsymbol angezeigt.

- Soll ein anderes, bereits vorhandenes Symbol ausgewählt werden, muß der Benutzer 'Gruppe suchen' anklicken, damit das Dialogfenster 'Gruppe suchen' angezeigt wird. Weitere Informationen hierzu können dem Abschnitt „Symbol über das Dialogfenster 'Gruppe suchen' angeben“ auf Seite 196 entnommen werden.
- Sollen neue Symbole erstellt werden, 'Symbol bearbeiten' auswählen, um den Symboleditor auszuführen.

15. Den Befehl im Textfeld 'Befehl zum Öffnen dieses Datentyps' überprüfen. Dieser Befehl wird ausgeführt, wenn der Benutzer eine Datendatei doppelt anklickt.

16. Optional: Stellt der Anwendung einen Druckbefehl für das Drucken von Datendateien über die Befehlszeile zur Verfügung, den Befehl in das Textfeld 'Befehl zum Drucken dieses Datentyps' unter Verwendung der Syntax \$n für ein Dateiarargument einzugeben.

17. Die Datentypdefinition auf eine der folgenden Weisen sichern:

- 'OK' anklicken, um den Datentyp zu sichern, und das Dialogfenster 'Datentyp hinzufügen' schließen.
- 'Anwenden' anklicken, um den Datentyp ohne Schließen des Dialogfensters 'Datentyp hinzufügen' zu sichern. Der Benutzer kann sofort mit der Definition eines weiteren Datentyps für die Aktion fortfahren.

Symbol über das Dialogfenster 'Gruppe suchen' angeben

Das Dialogfenster 'Gruppe suchen' wird angezeigt, wenn der Benutzer 'Gruppe suchen' im Hauptfenster 'Aktion erstellen' oder im Fenster 'Datentyp hinzufügen' anklickt. Dieses Dialogfenster wird verwendet, um das Symbol anzugeben, das für die Aktion oder den Datentyp verwendet wird.

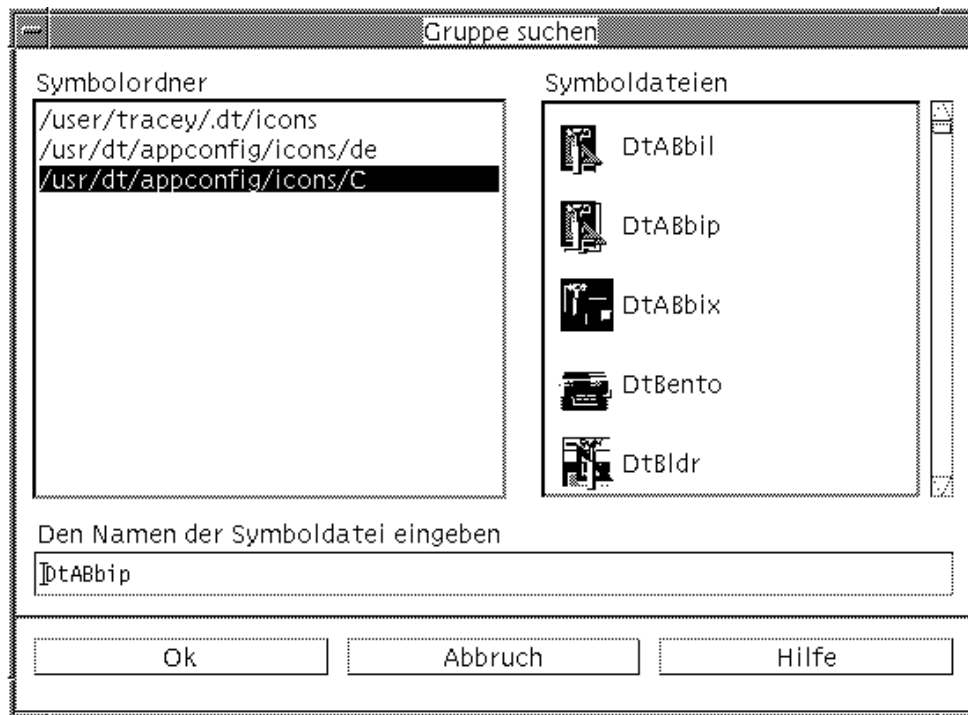


Abbildung 11-10 Dialogfenster 'Gruppe suchen'

Das Dialogfenster 'Gruppe suchen' ermöglicht dem Benutzer, eine Gruppe von Symbolabbilddateien in folgenden Positionen zu erstellen:

- In einem Verzeichnis im Symbolsuchpfad. Die Liste 'Symbolordner' enthält alle Verzeichnisse im Symbolsuchpfad.

- In einem Registrierungspaket, das mit dem Desktop mit `dtappintegrate` integriert wird. Diese Symbole befinden sich noch nicht in einem Verzeichnis im Symbolsuchpfad, werden aber von `dtappintegrate` dorthin gestellt.

Hinweis - Die Definitionen für Aktionen und Datentyp, die mit 'Aktion erstellen' erstellt wurden, verwenden den Dateinamen ohne Pfad für die Symboldateien (den Dateinamen ohne die Erweiterungen für Größe und Typ). Symbole für Aktionen und Datentypen, die mit 'Aktion erstellen' erstellt wurden, werden letztendlich in Verzeichnisse im Symbolsuchpfad gestellt.

Gruppen von Symbolen im Symbolsuchpfad angeben

1. **In der Liste 'Symbolordner' des Dialogfensters 'Gruppe suchen' auf dem Verzeichnispfad, der das Symbol enthält, doppelklicken.**
Die Liste 'Symboldateien' zeigt alle Symboldateien in diesem Ordner an.
2. **In der Liste 'Symboldateien' das Symbol anklicken, das verwendet werden soll.**
Dadurch wird der Dateiname ohne Pfad der Symboldatei in das Textfeld 'Den Namen der Symboldatei eingeben' gestellt.
3. **'OK' anklicken.**

Symbole in einem Registrierungspaket angeben

Handelt es sich bei dem Benutzer um einen Systemadministrator oder um einen Programmierer, der ein Registrierungspaket erstellt, werden die Symbolabbilddateien zunächst in ein Verzeichnis im Registrierungspaket gestellt:

```
app_root/dt/appconfig/icons/  
sprache
```

Nach der Registrierung mit `dtappintegrate` werden die Symboldateien in das Verzeichnis `/etc/dt/appconfig/icons/ sprache`, das sich im Symbolsuchpfad befindet, kopiert.

Diese Prozedur verwenden, um Symbole anzugeben, die Bestandteil eines Registrierungspakets sind:

1. **Im Textfeld 'Den Namen der Symboldatei eingeben' des Dialogfensters 'Gruppe suchen' den Dateinamen ohne Pfad der Symboldatei eingeben.**
2. **'OK' anklicken.**
'Aktion erstellen' zeigt ein Dialogfenster an, in dem der Benutzer darüber informiert wird, daß die Symbole in Verzeichnissen im Symbolpfad nicht gefunden wurden.

3. In dem Dialogfenster 'Information', das angezeigt wird, 'Keine Änderung' auswählen.

Aktionen manuell erstellen

Es gibt zwei Möglichkeiten, Aktionen zu erstellen:

- Mit der Desktop-Anwendung 'Aktion erstellen'.
- Durch das manuelle Erstellen einer Aktionsdefinition.

Um eine Aktion manuell zu erstellen, müssen Sie eine Datenbankdatei bearbeiten. Dieses Kapitel beschreibt, wie Aktionsdefinitionen manuell erstellt werden.

- „Gründe, eine Aktion manuell zu erstellen“ auf Seite 200
- „Eine Aktion manuell erstellen: Allgemeine Schritte “ auf Seite 201
- „Ausführungszeichenfolge für eine COMMAND-Aktion erstellen“ auf Seite 211
- „Fensterunterstützung und Terminal-Emulationen für COMMAND-Aktionen“ auf Seite 219
-
- „Aktionen erstellen, die Anwendungen auf entfernten Systemen ausführen“ auf Seite 224
- „Variablen in Aktions- und Datentypdefinitionen verwenden“ auf Seite 226
- „Aktionen an der Befehlszeile aufrufen “ auf Seite 227
- „Lokalisierte Aktionen erstellen“ auf Seite 229
- „Aktionen für ToolTalk-Anwendungen erstellen“ auf Seite 230
- Eine Einführung in die Arbeit mit Aktionen finden Sie in “.
- Informationen über die Anwendung 'Aktion erstellen' finden Sie in “.
- Referenzinformationen zu Aktionsdefinitionen können Sie der Hilfeseite dtactionfile(4) entnehmen.

Gründe, eine Aktion manuell zu erstellen

Es gibt drei verschiedene Aktionstypen:

- COMMAND
- MAP
- TT_MSG

Mit dem Werkzeug 'Aktion erstellen' können bestimmte Arten von `COMMAND`- und `MAP`-Aktionen erstellt werden. Alle `TT_MSG`-Aktionen (ToolTalk-Meldungen) müssen manuell erstellt werden.

Weitere Informationen finden sie im Abschnitt.

COMMAND-Aktionen

Eine *Befehlsaktion* führt einen Befehl aus, der eine Anwendung oder ein Hilfsprogramm startet, ein Shell-Skript ausführt oder einen Betriebssystembefehl aufruft. Die Definition der Aktion besteht aus dem auszuführenden Befehl (`EXEC_STRING`).

Mit dem Werkzeug 'Aktion erstellen' können die am häufigsten verwendeten Befehlsaktionen erstellt werden. Es gibt jedoch Situationen, in denen eine `COMMAND`-Aktion manuell erstellt werden muß:

- Es werden mehrere Dateien mit jeweils eigener Eingabeaufforderung als Argument angegeben.
- In der Aktion werden andere Aktionen aufgerufen.
- Die Aktion führt je nach Anzahl der Dateiangumente unterschiedliche Funktionen durch.
- Eine Anwendung wird auf einem anderen System als dem mit der Aktionsdefinition ausgeführt (Fernausführung).
- Die Aktion wird für einen anderen Benutzer ausgeführt (um beispielsweise das Root-Paßwort abzufragen und als Benutzer 'Root' auszuführen).

MAP-Aktionen

Bei einer *Zuordnungsaktion* wird eine Aktion einer anderen Aktion "zugeordnet", anstatt direkt einen Befehl oder eine ToolTalk-Meldung anzugeben.

Die Zuordnung ermöglicht es dem Benutzer, alternative Namen für Aktionen anzugeben. So startet beispielsweise eine integrierte Befehlsaktion mit dem Namen IconEditor den Symboleditor. Die Datenbank enthält auch eine Öffnen-Aktion, die in der Definition auf Bitmap- und Pixmap-Dateien (im Feld ARG_TYPE) beschränkt und der Aktion IconEditor zugeordnet ist. So kann der Benutzer den Symboleditor starten, indem eine Bitmap- oder Pixmap-Datei im Dateimanager ausgewählt und dann 'Öffnen' im Menü 'Ausgewählt' gewählt wird.

Die Anwendung 'Aktion erstellen' ermöglicht eine eingeschränkte Zuordnung für die Aktionen 'Öffnen' und 'Drucken'. Alle anderen Zuordnungsaktionen müssen manuell erstellt werden.

TT_MSG-Aktionen (ToolTalk-Meldung)

TT_MSG-Aktionen senden eine ToolTalk-Meldung. Alle TT_MSG-Aktionen müssen manuell erstellt werden.

Eine Aktion manuell erstellen: Allgemeine Schritte

In diesem Abschnitt wird erläutert, wie eine Konfigurationsdatei für eine Aktionsdefinition erstellt wird.

Konfigurationsdateien für Aktionen

Konfigurationsdateien mit Aktionsdefinitionen müssen folgende Bedingungen erfüllen:

- Die Dateien müssen die Namenskonvention *name.dt* verwenden.
- Die Dateien müssen sich im Datenbanksuchpfad (Aktionen und Datentypen) befinden. Der Standardpfad lautet:

Persönliche Aktionen - *Home-Verzeichnis* / .dt/types

Systemweite Aktionen - /etc/dt/appconfig/types/ *Sprache*

Integrierte Aktionen - /usr/dt/appconfig/types/ *Sprache*. Dieses Verzeichnis sollte nicht verwendet werden.

Informationen über das Ändern des Suchpfads für Aktionen und Datentypen finden Sie im Abschnitt „Wert eines Suchpfads setzen“ auf Seite 147.

▼ Eine Aktion manuell erstellen

1. Öffnen Sie eine vorhandene Datenbankdatei, oder erstellen Sie eine neue Datei.

Informationen finden Sie im vorhergehenden Abschnitt „Konfigurationsdateien für Aktionen ” auf Seite 201”.

2. Erstellen Sie die Aktionsdefinition mit folgender Syntax:

```
ACTION Aktionsname
{
    TYPE Aktionstyp
    Aktionsfeld
    ...
}
```

Die Parameter haben folgende Bedeutung:

Aktionsname - Der Name, mit dem die Aktion ausgeführt wird.

Aktionstyp - COMMAND (Standardwert), MAP oder TT_MSG.

Aktionsfeld - Eines der erforderlichen oder optionalen Felder für diesen Aktionstyp. Alle Felder bestehen aus einem Schlüsselwort und einem Wert.

Viele dieser Aktionsfelder werden in diesem Kapitel beschrieben. Weitere Informationen finden Sie auf der Hilfeseite `dtactionfile(4)`.

3. Speichern Sie die Datei.

4. Soll das Aktionssymbol eine eindeutige Grafik verwenden, erstellen Sie die Symbole für die Aktion. Die Symbole befinden sich normalerweise im folgenden Verzeichnis:

- Persönliche Symbole: *Home-Verzeichnis* / `.dt/icons`
- Systemweite Symbole: `/etc/dt/appconfig/icons/ sprache`. Der Standardwert für *sprache* ist C.

Weitere Informationen finden Sie im Abschnitt „Das Symbol für eine Aktion festlegen ” auf Seite 207.

5. Doppelklicken Sie auf 'Aktionen neu laden' in der Anwendungsgruppe Desktop-Werkzeuge.

6. Erstellen Sie eine Aktionsdatei für die Aktion. Diese Datei erstellt für die Aktion ein Symbol im Dateimanager oder Anwendungsmanager (wird mit der Aktion eine Anwendung gestartet, heißt das Symbol *Anwendungssymbol*).

Um eine Aktionsdatei zu erstellen, muß eine ausführbare Datei mit dem Namen *Aktionsname* angelegt werden. Sie können die Datei in jedes Verzeichnis einfügen,

für das Sie Schreibzugriffsrechte haben. Es können beliebig viele Aktionsdateien erstellt werden.

Beispiel für das Erstellen einer COMMAND-Aktion

In diesem Beispiel wird eine persönliche Aktion erstellt, mit der eine Fax-Anwendung auf dem entfernten System AppServerA gestartet werden kann. Der Befehl zum Starten der Fax-Anwendung lautet:

```
/usr/fax/bin/faxcompose [Dateiname]
```

1. Erstellen Sie die Datei *Home-Verzeichnis* / .dt/types/Fax.dt.

2. Fügen Sie die folgenden Aktionsdefinitionen in die Datei ein:

```
ACTION FaxComposer
{
    TYPE      COMMAND
    ICON      fax
    WINDOW_TYPE  NO_STDIO
    EXEC_STRING  /usr/fax/bin/faxcompose -c %Arg_1%
    EXEC_HOST    AppServerA
    DESCRIPTION  Startet das Faxprogramm
}
}
```

Die Felder WINDOW_TYPE und EXEC_STRING beschreiben die Funktionsweise der Aktion.

WINDOW_TYPE - Das Schlüsselwort NO_STDIO legt fest, daß die Aktion nicht in einem Terminalemulationsfenster ausgeführt werden muß.

Weitere Informationen finden Sie im Abschnitt „Fensterunterstützung für die Aktion festlegen“ auf Seite 219.

EXEC_STRING - Die Syntax %Arg_1% akzeptiert eine übergebene Datei. Beim Doppelklicken auf das Aktionssymbol wird ein leeres Faxfenster geöffnet.

Weitere Informationen finden Sie im Abschnitt „Ausführungszeichenfolge für eine COMMAND-Aktion erstellen“ auf Seite 211.

3. Speichern Sie die Datei.
4. Erstellen Sie mit dem Symboleditor die folgenden Symboldateien im Verzeichnis *Home-Verzeichnis*/.dt/icons:
 - fax.m.pm, Größe 32 x 32 Pixel
 - fax.t.pm, Größe 16 x 16 Pixel
5. Doppelklicken Sie auf 'Aktionen neu laden' in der Anwendungsgruppe Desktop-Werkzeuge.
6. Erstellen Sie eine ausführbare Datei mit dem Namen `FaxComposer` in einem Verzeichnis, für das Sie über Schreibzugriffsrechte verfügen (zum Beispiel in Ihrem Home-Verzeichnis).

Beispiel für das Erstellen einer MAP-Aktion

Das Beispiel setzt voraus, daß die meisten zu faxenden Dateien mit dem Texteditor erstellt werden und den Datentyp TEXTFILE (Dateien mit dem Namen *.txt) besitzen.

Mit den folgenden Arbeitsschritten wird der Menüpunkt 'Fax' dem Menü 'Ausgewählt' des Datentyps hinzugefügt.

1. Öffnen Sie die im letzten Beispiel erstellte Datei *Home-Verzeichnis*/.dt/types/Fax.dt.

2. Fügen Sie folgende Zuordnungsdefinition hinzu:

```

ACTION Fax
{
    ARG_TYPE    TEXTFILE
    TYPE        MAP
    MAP_ACTION  FaxComposer
}

```

3. Speichern Sie die Datei.
4. Kopieren Sie die Datenattributdefinition für TEXTFILE aus dem Verzeichnis /usr/dt/appconfig/types/*Sprache* /dtpad.dt in die neue Datei *Home-Verzeichnis* /.dt/types/textfile.dt. Fügen Sie die Aktion 'Fax' dem Feld ACTIONS hinzu.

```
DATA_ATTRIBUTES TEXTFILE
{
    ACTIONS    Open,Print,Fax
    ICON      Dtpenpd
    ...
}
```

5. **Speichern Sie die Datei.**
6. **Öffnen Sie den Anwendungsmanager, und doppelklicken Sie auf 'Aktionen neu laden' in der Anwendungsgruppe Desktop-Werkzeuge.**

▼ Datenbank für Aktionen und Datentypen neu laden

Damit neue oder bearbeitete Aktionsdefinitionen wirksam werden, muß die Datenbank erneut gelesen werden.

- ◆ **Öffnen Sie die Anwendungsgruppe Desktop-Werkzeuge, und doppelklicken Sie auf 'Aktionen neu laden'.**
- ◆ **Oder führen Sie den folgenden Befehl aus:**

```
dtaction ReloadActions
```

ReloadActions ist der Name der Aktion mit dem Namen 'Aktionen neu laden'.

Die Datenbank wird auch erneut gelesen, wenn der Benutzer folgende Aktionen durchführt:

- Anmelden
- Neustarten des Workspace Manager
- Speichern einer Aktion mit dem Befehl 'Speichern' im Menü 'Datei' des Fensters 'Aktion erstellen'.

Aktionsdatei (Symbol) für eine Aktion erstellen

Eine *Aktionsdatei* ist eine sichtbare Repräsentation der Aktion im Dateimanager oder Anwendungsmanager.

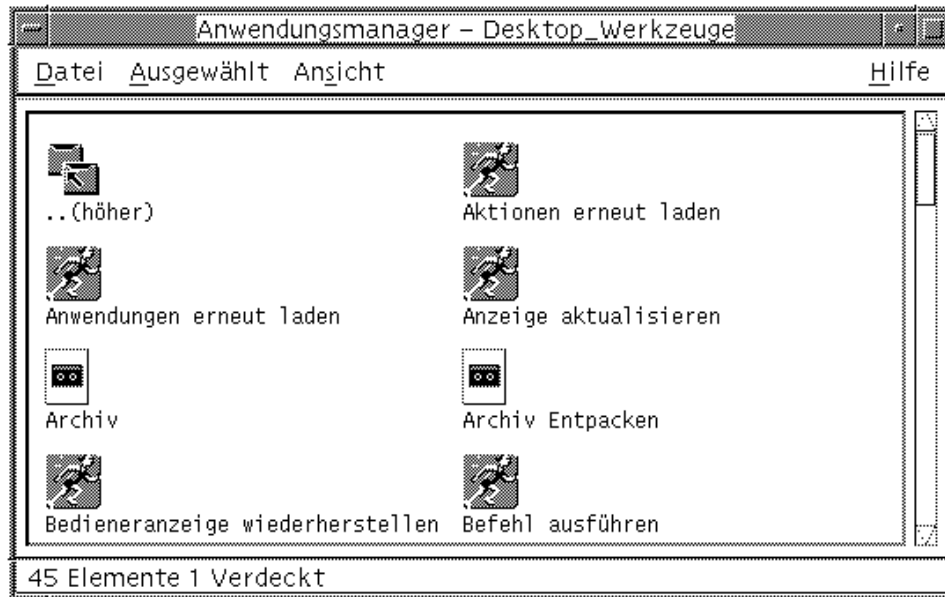


Abbildung 12-1 Aktionsdateien (Aktionssymbole) im Anwendungsmanager

Da das Symbol einer Aktionsdatei eine Aktion repräsentiert, wird es manchmal als *Aktionsdatei* bezeichnet. Wird durch die Aktion eine Anwendung gestartet, nennt man das Symbol *Anwendungssymbol*.

Durch Doppelklicken auf das Aktionssymbol wird die Aktion ausgeführt. Das Aktionssymbol kann auch eine Ablagezone sein.

Eine Aktionsdatei (Aktionssymbol) erstellen

- ◆ **Erstellen Sie eine ausführbare Datei mit dem Namen der Aktion. Der Inhalt der Datei ist nicht von Bedeutung.**

Ein Beispiel: Die Aktion ist folgendermaßen definiert:

```

ACTION MyFavoriteApp
{
    EXEC_STRING    Mfa -file %Arg_1%
    DESCRIPTION    Startet MyFavoriteApp
    ICON           Mfapp
}

```

Die Aktionsdatei ist dann eine ausführbare Datei mit dem Namen `MyFavoriteApp`. Im Dateimanager und Anwendungsmanager wird dann für die Datei `MyFavoriteApp` das Symbol `Mfapp` Größe. *Typ* verwendet. Durch Doppelklicken auf das Symbol `MyFavoriteApp` wird die Ausführungszeichenfolge ausgeführt. Als Online-Hilfe des Symbols wird der Inhalt des Feldes `DESCRIPTION` angezeigt (“Startet `MyFavoriteApp`”).

Aktionsbeschriftungen

Enthält die Aktionsdefinition das Feld `LABEL`, erhält die Aktionsdatei im Datei- und Anwendungsmanager den Inhalt dieses Feldes als Beschriftung und nicht den Dateinamen (*Aktionsname*). Enthält die Aktionsdefinition beispielsweise die Zeile

```
ACTION MyFavoriteApp
{
    LABEL      Lieblingsanwendung
    ...
}
```

hat das Aktionssymbol die Bezeichnung “Lieblingsanwendung”.

Das Symbol für eine Aktion festlegen

Im Feld `ICON` können Sie das Symbol angeben, das im Datei- und Anwendungsmanager für die Aktion verwendet wird.

Geben Sie kein Symbol an, werden die Standardsymboldateien für Aktionssymbole verwendet (`/usr/dt/appconfig/icons/ Sprache/Dtactn.*`).



Abbildung 12-2 Standardaktionssymbol

Das Standardsymbol einer Aktion kann mit folgender Ressource geändert werden:

```
*actionIcon: Symbol_Dateiname
```

Der Parameter *Symbol-Dateiname* kann als Basisdateiname oder als absoluter Pfad angegeben werden.

Das Feld `ICON` kann folgende Werte annehmen:

- Basisdateiname

Der Basisdateiname ist der Name der Datei mit der Symbolgrafik ohne Namenserweiterungen für die Größe (m und t) und den Grafiktyp (bm und pm). Verwenden Sie beispielsweise für die Dateien `GameIcon.m.pm` und `GameIcon.t.pm` den Basisnamen `GameIcon`.

Wenn Sie einen Basisdateinamen angeben, müssen sich die Dateien im Suchpfad für Symbole befinden:

- Persönliche Symbole: *Home-Verzeichnis* / `.dt/icons`
 - Systemweite Symbole: `/etc/dt/appconfig/icons/` *Sprache*
- Absoluter Pfad der Symboldatei (einschließlich des vollständigen Dateinamens)
- Der absolute Pfad sollte nur angegeben werden, wenn sich die Symboldatei nicht im Suchpfad für Symbole befinden. Liegt die Symboldatei `GameIcon.m.pm` im Verzeichnis `/doc/projects`, das nicht im Suchpfad enthalten ist, muß für das Feld `ICON` der Wert `/doc/projects/GameIcon.m.pm` angegeben werden.

Tabelle 12-1 enthält die zu erstellenden Symbolgrößen und die zugehörigen Dateinamen.

TABELLE 12-1 Symbolnamen und -größen für Aktionssymbole

Größe in Pixel	Bitmap-Name	Pixmap-Name
48 x 48	<i>name</i> .l.bm	<i>name</i> .l.pm
32 x 32	<i>name</i> .m.bm	<i>name</i> .m.pm
16 x 16	<i>name</i> .t.bm	<i>name</i> .t.pm

▼ Eine vorhandene Aktionsdefinition ändern

Sie können alle im System verfügbaren Aktionen ändern, auch die integrierten.

Hinweis - Gehen Sie beim Ändern der integrierten Aktionen vorsichtig vor. Sie sind genau auf die Zusammenarbeit mit den Desktop-Anwendungen abgestimmt.

1. Suchen Sie die Definition der zu ändernden Aktion.

Die Aktionsdefinitionen befinden sich in folgenden Standardverzeichnissen:

- Integrierte Aktionen: `/usr/dt/appconfig/types/` *Sprache*
- Systemweite Aktionen: `/etc/dt/appconfig/types/` *Sprache*

- Persönliche Aktionen: *Home-Verzeichnis* / `.dt/types`
Es können auch weitere Verzeichnisse im System definiert sein. Eine Liste der für Aktionen verwendeten Verzeichnisse erhalten Sie mit dem Befehl:

```
dtsearchpath -v
```

Das System verwendet die unter `DTDATABASESEARCHPATH` aufgeführten Verzeichnisse.

2. **Kopieren Sie gegebenenfalls den Text der Aktionsdefinition in eine neue oder bereits vorhandene Datei in einem der folgenden Verzeichnisse:**
 - Systemweite Aktionen: `/etc/dt/appconfig/types/ Sprache`
 - Persönliche Aktionen: *Home-Verzeichnis* / `.dt/types`
Integrierte Aktionen müssen zunächst kopiert werden, da Dateien im Verzeichnis `/usr/dt/appconfig/types/ Sprache` nicht bearbeitet werden sollten.
3. **Bearbeiten Sie die Aktionsdefinition.**
4. **Speichern Sie anschließend die Datei.**
5. **Doppelklicken Sie auf 'Aktionen neu laden' in der Anwendungsgruppe Desktop-Werkzeuge.**

Reihenfolge der Aktionsdefinitionen

Ruft der Benutzer eine Aktion auf, durchsucht das System die Datenbank nach dem betreffenden Aktionsnamen. Ist mehr als eine Aktion mit diesem Namen vorhanden, beachtet das System bestimmte Regeln, um die zu verwendende Aktion zu bestimmen.

- Werden keine anderen Regeln angewendet, ist die Position der Definitionen entscheidend. In der folgenden Liste sind die Aktionen von oben nach unten angeordnet:
 - Persönliche Aktionen (*Home-Verzeichnis* / `.dt/types`)
 - Systemweite lokale Aktionen (`/etc/dt/appconfig/types/ Sprache`)
 - Systemweite entfernte Aktionen (*Rechnername* : `/etc/dt/appconfig/types/ Sprache`). Es wird auf den entfernten Rechnern gesucht, die sich im Suchpfad für Anwendungen befinden.
 - Integrierte Aktionen (`/usr/dt/appconfig/types/ Sprache`)
- In einem gegebenen Verzeichnis werden die `*.dt`-Dateien in alphabetischer Reihenfolge gelesen.

- Aktionen, die durch die Felder ARG_CLASS, ARG_TYPE, ARG_MODE oder ARG_COUNT beschränkt sind, haben Vorrang vor den nicht beschränkten Aktionen (der Standardwert für diese vier Felder ist *.)

Bei mehreren Beschränkungen lautet die Rangfolge von oben nach unten:

- ARG_CLASS
- ARG_TYPE
- ARG_MODE
- ARG_COUNT

Sind mehrere einschränkende ARG_COUNT-Felder vorhanden, lautet die Rangfolge von oben nach unten:

- Spezifischer ganzzahliger Wert n
- $<n$
- $>n$
- *

Im folgenden Beispiel sind Abschnitte verschiedener Aktionsdefinitionen aufgeführt:

```

ACTION EditGraphics
    # EditGraphics-1
    {
        ARG_TYPE    XWD
        ...
    }

```

```

ACTION EditGraphics
    # EditGraphics-2
    {
        ARG_COUNT    0
        ...
    }

```

```

ACTION EditGraphics
    # EditGraphics-3

```

```
{  
  ARG_TYPE *  
  ...  
}
```

Durch das Doppelklicken auf das Aktionssymbol EditGraphics wird EditGraphics-2 gestartet, da kein Argument angegeben ist und ARG_COUNT 0 Vorrang hat. Wird ein Dateiargument des Typs XWD angegeben, wird EditGraphics-1 verwendet, da hier XWD ARG_TYPE angegeben ist. EditGraphics-3 wird für alle anderen Dateiargumente verwendet.

Ausführungszeichenfolge für eine COMMAND-Aktion erstellen

Bei einer COMMAND-Aktion müssen mindestens zwei Felder angegeben werden - ACTION und EXEC_STRING.

ACTION *Aktionsname*

```
{  
  EXEC_STRING Ausführungszeichenfolge  
}
```

Die Ausführungszeichenfolge ist der wichtigste Teil einer COMMAND-Aktionsdefinition. Die verwendete Syntax entspricht der Befehlszeile in einem Terminal-Fenster, enthält aber zusätzliche Elemente für Datei- und Zeichenfolgenargumente.

Allgemeine Merkmale von Ausführungszeichenfolgen

Ausführungszeichenfolgen können folgendes enthalten:

- Dateiargumente und andere Parameter
- Shell-Syntax
- Absolute Pfade oder Namen ausführbarer Dateien

Aktionsargumente

Bei einem Argument handelt es sich um Informationen, die von einem Befehl oder einer Anwendung für die Ausführung benötigt werden. So kann beispielsweise mit der folgenden Befehlszeile eine Datei im Texteditor geöffnet werden:

```
dtpad Dateiname
```

In diesem Befehl ist *Dateiname* ein Dateiarargument für den Befehl `dtpad`.

Aktionen können wie Anwendungen und Befehle mit Argumenten aufgerufen werden. Bei einer `COMMAND`-Aktion können dabei zwei Arten von Daten verwendet werden:

- Dateien
- Zeichenfolgen

Shells in Ausführungszeichenfolgen verwenden

Die Ausführungszeile wird direkt und nicht durch eine Shell ausgeführt. Es kann jedoch explizit eine Shell in der Ausführungszeichenfolge aufgerufen werden.

Ein Beispiel:

```
EXEC_STRING  \
    /bin/sh -c \
    'tar -tvf %(File)Arg_1% 2>&1 | \${PAGER:-more};\
    echo "\n*** Zum Schließen im Menü 'Fenster' die Option 'Schließen' wählen ***'
```

Name oder absoluter Pfad der ausführbaren Datei

Befindet sich die Anwendung in einem Verzeichnis, das in der Variablen `PATH` aufgeführt ist, muß nur der Name der ausführbaren Datei angegeben werden. Ist das Verzeichnis nicht im Pfad enthalten, muß der absolute Pfadname der ausführbaren Datei verwendet werden.

Eine Aktion ohne Argumente erstellen

Verwenden Sie für `EXEC_STRING` dieselbe Syntax wie für den Start der Anwendung von der Befehlszeile.

Beispiele

- Die folgende Ausführungszeichenfolge gehört zu einer Aktion, die den X-Client `xcutsel` startet.

```
EXEC_STRING xcutsel
```

- Die nächste Zeichenfolge startet den Client `xclock` mit einer Digitaluhr. Die Befehlszeile enthält eine Befehlszeilenoption, aber es werden keine Argumente benötigt.

```
EXEC_STRING xclock -digital
```

Eine Aktion erstellen, die eine abgelegte Datei akzeptiert

Verwenden Sie folgende Syntax für das Dateiarargument:

```
%Arg_n%
```

Oder

```
%(File)Arg_n%
```

`(File)` ist optional, da an `Arg_n` übergebene Werte immer als Dateien betrachtet werden (im Abschnitt „Ein Dateiarargument als Zeichenfolge interpretieren“ auf Seite 215 finden Sie Informationen zur Verwendung der Syntax `%(String)Arg_n%`).

Mit dieser Syntax kann der Benutzer ein Dateiojekt auf dem Aktionssymbol ablegen, um die Aktion mit dem übergebenen Dateiarargument zu starten. Dabei wird das *n*te Argument in die Befehlszeile übernommen. Die Datei kann sich auf dem lokalen oder auf einem entfernten System befinden.

Beispiele

- Die folgende Ausführungszeichenfolge führt `wc -w` aus. Dabei wird eine abgelegte Datei im Parameter `-load` verwendet.

```
EXEC_STRING wc -w %Arg_1%
```

- Im folgenden Beispiel können nur Verzeichnisargumente verwendet werden. Wird ein Verzeichnis auf dem Aktionssymbol abgelegt, zeigt die Aktion eine Liste aller Dateien im Verzeichnis an, für die Lese-/Schreibzugriffsrechte vorliegen.

```
ACTION Beschreibbare_Dateien_anzeigen
```

```
{
```

```

ARG_TYPE      FOLDER

EXEC_STRING /bin/sh -c 's -l %Arg_1% | grep rw-'

...

}

```

Eine Aktion erstellen, die zur Angabe eines Dateiaruments auffordert

Verwenden Sie für das Dateiarument folgende Syntax:

```
%(File)"Eingabeaufforderung"%
```

Mit dieser Syntax wird eine Aktion erstellt, die eine Aufforderung zur Eingabe eines Dateinamens anzeigt, sobald der Benutzer doppelt auf das Aktionssymbol klickt.

So zeigt beispielsweise folgende Ausführungszeichenfolge ein Dialogfenster an, in dem der Benutzer zur Eingabe des Dateiaruments für den Befehl `wc --w` aufgefordert wird:

```
EXEC_STRING wc -w %(File)"Wörter zählen in Datei:"%
```

Eine Aktion erstellen, die eine abgelegte Datei akzeptiert oder zur Angabe eines Dateiaruments auffordert

Verwenden Sie für das Dateiarument folgende Syntax:

```
%Arg_n"Eingabeaufforderung
```

```
"%
```

Oder

```
%(File)Arg_n"Eingabeaufforderung
```

```
"%
```

Mit dieser Syntax wird eine Aktion erstellt, die

- eine abgelegte Datei als Argument akzeptiert oder

- ein Dialogfenster mit einer Aufforderung zur Eingabe eines Dateinamens anzeigt, sobald der Benutzer doppelt auf das Aktionssymbol klickt.

So führt beispielsweise folgende Ausführungszeichenfolge den Befehl `lp -oraw` für die abgelegte Datei aus. Wird die Aktion durch Doppelklicken auf das Symbol gestartet, erscheint ein Dialogfenster mit einer Aufforderung zur Eingabe des Dateinamens.

```
EXEC_STRING lp -oraw %Arg_1"Zu druckende Datei:"%
```

Eine Aktion erstellen, die zur Angabe eines Nicht-Dateiarguments auffordert

Verwenden Sie für Nicht-Dateiargumente folgende Syntax:

```
% "Eingabeaufforderung" %
```

Oder

```
%(String) "Eingabeaufforderung" %
```

`(String)` muß nicht angegeben werden, da in Anführungszeichen gesetzter Text standardmäßig als Zeichenfolge interpretiert wird. Mit dieser Syntax wird ein Dialogfenster angezeigt, in dem der Benutzer zur Eingabe aufgefordert wird. Verwenden Sie sie nicht, um den Benutzer zur Eingabe eines Dateinamens aufzufordern.

Im folgenden Beispiel wird der Befehl `xwd` ausgeführt. Er fordert zur Eingabe eines Werts auf, der jedem Pixel hinzugefügt wird:

```
EXEC_STRING xwd -add %"Wert eingeben:"% -out %Arg_1"Dateiname:"%
```

Ein Dateiargument als Zeichenfolge interpretieren

Verwenden Sie für das Argument folgende Syntax:

```
%(String)Arg_n%
```

So druckt beispielsweise folgende Ausführungszeichenfolge mit dem Befehl `lp -tbanner Dateiname` eine Datei mit einem Deckblatt, das den Dateinamen enthält.

```
EXEC_STRING lp -t%(String)Arg_1% %(File)Arg_1"Zu druckende Datei:"%
```

Shell-Funktionen in einer Aktion bereitstellen

Geben Sie die Shell folgendermaßen in der Ausführungszeichenfolge an:

```
/bin/sh -c 'Befehl'
```

```
/bin/ksh -c 'Befehl'
```

```
/bin/csh -c 'Befehl'
```

Beispiele

- Das folgende Beispiel zeigt den Einsatz des Shell-Piping in einer Aktion.

```
EXEC_STRING /bin/sh -c 'ps | lp'
```

- In diesem Beispiel wird eine komplexere Ausführungszeichenfolge verwendet, die eine Shell-Verarbeitung voraussetzt und ein Dateiarargument akzeptiert.

```
EXEC_STRING /bin/sh -c 'tbl %Arg_1"Hilfeseite:"% | troff -man'
```

- Die folgende Ausführungszeichenfolge benötigt als Argument eine komprimierte Datei. Die Aktion entkomprimiert die Datei und druckt sie mit dem Befehl `lp -oraw`.

```
EXEC_STRING /bin/sh -c 'cat %Arg_1 "Zu druckende Datei:"% | \  
uncompress | lp -oraw'
```

- Die folgende Ausführungszeichenfolge startet ein Shell-Skript.

```
EXEC_STRING /usr/local/bin/StartGnuClient
```

COMMAND-Aktionen für mehrere Dateiarargumente erstellen

Aktionen können mehrere Dateiarargumente auf drei verschiedene Arten bearbeiten:

- Die Aktion wird für jedes Argument einmal ausgeführt. Enthält das Feld `EXEC_STRING` nur ein Dateiarargument und werden mehrere Dateiarargumente durch Ablegen auf das Aktionssymbol übergeben, wird die Aktion für jedes Dateiarargument separat ausgeführt.

Werden beispielsweise bei folgender Aktionsdefinition mehrere Dateiarargumente angegeben

```
ACTION DisplayScreenImage  
{  
    EXEC_STRING xwud -in %Arg_1%
```



```
...
}
```

wird die Aktion `DisplayScreenImage` mehrmals ausgeführt.

- Die Aktion verwendet zwei oder mehr nicht austauschbare Dateiargumente. So setzt beispielsweise

```
xsetroot -cursor Cursordatei Maskendatei
```

zwei eindeutige Dateien in der angegebenen Reihenfolge voraus.

- Die Aktion führt denselben Befehl nacheinander für alle übergebenen Dateiargumente aus. So druckt beispielsweise

```
pr Dateiname [Dateiname
```

```
...]
```

eine oder mehrere Dateien in einem Druckauftrag.

Eine Aktion mit nicht austauschbaren Argumenten erstellen

Verwenden Sie eine der folgenden Syntaxkonventionen:

- Soll zur Eingabe der Dateinamen aufgefordert werden, verwenden Sie für jedes Dateiargument folgende Syntax:

```
%(File) "Eingabeaufforderung" %
```

Verwenden Sie für jedes Argument eine andere *Eingabeaufforderung* .

So fordert beispielsweise folgende Ausführungszeichenfolge zur Eingabe zweier Dateinamen auf.

```
EXEC_STRING xsetroot -cursor %(File)"Cursor-Bitmap:"% \
```

```
%(File)"Masken-Bitmap:"%
```

- Sollen abgelegte Dateien akzeptiert werden, verwenden Sie folgende Syntax:

```
%Arg_n%
```

Verwenden Sie dabei für jedes Argument einen anderen Wert für *n*. Ein Beispiel:

```
EXEC_STRING diff %Arg_1% %Arg_2%
```

Eine Aktion mit austauschbaren Dateiargumenten erstellen

Verwenden Sie eine der folgenden Syntaxkonventionen:

- Um eine Aktion zu erstellen, die abgelegte Dateien akzeptiert und einen Befehl der Form *Befehl Datei 1 Datei 2 ...* ausführt, verwenden Sie folgende Syntax für die Dateiarumente:

```
%Args%
```

- Soll eine Aktion erstellt werden, die mehrere abgelegte Dateien akzeptiert oder bei einem Doppelklick eine Aufforderung zur Eingabe eines Dateinamens anzeigt, verwenden Sie folgende Syntax für die Dateiarumente:

```
%Arg_1" Eingabeaufforderung"% %Args%
```

Die Aktion gibt einen Befehl der Form *Befehl Datei 1 Datei 2 ...* aus.

Beispiele

- Die folgende Ausführungszeichenfolge erstellt eine Aktion, die den Befehl

```
pr Datei 1 Datei 2
```

mit mehreren Dateiarumenten ausführt.

```
EXEC_STRING pr %Args%
```

- Die folgende Ausführungszeichenfolge erstellt eine ähnliche Aktion wie im vorhergehenden Beispiel. Jedoch wird hier eine Eingabeaufforderung angezeigt, sobald doppelt auf das Aktionssymbol geklickt wird (keine Dateiarumente).

```
EXEC_STRING pr %Arg_1"Zu druckende Datei(en):"% %Args%
```

Eine Aktion für mehrere abgelegte Dateien erstellen

Um das Ablegen mehrerer Dateien zu akzeptieren und einen Befehl der Form

```
Befehl Datei 1  
Datei 2 ...
```

auszuführen, verwenden Sie folgende Syntax:

```
%Args%
```

Beispiele

- Die folgende Ausführungszeichenfolge startet das Skript Checkout für mehrere Dateien:

```
EXEC_STRING /usr/local/bin/Checkout \
```

```
%Arg_1"Welche Datei?"% %Args%
```

- In diesem Beispiel wird der Befehl `lp -oraw` mit mehreren Dateien ausgeführt:

```
EXEC_STRING    lp -oraw %Arg_1"Zu druckende Datei:"% %Args%
```

Fensterunterstützung und Terminal-Emulationen für COMMAND-Aktionen

Es gibt mehrere Möglichkeiten für die Fensterunterstützung auf dem Desktop durch COMMAND-Aktionen.

- Hat die Anwendung ein eigenes Fenster, muß in der Aktion keine weitere Fensterunterstützung zur Verfügung gestellt werden. Diese Option wird auch verwendet, wenn eine Aktion einen Befehl ohne direkte Benutzereingabe und ohne Ausgaben ausführt.
- Muß die Anwendung in einem Terminal-Emulatorfenster ausgeführt werden, kann mit der Aktion ein Fenster geöffnet und dann die Anwendung gestartet werden. Es stehen verschiedene Terminal-Optionen zur Verfügung.

Fensterunterstützung für die Aktion festlegen

Mit dem Feld `WINDOW_TYPE` können Sie die Art der Fensterunterstützung für die Aktion festlegen (siehe Tabelle 12-2).

TABELLE 12-2 WINDOW_TYPE-Feld und Fensterunterstützung

WINDOW_TYPE	Bereitgestellte Fensterunterstützung
NO_STDIO	Keine. Verwenden Sie NO_STDIO, wenn die Anwendung über ein eigenes Fenster verfügt oder der Befehl keine sichtbare Ausgabe hat.
PERM_TERMINAL	Permanentes Terminal-Emulatorfenster. Die Aktion öffnet ein Terminal-Fenster, das geöffnet bleibt, bis es vom Benutzer explizit geschlossen wird. In das Fenster können Daten eingegeben werden. Verwenden Sie diesen Typ für Befehle, die Eingaben entgegennehmen, Ausgaben erzeugen und dann beendet werden (z. B. <code>ls Verzeichnis</code>).
TERMINAL	Temporäres Terminal-Emulatorfenster. Die Aktion öffnet ein Terminal-Fenster, das sofort nach Beendigung des Befehls wieder geschlossen wird. Verwenden Sie diesen Typ für Befehle, die den ganzen Bildschirm nutzen (z. B. <code>vi</code>).

Befehlszeilenoptionen für den Terminal-Emulator angeben

Mit dem Feld `TERM_OPTS` können Sie in der Aktionsdefinition die Befehlszeilenoptionen für den Terminal-Emulator angeben.

Die folgende Aktion fordert beispielsweise zur Angabe des ausführenden Rechners auf:

```
ACTION OpenTermOnSystemUserChooses
{
    WINDOW_TYPE      PERM_TERMINAL
    EXEC_HOST        %(String)"Entferntes Terminal auf:"%
    TERM_OPTS        -title %(String)"Fenstertitel:"%
    EXEC_STRING      $$SHELL
}
```

Einen anderen Standard-Terminal-Emulator festlegen

Der von Aktionen verwendete Standard-Emulator ist `dtterm`. Sie können aber auch einen anderen Terminal-Emulator wählen. Der Standard-Emulator wird automatisch

verwendet, wenn in der Aktion nicht explizit ein bestimmter Terminal-Emulator angegeben wird.

Für den von einer Aktion verwendeten Terminal-Emulator müssen folgende Befehlszeilenoptionen angegeben werden:

- `-title Fenstertitel`
- `-e Befehl`

Der Standard-Terminal-Emulator für Aktionen wird durch zwei Ressourcen bestimmt:

- Die Ressource `localTerminal` gibt den von lokalen Anwendungen verwendeten Terminal-Emulator an.

```
*localTerminal: Terminal
```

Ein Beispiel:

```
*localTerminal: xterm
```

- Die Ressource `remoteTerminal` gibt den von Anwendungen auf entfernten Rechnern verwendeten Terminal-Emulator an.

```
*remoteTerminal: Host:  
Terminal [ , Host: Terminal  
...]
```

Ein Beispiel:

```
*remoteTerminal: sysibm1:/usr/bin/xterm,sysibp2:/usr/bin/yterm
```

Aktionen auf bestimmte Argumente beschränken

Durch das Beschränken auf einen bestimmten Argumenttyp ist eine genauere Definition einer Aktion möglich. So sollte beispielsweise eine Aktion auf PostScript-Dateiargumente beschränkt werden, wenn sie ein Anzeigeprogramm für PostScript-Dateien aufruft. Die Aktion zeigt eine Fehlermeldung an, wenn eine Datei angegeben wird, die nicht das PostScript-Format hat.

Aktionen können anhand folgender Bedingungen eingeschränkt werden:

- Datentyp des Dateiarguments.

- Anzahl der Dateiargumente (z. B. kein Argument im Gegensatz zu einem oder mehreren Argumenten). Auf diese Weise können beim Ablegen oder Doppelklicken auf ein Aktionssymbol verschiedene Funktionsweisen implementiert werden.
- Lese-/Schreibmodus des Arguments.

Eine Aktion auf einen bestimmten Datentyp beschränken

Mit dem Feld `ARG_TYPE` kann festgelegt werden, für welche Datentypen die Aktion verwendet werden kann. Verwenden Sie den Datenattributnamen.

Sie können auch eine Liste mit mehreren Datentypen angeben. Trennen Sie dann die Einträge durch Kommas voneinander.

Die folgende Aktionsdefinition geht beispielsweise davon aus, daß der Datentyp GIF existiert.

```
ACTION Open_Gif
{
    TYPE      COMMAND
    LABEL     " GIF anzeigen"
    WINDOW_TYPE  NO_STDIO
    ARG_TYPE  Gif
    ICON      xgif
    DESCRIPTION  Zeigt GIF-Dateien an
    EXEC_STRING  xgif
}
```

Eine Aktion anhand der Anzahl der Argumente beschränken

Mit dem Feld `ARG_COUNT` können Sie die Anzahl der Argumente für die Aktion festlegen. Folgende Werte können angegeben werden:

- * (Standard) - Beliebige Anzahl Argumente. Alle anderen Werte haben Vorrang vor *.
- n* - Beliebige positive Ganzzahl (einschließlich 0).
- >*n* - Mehr als *n* Argumente.
- <*n* - Weniger als *n* Argumente.

Mit Hilfe von ARG_COUNT können abhängig davon, ob doppelt auf das Symbol geklickt oder eine Datei abgelegt wird, unterschiedliche Funktionsweisen für das Aktionssymbol implementiert werden (siehe nächster Abschnitt "Unterschiedliche Funktionsweisen für Doppelklicken oder Ablegen implementieren").

▼ Unterschiedliche Funktionsweisen für Doppelklicken oder Ablegen implementieren

Folgendermaßen kann eine Aktion erstellt werden, die eine abgelegte Datei akzeptiert, jedoch bei einem Doppelklick auf das Aktionssymbol keine Aufforderung zur Eingabe eines Dateinamens anzeigt.

1. Erstellen Sie eine Aktionsdefinition für einen Doppelklick auf das Symbol.

Geben Sie im Feld ARG_COUNT 0 Argumente an. Verwenden Sie eine Syntax für das Feld EXEC_STRING , die abgelegte Dateiangaben nicht akzeptiert.

2. Erstellen Sie eine zweite Aktionsdefinition für das Ablegen eines Arguments.

Geben Sie mit dem Feld ARG_COUNT >0 Argumente an. Verwenden Sie eine Syntax für das Feld EXEC_STRING, die abgelegte Dateiangaben akzeptiert.

Ein Beispiel: mit den folgenden beiden Befehlszeilen kann der Editor `vedit` gestartet werden:

- Mit dem folgenden Befehl wird der Editor ohne Dateiangabe aufgerufen:

```
vedit
```

- Der folgende Befehl übergibt dem Editor als Argument eine Datei, die schreibgeschützt geöffnet wird:

```
vedit -R Dateiname
```

Die folgenden beiden Aktionen stellen die Ablegen- und Doppelklickfunktionalität für eine Aktion mit dem Namen Vedit zur Verfügung. Die erste Aktion hat Vorrang, wenn die Datenbank nach einer Übereinstimmung durchsucht wird, da ARG_COUNT 0 spezifischer ist als das implizierte ARG_COUNT * der Definition für die Ablagefunktionalität.

```
# Doppelklicken

ACTION Vedit
{
    TYPE          COMMAND
    ARG_COUNT     0
    WINDOW_TYPE   PERM_TERMINAL
    EXEC_STRING    vedit
}
```

```

# Ablegen
ACTION Vedit
{
    TYPE          COMMAND
    WINDOW_TYPE   PERM_TERMINAL
    EXEC_STRING   vedit -R %Arg_1%
}

```

Eine Aktion über den Argumentmodus beschränken

Geben Sie mit dem Feld `ARG_MODE` den Lese-/Schreibmodus des Arguments an. Folgende Werte können verwendet werden:

- * (Standard) - Beliebiger Modus
- !w - Nicht schreibbar
- w - Schreibbar

Aktionen erstellen, die Anwendungen auf entfernten Systemen ausführen

Wenn von Aktionen und Ausführung auf fernen Systemen die Rede ist, werden zwei Begriffe häufig verwendet:

Datenbank-Host - Das System mit der Aktionsdefinition.

Ausführungs-Host - Das System mit der ausführbaren Datei.

Meistens befinden sich Aktionen und ihre Anwendungen auf demselben System. Da der Standard-Ausführungs-Host für eine Aktion der Datenbank-Host ist, ist keine spezielle Syntax erforderlich.

Handelt es sich jedoch bei Datenbank- und Ausführungs-Host um unterschiedliche Systeme, muß in der Aktionsdefinition angegeben werden, wo die Ausführungszeichenfolge gestartet werden soll.

Die Fähigkeit, Aktionen und Anwendungen auf verschiedenen Systemen zu suchen, ist Bestandteil der Client/Server-Architektur des Desktop. Weiterführende Informationen zu Netzwerkanwendungen finden Sie im Abschnitt „Anwendungs-Services verwalten“ auf Seite 129.

Eine Aktion erstellen, die eine entfernte Anwendung ausführt

Mit dem Feld `EXEC_HOST` können Sie in einer Aktionsdefinition angeben, wo sich die Anwendung befindet.

Folgende Werte können für `EXEC_HOST` verwendet werden:

`%DatabaseHost%` - Der Host, auf dem die Aktion definiert ist.

`%LocalHost%` - Der Host, auf dem die Aktion aufgerufen wird (der *Sitzungs-Server*).

`%DisplayHost%` - Der Host, auf dem der X-Server läuft (nicht zulässig für X-Terminals).

`%SessionHost%` - Der Host, auf dem der kontrollierende Login-Manager ausgeführt wird.

Hostname - Der benannte Host. Verwenden Sie diesen Wert für Umgebungen, in denen die Aktion immer auf einem bestimmten System ausgeführt werden soll.

`%"Eingabeaufforderung" %` - Fordert den Benutzer bei jeder Ausführung der Aktion zur Eingabe des Hostnamens auf.

Der Standardwert ist `%DatabaseHost%`, `%LocalHost%`. Die Aktion führt den Befehl also auf dem Host mit der Aktionsdefinition aus, wenn das Feld `EXEC_HOST` nicht angegeben wird. Ist dies nicht möglich, führt die Aktion den Befehl auf dem Sitzungs-Server aus.

Beispiele

- Das Feld gibt im folgenden Beispiel den Host `ddsyd` an:

```
EXEC_HOST ddsyd
```

- In diesem Beispiel wird der Benutzer aufgefordert, einen Hostnamen einzugeben:

```
EXEC_HOST %"Auf welchem Host befindet sich die Anwendung:"%
```

- Im folgenden Beispiel wird festgelegt, daß die Aktion zuerst versuchen soll, die Anwendung auf dem System mit der Aktionsdefinition auszuführen. Ist das nicht möglich, wird versucht, die Anwendung auf dem Host `ddsyd` zu starten.

```
EXEC_HOST %DatabaseHost%, ddsyd
```

Variablen in Aktions- und Datentypdefinitionen verwenden

In Definitionsdateien für Aktionen und Datentypen können Zeichenfolge- und Umgebungsvariablen verwendet werden.

Zeichenfolgevariablen in einer Aktion verwenden

Der Gültigkeitsbereich einer Zeichenfolgevariable erstreckt sich von ihrer Definition bis zum Ende der Datei. Es gibt keine globalen Zeichenfolgevariablen für die Datenbank.

Sind eine Zeichenfolge- und eine Umgebungsvariable mit demselben Namen vorhanden, wird die Zeichenfolgevariable verwendet.

Eine Zeichenfolgevariable definieren

- ◆ **Verwenden Sie folgende Syntax:**

```
set Variablenname=  
Wert
```

Variablennamen können beliebige alphanumerische Zeichen und Unterstriche (`_`) enthalten. Jede Variable muß in einer eigenen Zeile definiert werden.

Ein Beispiel:

```
set Remote_Application_Server=sysapp  
set Remote_File_Server=sysdata
```

Eine Zeichenfolgevariable referenzieren

- ◆ **Verwenden Sie folgende Syntax:**

```
#{ } Variablenname [ ] }
```

Ein Beispiel:

```
EXEC-HOST    $Remote_Application_Server  
CWD          /net/#{Remote_File_Server}/doc/project
```

Umgebungsvariablen in Aktionen und Datentypen verwenden

◆ Mit folgender Syntax kann eine Umgebungsvariable referenziert werden:

```
#{ [ ] Variable [ ] } .
```

Die Variable wird beim Laden der Datenbank erweitert (durch ihren Wert ersetzt). Sind eine Zeichenfolge- und eine Umgebungsvariable mit demselben Namen vorhanden, wird die Zeichenfolgevariable verwendet.

So druckt beispielsweise folgende Ausführungszeichenfolge eine Datei mit einem Deckblatt, das den Anmeldenamen enthält.

```
EXEC-STRING lp -t$LOGNAME %(File)Arg_1%
```

Aktionen an der Befehlszeile aufrufen

Das Desktop stellt den Befehl `dtaction` zur Verfügung, mit dem Aktionen an der Befehlszeile gestartet werden können. Aktionen können mit `dtaction` folgendermaßen ausgeführt werden:

- In einem Skript
- In einer anderen Aktion
- In der Befehlszeile eines Terminal-Emulatorfensters

Syntax von `dtaction`

```
dtaction [-user
```

```
Benutzername] [-execHost Hostname] \  
Aktionsname [Argument [  
Argument]...]
```

-user *Benutzername* - Ermöglicht das Ausführen der Aktion für einen anderen Benutzer. Wird `dtaction` von einem anderen Benutzer als *Benutzername* aufgerufen, wird eine Paßwortabfrage angezeigt.

-execHost *Hostname* - Diese Option wird nur für `COMMAND`-Aktionen verwendet und gibt den Host an, auf dem der Befehl ausgeführt wird.

Argument - Die Argumente für die Aktion (normalerweise Dateiangaben).

Der Client `dtaction` verfügt über weitere Befehlszeilenoptionen. Weitere Informationen können Sie der Hilfeseite `dtaction(1)` entnehmen.

Eine Aktion erstellen, die andere Aktionen ausführt

Verwenden Sie `dtaction` im Feld `EXEC_STRING` der Aktion.

Die Aktion im folgenden Beispiel verwendet eine integrierte Aktion mit dem Namen `Spell` (die Aktion hat im Anwendungsmanager die Bezeichnung 'Rechtschreibprüfung'). Die neue Aktion führt den Texteditor und die Aktion `Spell` aus und zeigt die Rechtschreibfehler in einem separaten Terminal-Emulatorfenster an.

```
ACTION EditAndSpell
{
    WINDOW_TYPE    NO_STDIO

    EXEC_STRING    /bin/sh -c 'dtaction Spell \
                    %Arg_1"Datei:"%; dtpad %Arg_1%'
}
```

Eine Aktion erstellen, die für einen anderen Benutzer ausgeführt wird

Verwenden Sie folgende Syntax im Feld `EXEC_STRING`:

```
EXEC_STRING    dtaction -user Benutzername
```

Aktionsname [*DDateiangabe*]

Der neue Benutzer (*Benutzername*) muß über einen der folgenden Mechanismen den Anzeigezugriff auf das System besitzen:

- Leserecht für die Datei `.Xauthority` des angemeldeten Benutzers
- Oder `xhost`-Rechte

So ermöglichen es beispielsweise die folgenden beiden Aktionen dem Benutzer, sich als Root anzumelden und die Datei app-defaults zu bearbeiten.

```
ACTION AppDefaults
{
    WINDOW_TYPE    NO_STUDIO

    EXEC_STRING    /usr/dt/bin/dtaction -user root \
                  EditAppDefaults %Arg_1"Datei:"%
}

ACTION EditAppDefaults
{
    WINDOW_TYPE    TERMINAL

    EXEC_STRING    /bin/sh -c 'chmod +w %Arg_1%; \
                  vi %Arg_1%; chmod -w %Arg_1%'
}
```

Lokalisierte Aktionen erstellen

Der Suchpfad für Datentypen enthält auch sprachabhängige Verzeichnisse. Der Wert der Umgebungsvariablen LANG bestimmt, wo Datentypdefinitionen gesucht werden.

Verzeichnisse für lokalisierte Aktionen

Lokalisierte Aktionsdefinitionen müssen sich in den richtigen sprachabhängigen Verzeichnissen des Aktionssuchpfads befinden.

Folgender Standardsuchpfad wird verwendet:

- Persönliche Aktionen: *Home-Verzeichnis* / .dt/types
- Systemweite Aktionen: /etc/dt/appconfig/types/ *Sprache*
- Integrierte Aktionen: /usr/dt/appconfig/types/ *Sprache*

▼ Eine vorhandene Aktion sprachabhängig einrichten

1. Erstellen Sie eine Datei im betreffenden sprachabhängigen Verzeichnis (beispielsweise in `/etc/dt/appconfig/types/japanese`).
2. Kopieren Sie die Aktionsdefinition in die sprachabhängige Konfigurationsdatei.
Kopieren Sie beispielsweise eine Aktionsdefinition aus dem Verzeichnis

```
app_root/dt/appconfig/types/C/  
Datei.dt
```

in das Verzeichnis

```
app_root/dt/appconfig/types/japanese/  
Neue_Datei.dt
```

3. Fügen Sie ein LABEL-Feld hinzu, oder ändern Sie das vorhandene Feld LABEL.

`LABEL` *Zeichenfolge*

Die Beschriftung (LABEL) wird im Anwendungsmanager und Dateimanager verwendet, um das Aktionssymbol zu bezeichnen.

4. Passen Sie die folgenden Felder in der Aktionsdefinition an:
 - Symbole: `ICON`
 - Online-Hilfe: `DESCRIPTION`
 - Eingabeaufforderungen: der Text in Anführungszeichen im Feld `EXEC_STRING`

Aktionen für ToolTalk-Anwendungen erstellen

Hinweis - Die folgenden Informationen betreffen nur Anwendungen, die das Senden von Nachrichten mit ToolTalk unterstützen.

Erstellen Sie mit dem Aktionstyp `TT_MSG` eine Aktion, die eine ToolTalk-Meldung sendet.

ACTION *Aktionsname*

```
{  
  TYPE TT_MSG  
  ...  
}
```

Die Felder `addressing` und `disposition`

- Das ToolTalk-Feld `addressing` ist immer auf `TT_PROCEDURE` gesetzt.
- Das ToolTalk-Feld `disposition` verwendet als Standardwert das statische Meldungsmuster.

Nicht unterstützte Meldungen

Die folgenden Meldungen werden von Aktionen des Typs `TT_MSG` nicht unterstützt:

- Objektbezogene ToolTalk-Meldungen
- Kontextargumente in Meldungen

Schlüsselwörter für `TT_MSG`-Aktionen

Tabelle 12-3 enthält Informationen über die Schlüsselwörter und deren Verwendung in `TT_MSG`-Aktionen.

TABELLE 12-3 Schlüsselwörter für `TT_MSG`-Aktionen und deren Verwendung

Schlüsselwort	Verwendung
<code>TT_CLASS</code>	Definiert den Wert des ToolTalk -Meldungsfelds <code>class</code> .
<code>TT_SCOPE</code>	Definiert den Wert des ToolTalk -Meldungsfelds <code>scope</code> .
<code>TT_OPERATION</code>	Definiert den Wert des ToolTalk -Meldungsfelds <code>operation</code> .
<code>TT_FILE</code>	Definiert den Wert des ToolTalk -Meldungsfelds <code>file</code> .
<code>TT_ARGn_MODE</code>	Definiert den Wert des ToolTalk -Attributs <code>mode</code> des <i>n</i> ten Meldungsarguments.

TABELLE P-3 Schlüsselwörter für TT_MSG-Aktionen und deren Verwendung *(fortgesetzt)*

Schlüsselwort	Verwendung
TT_ARGn_VTYPE	Definiert den Wert des ToolTalk -Attributs vtype des <i>n</i> ten Meldungsarguments.
TT_ARGn_VALUE	Definiert den Wert des <i>n</i> ten Meldungsarguments.

Datentypen manuell erstellen

Es gibt zwei Möglichkeiten, eine Datentypdefinition zu erstellen:

- in dem Werkzeug 'Aktion erstellen'. Die Verwendung von 'Aktion erstellen' wird in Kapitel 11."ausführlich behandelt.
- die Datentypdefinition manuell erstellen.

Um einen Datentyp manuell zu erstellen, ist es erforderlich , eine Datenbankdatei zu bearbeiten.

Im folgenden Kapitel wird beschrieben, wie Datentypdefinitionen manuell erstellt werden.

- „Gründe, einen Datentyp manuell zu erstellen“ auf Seite 234
- „Komponenten einer Datentypdefinition: Kriterien und Attribute“ auf Seite 234
- „Datentyp manuell erstellen: Allgemeine Schritte“ auf Seite 235
- „Beispiel für die Erstellung einer persönlichen Aktion und eines Datentyps“ auf Seite 237
- „Datenkriterien für einen Datentyp definieren“ auf Seite 242

Weiterführende Informationen

- Eine Einführung in Datentypen enthält Kapitel 10.
- Referenzinformationen zu Datentypdefinitionen können der Hilfeseite `dtddsfile(4)` entnommen werden.

Gründe, einen Datentyp manuell zu erstellen

Wird ein Datentyp manuell erstellt, können alle Möglichkeiten, die die Syntax von Datentypdefinitionen bietet, ausgeschöpft werden.

Sollen die folgenden Merkmale von Datentypen verwendet werden, muß der Datentyp manuell erstellt werden:

- Datentypzuordnung auf der Grundlage der Position (des Pfads).
- Die Möglichkeit, dem Datentyp neben 'Öffnen' und 'Drucken' weitere Aktionen zuzuordnen.
- Mehrere Namens-, Muster- oder Inhaltskriterien werden für denselben Datentyp, beispielsweise ein Datentyp, der auf Dateien mit der Erweiterung *.abc oder *.def endet, verwendet.
- Datentypzuordnung auf der Grundlage von Verweisen.

Komponenten einer Datentypdefinition: Kriterien und Attribute

Eine Datentypdefinition besteht aus zwei separaten Datenbankdefinitionen:

- Die Definition `DATA_ATTRIBUTES`.

Die Definition `DATA_ATTRIBUTES` beschreibt den Namen des Datentyps und das Erscheinungsbild und Verhalten von Dateien dieses Typs.

- Die Definition `DATA_CRITERIA`.

Die Definition `DATA_CRITERIA` beschreibt, nach welchen Kriterien der Datentyp zugeordnet wird. Jede Kriteriendefinition gibt an, auf welche Definition `DATA_ATTRIBUTES` sie angewendet wird.

Für jede Definition `DATA_ATTRIBUTES` muß mindestens eine Definition `DATA_CRITERIA` vorliegen. Es können ihr aber auch mehr Definitionen `DATA_CRITERIA` zugeordnet sein.

Beispielsweise könnte ein Benutzer eine Attributdefinition für PostScript-Dateien erstellen, in der beschrieben wird, wie PostScript-Dateien im Dateimanager angezeigt werden und wie sie sich verhalten. Anschließend könnte der Benutzer zwei verschiedene Kriterien für den Datentyp PostScript erstellen, einen auf der Grundlage des Dateinamens, und einen auf der Grundlage des Dateiinhalts.

Weitere Informationen können dem Abschnitt „Datenkriterien für einen Datentyp definieren“ auf Seite 242 entnommen werden.

Datentyp manuell erstellen: Allgemeine Schritte

In diesem Abschnitt wird beschrieben, wie eine Datentypkonfigurationsdatei erstellt wird.

Konfigurationsdateien für Datentypen

Konfigurationsdateien, die Datentypdefinitionen enthalten, müssen folgende Voraussetzungen erfüllen:

- Die Dateien müssen der Namenskonvention *name*.dt entsprechen.
- Die Dateien müssen sich im Datenbanksuchpfad befinden. Der Standardsuchpfad lautet:

Persönliche Datentypen - *Home-Verzeichnis* / .dt/types

Systemweite Datentypen - /etc/dt/appconfig/types/ *Sprache*

Integrierte Datentypen - /usr/dt/appconfig/types/ *Sprache*. Dieses Verzeichnis sollte nicht verwendet werden.

Weitere Informationen zum Ändern des Datenbanksuchpfads können dem Abschnitt „Wert eines Suchpfads setzen“ auf Seite 147 entnommen werden.

▼ Datentypdefinition erstellen

1. **Eine bereits vorhandene Datenbankdatei öffnen oder eine neue erstellen.**

Weitere Informationen können dem vorherigen Abschnitt „Konfigurationsdateien für Datentypen“ auf Seite 235 entnommen werden.

2. **Die Datenattribute für den Datentyp unter Verwendung folgender Syntax definieren:**

```
DATA_ATTRIBUTES daten_typ_name
{
    ICON      bild_name
    DESCRIPTION  zeichenfolge
```

```

    attribute_field
    attribute_field
    ...
}

```

wobei die Parameter folgende Bedeutung haben:

daten_typ_name - Ein eindeutiger Name, der diesem Datentyp zugeordnet wird.

bild_name - Dateiname oder Pfad einer Symboldatei. Den Dateinamen ohne Pfad für die Datei verwenden. Beispielsweise wird für die Symboldateien `yimage.m.pm` und `myimage.t.pm` der Name `myimage` verwendet.

attribut_feld - Feld, das das Erscheinungsbild oder Verhalten des Datentyps definiert.

zeichenfolge - Zeichenfolge. Der Inhalt dieser Zeichenfolge wird für diesen Datentyp als Kontexthilfe angezeigt.

Weitere Informationen hierzu können dem Abschnitt „Beispiel für die Erstellung einer persönlichen Aktion und eines Datentyps“ auf Seite 237 entnommen werden.

3. Die Datenkriterien unter Verwendung folgender Syntax definieren:

```

DATA_CRITERIA kriterien_name
{
    DATA_ATTRIBUTES_NAME      daten_typ_name
    kriterien_feld
    kriterien_feld
    ...
}

```

wobei die Parameter folgende Bedeutung haben:

kriterien_name Eindeutiger Name für diese Kriteriendefinition.

daten_typ_name Name, der in der Definition `DATA_ATTRIBUTES` verwendet wird.

kriterien_feld Feld, in dem das Kriterium für die Zuordnung einer Datei zu diesem Datentyp definiert wird.

Weitere Informationen hierzu können dem Abschnitt „Datenattribute eines Datentyps definieren“ auf Seite 239 entnommen werden.

4. Die Datenbankdatei sichern.

5. Die Symbole für den Datentyp erstellen.

Weitere Informationen hierzu können dem Abschnitt „Symbolbild angeben, das für einen Datentyp verwendet wird“ auf Seite 239 entnommen werden.

6. Falls erforderlich, die Aktionen, die im Feld `ACTIONS` der Attributdefinition aufgelistet sind, erstellen.

7. 'Aktionen erneut laden' in der Gruppe von Anwendungen

'Desktop-Werkzeuge' doppelklicken, um die Datenbank erneut zu laden.

Beispiel für die Erstellung einer persönlichen Aktion und eines Datentyps

Angenommen, das System enthält eine Anwendung mit dem Namen `xgif`, die GIF-Bilder anzeigt. Normalerweise wird das Programm ausgeführt, indem folgendes eingegeben wird:

```
xgif dateiname
```

Nun sollen GIF-Bilder auf verschiedene Weise angezeigt werden:

- durch Doppelklicken einer GIF-Datendatei
- durch Auswählen der Datendatei und Auswählen der Anwendung im Menü 'Ausgewählt'

1. Eine neue Datei **Home-Verzeichnis** `/.dt/types/GifViewer.dt` zum Bearbeiten öffnen.

2. Die Datentypdefinitionen eingeben:

```
DATA_ATTRIBUTES Gif
{
    DESCRIPTION    Gif image file.
    ICON           GifIcon
    ACTIONS        View
}
DATA_CRITERIA Gif_Criteria
{
    DATA_ATTRIBUTES_NAME    Gif
    NAME_PATTERN              *.gif
}
```

3. Die Aktionsdefinition für die Aktion 'GifViewer' eingeben:

```
ACTION GifViewer
{
    EXEC_STRING      xgif %(File)Arg_1"Gif-Datei anzeigen:"
    WINDOW_TYPE      NO_STDIO
    DESCRIPTION      Eine Datei doppelklicken oder Übergeben,
                    um die GIF-Anzeige zu starten.
}
```

Da die Definition kein Feld `ICON` enthält, verwendet die Aktion das Standardsymbol des Systems.

4. Die folgende Zuordnungsaktion eingeben, um die Aktion GifViewer mit der Aktion View, die in der Datentypdefinition aufgelistet ist, zu verbinden. Mit dem Feld `ARG_TYPE` wird die Aktion View auf Dateien vom Typ Gif eingeschränkt.

```
ACTION View
{
    ARG_TYPE      Gif
    TYPE          MAP
    MAP_ACTION    GifViewer
}
```

5. Die Datei sichern.

6. Auf 'Aktionen erneut laden' in der Gruppe von Anwendungen Desktop-Werkzeuge doppelklicken, um die Datenbank erneut zu lesen.

Datenattribute eines Datentyps definieren

Die Definition `DATA_ATTRIBUTES` definiert das Erscheinungsbild und das Verhalten des Datentyps. Sie gibt den Namen des Datentyps an und bietet die Möglichkeit, zusätzlich folgendes anzugeben:

- das Dateimanagersymbol (Feld `ICON`)
- das Verhalten bei Doppelklicken und den Inhalt des Menüs 'Ausgewählt' (Feld `ACTIONS`)
- die Kontexthilfe des Datentyps (Feld `DESCRIPTION`)

Symbolbild angeben, das für einen Datentyp verwendet wird

Das Feld `ICON` wird verwendet, um das Symbol anzugeben, das im Dateimanager verwendet wird. Wird kein Symbolbild angegeben, wird im Dateimanager nur eine Bezeichnung angezeigt.

Bei dem Wert des Felds `ICON` kann es sich um folgendes handeln:

- Einen Dateinamen ohne Pfad.

Der Dateiname ohne Pfad ist der Name der Datei, die das Symbolbild enthält, ohne die Dateinamenerweiterungen für Größe (`m` und `t`) und Abbildtyp (`bm` und `pm`). Beispielsweise wird für Dateien mit den Namen `GameIcon.m.pm` und `GameIcon.t.pm` der Name `GameIcon` verwendet.

Wird der Dateiname ohne Pfad verwendet, müssen die Symboldateien in ein Verzeichnis im Symbolsuchpfad gestellt werden:

- Persönliche Symbole: *Home-Verzeichnis* / `.dt/icons`
- Systemweite Symbole: `/etc/dt/appconfig/icons/` *sprache*
- Einen absoluten Pfad zu der Symboldatei, einschließlich des vollständigen Dateinamens.

Der absolute Pfad sollte nur verwendet werden, wenn die Symboldatei sich nicht im Symbolsuchpfad befindet. Befindet sich beispielsweise die Symboldatei `GameIcon.m.pm` im Verzeichnis `/doc/projects`, das sich nicht im Symbolsuchpfad befindet, wird im Feld `ICON` der Wert `/doc/projects/GameIcon.m.pm` gesetzt.

Tabelle 13-1 enthält eine Liste mit Symbolgrößen, die der Benutzer erstellen sollte, und den zugehörigen Dateinamen.

TABELLE 13-1 Symbolnamen und Größen für Datentypsymbole

Größe in Pixel	Bitmap-Name	Pixmap-Name
32 x 32	<i>name</i> .m.bm	<i>name</i> .m.pm
16 x 16	<i>name</i> .t.bm	<i>name</i> .t.pm

Datentypen Aktionen zuordnen

Es gibt zwei Möglichkeiten, Datentypen Aktionen zuzuordnen:

- In dem Feld `ACTIONS` in der Definition `DATA_ATTRIBUTES` werden die Aktionen aufgelistet, die im Menü 'Ausgewählt' des Dateimanagers erscheinen. Bei der ersten Aktion in der Liste handelt es sich um die Standardaktion, die auch durch Doppelklicken aktiviert werden kann.
- Aktionen können mit dem Feld `ARG_TYPE` der Aktionsdefinition auf angegebene Datentypen beschränkt werden.

Beispielsweise erstellt die folgende Datentypdefinition einen Datentyp für spezielle, vom Systemadministrator erstellte "Readme"-Dateien mit der Namenskonvention `*.rm`.

```
DATA_ATTRIBUTES SysReadmeFile
{
    ICON          SysReadMe
    ACTIONS       Open, Respond
}
DATA_CRITERIA SysReadmeFileCriteria
{
    NAME_PATTERN  *.rm
    DATA_ATTRIBUTES_NAME SysReadmeFile
}
```

Eine spezielle Aktion 'Respond' für die Datei wird im folgenden definiert. Dadurch wird eine schreibbare Kopie der Datei im Texteditor geöffnet. Wird die

Datei gesichert und der Texteditor verlassen, wird die Datei über elektronische Post an den Systemadministrator gesendet (Adresse `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
}
```

Dateien auf der Grundlage des Datentyps verdecken

Ist eine Datei ein verdeckter Datentyp, wird sie nicht im Dateimanager angezeigt.

Im Feld `PROPERTIES` in der Definition `DATA_ATTRIBUTES` wird angegeben, daß Objekte dieses Typs verdeckt werden:

```
PROPERTIES    invisible
```

Verhalten der Datei bei Bearbeitung angeben

Die folgenden Felder `DATA_ATTRIBUTES` werden in erster Linie von Anwendungsprogrammierern verwendet. Hier wird angegeben, wie sich Dateien verhalten, wenn der Benutzer verschiedene Desktop-Aktivitäten durchführt.

Weitere Informationen können der Publikation *Common Desktop Environment Programmer's Guide*, entnommen werden, das Bestandteil der Entwicklerumgebungsdokumentation ist.

TABELLE 13-2 DATA_ATTRIBUTES Felder und Beschreibungen

Feld	Beschreibung
MOVE_TO_ACTION	Für Container, wie beispielsweise Verzeichnisse. Gibt die auszuführende Aktion an, wenn eine Datei in einen Container dieses Datentyps verschoben wird.
COPY_TO_ACTION	Für Container, wie beispielsweise Verzeichnisse. Gibt die auszuführende Aktion an, wenn eine Datei in einen Container dieses Datentyps kopiert wird.
LINK_TO_ACTION	Gibt die auszuführende Aktion an, wenn eine Datei mit einer Datei dieses Datentyps verknüpft wird.
IS_TEXT	Gibt an, daß Dateien dieses Datentyps Text enthalten, der in einem Textfeld angezeigt werden kann.
MEDIA	Gibt den entsprechenden ToolTalk-Medientyp an.
MIME_TYPE	Gibt den entsprechenden MIME-Typ an.
X400_TYPE	Gibt den entsprechenden X400-Typ an.

Datenkriterien für einen Datentyp definieren

Die Definition `DATA_CRITERIA` definiert die Kriterien für die Zuordnung eines Objekttyps zu einer Datei oder einem Verzeichnis

Sie können die Kriterien in Tabelle 13-3 für die Objektzuordnung verwenden.

TABELLE 13-3 DATA_CRITERIA - Kriterien und Beschreibungen

Kriterium	Beschreibung
Dateiname	Der Dateiname muß einem angegebenen Muster entsprechen. Verwenden Sie das Feld NAME_PATTERN.
Dateiposition	Der Pfad muß einem angegebenen Muster entsprechen. Verwenden Sie das Feld PATH_PATTERN.
Dateiinhalte	Ein angegebener Teil des Dateinhalts muß den angegebenen Daten entsprechen. Verwenden Sie das Feld CONTENT.
Dateimodus	Die Datei muß über die angegebenen Berechtigungen verfügen (Lesen, Schreiben, Ausführen, Verzeichnis). Verwenden Sie das Feld MODE.
Symbolische Verbindungen	Die Typzuordnung basiert auf der Datei, mit der das Objekt verbunden ist.

Für einen Datentyp kann mehr als ein Kriterium verwendet werden. Aber die Kriterien NAME_PATTERN und PATH_PATTERN sollten nicht in demselben Datentyp verwendet werden.

Datentypenzuordnung auf der Grundlage des Dateinamens

Das Feld NAME_PATTERN verwenden, um die erforderlichen Merkmale des Dateinamens anzugeben. Das Feld kann die folgenden Platzhalterzeichen enthalten:

- ? - Entspricht einem beliebigen einzelnen Zeichen
- * - Entspricht einer beliebigen Folge von Zeichen (einschließlich einer leeren Zeichenfolge)
- [cc...] - Entspricht einem beliebigen der in Klammern gesetzten Zeichen (c)
- [c- c] - Entspricht einem beliebigen Zeichen aus dem Bereich von c bis c

Beispiele

- Die folgende Datentypdefinition erstellt einen Datentyp auf der Grundlage des Dateinamens. Der Dateiname muß mit QS anfangen und mit .doc enden.

```
DATA_ATTRIBUTES QS_Doc
```

```
{
```

```

DESCRIPTION   Diese Datei enthält ein Dokument zum QS \ projekt.
ICON          Word_Doc
ACTIONS       Open
}

```

```

DATA_CRITERIA QS_Doc_Criteria
{
NAME_PATTERN   QS*.doc
DATA_ATTRIBUTES_NAME  QS_Doc
}

```

- Die folgende Definition erstellt einen Datentyp für Verzeichnisse mit dem Namen Demo_*n*, wobei *n* den Zahlen 0 bis 9 entspricht.

```

DATA_ATTRIBUTES Demo_directory
{
DESCRIPTION   Dies ist ein Verzeichnis. Zum Öffnen doppelklicken.
ICON          Demo
ACTIONS       OpenInPlace,OpenNewView
}

```

```

DATA_CRITERIA Demo_directory_criteria
{
NAME_PATTERN   Demo_[0-9]
MODE           d
DATA_ATTRIBUTES_NAME  Demo_directory
}

```

Datentypenzuordnung auf der Grundlage der Position

Das Feld `PATH_PATTERN` verwenden, um den Pfad anzugeben. Es können dieselben Platzhalterzeichen verwendet werden, wie bei `NAME_PATTERN`.

Beispielsweise verwendet der folgende Datentyp ein Kriterium auf der Grundlage des Pfads.

```
DATA_ATTRIBUTES Project_Graphics
{
    DESCRIPTION    Grafikdatei zum QS-Projekt. Symbol doppelklicken, um die Grafik anzuzeigen.
    ICON          QSgraphics
}
DATA_CRITERIA Project_Graphics_Criteria
{
    DATA_ATTRIBUTES_NAME    Project_Graphics
    PATH_PATTERN             */projects/QS/graphics/*
}
```

Datentypenzuordnung auf der Grundlage des Dateinamens und der Position

Soll ein Datentyp auf der Grundlage des Dateinamens als auch der Position erstellt werden, muß der Name in dem Wert für das Feld `PATH_PATTERN` enthalten sein. `NAME_PATTERN` und `PATH_PATTERN` können nicht in derselben Kriteriendefinition verwendet werden.

Beispiele

- Der im folgenden definierte Datentyp `QS_Source_Files` gilt für alle Dateien mit dem Namen `app n .c`, wobei $n = 1$ bis 9, die sich in den Unterverzeichnissen von `*/projects/QS` befinden.

```
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
}

```

- Der folgende Datentyp gilt für alle Dateien im Verzeichnis `/doc/project1` mit dem Namen `chnm .xxx`, wobei *n* 0 bis 9 ist, und *xxx* eine beliebige aus drei Zeichen bestehende Dateinamenerweiterung ist.

```

DATA_ATTRIBUTES ChapterFiles
{
  DESCRIPTION      Kapiteldatei für das Projektdokument.
  ICON             chapter
  ACTIONS          Edit,Print
}

DATA_CRITERIA Chapter_Criteria
{
  PATH_PATTERN      /doc/project1/ch[0-9][0-9].???
  DATA_ATTRIBUTES_NAME  ChapterFiles
}

```

Dateimodi als Kriterium der Datentypzuordnung verwenden

Im Feld `MODE` die erforderlichen Berechtigungen angeben.

Moduskriterien werden normalerweise in Kombination mit der Datentypzuordnung auf der Grundlage von Namen, Position oder Inhalt verwendet. Sie ermöglichen dem Benutzer, einen Datentyp auf eine Datei oder ein Verzeichnis einzuschränken oder die erforderlichen Lese-, Schreib- und Ausführungsberechtigungen anzugeben.

Das Feld `MODE` kann die folgenden logischen Operatoren (Tabelle 13-4) und Zeichen (Tabelle 13-5) enthalten:

TABELLE 13-4 Logische Operatoren und Beschreibungen für das Feld MODE

Operator	Beschreibung
!	Logischer Operator NICHT
&	Logischer Operator UND
	Logisches ODER

TABELLE 13-5 Zeichen und Beschreibungen für das Feld MODE

Zeichen	Beschreibung
f	Der Datentyp gilt nur für Dateien
d	Der Datentyp gilt nur für Verzeichnisse
r	Die Datei kann von jedem Benutzer gelesen werden
w	In die Datei kann von jedem Benutzer geschrieben werden
x	Die Datei kann von jedem Benutzer ausgeführt werden
l	Die Datei stellt eine Verbindung dar

Standardmäßig hat der Modus keinen Einfluß.

Beispiele

- Die folgenden Modusfelder schränken den Datentyp wie folgt ein:

f&!w - Schreibgeschützte Dateien

!w - Schreibgeschützte Dateien und Verzeichnisse

f&x - Ausführbare Dateien

f&r&x - Dateien, in die geschrieben werden kann, und die ausführbar sind

x|!w - Dateien, die ausführbar oder schreibgeschützt sind

- Die folgende Datentypdefinition erstellt einen Datentyp für schreibgeschützte, nicht ausführbare Dateien, deren Namen der Namenskonvention *.doc entsprechen. Es wird angenommen, daß eine Aktion View für den Datentyp definiert wurde.

```
DATA_ATTRIBUTES ReadOnlyDocument
{
    ICON      read_only

    DESCRIPTION  Dieses Dokument kann nicht beschrieben werden. Ein Doppelklick
    führt den Editor mit einer schreibgeschützten
    Kopie der Datei aus..

    ACTIONS     View
}

DATA_CRITERIA ReadOnlyDocument_Criteria
{
    NAME_PATTERN      *.doc

    MODE              !d&!x&!w

    DATA_ATTRIBUTES_NAME  ReadOnlyDocument
}

```

Datentypzuordnung auf der Grundlage des Inhalts

Im Feld CONTENT wird angegeben, daß die Datentypzuordnung auf der Grundlage des Inhalts der Datei erfolgen soll. Die Datentypzuordnung auf der Grundlage des Inhalts einer Datei kann in Kombination mit der Datentypzuordnung auf der Grundlage von Namen oder Position verwendet werden.

Die Zuordnung kann auf der Grundlage einer Zeichenfolge oder des numerischen Inhalts einer Datei erfolgen. Das erste Byte in der Datei erhält die Zahl 0.

- Für die Zuordnung auf der Grundlage einer Zeichenfolge folgende Syntax verwenden:

```
CONTENT start_byte string zeichenfolge
```

- Für die Zuordnung auf der Grundlage des numerischen Inhalts folgende Syntax verwenden:

```
CONTENT start_byte byte nummer
```



```
CONTENT start_byte short nummer
```

```
CONTENT start_byte long nummer
```

- Für die Zuordnung auf der Grundlage des Inhalts eines Verzeichnisses folgende Syntax verwenden:

```
CONTENT 0 filename "datei_name"
```

Standard-C-Notation für Oktal- (beginnend mit `o`) und Hexadezimalzahlen (beginnend mit `oX`) verwenden.

Hinweis - Die Datentypzuordnung auf der Grundlage des Inhalts beeinträchtigt die Systemleistung. Der Benutzer sollte, wenn möglich, die Datentypzuordnung auf der Grundlage von Namen und Position verwenden.

Beispielsweise gilt der folgende Datentyp `Writable_Wingz` für alle Dateien mit Schreibberechtigung, die am Anfang der Datei die Zeichenfolge `WNGZ` enthalten.

```
DATA_ATTRIBUTES Writable_Wingz
```

```
{  
  ...  
}
```

```
DATA_CRITERIA Writable_Wingz_Criteria
```

```
{  
  CONTENT      0 string WNGZ  
  MODE         w&l d  
  DATA_ATTRIBUTES_NAME Writable_Wingz  
}
```

▼ Datentypen mit mehreren unabhängigen Kriterien erstellen

Es kann ein Datentyp mit mehreren unabhängigen Kriterien erstellt werden, d.h. ein Datentyp, dem die Datei zugeordnet wird, wenn sie *einem* (oder beiden) Kriterien entspricht.

1. **Die Definition** `DATA_ATTRIBUTES` **für den Datentyp erstellen.**
2. **Eine Definition** `DATA_CRITERIA` **für jedes Kriterium erstellen.**

Das Feld DATA_ATTRIBUTES_NAME verwenden, um jedes Kriterium mit derselben Definition DATA_ATTRIBUTES zu verbinden.

Beispielsweise erstellen die folgenden Definitionen den Datentyp Mif. Die Datentypzuordnung erfolgt auf der Grundlage von Namen oder Inhalt.

```
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 zeichenfolge MIFFile  
}
```

Sprachenabhängige Datentypen erstellen

Der Suchpfad für Datentypen enthält sprachenabhängige Positionen. Das Desktop verwendet den Wert von LANG, um festzulegen, welche Positionen nach Datentypdefinitionen durchsucht werden.

Positionen für sprachenabhängige Datentypen

Sprachenabhängige Datentypdefinitionen müssen in die entsprechenden sprachenabhängigen Verzeichnisse im Aktionssuchpfad gestellt werden.

Der Standardsuchpfad lautet:

- Persönliche Aktionen: *Home-Verzeichnis* / `.dt/types`
- Systemweite Aktionen: `/etc/dt/appconfig/types/ sprache`
- Integrierte Aktionen: `/usr/dt/appconfig/types/ sprache`

▼ Datentypen sprachenabhängig erstellen

1. **Eine Datei in dem zugehörigen sprachenabhängigen Verzeichnis erstellen (zum Beispiel in `/etc/dt/appconfig/types/japanese`).**
2. **Die Datentypdefinition in die sprachenabhängige Konfigurationsdatei kopieren.**
3. **Ein oder mehrere Felder der Datentypdefinition sprachenabhängig anpassen.**

Symbole für das Desktop erstellen

Folgenden Objekten werden Desktop-Symbole zugeordnet:

- Aktionsdateien und Datentypen im Dateimanager und im Anwendungsmanager
- Bedienelemente im Bedienfeld
- Anwendungsfenster in Symbolgröße
- Grafiken, die von Anwendungen verwendet werden, wie zum Beispiel Paletten und Funktionsleisten
- Hintergrundmuster des Arbeitsbereichs
- „Symbolbilddateien“ auf Seite 253
- „Symbolzuordnungen“ auf Seite 256
- „Empfehlungen zur Erstellung von Symbolen“ auf Seite 260

Hinweis - Die Dokumentation zur Entwicklungsumgebung enthält zusätzliche Informationen zu Desktop-Symbolen. Weitere Informationen können Kapitel 4, „Visual Design,“ der Publikation *Common Desktop Environment: Style Guide und Certification Checklist* entnommen werden.

Symbolbilddateien

Damit das Desktop ein Symbolbild verwenden kann, muß die Symbolbilddatei:

- im korrekten Format sein.
- die korrekten Dateinamenskonventionen verwenden.
- den Desktop-Größenkonventionen entsprechen.

- sich in einem Verzeichnis im Symbolsuchpfad befinden.
- vom Desktop-Konstrukt mit der korrekten Syntax aufgerufen werden. Wird beispielsweise ein neues Bedienelement für das Bedienfeld erstellt, wird im Feld `ICON` in der Bedienfelddefinition definiert, welches Symbolbild für das Bedienelement verwendet wird.

Symboldateiformate

Für einen Farbbildschirm sollten Symboldateien im Format X-Pixmap (XPM) verwendet werden, die normalerweise die Erweiterung `.pm` haben. Ansonsten werden Dateien im Format X-Bitmap (XBM) verwendet, die die Erweiterung `.bm` haben. Wird in der Pixmap-Datei Transparenz verwendet, wird bei der Erstellung einer `.bm`-Datei eine Maskendatei (`_m.bm`) generiert. Weitere Informationen darüber, wie das Desktop diese Dateien findet, können dem Abschnitt „Symbolsuchpfad“ auf Seite 156 entnommen werden.

Symboldateinamen

Jedes Symbol- und Hintergrundmusterbild ist in einer eigenen Datei gespeichert. Normalerweise wird ein Symbol mit dem Dateinamen ohne Pfad angegeben. Beispielsweise wird ein Symbol möglicherweise als `mail` bezeichnet, wenn die Datei unter folgendem Namen gespeichert ist:

```
/usr/dt/appconfig/icons/sprache/mail.l.pm
```

Die Dateinamenskönvention des Hinzufügens von Erweiterungen ermöglicht, Symbole nach Größe und Typ in Gruppen zusammenzufassen. Symbolnamen für Desktop-Komponenten haben im allgemeinen folgende Formate:

dateiname ohne pfad . gröÙe . format

oder

dateiname ohne pfad . format

wobei die Parameter folgende Bedeutung haben:

dateiname ohne pfad— Der Dateiname ohne Pfad des Bilds, mit dem auf das Bild bezug genommen wird

gröÙe— Ein Buchstabe, der die Größe angibt: `l` (groß) `m` (mittel) `s` (klein) `t` (winzig)

format— Das Dateiformat: `pm` (Pixmap) `bm` (Bitmap)

Symbolgrößenkonventionen

Tabelle 14–1 zeigt die empfohlene Größe in Pixel für Desktop-Symbole.

TABELLE 14–1 Symbolgrößen und Dateinamen

Symbolgröße	Bitmap-Name	Pixmap-Name
16 x 16 (winzig)	<i>name.t.bm</i>	<i>name.t.pm</i>
24 x 24 (klein)	<i>name.s.bm</i>	<i>name.s.pm</i>
32 x 32 (mittel)	<i>name.m.bm</i>	<i>name.m.pm</i>
48 x 48 (groß)	<i>name.l.bm</i>	<i>name.l.pm</i>

Tabelle 14–2 zeigt die Symbolgrößen, die die Desktopkomponenten verwenden. In einigen Fällen ist die Größe des verwendeten Symbols vom Auflösungsvermögen des Bildschirms abhängig.

TABELLE 14–2 Desktop-Komponenten und deren Symbolgrößen

Desktop-Komponente	Hohe Auflösung	Mittlere Auflösung	Niedrige Auflösung
Dateimanager und Anwendungsmanager (Ansicht mit Name und Symbol)	mittel	mittel	mittel
Dateimanager und Anwendungsmanager (Ansicht mit Name und kleinem Symbol)	winzig	winzig	winzig
Bedienelemente im Hauptbedienfeld	groß	groß	mittel
Bedientafeln des Bedienfelds	mittel	mittel	winzig
Umschaltsymbole im Bedienfeld	klein	klein	winzig
Fenster in Symbolgröße	groß	groß	mittel

Wird beispielsweise ein Symbol mit dem Namen mail für einen Datentypen an einem Farbbildschirm definiert und ist die Vorgabe für den Dateimanager auf kleine Symbole gesetzt, wird das Symbolbild `mail.t.pm` verwendet.

Symbolsuchpfad

Das Desktop sucht eine Symboldatei oder ein Bild, indem es eine Liste von Verzeichnissen nach der Datei durchsucht. Diese Liste von Verzeichnissen, die *Symbolsuchpfad* genannt wird, wird durch den Wert mehrerer Umgebungsvariablen bestimmt. Welche Variablen verwendet werden, und wie sie zum Symbolsuchpfad zusammengestellt werden, wird im Abschnitt erläutert.

Der Standardsuchpfad lautet:

- Integrierte Symbole: `/usr/dt/appconfig/icons/ sprache`
- Systemweite Symbole: `/etc/dt/appconfig/icons/ sprache`
- Persönliche Symbole: *Home-Verzeichnis* `/.dt/icons`

Auf Symbole über das Netzwerk zugreifen

Das Desktop kann auf Symbole in anderen Systemen zugreifen. Weitere Informationen zur Erstellung eines Symbol-Servers können dem Abschnitt „Datenbank-, Symbol- und Hilfe-Services “ auf Seite 131 entnommen werden.

Symbolzuordnungen

Um eine schnellere Objekterkennung zu erreichen, können Symbole folgenden Objekten zugeordnet werden:

- Aktionen und Datentypen
- Bedienelementen im Bedienfeld und den Bedientafeln
- Anwendungsfenstern in Symbolgröße

Symboldateien angeben

Für Symbole, die für Aktionen, Datentypen und im Bedienfeld oder den Bedientafeln verwendet werden, muß nur der Dateiname ohne Pfad des Symbols angegeben werden (keine Erweiterungen). Die korrekten Erweiterungen werden je nach

Auflösung des Bildschirms, Farbunterstützung, und den Ansichtsoptionen des Dateimanagers (zum Beispiel 'Mit kleinen Symbolen') automatisch hinzugefügt.

Um den Suchpfad außer Kraft zu setzen, den vollständigen Pfad und Namen des Symbols angeben.

▼ Symbole einer Aktion oder einem Datentyp zuordnen

1. Das Symbol in dem Feld `ICON` angeben.

Wird entsprechend den Namenskonventionen für Symboldateien vorgegangen, muß nur der Dateiname ohne Pfad des Symbols angegeben werden. Das korrekte Symbol wird entsprechend der Auflösung und der Farbunterstützung des Bildschirms angezeigt.

2. Die folgenden Symbolgrößen erstellen:

- Aktionen: groß, mittel, und winzig
- Datentypen: mittel und winzig

Beispiel für eine Aktionsdefinition

Bei dem folgenden Beispiel handelt es sich um eine Aktionsdefinition zum Starten des Zeichenwerkzeugs Island Paint. Die Symbole `Ipaint.l` und `Ipaint.s` werden der Aktion zugeordnet.

```
ACTION      IslandPaintOpenDoc
{
    WINDOW_TYPE    NO-STDIO
    ICON      Ipaint
    EXEC_STRING    /usr/bin/IslandPaint %Arg_1"Datei öffnen: "%
}
```

Werden farbige Symbole verwendet, fügt das Desktop zunächst `.pm` an, wenn es nach den zugehörigen Symboldateien sucht. Ansonsten (oder wenn keine Entsprechung mit `.pm` gefunden wird) fügt das Desktop `.bm` an.

Beispiel für eine Datentypdefinition

Die folgende Datentypdefinition ordnet die Symbole `comprsd.l` und `comprsd.s` komprimierten Dateien zu.

```
DATA_ATTRIBUTES    COMPRESSED
{
```

```

ICON      comprsd

ACTIONS   Uncompress

DESCRIPTION  EINE KOMPRIMIERTER Datei wurde mit dem Befehl

'compress' komprimiert,

um weniger Speicherplatz zu belegen.

}

```

▼ Symbole in einem Bedienelement im Bedienfeld anzeigen

1. Mit dem Feld `ICON` den Symbolnamen angeben.

Überwacht das Bedienelement eine Datei (das Feld `MONITOR_TYPE` ist auf `mail` oder `file` gesetzt), im Feld `ALTERNATE_ICON` angeben, welches Symbol verwendet werden soll, wenn eine Änderung festgestellt wird.

Der Animationseffekt kann auch für Tasten und Übergabebereichbedienelemente zur Verfügung gestellt werden.

2. Die folgenden Symbolgrößen erstellen:

- Vorderes Bedienfeld und Bedientafeln: groß, mittel und winzig
- Arbeitsbereichumschalter: klein

Beispiel

Das folgende Bedienelement ändert sein Erscheinungsbild, wenn eine Datei mit dem Namen `report` in das Verzeichnis `/doc/ftp/pub/` gestellt wird. Befindet sich die Datei nicht dort, wird das Symbol `NoReport.pm` angezeigt. Befindet sich die Datei dort, wird das Symbol `Report.pm` angezeigt.

```

CONTROL MonitorReport

{

CONTAINER_NAME   container_name

TYPE             ICON

MONITOR_TYPE     file

FILE_NAME        /doc/ftp/pub/report

ICON             NoReport

```

}

▼ Symbole einem Anwendungsfenster zuordnen

1. Die Ressource `iconImage` für die Arbeitsbereichssteuerung wie folgt setzen:

```
Dtwm*clientname *iconImage: symbol_datei_name
```

Um den korrekten Wert für `clientname` festzustellen, den Anwendungsmanager öffnen und 'Fenstereigenschaften' in der Gruppe von Anwendungen 'Desktop-Werkzeuge' doppelklicken. Wird ein Fenster ausgewählt, werden dessen Eigenschaften aufgelistet. Das Merkmal `WM_CLASS` zeigt den Klassennamen des Fensters in Anführungszeichen an.

Weitere Informationen zum Festlegen von Ressourcen können dem Abschnitt entnommen werden.

2. 'Arbeitsbereichsmanager neu starten' im Arbeitsbereichsmenü auswählen.

Um zu überprüfen, ob das Symbol vom Arbeitsbereichsmanager erkannt wurde, das Fenster, dessen Symbol geändert werden soll, auf Symbolgröße verkleinern.

Hinweis - Bei einigen Anwendungen ist es nicht möglich, das Standardfenstersymbol außer Kraft zu setzen.

▼ Dateimanager zur Anzeige von Symbolen verwenden

1. Die Datei `/usr/dt/examples/sprache /IconBrowse.dt` in das Verzeichnis **Home-Verzeichnis** `/.dt/types/Iconbrowse.dt` kopieren.

2. Den Anwendungsmanager öffnen und auf 'Aktionen erneut starten' in der Gruppe von Anwendungen 'Desktop-Werkzeuge' doppelklicken.

Wechselt der Benutzer in ein Verzeichnis, das Symbole enthält (`.bm-` und `.pm-` Dateien), wird jedes Symbol mit seinem Namen dargestellt Wechselt der Benutzer beispielsweise in das Verzeichnis

`/usr/dt/appconfig/icons/sprache`, werden viele Desktop-Symbole angezeigt.

Hinweis - Die Aktivierung der Anzeige von Symbolen an Systemen mit geringer Speicherkapazität kann die Anzeige von Verzeichnissen im Dateimanager verlangsamen. Bilddateien, die größer als 256 x 256 sind, werden in der Standardkonfiguration nicht angezeigt.

Um die Anzeige von Symbolen zu inaktivieren, muß folgendermaßen vorgegangen werden:

1. **Die persönliche Kopie des Benutzers der Datei `IconBrowse.dt` löschen.**
2. **Den Anwendungsmanager öffnen und auf 'Aktionen erneut laden' in der Gruppe von Anwendungen Desktop-Werkzeuge doppelklicken.**

Empfehlungen zur Erstellung von Symbolen

Symbole, die miteinander in Bezug stehen, ähnlich gestalten. Beispielsweise empfiehlt es sich, bei der Erstellung von Symbolen für eine Anwendung darauf zu achten, daß zwischen dem Symbol der Anwendung und den Symbolen für Datendateien zweckmäßige Ähnlichkeiten bestehen.

Sicherstellen, daß die zweifarbige Version eines farbigen Symbols akzeptabel ist. Wird das Symbol an einem Monochrombildschirm oder einem Graustufen-Monitor angezeigt (oder sind an einem Farbbildschirm nicht genügend Farben definiert), wird das Symbol automatisch in seiner zweifarbigen Version angezeigt.

Um die verwendeten Farben im System zu erhalten, sollte der Benutzer die Verwendung von Farben auf diejenigen Farben einschränken, die das Desktop zur Verfügung stellt (Symbole, die mit dem Symboleditor erstellt wurden, verwenden nur Desktop-Farben)

Die Symbolgrößen, die die Desktop-Komponenten verwenden, können Tabelle 14-1 entnommen werden.

Farbpalette

Für Desktop-Symbole steht eine Palette mit folgenden 22 Farben zur Verfügung:

- Acht statische Grautöne
- Acht statische Farben: Rot, Blau, Grün, Zyanblau, Magenta, Gelb, Schwarz und Weiß
- Sechs dynamische Farben: Vordergrund, Hintergrund, obere Schattierung, untere Schattierung, Auswahl und Transparent

Mit dieser Palette werden attraktive, benutzerfreundliche Symbole erstellt, ohne Farbressourcen, die von anderen Anwendungen benötigt werden, zu übernehmen. Die meisten Symbole, die das Desktop zur Verfügung stellt, verwenden die Farbe Grau, die mit einer anderen Farbe abgesetzt wird.

Die transparente Farbe wird verwendet, um Symbole zu erstellen, die für den Betrachter auf den ersten Blick nicht rechteckig sind, da die Farbe hinter dem Symbol durchscheint.

Erweiterte Anpassung des vorderen Steuerbereichs

Der vordere Steuerbereich kann mit Hilfe seiner Pop-Up-Menüs und der 'Symbol installieren'-Steuerung in den zusätzlichen Steuerbereichen angepaßt werden.

Dieses Kapitel enthält Informationen über das Anpassen des vorderen Steuerbereichs durch das Erstellen und Bearbeiten der Konfigurationsdateien.

- „Konfigurationsdateien des vorderen Steuerbereichs“ auf Seite 264
- „Die Anpassungen der Benutzeroberfläche verwalten“ auf Seite 266
- „Aufbau der Definition des vorderen Steuerbereichs“ auf Seite 268
- „Den Hauptsteuerbereich ändern “ auf Seite 272
- „Zusätzliche Steuerbereiche erstellen und ändern“ auf Seite 278
- „Steuerungen im vorderen Steuerbereich definieren“ auf Seite 283
- „ Den Wechselbereich anpassen“ auf Seite 292
- „Allgemeine Konfiguration des vorderen Steuerbereichs“ auf Seite 294
- Ausführliche Informationen über die Steuerungen sowie die Konfiguration des vorderen Steuerbereichs finden Sie auf der Hilfeseite `dtfpfile(4X)`.
- Ausführliche Informationen über den Workspace Manager können Sie den Hilfeseiten `dtwm(1)` und `dtwmrc (4)` entnehmen.

Konfigurationsdateien des vorderen Steuerbereichs

Der vordere Steuerbereich wird in einer Datenbank mit Konfigurationsdateien definiert.

Mit Hilfe dieser Konfigurationsdateien kann der vordere Steuerbereich angepaßt werden. Bestimmte Änderungen können nur durch Bearbeiten einer Konfigurationsdatei durchgeführt werden. Dazu gehören beispielsweise folgende Operationen:

- Hinzufügen einer neuen Steuerung zum Hauptsteuerbereich.
- Hinzufügen spezieller Steuerungen (z. B. Client-Fenster).
- Ändern bestimmter Standardfunktionsweisen (z. B. ob die Steuerungen auf einen Einzel- oder Doppelklick reagieren).

Um dem Benutzer einen möglichst großen Spielraum bei der Konfiguration des Bedienfelds zu lassen, können die Konfigurationsdateien für einen einzelnen Benutzer definiert (persönlich), für das System definiert (systemweit) oder auf anderen Systemen gespeichert sein.

Der vordere Steuerbereich wird vom Workspace Manager erstellt und verwaltet.

Standardkonfigurationsdatei für den vorderen Steuerbereich

Der vordere Steuerbereich wird mit der Konfigurationsdatei
`/usr/dt/appconfig/types/Sprache /dtwm.fp` definiert.

Diese Datei sollte nicht geändert werden.

Suchpfad für die Konfigurationsdateien des vorderen Steuerbereichs

Die Definition des vorderen Steuerbereichs kann auf beliebig viele Dateien auf dem lokalen oder auf entfernten Systemen verteilt werden.

Die Definitionsdateien müssen folgende Bedingungen erfüllen:

- Die Datei muß die Erweiterung `.fp` aufweisen (z. B. `mail.fp`).
- Die Datei muß sich im Suchpfad der Aktionsdatenbank befinden.

Der Standardsuchpfad enthält folgende Verzeichnisse, die in der angegebenen Reihenfolge durchsucht werden:

Persönliche Anpassungen - *Home-Verzeichnis* / .dt/ty

Systemweite Anpassungen - /etc/dt/appconfig/types/ *Sprache*

Integrierter Steuerbereich und Steuerungen - /usr/dt/appconfig/types/ *Sprache*

Ein weiteres Verzeichnis, das *Home-Verzeichnis* / .dt/types/tp_dynamic, wird für persönliche Anpassungen verwendet, die mit der Benutzeroberfläche durchgeführt werden. Verwenden Sie es nicht für manuelle Anpassungen.

In den Datenbanksuchpfad können auch weitere Verzeichnisse aufgenommen werden, die für die Netzwerkkonfiguration des Systems verwendet werden. Dies sind insbesondere entfernte Verzeichnisse, wenn das System für den Zugriff auf einen Anwendungsserver konfiguriert ist. Weitere Informationen hierzu finden Sie im Abschnitt „Datenbanksuchpfad (Aktionen/Datentypen) ” auf Seite 153.

Den vorderen Steuerbereich konfigurieren: Vorrangsregeln

Der vordere Steuerbereich wird mit den Konfigurationsdateien im Aktionsdatenbanksuchpfad konfiguriert.

Tritt ein Konflikt zwischen Komponenten in der Definition auf, wird die zu verwendende Definition anhand bestimmter Regeln ausgewählt. Ein Konflikt zwischen zwei Komponenten entsteht in den folgenden Situationen:

- Die Komponenten haben den gleichen Steuerungsnamen CONTAINER_NAME und Typ CONTAINER_TYPE.
- Die Komponenten haben dieselbe Position (unterschiedliche Namen, aber identische Werte für CONTAINER_NAME , CONTAINER_TYPE und POSITION_HINTS).

Für den vorderen Steuerbereich gelten folgende Vorrangsregeln:

- Haben Komponenten denselben Steuerungsnamen sowie einen identischen Container-Namen und -Typ, wird die zuerst gelesene Komponente verwendet.

Enthalten beispielsweise eine systemweite und eine integrierte Steuerung die folgenden Felder, sind aber ansonsten unterschiedlich:

```
CONTROL TextEditor
{
    CONTAINER_TYPE    BOX
    CONTAINER_NAME    Top
    ...
}
```

```
}
```

wird die systemweite Steuerung verwendet.

- Sind zwei Komponenten für dieselbe Position definiert, werden sie in der Reihenfolge plazierte, in der sie gelesen werden.

Erstellt ein Benutzer beispielsweise eine neue persönliche Steuerung für den Hauptsteuerbereich (`CONTAINER_TYPE BOX` und `CONTAINER_NAME Top`) und weist dieser den Wert `POSITION_HINTS 5` zu, verschiebt die persönliche Steuerung die integrierte und alle anderen Steuerungen mit höheren Positionsnummern um eine Position nach rechts.

Hinweis - Wird eine Steuerung durch Erstellen einer neuen systemweiten oder persönlichen Version geändert, muß in der neuen Definition derselbe Steuerungsname (`CONTAINER_NAME` und `CONTAINER_TYPE`) angegeben werden. Andernfalls werden beide Steuerungen angezeigt.

Dynamisch erstellte Dateien für den vorderen Steuerbereich

Paßt der Benutzer den vorderen Steuerbereich mit der Steuerung 'Symbol installieren' und den Pop-Up-Menüs an, werden die entsprechenden Dateien in das Verzeichnis *Home-Verzeichnis* / `.dt/types/ep_dynamic` geschrieben.

Der vordere Steuerbereich erstellt eine weitere Datei, *Home-Verzeichnis* / `.dt/sessions/dtwmfp.session`, mit der der aktuelle Status des angepaßten Steuerbereichs für jede Sitzung gesichert und wiederhergestellt werden kann.

Die Anpassungen der Benutzeroberfläche verwalten

Benutzer können mit Hilfe der Pop-Up-Menüs und der 'Symbol installieren'-Steuerungen den vorderen Steuerbereich weitgehend an die jeweiligen Anforderungen anpassen.

Dieser Abschnitt enthält Informationen zu den folgenden Themen:

- Verhindern bestimmter persönlicher Anpassungen. Es kann beispielsweise ausgeschlossen werden, daß ein Benutzer eine Steuerung löschen kann.

- Rückgängigmachen persönlicher Anpassungen. Ein Benutzer könnte Sie beispielsweise auffordern, eine versehentlich gelöschte Steuerung wiederherzustellen.

▼ Persönliche Anpassungen verhindern

1. **Handelt es sich um eine integrierte Steuerung, kopieren Sie deren Definition aus `/usr/dt/appconfig/types/Sprache /dtwm.fp` in das Verzeichnis `/etc/dt/appconfig/types/Sprache/Name.fp`.**
2. **Fügen Sie folgende Zeile zur Definition der Steuerung hinzu:**

```
LOCKED True
```

▼ Eine gelöschte Steuerung oder einen gelöschten zusätzlichen Steuerbereich wiederherstellen

Mit der Aktion 'Vorderen Steuerbereich wiederherstellen' in der Anwendungsgruppe Desktop-Werkzeuge können alle an der Benutzeroberfläche vorgenommenen Änderungen rückgängig gemacht werden. Der Benutzer kann auf diese Weise alle Anpassungen widerrufen, die mit den Pop-Up-Menüs des vorderen Steuerbereichs durchgeführt wurden.

Folgendermaßen kann eine Steuerung wiederhergestellt werden.

- ◆ **Löschen Sie im Verzeichnis *Home-Verzeichnis* `/.dt/types/fp_dynamic` die Datei, die beim Löschen der Steuerung durch den Benutzer erstellt wurde. Die Steuerung erhält den Namen der ursprünglich gelöschten.**

Löscht der Benutzer beispielsweise die Steuerung 'Symboleditor', ist eine Datei mit folgendem Inhalt im Verzeichnis `fp_dynamic` vorhanden:

```
CONTROL IconEditor
{
...
DELETE True
}
```

Löscht der Benutzer einen zusätzlichen Steuerbereich, wird für diesen und für jede Steuerung im Bereich eine eigene dynamische Datei erstellt.

Aufbau der Definition des vorderen Steuerbereichs

Der vordere Steuerbereich wird durch die Definitionen seiner Komponenten erstellt. Für jede dieser Komponenten ist eine bestimmte Syntax erforderlich. Sie definiert, wo sich die Komponente im Steuerbereich befindet, wie sie aussieht und wie sie sich verhält.

Komponenten des vorderen Steuerbereichs

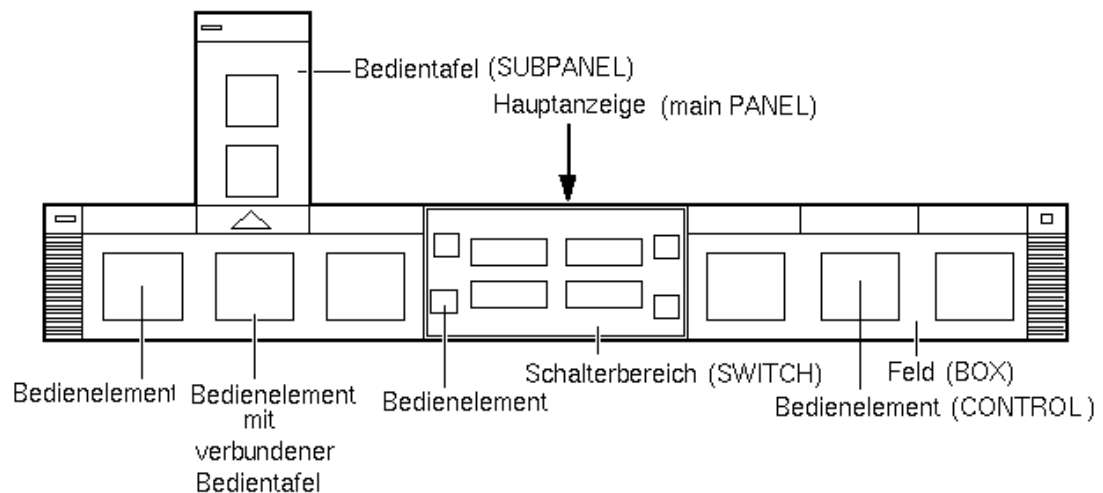


Abbildung 15-1 Steuerbereichskomponenten

Der Steuerbereich wird von außen nach innen erstellt:

- PANEL ist der Container auf oberster Ebene (das übergeordnete Objekt) für den gesamten Steuerbereich.
- PANEL ist ein Container für ein oder mehrere BOX-Objekte.
- BOX ist ein Container für ein oder mehrere CONTROL-Objekte.

Es gibt zwei spezielle Container-Typen:

- Ein zusätzlicher Steuerbereich (SUBPANEL) ist einer bestimmten Steuerung zugeordnet (die Steuerung ist der Container des zusätzlichen Steuerbereichs). Der Steuerbereich öffnet sich über der Steuerung, zu der er gehört.

- Ein SWITCH enthält Schaltflächen zum Wechseln des Arbeitsbereichs und weitere Steuerungen.

Allgemeine Syntax der Definition des vorderen Steuerbereichs

Jede Komponente im vorderen Steuerbereich wird mit folgender Syntax einzeln definiert:

COMPONENT Name

```
{
  KEYWORDWert
  KEYWORDWert
  ...
}
```

Einige Schlüsselwörter müssen angegeben werden, andere sind optional. Weitere Informationen finden Sie auf der Hilfeseite `dtfpfile(4X)`.

PANEL-Definition

Das Objekt `PANEL` ist die Komponente der obersten Ebene. Bei der Definition müssen folgende Informationen angegeben werden:

- Der Name des vorderen Steuerbereichs.
- Verschiedene Felder, die das Erscheinungsbild und die Funktionsweise des Steuerbereichs beschreiben.

PANEL Vorderer_Steuerbereichsname

```
{
  KEYWORD Wert
  KEYWORD Wert
  ...
}
```

Vorderer_Steuerbereichsname ist ein eindeutiger Name für den vorderen Steuerbereich (Standardname "FrontPanel").

BOX-Definitionen

Bei der Definition eines BOX-Objekts müssen folgende Informationen angegeben werden:

- Der Name des BOX-Objekts
- In welchem PANEL-Objekt es sich befindet (CONTAINER_NAME)
- Die Position innerhalb des PANEL-Objekts (POSITION_HINTS)
- Verschiedene Felder, die das Erscheinungsbild und die Funktionsweise des gesamten Objekts beschreiben.

BOX *Feldname*

```
{  
    CONTAINER_NAME      Vorderer_Steuerbereichsname  
    POSITION_HINTS       Position  
    KEYWORD             Wert  
    KEYWORD             Wert  
    ...  
}
```

CONTROL-Definitionen

Bei der Definition eines CONTROL-Objekts müssen folgende Informationen angegeben werden:

- Der Name des CONTROL-Objekts
- Ob sich die Steuerung in einem Feld (BOX), einem zusätzlichen Steuerbereich (SUBPANEL) oder Wechselbereich (SWITCH) befindet (CONTAINER_TYPE)
- In welchem Feld, zusätzlichen Steuerbereich oder Wechselbereich sich die Steuerung befindet (CONTAINER_NAME)
- Die Position der CONTROL-Komponente innerhalb des BOX-Objekts (POSITION_HINTS)
- Verschiedene Felder, die das Erscheinungsbild und die Funktionsweise des gesamten Objekts beschreiben.

CONTROL *Steuerungsname*

```
{  
    CONTAINER_TYPE      BOX oder SUBPANEL  
  
    oder SWITCH  
    CONTAINER_NAME      Feldname oder
```

```

Steuerbereichsname oder Wechselbereichsname

TYPE          Steuerungstyp

POSITION_HINTS  Position

KEYWORD        Wert

KEYWORD        Wert

...

}

```

SUBPANEL-Definitionen

Bei der Definition eines SUBPANEL-Objekts müssen folgende Informationen angegeben werden:

- Der Name des SUBPANEL-Objekts
- Der Name der Steuerung, der der zusätzliche Steuerbereich zugeordnet ist (CONTAINER_NAME)
- Verschiedene Felder, die das Erscheinungsbild und die Funktionsweise des gesamten Objekts beschreiben.

```

SUBPANEL Zusätzlicher_Steuerbereichsname

{

CONTAINER_NAME  Steuerungsname

KEYWORD        Wert

KEYWORD        Wert

...

}

```

SWITCH-Definition

Bei der Definition eines SWITCH-Objekts müssen folgende Informationen angegeben werden:

- Der Name des SWITCH-Objekts
- In welcher BOX-Komponente sich das SWITCH-Objekt befindet (CONTAINER_NAME).

- Die Position des SWITCH-Objekts innerhalb der BOX-Komponente (POSITION_HINTS)
- Verschiedene Felder, die das Erscheinungsbild und die Funktionsweise des gesamten Objekts beschreiben.

SWITCH *Wechselbereichsname*

```
{
CONTAINER_NAME    Feldname
POSITION_HINTS    Position
KEYWORD           Wert
KEYWORD           Wert
...
}
```

Den Hauptsteuerbereich ändern

Der Hauptsteuerbereich ist der vordere Steuerbereich ohne die zusätzlichen Steuerbereiche.

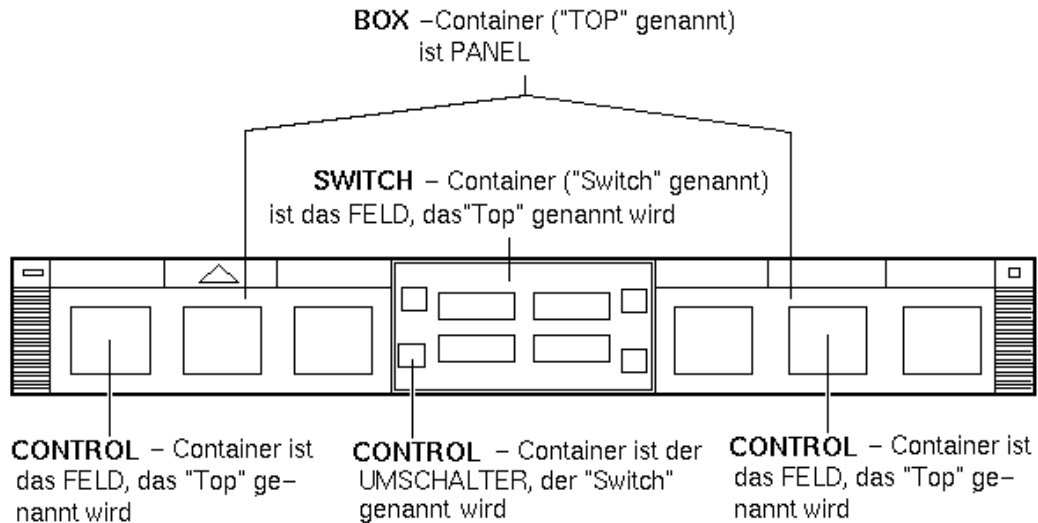


Abbildung 15-2 Hauptsteuerbereich-Container

Folgende Änderungen können vorgenommen werden:

- Steuerungen hinzufügen oder entfernen.
- Die Positionen der Steuerungen ändern.

▼ Eine Steuerung dem Hauptsteuerbereich hinzufügen

1. Erstellen Sie eine Konfigurationsdatei für den vorderen Steuerbereich:

- Systemweit: `/etc/dt/appconfig/types/ Sprache/*.fp`
- Persönlich: `Home-Verzeichnis/.dt/types/*.fp`

2. Definieren Sie die Steuerung in der Datei.

Geben Sie mit den Feldern `CONTAINER_NAME` und `CONTAINER_TYPE` den Container der Steuerung an:

```
CONTAINER_NAME  Top
                CONTAINER_TYPE  BOX
```

Mit dem Feld `POSITION_HINTS` wird die Position der Steuerung festgelegt. Da Anpassungen Vorrang vor integrierten Steuerungen haben, schiebt die neue Steuerung bereits vorhandene um eine Position nach rechts.

3. Speichern Sie die Konfigurationsdatei.

4. Erstellen Sie ein Symbol für die Steuerung im vorderen Steuerbereich.

Weitere Informationen finden Sie im Abschnitt „Das von einer Steuerung verwendete Symbol festlegen“ auf Seite 277.

5. Wählen Sie 'Workspace Manager neu starten' im Menü.

Die folgende Steuerungsdefinition in der Datei `/etc/dt/appconfig/types/ Sprache/audio.fp` fügt beispielsweise eine Audio-Steuerung zwischen die Uhr- und die Kalender-Steuerung ein.

```
CONTROL AudioApplication
{
    TYPE          icon
    CONTAINER_NAME  Top
    CONTAINER_TYPE  BOX
    ICON          AudioApp
    POSITION_HINTS  2
```

```

PUSH_ACTION      StartAudioApplication
PUSH_RECALL      true
}

```

▼ Eine Steuerung entfernen

1. Erstellen Sie eine Konfigurationsdatei für den vorderen Steuerbereich:

- Systemweit: `/etc/dt/appconfig/types/Sprache/Name.fp`
- Persönlich: `Home-Verzeichnis/.dt/types/Name.fp`

2. Kopieren Sie die Definition der zu entfernenden Steuerung in die neue Datei.

Handelt es sich um eine integrierte Steuerung, finden Sie die Definition in der Datei `/usr/dt/appconfig/types/Sprache/dtwm.fp`.

Sie müssen nicht die gesamte Definition kopieren. Es werden jedoch die Felder `CONTAINER_NAME` und `CONTAINER_TYPE` benötigt.

3. Fügen Sie das Feld `DELETE` der Definition hinzu:

```
DELETE      True
```

4. Speichern Sie die Konfigurationsdatei.

5. Wählen Sie 'Workspace Manager neu starten' im Menü.

Die folgende Definition in der Datei

`/etc/dt/appconfig/types/Sprache/TrashCan.fp` entfernt beispielsweise den Papierkorb aus dem vorderen Steuerbereich.

```

CONTROL Trash
{
    CONTAINER_NAME  Top
    CONTAINER_TYPE  BOX
    DELETE          True
}

```

▼ Eine Steuerung ändern

Folgendermaßen kann die Definition einer Steuerung geändert werden (um beispielsweise ein anderes Symbol zu verwenden).

1. **Kopieren Sie die gesamte Definition der Steuerung aus** `/usr/dt/appconfig/types/Sprache/dtwm.fp` **in die Datei:**
 - Systemweit: `/etc/dt/appconfig/types/Sprache/Name.fp`
 - Persönlich: `Home-Verzeichnis/.dt/types/Name.fp`.
2. **Bearbeiten Sie die Felder, die geändert werden sollen. Es können auch Felder hinzugefügt werden.**
3. **Speichern Sie die Datei.**
4. **Wählen Sie 'Workspace Manager neu starten' im Menü.**

▼ Die Position von Steuerungen ändern

1. **Kopieren Sie die Definitionen der gewünschten Steuerungen aus** `/usr/dt/appconfig/types/Sprache/dtwm.fp` **in die Datei:**
 - Systemweit: `/etc/dt/appconfig/types/Sprache/Name.fp`
 - Persönlich: `Home-Verzeichnis/.dt/types/Name.fp`.

Für jede Steuerung, deren Position geändert werden soll, muß die gesamte Definition kopiert werden.
2. **Vertauschen Sie die Werte der POSITION_HINTS-Felder in den Steuerungsdefinitionen.**
3. **Speichern Sie die Datei.**
4. **Wählen Sie 'Workspace Manager neu starten' im Menü.**

Die folgenden Definitionen in der Datei `/etc/dt/appconfig/types/C/MailHelp.fp` vertauschen beispielsweise die Positionen der Mail- und der Hilfe-Manager-Steuerung und sperren diese Komponenten gegen Änderungen durch Benutzer.

```
CONTROL Mail
{
    POSITION_HINTS      12
    LOCKED             True
    ...Restliche Definition der Steuerung
```

```
}
```

CONTROL Help

```
{  
    POSITION_HINTS    5  
    LOCKED           True  
    ...Restliche Definition der Steuerung  
}
```

▼ Eine Steuerung im vorderen Steuerbereich ersetzen

- ◆ Erstellen Sie eine neue Steuerungsdefinition mit identischen Angaben für:

- *Steuerungsname*

- CONTAINER_NAME (Wert)

So sind beispielsweise die folgenden beiden Steuerungen in zwei verschiedenen Konfigurationsdateien definiert. Sie haben identische Steuerungs- und Container-Namen und gelten daher als dieselbe Steuerung.

- Definition in `/etc/dt/appconfig/types/C/SysControls.fp`:

```
Control ImportantApplication  
  
{  
    CONTAINER_NAME    Top  
    CONTAINER_TYPE    BOX  
    POSITION_HINTS     2  
    ...  
}
```

- Definition in *Home-Verzeichnis* `/.dt/types/MyControls.fp`:

```
Control ImportantApplication  
  
{
```

```

CONTAINER_NAME      Top
CONTAINER_TYPE      BOX
POSITION_HINTS      6
...
}

```

Die persönliche Steuerung hat Vorrang und wird daher an Position 6 eingefügt.

Das von einer Steuerung verwendete Symbol festlegen

Das Feld `ICON` der Steuerungsdefinition gibt die Symbolgrafik für die Steuerung an.

Das Feld `ICON` kann folgende Werte annehmen:

- Basisdateiname

Der Basisdateiname ist der Name der Datei mit der Symbolgrafik ohne die Erweiterungen für Größe (`m` und `t`) und Grafiktyp (`bm` und `pm`). Geben Sie beispielsweise für die Dateien `MyGame.l.pm` und `MyGame.m.pm` den Wert `MyGame` an.

Wird der Basisdateiname verwendet, müssen sich die Symboldateien in einem Verzeichnis des Symbolsuchpfads befinden:

- Persönliche Symbole: *Home-Verzeichnis* / `.dt/icons`
- Systemweite Symbole: `/etc/dt/appconfig/icons/` *Sprache*

- Absoluter Pfad (einschließlich des vollständigen Dateinamens)

Der absolute Pfad sollte nur angegeben werden, wenn sich die Grafikdatei nicht im Symbolsuchpfad befindet.

Die erforderliche Symbolgröße ist von der Position der Steuerung abhängig:

Position Größe

Hauptsteuerbereich – 48 x 48 Pixel (*Name.l.pm* oder *Name.l.bm*)

Zusätzlicher Steuerbereich – 24 x 24 Pixel (*Name.s.pm* oder *Name.s.bm*)

Plazieren Sie die Symboldatei in einem der folgenden Verzeichnisse:

- Persönliche Symbole: *Home-Verzeichnis* / `.dt/icons`
- Systemweite Symbole: `/etc/dt/appconfig/icons/` *Sprache*

Zusätzliche Steuerbereiche erstellen und ändern

Benutzer können zusätzliche Steuerbereiche mit den Pop-Up-Menüs des vorderen Steuerbereichs erstellen oder ändern.

In diesem Abschnitt finden Sie Informationen über systemweite Anpassungen, die mit den Konfigurationsdateien des vorderen Steuerbereichs vorgenommen werden müssen.

Ein zusätzlicher Steuerbereich wird einer Steuerung im Hauptsteuerbereich zugeordnet.

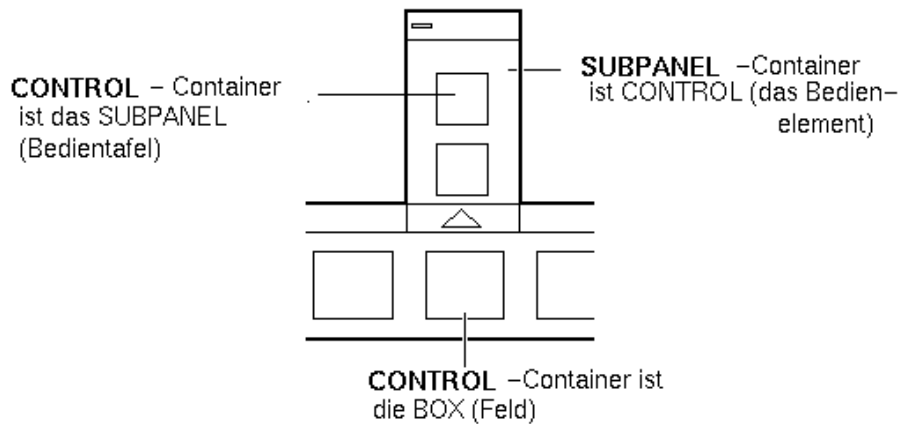


Abbildung 15-3 Der Container eines zusätzlichen Steuerbereichs ist die Steuerung, der er zugeordnet ist

Die Zuordnung wird in der Definition des zusätzlichen Steuerbereichs vorgenommen. Das Feld `CONTAINER_NAME` gibt die betreffende Steuerung an:

```
CONTROL Steuerungsname
```

```
{  
    ...  
}
```

```
SUBPANEL Zusätzlicher_Steuerbereich
```

```

{
CONTAINER_NAME      Steuerungsname
...
}

```

▼ Einen neuen systemweiten zusätzlichen Steuerbereich erstellen

1. **Ermitteln Sie den *Steuerungsname* der Steuerung im Hauptsteuerbereich, der Sie den zusätzlichen Steuerbereich zuordnen wollen.**

Handelt es sich um eine integrierte Steuerung, finden Sie die Definition in der Datei `/usr/dt/appconfig/types/Sprache/dtwm.fp`.

2. **Erstellen Sie eine neue Datei `/etc/dt/appconfig/types/Sprache/*.fp`.**

3. **Definieren Sie den zusätzlichen Steuerbereich:**

```

SUBPANEL Zusätzlicher_Steuerbereich
{
CONTAINER_NAME      Steuerungsname
TITLE               Wert
KEYWORD             Wert
...
}

```

4. **Speichern Sie die neue Konfigurationsdatei.**
5. **Wählen Sie 'Workspace Manager neu starten' im Menü.**

Integrierte zusätzliche Steuerbereiche anpassen

Sie können die allgemeinen Eigenschaften (z. B. Titel) und den Inhalt der integrierten zusätzlichen Steuerbereiche ändern.

Allgemeine Eigenschaften eines integrierten zusätzlichen Steuerbereichs ändern

1. Erstellen Sie eine neue Konfigurationsdatei für den vorderen Steuerbereich im Verzeichnis:

- Systemweit: `/etc/dt/appconfig/types/Sprache/Name.fp`
- Persönlich: `Home-Verzeichnis / .dt/types/Name.fp`.

2. Kopieren Sie die gesamte SUBPANEL-Standarddefinition aus `/usr/dt/appconfig/types/Sprache /dtwm.fp` in die neue Datei:

```
SUBPANEL    Zusätzlicher_Steuerbereich

{

    ...

}
```

3. Ändern Sie die Definition des zusätzlichen Steuerbereichs.

4. Speichern Sie die neue Konfigurationsdatei.

5. Wählen Sie 'Workspace Manager neu starten' im Menü.

Die folgende Definition in der Datei

`/users/janice/.dt/types/PerApps.fp` ändert beispielsweise den Namen des zusätzlichen Steuerbereichs 'Persönliche Anwendungen':

```
SUBPANEL PersAppsSubpanel

{

    CONTAINER_NAME    TextEditor

    TITLE             Meine Anwendungen

}
```

Eine systemweite Steuerung einem integrierten zusätzlichen Steuerbereich hinzufügen

1. Erstellen Sie eine neue Konfigurationsdatei für den vorderen Steuerbereich im Verzeichnis `/etc/dt/appconfig/types/Sprache/Name.fp`.

2. Definieren Sie die systemweite Steuerung.

Geben Sie in den Feldern `CONTAINER_NAME` und `CONTAINER_TYPE` den Container für die Steuerung an:


```

CONTROL Steuerungsname
{
    CONTAINER_NAME      Zusätzlicher_Steuerbereich
    CONTAINER_TYPE      SUBPANEL
    ...
}

```

Weitere Informationen finden Sie im Abschnitt „Steuerungen im vorderen Steuerbereich definieren“ auf Seite 283.

3. Speichern Sie die Konfigurationsdatei.

4. Wählen Sie 'Workspace Manager neu starten' im Menü.

Die folgende Steuerungsdefinition in der neuen Datei `/etc/dt/appconfig/types/Sprache/DigitalClock.fp` fügt beispielsweise die Steuerung 'Digitale Uhr' (in der Anwendungsgruppe 'Desktop-Werkzeuge') dem zusätzlichen Steuerbereich 'Persönliche Anwendungen' aller Benutzer hinzu.

```

CONTROL DigitalClockControl
{
    TYPE      icon
    CONTAINER_NAME      PerAppsSubpanel
    CONTAINER_TYPE      SUBPANEL
    ICON      Dtdgclk
    PUSH_ACTION      DigitalClock
    PUSH_RECALL      True
}

```

Eine Steuerung aus einem integrierten zusätzlichen Steuerbereich entfernen

- ◆ **Gehen Sie wie beim Entfernen einer Steuerung aus dem Hauptsteuerbereich vor (siehe „Eine Steuerung entfernen“ auf Seite 274).**

Die Steuerung 'Symbol installieren' entfernen

- ◆ Fügen Sie das folgende Feld der Definition des zusätzlichen Steuerbereichs hinzu:

```
CONTROL_INSTALL      False
```

▼ Die Selbstschließfunktion zusätzlicher Steuerbereiche ändern

Zusätzliche Steuerbereiche werden standardmäßig automatisch geschlossen, sobald der Benutzer eine Steuerung auswählt, wenn der Steuerbereich nicht von seiner ursprünglichen Position verschoben wurde.

Der vordere Steuerbereich kann so konfiguriert werden, daß zusätzliche Steuerbereiche offen bleiben, bis sie vom Benutzer explizit geschlossen werden.

1. Erstellen Sie eine neue Konfigurationsdatei für den vorderen Steuerbereich im Verzeichnis:

- Systemweit: `/etc/dt/appconfig/types/Sprache/*.fp`
- Persönlich: `Home-Verzeichnis/.dt/types/*.fp`

2. Kopieren Sie die PANEL-Standarddefinition aus

`/usr/dt/appconfig/types/Sprache/dtwm.fp` in die neue Datei:

```
PANEL FrontPanel
{
    ...
}
```

3. Fügen Sie folgendes Feld der PANEL-Definition hinzu:

```
SUBPANEL_UNPOST      False
```

4. Speichern Sie die neue Konfigurationsdatei.

5. Wählen Sie 'Workspace Manager neu starten' im Menü.

Steuerungen im vorderen Steuerbereich definieren

Der Benutzer kann persönliche Steuerungen durch Ablegen der entsprechenden Symbole auf der Steuerung 'Symbol installieren' erstellen.

Diese Vorgehensweise ermöglicht zwar eine einfache Anpassung der Benutzeroberfläche, jedoch steht nicht der volle Funktionsumfang der Steuerungen im vorderen Steuerbereich zur Verfügung. Eine mit 'Symbol installieren' erstellte Steuerung verfügt beispielsweise nicht über folgende Merkmale:

- Animation
- Anzeigen eines Client-Fensters
- Ändern des Erscheinungsbilds bei einem bestimmten Ereignis (z. B. beim Empfangen neuer Post)

Dieser Abschnitt beschreibt, wie Steuerungen im vorderen Steuerbereich manuell erstellt werden können.

Vollständige Informationen zur Syntax für das Erstellen von Steuerungen sind auf der Hilfeseite `dtfpfile(4X)` zu finden.

Steuerungsdefinitionen

Die Struktur der Definition einer Steuerung im vorderen Steuerbereich lautet:

CONTROL *Steuerungsname*

```
{  
    TYPE           Steuerungstyp  
    CONTAINER_NAME Wert  
    CONTAINER_TYPE Wert  
    andere Felder zum Definieren des Erscheinungsbilds und der Funktionsweise  
}
```

Steuerungstypen

Das Feld `TYPE` der Definition bestimmt das grundlegende Verhalten der Steuerung.

TYPE (Steuerungstyp)

Funktionsweise

Erweiterte Anpassung des vorderen Steuerbereichs **283**

icon (Standard).	Die Steuerung führt eine bestimmte Aktion aus, sobald auf sie geklickt bzw. eine Datei auf ihr abgelegt wird.
blank	Platzhalter zum Festlegen des Abstands zwischen den Steuerungen.
busy	Betriebsanzeige. Die Steuerung blinkt (schaltet die Grafik um), wenn eine Aktion aufgerufen wird.
client	Ein Client-Fenster im vorderen Steuerbereich.
clock	Die Steuerung zeigt die Uhrzeit an.
date	Die Steuerung zeigt das aktuelle Datum an.
file	Die Steuerung repräsentiert eine Datei. Durch Auswählen der Steuerung wird die Standardaktion für die Datei ausgeführt.

▼ Eine neue Steuerung erstellen

Dieser Abschnitt enthält Informationen über die allgemeinen Schritte zum Definieren einer Steuerung und zum Erstellen der verschiedenen Steuerungstypen.

- 1. Soll die Steuerung über eine `PUSH_ACTION` und/oder `DROP_ACTION` verfügen, erstellen Sie die Aktionsdefinitionen. Diese Aktionen werden ausgeführt, wenn der Benutzer auf die Steuerung klickt oder eine Datei auf ihr ablegt.**
- 2. Erstellen Sie die Symboldateien für die Steuerung.**
Informationen über die Größe, Namen und Verzeichnisse der Symbole finden Sie im Abschnitt „Symbolbilddateien“ auf Seite 253.
- 3. Erstellen Sie eine neue Konfigurationsdatei für den vorderen Steuerbereich im Verzeichnis:**
 - Systemweit: `/etc/dt/appconfig/types/ Sprache/*.fp`
 - Persönlich: *Home-Verzeichnis* `/.dt/types/*.fp`
- 4. Fügen Sie die Steuerungsdefinition der Datei hinzu.**
- 5. Speichern Sie die Datei.**
- 6. Wählen Sie 'Workspace Manager neu starten' im Menü.**

Eine Steuerung erstellen, die beim Klicken eine Aktion ausführt

Mit den folgenden Feldern wird die Funktionsweise der Steuerung definiert:

- TYPE: Hat den Wert 'icon'.
- PUSH_ACTION: Gibt den Namen der auszuführenden Aktion an.

Die folgende Steuerung wird dem zusätzlichen Steuerbereich 'Persönliche Anwendungen' hinzugefügt und startet ein Spiel:

```
CONTROL Ball
{
    TYPE          icon
    CONTAINER_NAME    PersAppsSubpanel
    CONTAINER_TYPE    SUBPANEL
    ICON           ball
    PUSH_ACTION      RunBallGame
    HELP_STRING      "Wählen Sie diese Steuerung, um 'Ball' zu spielen."
}
```

Die folgende Steuerung wird in der linken oberen Ecke des Wechselbereichs platziert und startet eine Aktion namens CutDisp.

```
CONTROL StartCutDisp
{
    TYPE          icon
    CONTAINER_NAME    Switch
    CONTAINER_TYPE    SWITCH
    POSITION_HINTS    first
    ICON           cutdisp
    HELP_STRING      "Wählen Sie diese Steuerung, um die Aktion cutdisp auszuführen."
    PUSH_ACTION      CutDisp
}
```

Eine Steuerung erstellen, die eine Datei öffnet

Mit den folgenden Feldern wird die Funktionsweise der Steuerung definiert:

- TYPE: Hat den Wert 'file'.

- `FILE_NAME`: Gibt den Pfad der zu öffnenden Datei an.
- `PUSH_ACTION`: Hat den Wert 'Open'.

Es muß eine Open-Aktion für den Datentyp der Datei existieren.

Im folgenden Beispiel wird eine Steuerung am rechten Rand des Hauptsteuerbereichs hinzugefügt. Sie ruft den Texteditor mit der Datendatei `/users/ellen/PhoneList.txt` auf. Die Open-Aktion für `*.txt`-Dateien ist in der Standardaktionsdatenbank enthalten.

```
CONTROL EditPhoneList
{
    TYPE          file
    FILE_NAME     /users/ellen/PhoneList.txt
    CONTAINER_NAME Top
    CONTAINER_TYPE BOX
    POSITION_HINTS last
    ICON         PhoneBook
    HELP_STRING   "Diese Steuerung zeigt die Telefonliste von Ellen an."
    PUSH_ACTION   Open
}
```

Eine Steuerung als Ablagezone erstellen

Geben Sie im Feld `DROP_ACTION` die Aktion an, die beim Ablegen einer Datei auf der Steuerung ausgeführt wird. Die Aktion muß in der Lage sein, Dateiarumente zu verarbeiten.

Oftmals enthält eine Steuerungsdefinition sowohl das Feld `PUSH_ACTION` als auch das Feld `DROP_ACTION`. Es kann dann dieselbe Aktion für Klicken und Ablegen verwendet werden.

Die folgende Steuerung befindet sich im zusätzlichen Steuerbereich 'Persönliche Anwendungen' und startet den X-Client `xwud`, der ein Dateiarument benötigt.

```
CONTROL Run_xwud
{
    CONTAINER_NAME PerAppsSubpanel
    CONTAINER_TYPE SUBPANEL
    POSITION_HINTS 2
    ICON         XwudImage
}
```

```

PUSH_ACTION      RunXwud
DROP_ACTION      RunXwud
}

```

Eine Steuerung erstellen, die eine Datei überwacht

Mit den folgenden Feldern wird die Funktionsweise der Steuerung definiert:

- **TYPE:** Geben Sie einen der folgenden Werte an:
 - `icon` - Verwenden Sie diesen Typ, wenn eine `PUSH_ACTION` und/oder `DROP_ACTION` für die Steuerung definiert werden soll.
 - `file` - Verwenden Sie diesen Typ, wenn sich die Steuerung bei der Auswahl wie die entsprechende Datei verhalten soll, wenn im Dateimanager doppelt auf deren Symbol geklickt wird.
- **ICON und ALTERNATE_ICON :** Diese beiden Felder legen die Grafiken fest, die für den geänderten und den nicht geänderten Status der überwachten Datei angezeigt werden.
- **MONITOR_TYPE:** Dieses Feld beschreibt die Bedingungen für das Ändern der Grafik. Verwenden Sie hier einen der folgenden Werte:
 - `mail` - Die Steuerung ändert das Erscheinungsbild, wenn Informationen in die Datei eingefügt werden.
 - `file` - Die Steuerung ändert das Erscheinungsbild, wenn die überwachte Datei nicht mehr leer ist.
- **FILE_NAME:** Mit diesem Feld wird die zu überwachende Datei angegeben.

Die folgende Steuerung befindet sich am oberen Rand des zusätzlichen Steuerbereichs 'Persönliche Anwendungen' und sucht nach der Datei `meetings`, die per 'Anonymous FTP' auf das System übertragen wurde.

```

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
}

```

```
}
```

Eine Einzelinstanz-Steuerung (Schalter) erstellen

Eine Einzelinstanz-Steuerung prüft, ob der Prozeß bereits ausgeführt wird, der mit `PUSH_ACTION` gestartet wird. Wurde der Prozeß noch nicht gestartet, wird `PUSH_ACTION` ausgeführt. Läuft der Prozeß bereits, wird das betreffende Fenster in den Vordergrund des aktuellen Arbeitsbereichs gebracht.

Mit den folgenden Feldern wird die Funktionsweise der Steuerung definiert:

- `PUSH_RECALL`: Hat den Wert `True`.
- `CLIENT_NAME`: Gibt den Namen des Client an.

Der Wert von `CLIENT_NAME` muß mit der ersten Zeichenfolge (*res_name*) der Eigenschaft `WM_CLASS` des Hauptfensters der Anwendung übereinstimmen. Weitere Informationen hierzu finden Sie auf der Hilfeseite `xprop(1)`.

- `PUSH_ACTION`: Gibt die Aktion an, die beim Klicken auf die Steuerung ausgeführt wird.

Folgende Steuerung startet eine Instanz einer Anwendung, deren Aktion den Namen `MyEditor` hat.

```
CONTROL MyEditor
{
    TYPE          icon
    CONTAINER_NAME    Top
    CONTAINER_TYPE    BOX
    POSITION_HINTS    15
    PUSH_RECALL      True
    CLIENT_NAME      BestEditor
    PUSH_ACTION      StartMyEditor
    ICON            MyEd
}
```

Eine Client-Fenstersteuerung erstellen

Eine Client-Fenstersteuerung ist ein Anwendungsfenster im vorderen Steuerbereich. Es kann beispielsweise eine Systemauslastungsanzeige im vorderen Steuerbereich durch Erstellen einer `xload`-Client-Fenstersteuerung angezeigt werden.

1. Definieren Sie die Steuerung.

Mit den folgenden Feldern wird die Funktionsweise der Steuerung definiert:

- TYPE: Hat den Wert `client`.
- CLIENT_NAME: Gibt den zu startenden Client an.
Der Wert von `CLIENT_NAME` muß mit der ersten Zeichenfolge (*res_name*) der Eigenschaft `WM_CLASS` des Hauptfensters der Anwendung übereinstimmen. Weitere Informationen hierzu finden Sie auf der Hilfeseite `xprop(1)`.
- CLIENT_GEOMETRY: Gibt die Größe (in Pixel) an, die für den vorderen Steuerbereich des Client-Fensters benötigt wird.
Die Hilfeseite `xwininfo(1)` beschreibt, wie die Größe eines Fensters in Pixel ermittelt werden kann.

2. Wählen Sie 'Workspace Manager neu starten' im Menü.

3. Starten Sie den Client an der Befehlszeile eines Terminal-Emulatorfensters.

Die folgende Steuerung zeigt eine Auslastungsanzeige von 30×20 Pixel an.

```
CONTROL LoadMeter
{
    TYPE      client

    CONTAINER_NAME    Top
    CONTAINER_TYPE    BOX

    CLIENT_NAME      xload

    CLIENT_GEOMETRY  30x20
}
```

Wird der Client zwischen den Sitzungen nicht gespeichert und wiederhergestellt, sollten Sie die Steuerung so konfigurieren, daß sie beim Klicken des Benutzers auf das Symbol den Client startet. Sie können beispielsweise die Steuerung `LoadMeter` konfigurieren, den Client `xload` zu starten, indem Sie folgende Zeile zu der Definition hinzufügen:

```
PUSH_ACTION    StartXload
```

und diese Aktion erstellen:

```
ACTION StartXload
{
    WINDOW_TYPE    NO_STDIO

    EXEC_STRING    /usr/contrib/bin/X11/xload
}
```

Eine Steuerung animieren

Sie können einer Steuerung eine Animationssequenz zuordnen, die beim Klicken auf die oder Ablegen auf der Komponente verwendet wird.

Damit eine Animation verwendet werden kann, muß die Steuerung folgende Bedingungen erfüllen:

- Sie muß den Typ `icon` haben.
- Das Feld `PUSH_ACTION` oder `DROP_ACTION` muß definiert sein.

1. Die Animationssequenz wird mit der Komponente `ANIMATION` angegeben:

```
ANIMATION Animationsname
{
    Symbolgrafik [Verzögerung
}
Symbolgrafik [Verzögerung
}
...
}
```

Mit *Verzögerung* wird die Zeitverzögerung zwischen den Symbolen in der Sequenz in Millisekunden angegeben.

2. Fügen Sie das Feld `PUSH_ANIMATION` und/oder `DROP_ANIMATION` der Steuerungsdefinition hinzu, um den Namen der `ANIMATION`-Sequenz anzugeben.

Im folgenden Beispiel wird eine Steuerung animiert, mit der die Anwendung `BestEditor` gestartet werden kann. Die Zeitverzögerung zwischen den Symbolen beträgt 300 Millisekunden. Es wird vorausgesetzt, daß die Symboldateien `frame1`, `frame2` usw. vorhanden sind.

```
CONTROL BestEditor
{
    ...
    PUSH_ANIMATION BestEdAnimation
    ...
}
```

```
ANIMATION BestEdAnimation
```

```
{  
    frame1  300  
    frame2  
    ...  
}
```

Elementhilfe für die Steuerungen im vorderen Steuerbereich bereitstellen

Elementhilfe kann auf zwei Arten für eine Steuerung bereitgestellt werden:

- Durch eine Zeichenfolge in der Steuerungsdefinition.

Die Zeichenfolge wird angezeigt, wenn der Benutzer für die Steuerung Elementhilfe anfordert. Sie darf keine Formatierungsinformationen (z. B. Kopfzeilen) oder Verweise enthalten.

Die Hilfe-Zeichenfolge muß folgendermaßen in der Definition der Steuerung angegeben werden:

```
HELP_STRING      Hilfe_Zeichenfolge
```

- Durch Angabe eines Hilfethemas in einem registrierten Hilfeband.

Eine Hilfethema enthält Informationen, die mit den Funktionen des Hilfesystems erstellt wurden. Um ein Hilfethema zu erstellen, benötigen Sie das Help Developers Kit des Desktops.

Der Hilfeband und die ID des Themas müssen in der Steuerungsdefinition angegeben werden:

```
HELP_VOLUME      Hilfeband
```

```
HELP_TOPIC       Thema_ID
```

Den Wechselbereich anpassen

Der Bereich zum Wechseln des Arbeitsbereichs kann folgendermaßen angepaßt werden:

- Anzahl der Arbeitsbereiche ändern.
- Layout des Wechselbereichs ändern.
- Steuerungen im Wechselbereich ändern.

▼ Den Standardwert für die Anzahl der Arbeitsbereiche ändern

- ◆ Ändern Sie die folgende Ressource des Workspace Managers:

```
Dtwm*workspaceCount:      n
```

Weitere Informationen finden Sie im Abschnitt „Anzahl von Arbeitsbereichen für das gesamte System ändern“ auf Seite 303.

▼ Die Zeilenanzahl des Wechselbereichs ändern

- ◆ Die Anzahl der Zeilen kann mit dem Feld `NUMBER_OF_ROWS` in der SWITCH-Definition geändert werden.

Im folgenden Beispiel wird eine SWITCH-Komponente mit drei Zeilen definiert:

```
SWITCH Switch
{
    CONTAINER_NAME    Feldname
    NUMBER_OF_ROWS    3
    ...
}
```

▼ Steuerungen im Wechselbereich ändern oder hinzufügen

1. Erstellen Sie zuerst eine Konfigurationsdatei für den vorderen Steuerbereich mit der Steuerungsdefinition.

- Legen Sie fest, daß sich die Steuerung im Wechselbereich befinden soll:

```
CONTAINER_NAME    Switch
```

```
CONTAINER_TYPE    SWITCH
```

- Geben Sie die Position im Wechselbereich an:

```
POSITION_HINTS    n
```

Das Argument *n* ist eine Ganzzahl und gibt die gewünschte Position an. Die Positionen sind durchgehend von links nach rechts und von oben nach unten numeriert (der Standardbereich mit zwei Zeilen hat die Positionen 1 bis 4).

2. Erstellen Sie ein Symbol für die Steuerung. Die empfohlene Größe ist 16 x 16 Pixel.

Im folgenden Beispiel wird eine Terminal-Steuerung in den Wechselbereich eingefügt.

```
CONTROL SwitchTerminal
{
TYPE                icon
CONTAINER_NAME      Switch
CONTAINER_TYPE      SWITCH
POSITION_HINTS      3
ICON                Fpterm
LABEL               Terminal
PUSH_ACTION         Dtterm
HELP_TOPIC          FPOnItemTerm
HELP_VOLUME         FPanel
}
```

Die Steuerung verwendet ein integriertes Symbol und das von der Steuerung 'Terminal' im zusätzlichen Steuerbereich 'Persönliche Anwendungen' verwendete Hilfethema.

Allgemeine Konfiguration des vorderen Steuerbereichs

Die PANEL-Syntax des vorderen Steuerbereichs ermöglicht folgendes:

- Ändern der Position des vorderen Steuerbereichs.
- Ändern des Aussehens des Fensters.
- Einstellen des allgemeinen Erscheinungsbilds und der allgemeinen Funktionsweise der Steuerungen.

Die Standarddefinition von PANEL befindet sich in der Datei
`/usr/dt/appconfig/types/Sprache/dtwm.fp`.

Weitere Informationen finden Sie auf der Hilfeseite `dtfpfile(4X)`.

Allgemeine Arbeitsschritte

1. Erstellen Sie eine neue Konfigurationsdatei für den vorderen Steuerbereich im Verzeichnis `/etc/dt/appconfig/types/Sprache` oder *Home-Verzeichnis* `/.dt/types`.
2. Kopieren Sie die Standarddefinition von PANEL aus der Datei `/usr/dt/appconfig/types/Sprache/dtwm.fp` in die neue Datei.
3. Nehmen Sie in der PANEL-Definition die gewünschten Änderungen vor.
Die neue Definition hat Vorrang vor der Standarddefinition von PANEL.

▼ Die Standardposition des vorderen Steuerbereichs ändern

- ◆ Mit dem Feld `PANEL_GEOMETRY` der PANEL-Definition kann die Position des Steuerbereichs festgelegt werden.

Der folgende Steuerbereich befindet sich in der rechten oberen Ecke.

```
PANEL SpecialFrontPanel
{
    PANEL_GEOMETRY    -1+1
```

```
...  
}
```

▼ Steuerungen im Hauptsteuerbereich beschriften

1. **Fügen Sie folgende Zeile der `PANEL`-Definition hinzu:**

```
DISPLAY_CONTROL_LABELS    True
```

2. **Fügen Sie jeder Steuerung ein `LABEL`-Feld hinzu.**

Steuerungsname wird verwendet, wenn `LABEL` nicht angegeben wird.

▼ Das Klickverhalten von Steuerungen ändern

- ♦ **Mit dem Feld `CONTROL_BEHAVIOR` in der `PANEL`-Definition wird festgelegt, wie der Benutzer die `PUSH_ACTION` einer Steuerung starten kann. Folgende Werte können angegeben werden:**

`single_click` - Die `PUSH_ACTION` wird durch einen einzelnen Klick gestartet.

`double_click` - - Die `PUSH_ACTION` wird durch einen Doppelklick gestartet.

▼ Einen vorderen Steuerbereich neu erstellen

Das Erstellen eines neuen vorderen Steuerbereichs ist von Vorteil, wenn umfangreiche Änderungen vorgenommen werden sollen.

Damit keine Namenskonflikte mit den integrierten Steuerbereichskomponenten auftreten können, sollten neue eindeutige Namen für `PANEL` und die anderen Container vergeben werden.

1. **Erstellen Sie die `PANEL`-Komponente für den neuen Steuerbereich. Verwenden Sie einen eindeutigen Namen:**

```
PANEL Vorderer_Steuerbereich
```

```
{  
...  
}
```

2. **Erstellen Sie die neuen Felder und Steuerungen. Verwenden Sie dabei die neuen Container-Namen.**

Sollen auch bereits vorhandene Komponenten verwendet werden, kopieren Sie deren Definitionen, und ändern Sie anschließend den Wert des Feldes CONTAINER_NAME.

3. Wählen Sie 'Workspace Manager neu starten' im Menü.

Beispiel zur Erstellung eines persönlichen vorderen Steuerbereichs mit drei Zeilen

Im folgenden Beispiel wird der vordere Standardbereich so geändert, daß seine Steuerungen in drei Zeilen angezeigt werden.

1. Kopieren Sie die Datei /usr/dt/appconfig/types/ *Sprache*/dtwm.fp in das Verzeichnis *Home-Verzeichnis*/.dt/types/MyFrontPanel.fp. Weisen Sie für die Datei Schreibrechte zu.

Diese Konfigurationsdatei wird für den neuen Steuerbereich benötigt.

2. Ändern Sie den Namen des vorderen Steuerbereichs:

```
PANEL NewFrontPanel
```

3. Ändern Sie den Namen des Feldes 'Top' und seines Containers:

```
BOX NewFrontPanelTop
{
    CONTAINER_NAME    NewFrontPanel
    POSITION_HINTS     first
    ...
}
```

4. Fügen Sie Feldefinitionen für die mittlere und die untere Zeile hinzu:

```
BOX NewFrontPanelMiddle
{
    CONTAINER_NAME    NewFrontPanel
    POSITION_HINTS     second
}
```


BOX NewFrontPanelBottom

```
{  
    CONTAINER_NAME      NewFrontPanel  
    POSITION_HINTS       second  
}
```

5. **Ändern Sie das Feld** CONTAINER_NAME **der folgenden Steuerungen in den Wert** NewFrontPanelTop:

- Clock
- Date
- Home
- TextEditor
- Mail

6. **Ändern Sie das Feld** CONTAINER_NAME **der folgenden Steuerungen in den Wert** NewFrontPanelBottom:

- Printer
- Style
- Applications
- Help
- Trash

7. **Ändern Sie das Feld** CONTAINER_NAME **der SWITCH-Komponente in den Wert** NewFrontPanelMiddle.

8. **Setzen Sie die Ressource:**

```
Dtwm*frontPanel*name: NewFrontPanel
```

9. **Wählen Sie 'Workspace Manager neu starten' im Menü.**

Workspace Manager konfigurieren

In diesem Kapitel erfahren Sie Details zur Konfiguration des Workspace Managers.

- „Konfigurationsdateien des Workspace Managers“ auf Seite 300
- „Arbeitsbereiche anpassen“ auf Seite 303
- „Menüs im Workspace Manager“ auf Seite 306
- „Maustastenbelegung ändern“ auf Seite 309
- „Tastaturbelegung ändern“ auf Seite 311
- „Wechseln zwischen Standard- und benutzerdefinierter Funktionsweise“ auf Seite 314

Der Workspace Manager ist der zum Desktop gehörige Fenstermanager. Ebenso wie andere Fenstermanager steuert und verwaltet dieses Modul die folgenden Elemente:

- Aussehen von Fensterrahmen-Komponenten
- Verhalten von Fenstern, einschließlich Fokusverhalten und Lage auf dem Bildschirm
- Tastatur- und Maustastenbelegung
- Aussehen von Fenstern bei Verkleinerung auf Symbolgröße
- Menüs 'Arbeitsbereich' und 'Fenster'

Zusätzlich ist der Workspace Manager für die folgenden Desktop-Elemente verantwortlich:

- *Arbeitsbereiche*. Der Workspace Manager steuert die Anzahl der Arbeitsbereiche und überwacht, welche Fenster in den einzelnen Arbeitsbereichen geöffnet sind.
- *Hintergrund-Ablagen in Arbeitsbereichen*. Der Benutzer wechselt Hintergrund-Ablagen zwar mit dem Style-Manager, aber deren Verwaltung ist eine Funktion des Workspace Managers.

- *Vorderer Steuerbereich*. Obwohl der vordere Steuerbereich eigene Konfigurationsdateien besitzt, wird er vom Workspace Manager aufgebaut und verwaltet.

Für einige dieser Elemente sind Änderungen mit dem Style-Manager möglich. Im Style Manager können häufig benötigte Modifikationen schnell und einfach ausgeführt werden. Andere Ressourcen wiederum müssen manuell eingestellt werden.

Der Workspace Manager hat den Befehlsnamen `dtwm` und basiert auf dem Window Manager von OSF/Motif.

- Referenzinformationen über den Workspace Manager finden Sie auf den Handbuchseiten `dtwm(1)` und `dtwmrc(4)`.
- Hinweise zur Konfiguration der Workspace Manager—Ressourcen gibt der Abschnitt .
- Die Konfigurationsdateien des vorderen Steuerbereichs sind beschrieben unter Kapitel 15.”

Details zur Einstellung von Ressourcen beschreibt das Thema .

Konfigurationsdateien des Workspace Managers

Informationen über die Fenstermenüs, Arbeitsbereichsmenüs, Tastaturbelegung und die Maustastenbelegung bezieht der Workspace Manager aus einer der Konfigurationsdateien.

Es handelt sich um folgende Dateien:

- Persönliche Datei: *HomeVerzeichnis* / `.dt/dtwmrc`
- Systemspezifische Anpassungsdatei: `/etc/dt/config/ sprache/sys.dtwmrc`
- Integrierte Datei: `/usr/dt/config/ sprache/sys.dtwmrc`

Die Suche nach einer Konfigurationsdatei erfolgt in der angegebenen Reihenfolge. Die zuerst gefundene wird verwendet.

Wenn ein Benutzer mehr als eine Sitzungssprache verwendet, kann eine persönliche, sprachabhängige Konfigurationsdatei namens

HomeVerzeichnis / `.dt/sprache /dtwmrc` angelegt werden. Diese hat dann Vorrang vor *HomeVerzeichnis* / `.dt/dtwmrc` .

▼ Persönliche Konfigurationsdateien anlegen und ändern

Die persönliche Konfigurationsdatei für den Workspace Manager heißt *HomeVerzeichnis/.dt/dtwmrc* . Wenn diese Datei vorhanden ist, wird sie verwendet.

1. Doppelklicken Sie in der Anwendungsgruppe 'Desktop_Tools' auf 'Dtwmrc bearbeiten'.

Wenn bereits eine persönliche Datei *dtwmrc* vorliegt, wird diese in den Editor geladen. Andernfalls wird *sys.dtwmrc* nach *HomeVerzeichnis/.dt/dtwmrc* kopiert und erst dann geladen.

2. Bearbeiten Sie die Datei.

3. Verlassen Sie den Editor.

Die Datei wird als Ihre persönliche Datei *dtwmrc* gespeichert, unabhängig von der ursprünglichen Quelle.

▼ Systemweite Konfigurationsdateien anlegen

Die systemweit gültige Konfigurationsdatei für den Workspace Manager hat den Namen */etc/dt/config/sprache/sys.dtwmrc*.

- ◆ **Kopieren Sie** */usr/dt/config/sprache/sys.dtwmrc* auf */etc/dt/config/sprache/sys.dtwmrc*.

Hinweis - Die Datei wird nicht verwendet, wenn *HomeVerzeichnis/.dt/dtwmrc* bereits vorhanden ist.

▼ Weitere Dateien einbinden

- ◆ **Richten Sie sich nach folgender Syntax:**

```
include
{
    Pfad
    Pfad
    ...
}
```

```
}
```

Beispielsweise bewirken die folgenden Zeilen die Einbindung der Datei

```
/users/ellen/mymenu:
```

```
include
```

```
{
```

```
    /users/ellen/mymenu
```

```
}
```

Include—Anweisungen ermöglichen die Erweiterung der Funktionalität, ohne die gesamte Konfigurationsdatei zu kopieren. Angenommen, ein Benutzer möchte ein neues Tastenkürzel definieren, ohne die Konfigurationsdatei komplett zu bearbeiten. Er würde dann lediglich eine Datei *HomeVerzeichnis/.dt/dtwmrc* mit folgendem Inhalt erstellen:

```
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"
```

```
    ...
```

```
}
```

▼ Workspace Manager neu starten

Änderungen an der Konfigurationsdatei wirken sich erst dann aus, wenn der Workspace Manager neu gestartet wird.

- ◆ Wählen Sie 'Workspace Manager neu starten' aus dem Menü 'Arbeitsbereich' (klicken Sie mit Maustaste 3 in den Arbeitsbereich).

Arbeitsbereiche anpassen

Die Anpassung von Arbeitsbereichen, etwa die Änderung von Name und Anzahl, läßt sich größtenteils in der Benutzeroberfläche des Desktop erledigen. Für die Definition systemweiter Standardwerte verfügt der Workspace Manager jedoch über zusätzliche Ressourcen.

▼ Anzahl von Arbeitsbereichen für das gesamte System ändern

Standardmäßig ist der Desktop für vier Arbeitsbereiche konfiguriert. Über ein Popup-Menü, das der Option 'Arbeitsbereiche' zugeordnet ist, kann der Benutzer die Anzahl erhöhen oder verringern.

In der Datei `/usr/dt/app-defaults/C/Dtwm` wird die Ressource `workspaceCount` auf die folgende Standardanzahl von Arbeitsbereichen gesetzt:

```
Dtwm*0*workspaceCount: 4
```

```
Dtwm*workspaceCount: 1
```

Für Bildschirm 0 sind mehrere Arbeitsbereiche vorgesehen, für alle anderen Bildschirme nur ein einziger.

Wenn diese Standardwerte für alle neuen Benutzer einer Workstation geändert werden sollen, können Sie dazu eine Datei `/etc/dt/config/C/sys.resources` anlegen bzw. ändern, falls schon vorhanden.

- ◆ Mit der Ressource `0*workspaceCount` definieren Sie einen systemweit gültigen Standardwert auf dem primären Bildschirm :

```
Dtwm*0*workspaceCount: anzahl
```

Im folgenden Beispiel wird die Anzahl der Arbeitsbereiche auf dem primären Bildschirm systemweit mit sechs festgelegt:

```
Dtwm*0*workspaceCount: 6
```

Informationen zur Einstellung von Ressourcen für den Workspace Manager finden Sie unter .

Beispielsweise setzt die folgende Ressource die Anzahl der Arbeitsbereiche auf sechs:

```
Dtwm*workspaceCount: 6
```

▼ Systemweit gültige Namen von Arbeitsbereichen

Intern werden die Arbeitsbereiche nach dem Schema `ws n` nummeriert, wobei `n` für eine fortlaufende, bei 0 beginnende Zahl steht. Beispielsweise haben die vier Standardarbeitsbereiche die Namen `ws0` bis `ws3` .

- ◆ **Der Name eines bestimmten Arbeitsbereichs wird mit der Ressource `title` geändert:**

```
Dtwm*wsn: name
```

Informationen zur Einstellung von Ressourcen für den Workspace Manager finden Sie unter .

Im folgenden Beispiel werden die vier Standardarbeitsbereiche umbenannt:

```
Dtwm*ws0*title: Anna
```

```
Dtwm*ws1*title: Kurt
```

```
Dtwm*ws2*title Julia
```

```
Dtwm*ws3*title Peter
```

▼ Zusätzliche Hintergrund-Ablagen erzeugen

1. **Erstellen Sie die Hintergrund-Ablagen als Bild. Dabei kann es sich um Bitmap- oder Pixmap—Dateien handeln.**
2. **Speichern Sie die Hintergrund-Ablagen in einem der folgenden Verzeichnisse. (Die Verzeichnisse müssen eventuell zuerst angelegt werden.)**

- Systemweite Hintergrund-Ablagen: `/etc/dt/backdrops`
- Persönliche Hintergrund-Ablagen: *HomeVerzeichnis* / `.dt/backdrops`

3. Wählen Sie 'Workspace Manager neu starten' aus dem Menü 'Arbeitsbereich'.

Die systemweit verwendeten sowie die persönlichen Hintergrund-Ablagen werden gemeinsam mit den integrierten in `/usr/dt/backdrops` gespeichert.

Sie können eine vorhandene integrierte Hintergrund-Ablage ersetzen, indem Sie eine persönliche bzw. systemweite Hintergrund-Ablage mit demselben Namen erstellen.

▼ Hintergrund-Ablagen durch Grafiken ersetzen

Die Hintergrund-Ablagen werden über das Root-Fenster des Bildschirms gelegt. Im Dialogfeld 'Hintergrund-Ablage' des Style Managers finden Sie eine Option 'NoBackdrop', mit der die Hintergrund-Ablage transparent gemacht werden kann.

Hinter allen Hintergrund-Ablagen des Arbeitsbereichs befindet sich immer nur ein einziges Root-Fenster. Somit bleibt eine Grafik, die über dem Root-Fenster plazierte wurde, über alle Arbeitsbereiche hinweg bestehen. Sie können zwar festlegen, welche Arbeitsbereiche das Root-Fenster mit einer Hintergrund-Ablage abdecken, aber es ist in allen Arbeitsbereichen dieselbe Grafik sichtbar, wenn die Option NoBackdrop aktiviert ist.

1. Erstellen Sie die Grafik.

Das Grafikformat muß sich nach dem vorgesehenen Hilfsprogramm richten, mit dem die Grafik im Root-Fenster angezeigt wird. Wenn Sie beispielsweise `xsetroot` einsetzen möchten, müssen Sie eine Bitmap-Datei erzeugen.

2. Falls noch nicht vorhanden, legen Sie eine ausführbare Datei

HomeVerzeichnis / `.dt/sessions/sessionetc` an.

Die Datei `sessionetc` wird bei jeder Anmeldung des Benutzers ausgeführt.

3. Fügen Sie den Befehl zur Anzeige der Grafik in die Datei `sessionetc` ein.

Der folgende Befehl bewirkt beispielsweise, daß das Root-Fenster mit dem angegebenen Bitmap in Kachelstruktur gefüllt wird:

```
xsetroot -bitmap /users/ellen/.dt/icons/root.bm
```

Menüs im Workspace Manager

Der Workspace Manager verfügt über drei Standardmenüs:

Arbeitsbereich—Dieses Menü wird auch als Root-Menü bezeichnet. Es wird immer dann angezeigt, wenn der Benutzer die Maustaste 3 drückt, während sich der Mauszeiger über der Hintergrund-Ablage befindet. Das Menü ist der Maustaste über eine Maustastenbelegung zugeordnet.

Fenster—Dieses Menü wird angezeigt, wenn der Benutzer die Maustaste 1 oder 3 drückt, während sich der Mauszeiger über der Menüschaltfläche 'Fenster' befindet (obere linke Ecke des Fensterrahmens). Das Menü ist über die Ressource `windowMenu` mit der Maustaste verknüpft.

Vorderer Steuerbereich—Dieses Menü wird angezeigt, wenn der Benutzer die Maustaste 1 oder 3 drückt, während sich der Mauszeiger über der Menüschaltfläche 'Fenster' des vorderen Steuerbereichs befindet.

Menüsyntax im Workspace Manager

In den Menüs des Workspace Managers gilt folgende Syntax:

Menu *MenuName*

```
{  
    auswahl11 [mnemonic] [  
    tastenkürzel] funktion [  
    argument]  
  
    auswahl2 [mnemonic] [  
    tastenkürzel] funktion [  
    argument]  
  
    ...  
}
```

Die einzelnen Parameter können folgendermaßen belegt werden:

auswahl—Der Text oder das Bitmap, das im Menü angezeigt wird. Wenn ein Text Leerzeichen enthält, muß er in Anführungszeichen gesetzt werden. Für Bitmaps gilt die Syntax `@/pfad`.

mnemonic—Ein einzelnes Zeichen, das als Tastenkürzel fungieren soll, wenn das Menü geöffnet ist. Die Festlegung erfolgt nach dem Schema `_ zeichen`.

tastenkürzel—Eine Tastenkombination, die unabhängig davon aktiv ist, ob das Menü geöffnet ist oder nicht. Zur Definition von Tastenkürzeln gilt die Syntax `modifikator<Key> tastenname`. Als Modifikator stehen `Ctrl`, `Shift`, `Alt` oder `Lock`

zur Auswahl. Eine Liste aller Tastennamen enthält die Datei `keysymdef.h` im Verzeichnis "X11 include".

funktion—Die Funktion, die ausgeführt werden soll, wenn ein bestimmter Menüeintrag gewählt wird. Eine Liste der verfügbaren Funktionen finden Sie auf der Handbuchseite `dtwmrc(4)`.

argument—Die Funktionsargumente. Details enthält die Handbuchseite `dtwmrc(4)`.

Das folgende Beispiel zeigt einen Menüeintrag 'Wiederherstellen', der ein Fenster wieder in die normale Darstellung zurückversetzt. Sobald das Menü geöffnet ist, genügt auch die Eingabe des Buchstabens 'W', um denselben Effekt zu erzielen. Alternativ dazu kann die Funktion mit der Tastenkombination Alt+F5 ausgelöst werden.

```
Wiederherstellen  _W  Alt<Key> F5  f.normalize
```

Hinweis - Eine vollständige Beschreibung der Menüsyntax für den Workspace Manager finden Sie auf der Handbuchseite `dtwmrc(4)`.

▼ Das bestehende Menü 'Arbeitsbereich' (Root-Menü) ändern

1. Öffnen Sie die Datei, die bearbeitet werden muß:

- Persönlich: *HomeVerzeichnis* / `.dt/dtwmrc`
- Systemweit: `/etc/dt/config/sprache/sys.dtwmrc`
Hinweise zur Erstellung dieser Dateien finden Sie unter „Konfigurationsdateien des Workspace Managers“ auf Seite 300.

2. Bearbeiten Sie die Beschreibung des Menüs 'Arbeitsbereich'.

Die voreingestellte Definition des Menüs 'Arbeitsbereich' hat den Namen `DtRootMenu`.

```
Menu DtRootMenu
{
    "Workspace Menu"          f.title
    "Shuffle Up"              f.circle_up
    "Shuffle Down"           f.circle_down
    ...
}
```

▼ Ein neues Menü 'Arbeitsbereich' (Root-Menü) erstellen

1. Öffnen Sie die Datei, die bearbeitet werden muß:

- Persönlich: *HomeVerzeichnis* / .dt/dtwmrc
 - Systemweit: /etc/dt/config/ *sprache*/sys.dtwmrc
- Hinweise zur Erstellung dieser Dateien finden Sie unter „Konfigurationsdateien des Workspace Managers“ auf Seite 300.

2. Stellen Sie das neue Menü zusammen:

```
Menu menu_name
{
    ...
}
```

Siehe auch „Menüsyntax im Workspace Manager “ auf Seite 306.

3. Definieren oder ändern Sie die Maustastenbelegung zur Anzeige des neuen Menüs.

Wenn das neue Menü ein vorhandenes ersetzt, ändern Sie entsprechend die Maustastenzuordnung, über die das Menü 'Arbeitsbereich' angezeigt wird.

```
<Btn3Down> root f.menu menu_name
```

Wenn es sich um ein zusätzliches Menü handelt, definieren Sie eine neue Maustastenbelegung. Im folgenden Beispiel wird das Menü geöffnet, wenn mit der Maustaste 3 in die Hintergrund-Ablage geklickt und gleichzeitig die Shift-Taste gedrückt wird:

```
Shift<Btn3Down> root f.menu menu_name
```

4. Wählen Sie 'Workspace Manager neu starten' aus dem Menü 'Arbeitsbereich'.

▼ Ein neues Menü 'Fenster' erstellen

Hinweis - Das Menü 'Fenster' ist in den Workspace Manager integriert und sollte nicht wesentlich verändert werden, damit das Fensterverhalten in den einzelnen Anwendungen möglichst konsistent bleibt.

1. Öffnen Sie die Datei, die bearbeitet werden muß:

- **Persönlich:** *HomeVerzeichnis* / *.dt/dtwmrc*
- **Systemweit:** */etc/dt/config/ sprache/sys.dtwmrc*
Hinweise zur Erstellung dieser Dateien finden Sie unter „Konfigurationsdateien des Workspace Managers“ auf Seite 300.

2. Stellen Sie das neue Menü zusammen:

```
Menu menu_name
{
    ...
}
```

3. Benennen Sie das neue Menü über die Ressource `windowMenu`:

```
Dtwm>windowMenu: menu_name
```

4. Wählen Sie 'Workspace Manager neu starten' aus dem Menü 'Arbeitsbereich'.

Maustastenbelegung ändern

Bei einer *Maustastenbelegung* wird die Betätigung einer Maustaste und optional auch einer Modifikatortaste auf der Tastatur einer Funktion des Workspace Managers zugeordnet. Einmal definierte Maustastenbelegungen gelten für alle Arbeitsbereiche.

Die voreingestellten Maustastenbelegungen für den Desktop sind in der Konfigurationsdatei des Workspace Managers in einem Set namens `DtButtonBindings` zusammengefaßt:

```
Buttons DtButtonBindings
{
    ...
}
```

Syntax für die Maustastenbelegung

Die Syntax für Maustastenbelegungen hat folgende Form:

```
Buttons BelegungsSetName
```

```

{
    [Modifikator]<TastenameMouseaktion
> Kontext Funktion
[argument]

    [Modifikator]<TastenameMausaktion
> Kontext Funktion
[argument]

```

Die einzelnen Parameter können folgendermaßen belegt werden:

Tastename—Btn1—Linke Maustaste Btn2—Mittlere Maustaste (3-Tasten-Maus) bzw. beide Maustasten (2-Tasten-Maus) Btn3—Rechte Maustaste Btn4—Tasten 1 und 2 gleichzeitig auf 3-Tasten-Maus Btn5—Tasten 2 und 3 gleichzeitig auf 3-Tasten-Maus

Modifikator—Ctrl, Shift, Alt, Lock

Mausaktion—Down—Maustaste gedrückt halten; Up—Maustaste loslassen;
Click—Klick mit einer Maustaste; Click2—Doppelklick mit Maustaste;
Drag—Maus bewegen, während eine Maustaste gedrückt ist.

Kontext—Definiert den Bereich, in dem sich der Mauszeiger befinden muß, damit die Funktionszuordnung gültig wird. Bei Bedarf trennen Sie mehrere Kontexte mit dem Zeichen " |".

root—Das Arbeitsbereichsfenster; window—Client—Fenster oder der Fensterrahmen; frame—Fensterrahmen (ohne Inhalt); icon—Symbol;
title—Titelleiste; app—Client—Fenster (ohne Rahmen).

Funktion—Eine der Funktionen des Workspace Managers. Eine Liste der verfügbaren Funktionen finden Sie auf der Handbuchseite dtwmrc(4).

Argument—Argumente, die an eine Funktion des Workspace Managers übergeben werden müssen. Details finden Sie auf der Handbuchseite dtwmrc(4).

Die folgende Anweisungszeile bewirkt, daß das in DtRootMenu beschriebene Menü angezeigt wird, wenn Maustaste 3 gedrückt wird, während sich der Mauszeiger im Arbeitsbereichsfenster befindet (aber nicht in Client-Fenstern).

```
<Btn3Down> root f.menu DtRootMenu
```

Hinweis - Eine ausführliche Beschreibung der Syntax enthält die Handbuchseite dtwmrc(4).

▼ Neue Maustastenbelegungen hinzufügen

1. Öffnen Sie die Datei, die bearbeitet werden muß:

- Persönlich: *HomeVerzeichnis* / .dt/dtwmrc
- Systemweit: /etc/dt/config/ *sprache*/sys.dtwmrc

Hinweise zur Erstellung dieser Dateien finden Sie unter „Konfigurationsdateien des Workspace Managers“ auf Seite 300.

2. **Fügen Sie die Maustastenbelegung in die Definition** `DtButtonBindings` **ein.**
Ordnen Sie dem Klicken und dem Drücken einer bestimmten Maustaste keine verschiedenen Funktionen zu. Ebenso gilt, daß eine Maustaste in einem bestimmten Kontext immer nur eine Funktion haben sollte.
3. **Wählen Sie 'Workspace Manager neu starten' aus dem Menü 'Arbeitsbereich'.**

▼ Neue Sets für Maustastenbelegungen definieren

1. **Öffnen Sie die Datei, die bearbeitet werden muß:**
 - Persönlich: `HomeVerzeichnis / .dt/dtwmrc`
 - Systemweit: `/etc/dt/config/ sprache/sys.dtwmrc`
Hinweise zur Erstellung dieser Dateien finden Sie unter „Konfigurationsdateien des Workspace Managers“ auf Seite 300.
2. **Hinweise zur Definition eines neuen Sets für Maustastenbelegungen finden Sie unter „Syntax für die Maustastenbelegung “ auf Seite 309.**
3. **Geben Sie der Ressource** `buttonBindings` **einen neuen Namen:**

```
Dtwm*buttonBindings: BelegungsSetName
```

4. **Wählen Sie 'Workspace Manager neu starten' aus dem Menü 'Arbeitsbereich'.**

Hinweis - Die neue Belegungsdefinition ersetzt die bisherige. Wenn bestehende Belegungen erhalten bleiben sollen, kopieren Sie diese aus `DtButtonBindings` in das neue Set.

Tastaturbelegung ändern

Bei einer *Tastaturbelegung*, manchmal auch als Definition von *Tastenkürzeln* bezeichnet, wird eine Kombination von Tasten einer Funktion des Workspace Managers zugeordnet. Einmal definierte Tastaturbelegungen gelten für alle Arbeitsbereiche.

Hinweis - Vermeiden Sie es, häufig benutzten Tastenkombinationen eine neue Bedeutung zuzuordnen. Beispielsweise ist es wenig sinnvoll, die Kombination Shift+A mit einer Menüfunktion zu belegen, weil dann der Großbuchstabe A nicht mehr verfügbar ist.

Standard-Tastaturbelegungen für den Desktop

Die voreingestellten Tastaturbelegungen sind in der Konfigurationsdatei des Workspace Managers in einem Set namens `DtKeyBindings` definiert:

```
Keys DtKeyBindings
```

```
{  
  ...  
}
```

Syntax für Tastaturbelegungen

Die Syntax für Tastaturbelegungen hat folgende Form:

```
Keys BelegungsSetName
```

```
{  
  [Modifikatoren] <Key> Tastename Kontext Funktion  
  
  [Argument]  
  [Modifikatoren] <Key> Tastename Kontext Function  
  
  [argument]  
  ...  
}
```

Die einzelnen Parameter können folgendermaßen belegt werden

Modifikatoren—`Ctrl`, `Shift`, `Alt` und `Lock`. Es sind mehrere Modifikatoren erlaubt. Als Trenner dient das Leerzeichen.

tastename—Die Taste, mit der die Funktion ausgelöst werden soll. Bei allen Tasten, die mit einem Buchstaben beschriftet sind, ist diese Beschriftung der gültige **tastename**. Beispielsweise hat die Taste 'A' den Tastennamen "a", und die Taste "2" heißt "2". Die Tabulatortaste hat den Kurznamen "Tab". Die Funktionstasten heißen ebenfalls wie ihre Beschriftung auf der Tastatur (z.B. "F3").

Bei anderen Tasten wird die Bezeichnung in gesprochener Langform verwendet, z.B. `plus` für die Taste "+". Die Datei `keysymdef.h` enthält alle Tastendefinitionen. Sie ist in einem systemabhängigen Verzeichnis abgelegt.

kontext—Das Element, das den Eingabefokus besitzt, damit die zugehörige Aktion ausgeführt werden kann. Es lassen sich mehrere Kontexte verketteten, falls nötig. Als Trennzeichen dient dann "|".

`root`—Hintergrund-Ablage des Arbeitsbereichs; `window`—Client—Fenster;
`icon`—Symbol

funktion—Eine Funktion des Fenstermanagers. Die Handbuchseite `dtwmrc(4)` enthält die Liste gültiger Funktionen.

argument—Sämtliche Argumente, die von der jeweiligen Funktions des Fenstermanagers benötigt werden. Details finden Sie auf der Handbuchseite `dtwmrc(4)`.

Im folgenden Beispiel wird eine Tastaturbelegung definiert, um den Eingabefokus mit der Kombination `Alt+F6` an das nächste transiente Fenster zu übergeben.

```
Alt<Key>F6      window      f.next_key      transient
```

Hinweis - Vollständige Informationen zur Syntax von Tastaturbelegungen finden Sie auf der Handbuchseite `dtwmrc(4)`.

▼ Benutzerdefinierte Sets für die Tastaturbelegung erstellen

1. Öffnen Sie die Datei, die bearbeitet werden muß:

- Persönlich: *HomeVerzeichnis* / `.dt/dtwmrc`
- Systemweit: `/etc/dt/config/ sprache/sys.dtwmrc`
Hinweise zur Erstellung dieser Dateien finden Sie unter „Konfigurationsdateien des Workspace Managers“ auf Seite 300.

2. Sets mit Tastaturbelegungen sollten Sie immer eindeutig benennen (*BelegungsSetName*). Als Muster kann das vordefinierte `Desktop—Set DtKeyBindings` dienen.

3. Setzen Sie die Ressource `keyBindings` auf den Namen des neuen Sets:

```
Dtwm*keyBindings: KeyBindingSetName
```

4. Wählen Sie 'Workspace Manager neu starten' aus dem Menü 'Arbeitsbereich'.

Hinweis - Die neuen Tastaturbelegungen ersetzen Sie die bisherigen. Wenn vorhandene Zuordnungen bestehen bleiben sollen, kopieren Sie diese aus `DtKeyBindings` in Ihr neues Set.

Wechseln zwischen Standard- und benutzerdefinierter Funktionsweise

So schalten Sie zwischen dem standardmäßigen OSF/Motif 1.2- und dem CDE-Fensterverhalten um:

1. Drücken Sie Alt+Shift+Ctrl+!

2. Klicken Sie im Dialogfeld auf 'OK'.

Beim Wechsel zur Standardfunktionsweise werden der vordere Steuerbereich und sämtliche benutzerdefinierten Tastatur- und Maustastenbelegungen entfernt.

Anwendungsressourcen, Schriftarten und Farben verwalten

Mit Hilfe des Style-Managers oder durch Anpassung zusätzlicher Schriftarten und Farbressourcen kann aus einer großen Anzahl von Farben und Schriftarten für die Anzeige ausgewählt werden. Dieses Kapitel behandelt die Anpassung von Schriftarten und Farbressourcen.

Dieses Kapitel beschreibt auch, wie Umsetzungen für den `DtEditor` Fensterobjektanwendungen wie zum Beispiel den Texteditor des Desktop (`dtpad`) und Mailer (`dtmail`) angegeben werden. Das Kapitel beschreibt ebenfalls Alternativen für Direktaufrufe des `DtEditor` Fensterobjektanwendungs-menü die mit diesen Umsetzungen in Konflikt treten.

-
- „UNIX-Belegungen definieren“ auf Seite 316
- „Schriftarten verwalten“ auf Seite 322
- „Farben verwalten“ auf Seite 327
- „Schattenstärken für Anwendungsfenster einstellen“ auf Seite 336

Anwendungsressourcen einstellen

Ressourcen werden von Anwendungen dazu verwendet, um gewisse Aspekte des Erscheinungsbilds und des Verhaltens einzustellen. Beispielsweise ermöglichen Ressourcen des Style-Managers (`dtstyle`), dem System anzugeben, wo es nach Dateien mit Informationen über Farbpaletten suchen soll:

```
dtstyle*paletteDirectories: /usr/dt/palettes/C \
Home-Verzeichnis/.dt/palettes
```

Standarddateien für die Desktop-Anwendungen befinden sich im Verzeichnis:
`/usr/dt/app-defaults/sprache`

▼ Systemweite Ressourcen einstellen

- ◆ **Die Ressourcen der Datei `/etc/dt/config/sprache/sys.resources` hinzufügen. (Möglicherweise muß die Datei erstellt werden.)**

Wenn zum Beispiel in `/etc/dt/config/C/sys.resources` folgendes angegeben wird:

```
AnApplication*resource: wert
```

dann wird die Ressource `AnApplication*resource` bei der nächsten Anmeldung in die Eigenschaft `RESOURCE_MANAGER` von jedem Benutzer gestellt.

▼ Persönliche Ressourcen einstellen

1. **Die Ressourcen der Datei `Home-Verzeichnis/.Xdefaults` hinzufügen.**
2. **Ressourcen erneut laden in der Anwendungsgruppe 'Desktop_Werkzeuge' doppelklicken.**

Wie der Desktop Ressourcen lädt

Beim Neustart einer Session werden die Ressourcen von der Sessionmanager geladen. Mehr Informationen darüber, wie die Sessionmanager die Ressourcen in `RESOURCE_MANAGER` lädt, können dem Abschnitt „Sitzungsressourcen laden“ auf Seite 29 entnommen werden.

UNIX-Belegungen definieren

UNIX-Belegungen sind standardmäßig nicht aktiviert.

▼ EMACS-Umsetzungen angeben

Die folgende Prozedur gibt an:

- EMACS-Umsetzungen für DtEditor-Fensterobjektanwendungen, wie zum Beispiel den Texteditor des Desktop (dtpad) und Mailer (dtmail).
 - Alternativen für DtEditor-Direktaufrufe des Fensterobjektanwendungsmenüs, die mit diesen Umsetzungen in Konflikt treten.
1. **Folgende Zeile der Datei *Home-Verzeichnis* / .Xdefaults hinzufügen:**

```
#include "/usr/dt/app-defaults/sprache /UNIXbindings"
```

wobei *sprache* der Wert der LANG-Umgebungsvariablen ist.
 2. **Session neu starten.**

▼ EMACS-Umsetzungen ändern

1. **Die Inhalte dieser Datei einfügen:**

```
/usr/dt/app-defaults/ sprache /UNIXbindings
```

in ***Home-Verzeichnis* / .Xdefaults.**
2. **Die Belegungen in der Datei .Xdefaults bearbeiten.**
3. **Nach Beendigung die Session neu starten.**

UNIX-Belegungen über die UNIX-Belegungsdatei

Die Datei `/usr/dt/app-defaults/sprache /UNIXbindings` enthält die in den folgenden Tabellen beschriebenen Belegungen.

Hinweis - Die Löschtaste löscht das vorhergehende Zeichen, wenn die UNIX-Belegungen aktiviert sind, und die Tastenkombination Umschalttaste+Löschtaste löscht das nächste Zeichen.

Tabelle 17-1 zeigt eine Liste von neuen dtpad Belegungen für Menüdirektaufrufe und Direktaufruftext, die mit den UNIX-Belegungen in Konflikt treten.

TABELLE 17-1 dtpad Neue Belegungen

Menüdirektaufrufe und Direktaufruftext	Neue Belegung
Dtpad*fileMenu.print.acceleratorText:	
Dtpad*fileMenu.print.accelerator:	
Dtpad*editMenu.undo.acceleratorText:	Steuertaste+_
Dtpad*editMenu.undo.accelerator:	Steuertaste<Taste>_
Dtpad*editMenu.paste.acceleratorText:	Umschalttaste+Einfügetaste
Dtpad*editMenu.paste.accelerator:	Umschalttaste<Taste>osfInsert
Dtpad*editMenu.selectAll.acceleratorText:	Steuertaste+/ /
Dtpad*editMenu.selectAll.accelerator:	Steuertaste<Taste>/

Tabelle 17-2 zeigt eine Liste von neuen dtmail Belegungen beim Erstellen einer Nachricht für Menüdirektbeschleuniger und Direktaufruftext, die mit den UNIX-Belegungen in Konflikt treten.

TABELLE 17-2 dtmail Neue Belegungen beim Erstellen einer Nachricht

Menüdirektaufrufe und Direktaufruftext	Neue Belegung
Dtmail*ComposeDialog*menubar*Edit.Undo.acceleratorText:	Steuertaste+_
Dtmail*ComposeDialog*menubar*Edit.Undo.accelerator:	Steuertaste<Taste>_
Dtmail*ComposeDialog*menubar*Edit.Paste.acceleratorText:	Umschalttaste+Einfügetaste
Dtmail*ComposeDialog*menubar*Edit.Paste.accelerator:	Umschalttaste<Taste>osfInsert

TABELLE 17-2 dtmail Neue Belegungen beim Erstellen einer Nachricht (fortgesetzt)

Menüdirektaufrufe und Direktaufdrucktext	Neue Belegung
Dtmail*ComposeDialog*menubar*Edit.Find/Change.acceleratorText:	Steuertaste+S
Dtmail*ComposeDialog*menubar*Edit.Find/Change.accelerator:	Steuertaste<Taste>s

- Die folgenden Umsetzungen bieten (GNU-Darstellung) EMACS-Steuerung und Metatastenbelegungen, sowie einige zusätzliche Belegungen. Wenn zulässig, ermöglichen sie auch, daß die Umschalttaste in Kombination mit der normalen Belegung benutzt werden kann, um die Richtung des Vorgangs umzukehren. Beispielsweise bewegt dann die Tastenkombination Steuertaste+Umschalttaste+F den Cursor ein Zeichen zurück, da Steuertaste+Umschalttaste+F den Cursor normalerweise ein Zeichen vorwärts bewegen.

Die zusätzlichen Belegungen sind:

Strg+Komma - vorheriges Wort

Strg+Umschalt+Komma - nächstes Wort

Strg+Punkt - nächstes Wort

Strg+Umschalt+Punkt - vorheriges Wort

Strg+Eingabe - Dateiene

Strg+Umschalt+Eingabe - Dateianfang

GNU EMACS belegt die Löschtaste eher mit Zeichen, die zurück löschen (), als mit Zeichen, die vorwärts löschen (). Die Tastenkombination Meta+F ist üblicherweise das mnemonische Zeichen für das Dateimenü, so daß die Belegung mit nächstes Wort ignoriert wird. Für das nächste Wort sollte eine der anderen Belegungen verwendet werden (zum Beispiel Steuertaste+Punkt).

Tabelle 17-3 zeigt eine Liste der DtEditor.text Umsetzungen

TABELLE 17-3 DtEditor.text Umsetzungen

Kombinationstaste	Taste	Aktionsroutine
c ~s	<Taste>a:	Zeilenanfang()\n\
c s	<Taste>a:	Zeilenende()\n\
c ~s	<Taste>b:	letztes Zeichen()\n\

TABELLE P-3 DtEditor.text Umsetzungen (fortgesetzt)

Kombinationstaste Taste Aktionsroutine		
c s	<Taste>b:	nächstes Zeichen()\n\
c ~s	<Taste>b:	letztes Zeichen()\n\
c s	<Taste>b:	letztes Wort()\n\
m ~s	<Taste>b:	letztes Wort()\n\
m s	<Taste>b:	nächstes Wort()\n\
c ~s	<Taste>d:	nächstes Zeichen löschen()\n\
c s	<Taste>d:	letztes Zeichen löschen()\n\
m ~s	<Taste>d:	nächstes Wort löschen()\n\
m s	<Taste>d:	letztes Wort löschen()\n\
c ~s	<Taste>e:	Zeilenende()\n\
c s	<Taste>e:	Zeilenanfang()\n\
c ~s	<Taste>f:	nächstes Zeichen()\n\
c s	<Taste>f:	letztes Zeichen()\n\
m ~s	<Taste>f:	nächstes Wort()\n\
m s	<Taste>f:	letztes Wort()\n\
c	<Taste>j:	neue Zeile und einrücken()\n\
c ~s	<Taste>k:	bis Zeilenende löschen()\n\
c s	<Taste>k:	bis Zeilenanfang löschen()\n\
c	<Taste>l:	Anzeige neu aufbauen()\n\
c	<Taste>m:	neue Zeile()\n\
c s	<Taste>n:	Vorgang rückwärts()\n\
c ~s	<Taste>n:	Vorgang vorwärts()\n\
c	<Taste>o:	neue Zeile und sichern()\n\

TABELLE P-3 DtEditor.text Umsetzungen (fortgesetzt)

Kombinationstaste	Taste	Aktionsroutine
c ~s	<Taste>p:	Vorgang rückwärts()\n\
c s	<Taste>p:	Vorgang vorwärts()\n\
c ~s	<Taste>u:	bis Zeilenanfang löschen()\n\
c s	<Taste>u:	bis Zeilenende löschen()\n\
c ~s	<Taste>v:	nächste Seite()\n\
c s	<Taste>v:	letzte Seite()\n\
m ~s	<Taste>v:	letzte Seite()\n\
m s	<Taste>v:	nächste Seite()\n\
c	<Taste>w:	Auswahl löschen()\n\
c ~s	<Taste>y:	löschen rückgängig()\n\
m	<Taste>]:	nächster Absatz()\n\
m	<Taste>[:	letzter Absatz()\n\
c ~s	<Taste>comma:	letztes Wort()\n\
c s	<Taste>comma:	nächstes Wort()\n\
m	<Taste>\\<:	Dateianfang()\n\
c ~s	<Taste>period:	nächstes Wort()\n\
c s	<Taste>period:	letztes Wort()\n\
m	<Taste>\\>:	Dateiende()\n\
c ~s	<Taste>Return:	Dateiende()\n\
c s	<Taste>Return:	Dateianfang()\n\
~c ~s ~m ~a	<Taste>osfDelete:	letztes Zeichen löschen()\n\
~c s ~m ~a	<Taste>osfDelete:	nächstes Zeichen löschen()

Schriftarten verwalten

Mit dem Dialogfenster 'Schrift' des Style-Managers kann die Schriftgruppe und -größe für alle Anwendungen ausgewählt werden. Es können auch Schriftarten auf der Befehlszeile angegeben werden und Ressourcen verwendet werden, um:

- Schriftartressourcen für einzelne Anwendungen einzustellen
- verschiedene Schriftarten, die vom Dialogfenster Schriftart benutzt werden sollen, zuordnen.

Eine *Schriftart* ist eine Darstellungsart, bei der Textzeichen gedruckt oder angezeigt werden. Das Desktop enthält eine Vielfalt von Schriften in verschiedenen Arten und Größen.

Eine *Bitmap-Schriftart* besteht aus einer Matrix aus Punkten. Standardmäßig konfiguriert der Style-Manager nur Bitmap-Schriftarten. Die Schriftart ist vollständig in einer Datei enthalten. Für die Gesamtheit von Schriftgrößen, Zeichenneigungen und Schriftstärken werden viele Dateien benötigt.

Schriftarten sind angegeben als Werte von Ressourcen und als Befehlsparameter. Mit dem XLFD-Namen (X Logical Font Description) wird eine gewünschte Schriftart angefordert. Das System findet die Schriftart, die am besten zu der gegebenen Beschreibung passt.

Schriftartressourcen des Desktop einstellen

Das Dialogfenster 'Schrift' des Style-Managers erlaubt, Schriftarten (bis zu sieben Größen) für Texteingaben, Bezeichnungen usw. einzustellen. Außerdem können Sie Schriftartgruppen hinzufügen oder löschen.

Vom Schriftart-Dialogfenster eingestellte Ressourcen

Wenn eine Schriftart ausgewählt wird, werden die folgenden Ressourcen in die Eigenschaft `RESOURCE_MANAGER` geschrieben:

- `SystemFont` wird verwendet für Systembereiche, wie zum Beispiel Menüleisten, Menüteilfenster, Tasten, Umschalttasten und Bezeichnungen. Die folgende Ressource wird von `SystemFont` eingestellt:
 - *`FontList` - Angezeigt in Systembereichen von Desktop-Clients und anderen erstellten Clients, die die OSF/Motif-Hilfsprogramme benutzen.
- `UserFont` wird zur Texteingabe in Fenster verwendet. Folgende Ressourcen werden von `UserFont` eingestellt:
 - *`Font` - unterstützt frühere Versionen von X-Anwendungen

*FontSet - Haupteinstellung

*XmText*FontList - angezeigt in Texteingabefenstern

*XmTextField*FontList - angezeigt in Texteingabefenstern

Vom Dialogfenster 'Schrift' verwendete Schriftarten

Die Fonts, die für jede Auswahl im Dialogfenster 'Schrift' benutzt werden, sind in der Ressourcendatei `/usr/dt/app-defaults/Dtstyle` angegeben. Bis zu sieben Größen können angegeben werden.

NumFonts - Anzahl der Schriftartgrößen im Dialogfenster

SystemFont[1-7] - bis zu sieben Ressourcen, die eine bestimmte Schriftart einer Dialogfensterauswahl für SystemFont zuordnen

UserFont[1-7] - bis zu sieben Ressourcen, die eine bestimmte Schriftart einer Dialogfensterauswahl für UserFont zuordnen.

Hinweis - Die Standardschriftarten für diese Ressourcen wurden für Lesbarkeit auf verschiedenen Bildschirmen ausgewählt. Wenn für eine Anwendung eine besondere Schriftart benötigt wird, sollte die Schriftart mit einer Schriftartressource der Anwendung eingestellt werden und nicht mit einer Änderung der Schriftarten des Desktop.

Mehr Informationen zu Anwendungsschriftarten können den Handseiten `DtStdAppFontNames (5)` und `DtStdInterfaceFontNames (5)` entnommen werden.

▼ Verfügbare Schriftarten auflisten

1. Folgendes eingeben:

```
xlsfonts [-optionen ][-fn muster]
```

Eine Liste mit XLFD-Namen und Alias-Namen von Schriftarten, die auf dem System zur Verfügung stehen, wird angezeigt. Bitmap-Schriftarten zeigen Werte in allen vierzehn XLFD-Feldern. Skalierbare Schriften zeigen Nullen in den Positionen *Pixelgröße*, *Punktgröße*, *AuflösungX*, und *AuflösungY*.

2. Um nach bestimmten Schriftarten zu suchen, sucht

`xlsfonts` nach übereinstimmenden Mustern. **Mit Platzhalterzeichen wird der Teil des Musters ersetzt, der nicht übereinstimmen soll.**

3. Wenn `xlsfonts` keine Schriftartennamen zeigt, die mit `dt` beginnen, enthält der Schriftartenpfad keine Desktop-Schriftarten. Folgenden Befehl eingeben, um die Desktop-Schriftarten in die verfügbaren Schriftarten miteinzubeziehen:

```
xset +fp verzeichnisname
```

wobei *verzeichnisname* das Verzeichnis mit den Desktopschriftarten ist. Die Standardposition beim Neustart einer Session ist
`/usr/dt/config/xfonts/sprache`.

Zusätzliche Informationen:

- Die `xset` und `xlsfonts` Handseiten enthalten eine Liste der verfügbaren Optionen.
- *Das X-Fenster-System benutzen* erklärt Alias-Namen von Schriftarten und den `xset` Client.

▼ Schriftarten auf der Befehlszeile angeben

- ◆ **Mit der `-xrm` Befehlszeilenoption wird eine Schriftartenressource für einen bestimmten Client angegeben. Beispiel:**

```
anwendungsname -xrm "*bitstream-charter-medium-r-normal-8-88-75-75-p-45-iso8859-1"
```

X Logical Font Description (XLFD)

Eine Schriftart wird durch eine Liste mit vierzehn verschiedenen Kenndaten, die durch Bindestriche (-) getrennt sind, angegeben. Dies wird X Logical Font Description (XLFD) genannt. In einigen Fällen kann eine Eigenschaft in der Liste durch ein * Platzhalterzeichen ersetzt werden, und ein Zeichen in einer Eigenschaft kann durch ein ? Platzhalterzeichen ersetzt werden. Tabelle 15-4 enthält eine Liste von Spezifikationen der Eigenschaftszeichenfolgen für Schriftarten.

Die Form der Spezifikation der Eigenschaftszeichenfolge lautet:

*"Entwickler-Markename- Schriftstärke-Zeichenneigung- Zeichenbreite-
Zusätzliche Informationen
-Pixelgröße-Punktgröße -AuflösungX-AuflösungY -Zeichenabstand-Durchschnittsbreite -
Registrierungsberechtigter- Zeichensatz"*

TABELLE 17-4 Spezifikationen der Eigenschaftszeichenfolgen für Schriftarten

Eigenschafts- zeichenfolge	Definition
<i>Entwickler</i>	Zeichenfolge, die den Entwickler der Schriftart identifiziert
<i>Markenname</i>	Zeichenfolge, die den Markennamen der Schriftart identifiziert
<i>Schriftstärke</i>	Zeichenfolge, die die relative Schriftstärke der Schriftart angibt, z.B. fett
<i>Zeichenneigung</i>	Code, der die Richtung der Zeichenneigung beschreibt R (Roman–keine Neigung) I (Italic–Neigung rechts) O (Oblique–Neigung rechts) RI (Reverse Italic–Neigung links) RO (Reverse Oblique–Neigung links)
<i>Zeichenbreite</i>	Zeichenfolge, die die Breite angibt, z.B. komprimiert oder erweitert
<i>Zusätzliche Informationen</i>	Zeichenfolge mit zusätzlichen Informationen, die gebraucht werden, um die Schriftart eindeutig zu identifizieren
<i>Pixelgröße</i>	Eine ganze Zahl, die die Größe eines Quadrats in Pixel angibt
<i>Punktgröße</i>	Eine ganze Zahl, die die Größe eines Quadrats in Dezimalzeichen angibt
<i>AuflösungX</i>	Eine ganze Zahl, die die horizontale Auflösung in Pixel angibt
<i>AuflösungY</i>	Eine ganze Zahl, die die vertikale Auflösung in Pixel angibt
<i>Zeichenabstand</i>	Ein Code, der den Abstand zwischen den Zeichen angibt M (Monospace–feste Zeichenbreite) P (Proportionaler Abstand–variable Zeichenbreite) C (Zeichenrahmen)
<i>Durchschnittsbreite</i>	Eine ganze Zahl, die die Durchschnittsbreite in 1/10 Pixel angibt

TABELLE P-4 Spezifikationen der Eigenschaftszeichenfolgen für Schriftarten (fortgesetzt)

Eigenschafts- zeichenfolge	Definition
<i>Registrierungs- erechtigter</i>	Zeichenfolge, identifiziert den Registrierungsberechtigten, der den verschlüsselten Code registriert hat
<i>Zeichensatz</i>	Zeichenfolge, identifiziert den Zeichensatz im angegebenen Register

Beispiel

Der folgende XLFD-Name beschreibt die Schriftart charter von Bitstream, die die ISO8859-1 Standardverschlüsselung unterstützt:

```
-bitstream-charter-medium-r-normal--8-80-75-75-p-45-iso8859-1
```

Die Schriftart hat eine mittlere Schriftstärke, keine spezielle Neigung und eine normale Breite. Sie ist eine proportionale Schriftart mit einem quadratischen Rahmen von 8 Pixel oder 8,0 Punkten. Die horizontale und vertikale Auflösung ist bei beiden 75 Pixel. Die Durchschnittsbreite eines Zeichens beträgt $45 \frac{1}{10}$ Pixel oder 4,5 Pixel.

Teile des Strings können durch Platzhalterzeichen ersetzt werden. Das System verwendet die erste Schriftart, die dem angegebenen Muster entspricht.

Wenn nur eine acht-Pixel Schriftart charter benötigt wird, folgendes eingeben:

```
*-charter-*-*-*-8-*
```

Ausgewählte Schriftartengruppenattribute anzeigen

Mit der Schaltfläche "Attribute" im Dialogfenster "Schrift" im Style-Manager können Sie die folgenden Attribute von Schriftartengruppen anzeigen:

- Schriftartengruppe
- Größe
- Alias
- Alias XLFD
- Alias-Position
- Schriftart
- Schriftart XLFD

Benutzer-Schriftartengruppen im Dateisystem

Wenn ein Benutzer eine Schriftartengruppe hinzufügt, wird sie im folgenden Verzeichnis gespeichert:

Home-Verzeichnis / `.dt/sdttfonts/Host/ Sprache/ Schriftart-nnnnnn`

Host ist der Name des Host der lokalen Workstation.

Sprache gibt die aktuelle Sprachumgebung des Benutzers an (beispielsweise "C" oder "ja").

Schriftart-nnnnnn ist ein Name, der aus dem Namen der ausgewählten Schriftart abgeleitet und um eine eindeutige Zufallszahl erweitert wird.

Dieses Schriftartengruppenverzeichnis enthält die folgenden drei Dateien:

- `fonts.alias`
- `fonts.dir`
- `sdttfonts.group`

Die Dateien `fonts.alias` und `fonts.dir` sind normale X11-Schriftartendateien, die in den Schriftenpfad des X-Servers aufgenommen werden können. Die Datei `sdttfonts.group` enthält den vom Benutzer angegebenen Namen der Schriftartengruppe.

Erstellung von Schriftartengruppen durch den Systemverwalter

Damit andere Benutzer die Schriftartengruppen auf einer Workstation nutzen können, kann der Systemverwalter die Schriftartengruppen in das Verzeichnis `/etc/dt/sdttfonts/ Sprache` oder `/usr/openwin/lib/X11/stdttfonts /Sprache` kopieren. Der Session Manager sucht zunächst in *Home-Verzeichnis* `.dt/stdttfonts/Host /Sprache`, dann in `/etc/dt/sdttfonts /Sprache` und abschließend in `/usr/openwin/lib/X11/stdttfonts/Sprache` .

Farben verwalten

Dieser Abschnitt beschreibt:

- Wie der Style-Manager Bildschirmfarben einstellt.
- Wie der Style-Manager Ressourcen verwendet, um die Farben des Desktops zu steuern.

Farbpaletten

Eine Palette besteht aus einer Gruppe von Farbskalen. Die Farbskalen für die aktuelle Palette werden im Farbdialogfenster des Style-Manager gezeigt.

Für jede Palette existiert eine Datei. Die Ressource paletteDirectories gibt die Verzeichnisse an, die Palettendateien enthalten. Standardmäßig enthält diese Ressource:

- integrierte Paletten: /usr/dt/palettes
- systemweite Paletten: /etc/dt/palettes
- persönliche Paletten: *Home-Verzeichnis* / .dt/palettes

Farbskalen

Jede Farbskala in der aktuellen Palette wird durch eine Farbtaste im Farbdialogfenster des Style-Managers dargestellt. Jede Farbe wird durch eine Farbskala-Kennung identifiziert - eine Zahl von 1 bis 8.



Abbildung 17-1 Farbskala-Kennungswerte für HIGH_COLOR

Jede Farbskala ist aus maximal fünf Farben zusammengesetzt. Jede Farbtaste zeigt die Hintergrundfarbe der Farbskala. Die fünf Farben in jeder Farbskala stellen die folgenden Ressourcen für Anzeigekomponenten dar:

`foreground` - Der Vordergrund eines Anwendungsfensters oder Fensterrahmens. Er ist entweder schwarz oder weiß. Er wird im allgemeinen für Text in Fenstern und Titel benutzt.

`background` - Der Hintergrund der Anwendung oder des Fensterrahmens.

`topShadowColor` - Die Farbe der oberen und linken Schattierungen der Bedienelemente von Anwendungen (z.B. Tasten) und Fensterrahmen.

`bottomShadowColor` - Die Farbe der unteren und rechten Schattierungen der Bedienelemente von Anwendungen und Fensterrahmen.

`selectColor` - Die Farbe, die den aktiven Status von bestimmten Bedienelementen anzeigt, z.B. aktive Umschalttasten und Tasten.

Die Anzahl von Farbskalen, die von jeder Palette benutzt werden, wird bestimmt von der Ressource `colorUse`, die der Benutzer mit dem Dialogfenster Anzahl der benutzten Farben im Style-Manager einstellen kann.

Farbwert angeben

Der Style-Manager verwendet RGB-Werte, wenn er Farbinformationen in seine Palettendateien schreibt. Die Syntax für RGB-Zahlen lautet:

```
#RedGreenBlue
```

Red, *Green*, und *Blue* sind hexadezimale Zahlen, jede mit 1 bis 4 Stellen, die die Menge der benutzten Farbe anzeigen. Jede Farbe muß die gleiche Anzahl von Stellen haben. Deshalb bestehen gültige Farbwerte aus 3, 6, 9 oder 12 hexadezimalen Stellen.

Weiß könnte zum Beispiel auf jede der folgenden Möglichkeiten angegeben werden:

```
#fff
```

```
#ffffff
```

```
#ffffffffff
```

```
#fffffffffffffff
```

Wenn eine Farbressource direkt eingestellt wird, kann entweder der Farbname oder der RGB-Wert verwendet werden. Die Datei `/usr/lib/X11/rgb.txt` enthält eine Liste aller benannten Farben.

Wie Farbskalen Ressourcen zugeordnet werden

Das Desktop ordnet Farbskalen über Ressourcen verschiedenen Bildelementen zu. Folgende Zuordnungen werden gemacht:

Ressource Bildelement

`activeColorSetId` Farbe des aktiven Fensterrahmens

`inactiveColorSetId` Farbe des inaktiven Fensterrahmens

`textColorSetId` Bereiche zur Texteingabe

`primaryColorSetId` Haupthintergrundbereiche der Anwendung

`secondaryColorSetId` Menüleiste, Menüs und Dialogfenster der Anwendung

Diese Ressourcen nehmen eine Farbskalen-Kennung als ihren Wert an. Bildelemente mit Farbskalen-Kennungen farbig zu machen, ermöglicht dem Element, dynamisch zu dem neuen Farbschema zu wechseln, wenn eine neue Palette mit dem Style-Manager ausgewählt wird.

Diese Ressourcen können für einzelne Anwendungen verwendet werden. Beispielsweise zeigt die folgende Zeile, wie man alle `dtterm`-Fenster visuell gruppieren könnte, indem Farbskala 8 als Grundfarbe verwendet wird.

```
dtterm*primaryColorSetId: 8
```

Zuordnungen der Standardfarbskala

Die Farbskalen-Kennung, die für Bildelemente verwendet wird, hängt davon ab, wieviel Farben im Style-Manager eingestellt sind.

Tabelle 17-5 zeigt die Farbskalenkennungen für den Modus "Maximale Anzahl Farben" (acht Farbskalen) des Style-Managers (Einstellung "Mehr Farben für Desktop").

TABELLE 17-5 Maximale Anzahl Farben

Farbskalenkennung	Anzeigeelement
1	Rahmenfarbe des aktiven Fensters
2	Rahmenfarbe nicht aktiver Fenster
3	Nicht verwendet (Standard)

TABELLE P-5 Maximale Anzahl Farben *(fortgesetzt)*

Farbskalenkennung	Anzeigeelement
4	Bereiche zur Texteingabe
5	Haupthintergrundbereiche der Anwendung
6	Menüleiste, Menüs und Dialogfenster der Anwendung
7	Nicht verwendet (Standard)
8	Hintergrund des vorderen Bedienfeldes

Tabelle 17-6 zeigt die Farbskalenkennungen für den Modus “Mittlere Farbe” (vier Farbskalen) des Style-Managers (Einstellung “Mehr Farben für Anwendungen”).

TABELLE 17-6 Mittlere Farbe

Farbskalenkennung	Anzeigeelement
1	Rahmenfarbe des aktiven Fensters
2	Rahmenfarbe nicht aktiver Fenster
3	Hintergrundfarbe für Anwendung und vorderes Bedienfeld
4	Bereiche zur Texteingabe

Tabelle 17-7 zeigt die Farbskalenkennungen für den Modus “Niedrige Farbe” (vier Farbskalen) des Style-Managers (Einstellung “Die meisten Farben für Anwendungen”).

TABELLE 17-7 Niedrige Farbe

Farbskalenkennung	Anzeigeelement
1	Rahmen des aktiven Fensters, Auswahl Schaltflächen im Arbeitsbereich
2	Alle anderen Anzeigeelement

Farbe mit dem Style-Manager steuern

Mit dem Style-Manager können Farben für Desktop-Anwendungen und andere kooperierende Anwendungen dynamisch geändert werden. Die Vorder- und Hintergrundfarben, die vom Style-Manager eingestellt werden, stehen nicht-kooperierenden Anwendungen zur Verfügung.

- Damit ein Client die Farbänderungen des Style-Managers nutzen kann, muß er die Motifbibliothek des Desktop verwenden. Clients, die mit anderen Hilfsprogrammen geschrieben wurden, können die Farbe nicht dynamisch ändern. Farbänderungen für diese Clients haben keine Auswirkung, bevor der Client neu gestartet wird.
- Es dürfen keine anderen spezifischen Farbressourcen für den Client angewendet werden. Dies beinhaltet vom Benutzer angegebene Ressourcen, Standardwerte für Anwendungen und Ressourcen, die in die Anwendung integriert sind
- Clients können die Ressourcen `primaryColorSetId` und `secondaryColorSetId` angeben, um bestimmte Farben einer Desktop-Palette zu verwenden.

Anzahl der vom Style-Manager benutzten Farben

Die Anzahl der Farben, die vom Style-Manager benutzt werden, hängt von den Werten der folgenden Ressourcen ab:

`colorUse` - Konfiguriert die Anzahl der Farben, die vom Desktop benutzt werden.

`shadowPixmaps` - Befiehlt dem Desktop, die zwei Schattenfarben durch Pixmaps zu ersetzen.

`foregroundColor` - Gibt an, ob sich die Vordergrundfarbe dynamisch ändert.

`dynamicColor` - Steuert, ob Anwendungen beim Wechseln von Paletten die Farbe ändern.

Tabelle 17-8 zeigt eine Liste der maximalen Anzahl von Farben, die vom Desktop zugeordnet werden.

TABELLE 17-8 Anzahl der Desktopfarben

Anzeige	Maximale Anzahl der Farben	Anzahl abgeleitet von
B_W (SCHWARZ-WEIß)	2	Schwarz und Weiß
LOW_COLOR (NIEDRIGE FARBE)	12	Zwei Farbskalen mal fünf Farben plus Schwarz und Weiß
MEDIUM_COLOR (MITTLERE FARBE)	22	Vier Farbskalen mal fünf Farben plus Schwarz und Weiß
HIGH_COLOR (MAXIMALE ANZAHL FARBEN)	42	Acht Farbskalen mal fünf Farben plus Schwarz und Weiß

Die maximale Anzahl von Farben bestimmen:

1. Die Anzahl der Farbskalen in der Palette mit der Anzahl der Farben in jeder Farbskala multiplizieren.

2. 2 addieren (für Schwarz und Weiß).

Bei der folgenden Konfiguration wären jedoch nur zehn Farben in der Palette; vier Farbskalen mal zwei Farben in jeder Skala (background und selectColor) plus Schwarz und Weiß.

```
*colorUse: MEDIUM_COLOR
    *shadowPixmaps: True
    *foregroundColor: White
```

Hinweis - Mehrfarbige Symbole benötigen vierzehn zusätzliche Farben.

Ressource colorUse

Der Standardwert der Ressource `colorUse` ist `MEDIUM_COLOR` (mittlere Farbe). Der Wert dieser Ressource beeinflusst die Anzahl der Farbskalen in einer Palette. Andere Ressourcen beeinflussen die Anzahl der Farben für Schattierungen. Der Wert

der Ressource `colorUse` beeinflusst ebenfalls die Benutzung von mehrfarbigen Symbolen.

Wert	Beschreibung
B_W	Style-Manager Einstellung "Schwarz und Weiß" Anzeige mit 1 bis 3 Farbspeichern Anzahl der Farbskalen: 2 Maximale Anzahl der Farben: 2 Standardanzahl der Farben: 2 Keine mehrfarbigen Symbole
LOW_COLOR	Style-Manager Einstellung "Die meisten Farben für Anwendungen" Anzeige mit 4 bis 5 Farbspeichern Anzahl der Farbskalen: 2 Maximale Anzahl der Farben: 12 Standardanzahl der Farben: 12 Keine mehrfarbigen Symbole
MEDIUM_COLOR	Style-Manager Einstellung "Mehr Farben für Anwendung" Anzeige mit 6 Farbspeichern Anzahl der Farbskalen: 4 Maximale Anzahl der Farben: 22 Standardanzahl der Farben: 22 Mehrfarbige Symbole
HIGH_COLOR	Style-Manager Einstellung "Mehr Farben für das Desktop" Anzeige mit 7 oder mehr Farbspeichern Anzahl der Farbskalen: 8 Maximale Anzahl der Farben: 42 Standardanzahl der Farben: 42 Mehrfarbige Symbole\
default	Das Desktop wählt den richtigen Wert für die Anzeige. (Um die Anzahl der Farben zu verringern, die vom Desktop bei der Anzeige von maximalen Farben benutzt wird, ist die Standardressource <code>colorUse</code> auf <code>MEDIUM_COLOR</code> gesetzt.)

Ressource `shadowPixmaps`

Die Ressource `shadowPixmaps` ersetzt die zwei Schattenfarben auf dem Desktop durch Pixmaps. Diese Pixmaps mischen die Hintergrundfarbe mit Schwarz oder Weiß, um obere oder untere Schattierungswerte zu simulieren. Damit wird die Anzahl der benötigten Farben um zwei verringert, da Farbzellen nicht den Schattenfarben zugeordnet werden müssen.

Wert	Beschreibung
True	Das Desktop erstellt ein <code>topShadowPixmap</code> und ein <code>bottomShadowPixmap</code> anstatt die Schattenfarben zu verwenden.
False	<code>topShadowColor</code> und <code>bottomShadowColor</code> aus der Palette werden verwendet.

Der Standardwert für shadowPixmaps hängt von der verwendeten Ressource colorUse und der Hardwareunterstützung für den Bildschirm ab.

Ressource foregroundColor

Die Ressource foregroundColor gibt an, wie der Vordergrund in einer Palette konfiguriert ist.

Einstellung	Ergebnis
White	Der Vordergrund ist auf Weiß eingestellt.
Black	Der Vordergrund ist auf Schwarz eingestellt.
Dynamic (Default)	Dynamic Der (Standard)-Vordergrund ist dynamisch auf Schwarz oder Weiß eingestellt, abhängig von dem Wert von background . Wenn zum Beispiel weiße Buchstaben auf einem gelben Hintergrund schwer zu lesen sind, wählt das System schwarz.

Wenn foregroundColor entweder auf Schwarz oder Weiß eingestellt ist, wird die Anzahl der Farben in der Farbskala um eins verringert, und der Vordergrund ändert sich nicht, wenn sich die Hintergrundfarbe ändert.

Der Standardwert für foregroundColor ist Dynamic, ausgenommen colorUse hat den Wert B_W.

Ressource dynamicColor

Die Ressource dynamicColor steuert, ob Anwendungen die Farbe dynamisch ändern; das heißt, ob die Clients die Farbe ändern, wenn die Paletten gewechselt werden.

Wert	Beschreibung
True	Clients ändern die Farbe dynamisch, wenn eine neue Palette ausgewählt wird. Dies ist der Standardwert.
False	Clients ändern die Farbe nicht dynamisch. Wenn eine neue Palette ausgewählt wird, verwenden die Clients die neuen Farben erst nach einem Neustart der Session.

Wenn der Wert der Ressource `dynamicColor` `True` ist, ordnen Clients, die Farben nicht dynamisch ändern können (Nicht-Motif Anwendungen), andere Zellen der Farbtabelle zu als Clients, die die Farbe dynamisch ändern können.

Hinweis - Da sich alle Clients dieselben Farbzellen teilen können, wird die Anzahl der vom Desktop verarbeiteten Farben reduziert, wenn `dynamicColor` auf `False` eingestellt wird.

Schattenstärken für Anwendungsfenster einstellen

Das Desktop definiert eine Standardschattenstärke von einem Pixel für die Komponenten in Anwendungsfenstern, wie zum Beispiel Tastenschatten und die Hervorhebung von bestimmten Bereichen. Motif 1.2-Anwendungen verwenden diesen Ressourcenwert; andere Anwendungen können diesen Ressourcenwert nicht verwenden und erscheinen deshalb verschieden auf dem Bildschirm.

Die Schattenstärke für Nicht-Motif 1.2-Anwendungen auf ein Pixel einstellen:

1. **Als Benutzer `root` anmelden.**
2. **Die Datei `/etc/dt/config/sprache/sys.resources` erstellen.**

3. **Die applikationsspezifische Ressource in `/etc/dt/config/sprache/sys.resources` wie folgt angeben:**
`application_class_name*XmCascadeButton*shadowThickness: 1`

Mehr Informationen über die Neuebelegung von Standardressourcen des Systems und das Angeben zusätzlicher Ressourcen für alle Desktopbenutzer können dem Abschnitt „Sitzungsressourcen laden“ auf Seite 29 entnommen werden.

Länderspezifische Desktop-Sitzungen konfigurieren

Die Konfiguration länderspezifischer Desktop-Sitzungen umfaßt die folgenden Schritte:

- Die Umgebungsvariable `LANG` und weitere NLS-Umgebungsvariablen einstellen (NLS = National Language Support).
- Auf sprachabhängige Meldungskataloge und Ressourcendateien zugreifen.
- Fernausführung von Anwendungen über internationalisierte Systeme.
- „Umgebungsvariable `LANG` verwalten ” auf Seite 337
- „Schriftarten finden” auf Seite 340
- „Ressourcendatei `app-defaults` länderspezifisch anpassen” auf Seite 341
- „Aktionen und Datentypen länderspezifisch anpassen” auf Seite 341
- „Symbole und Bitmaps länderspezifisch anpassen” auf Seite 342
- „Hilfebände länderspezifisch anpassen” auf Seite 344
- „Meldungskataloge länderspezifisch anpassen” auf Seite 344
- „Fernausführung länderspezifischer Desktop-Anwendungen” auf Seite 344
- „Tastaturbelegung zurücksetzen” auf Seite 345

Umgebungsvariable `LANG` verwalten

Die Umgebungsvariable `LANG` muß gesetzt werden, damit der Desktop die sprachabhängigen Routinen des Betriebssystems verwenden kann. Der Desktop unterstützt folgende Sprachgruppen:

- Westliche und romanische Sprachen
- Japanisch
- Traditionelles Chinesisch
- Vereinfachtes Chinesisch
- Koreanisch

Hinweis - Möglicherweise wurde vom Anbieter Ihres Desktop die Unterstützung weiterer Sprachen installiert.

Die Variable `LANG` kann auf jeden Wert gesetzt werden, den das Betriebssystem unterstützt. Das Menü 'Optionen' des Anmeldebildschirms zeigt die Liste der unterstützten Sprachen und Gebiete.

Es gibt vier Methoden, `LANG` einen Wert zuzuweisen:

- Eine Ressource in der Datei `Xconfig` bearbeiten.
- Über das Menü 'Optionen' im Anmeldebildschirm.
- Ein ausführbares `sh`- oder `ksh`- Skript in `Xsession.d` erstellen. (Siehe auch „Das Skript `Xsession.d` ausführen " auf Seite 26.)
- Die Benutzerdatei `.dtprofile` bearbeiten.

Nachdem `LANG` gesetzt ist, verwendet der Desktop die folgenden sprachabhängigen Dateien, um die Oberfläche länderspezifisch anzupassen.

Farben—`/usr/dt/palettes/desc .sprache`

Hintergrundmuster—`/usr/dt/backdrops/desc .sprache`

Sprache für mehrere Benutzer einstellen

Wenn Sie die Sprache über eine `Xconfig`—Datei einstellen, wird der Anmeldebildschirm länderspezifisch angepaßt, und `LANG` wird für alle Benutzer gesetzt. Dies ist die einzige Möglichkeit, `LANG` für alle Bildschirme in Mehrplatzumgebungen zu ändern. (Damit `Xconfig` geändert werden kann, kopieren Sie die Datei `/usr/dt/config/Xconfig` nach `/etc/dt/config/Xconfig` .)

Die Sprache wird mit folgender Zeile ausgewählt, die Sie in die Datei `/etc/dt/config/Xconfig` einfügen:

```
dtlogin.host_bildschirm.language:
```

```
sprache
```

Im folgenden Beispiel wird `LANG` für den Bildschirm `my_host:0` auf `Swedish_locale` gesetzt.

```
dtlogin.my_host_0.language: Swedish_locale
```

Der Client `dtlogin` liest den entsprechenden Meldungskatalog für diese Sprache und zeigt den länderspezifischen Anmeldebildschirm an. Mit Hilfe der folgenden Ressourcen in der Datei `/etc/dt/config/Xresources` bestimmt der Client `dtlogin` anschließend die Liste der länderspezifischen Daten:

- `dtlogin*language`
- `dtlogin*languageList`
- `dtlogin*languageName`

Eventuell muß in der Datei `xconfig` die Umgebungsvariable `NLSPATH` an die gewählte Sprache angepaßt werden. Wenn diese Anpassung nicht erfolgt oder wenn Sie `NLSPATH` manuell einstellen möchten, finden Sie Hinweise unter „Umgebungsvariable `NLSPATH`“ auf Seite 340.

Sprache für eine einzige Sitzung einstellen

Wenn die Sprache nur für die Dauer einer einzigen Sitzung umgestellt werden soll, verwenden Sie das Menü 'Optionen' im Anmeldebildschirm. Damit wird der Anmeldebildschirm länderspezifisch angepaßt, und die Variable `LANG` wird gesetzt. Nach Abschluß der Sitzung nimmt `LANG` wieder den Standardwert an, der in `dtlogin` eingestellt wurde.

Sprache für einen bestimmten Benutzer einstellen

Der Benutzer kann die bei der Anmeldung gültige Einstellung von `LANG` in der Datei *HomeVerzeichnis* / `.dtprofile` überschreiben. Der Anmeldebildschirm wird dann nicht länderspezifisch angepaßt, und `LANG` wird für den Benutzer gesetzt.

- Wenn Sie `sh` oder `ksh` verwenden, hat die Wertzuweisung folgende Form:

```
LANG=sprache
```

```
export LANG
```

- Wenn Sie `csh` verwenden, hat die Wertzuweisung folgende Form:

```
setenv LANG sprache
```

Umgebungsvariable `LANG` und die Sitzungskonfiguration

Die Umgebungsvariable `LANG` ändert den Namen des Verzeichnisses, in dem nach sitzungsspezifischen Konfigurationsdateien gesucht wird.

Es gibt folgende sprachbezogene Konfigurationsdateien für eine Sitzung:

- /usr/dt/config/*sprache* /Xresources (Ressourcendatei für Login Manager)
- /usr/dt/config/ *sprache*/sys.font (Ressourcendatei für Session Manager)
- /usr/dt/config/ *sprache*/sys.resources (Ressourcendatei für Session Manager)
- /usr/dt/config/ *sprache*/sys.session (Ausführbare Shell für Session Manager)
- /usr/dt/config/ *sprache*/sys.dtwmrc (Ressourcendatei für Window Manager)
- /usr/dt/appconfig/types/ *sprache*/dtwm.fp (Vorderer Steuerbereich für Window Manager)

Weitere NLS-Umgebungsvariablen einstellen

Neben LANG gibt es weitere NLS-Umgebungsvariablen, wie beispielsweise LC_CTYPE und LC_ALL. Diese Variablen werden weder von der Sprachressource dtlogin noch vom Menü 'Optionen' des Anmeldebildschirms beeinflusst, sondern müssen in den folgenden Dateien eingestellt werden:

- Systemweite Variablen: /etc/dt/config/Xsession.d
- Persönliche Variablen: *HomeVerzeichnis* /.dtprofile

Umgebungsvariable NLSPATH

Die Umgebungsvariable NLSPATH definiert die Verzeichnispfade, in denen Anwendungen nach Meldungskatalogen suchen. Sowohl LANG als auch NLSPATH müssen für die Verwendung dieser Kataloge konfiguriert sein. Siehe auch „Meldungskataloge länderspezifisch anpassen“ auf Seite 344. Dort finden Sie Hinweise zur Verzeichnisposition länderspezifischer Meldungen. Die meisten Desktop-Clients stellen der Pfadangabe beim Start das Präfix NLSPATH voran.

Schriftarten finden

Die im Desktop enthaltenen Schriftarten befinden sich im Verzeichnis /usr/lib/X11/fonts. Jedes Verzeichnis enthält eine Verzeichnisdatei namens fonts.dir sowie eine Aliasdatei fonts.alias. Informationen zur Erstellung der Dateien fonts.dir und fonts.alias finden Sie in der Handbuchseite mkfontdir.

Mit dem Befehl xlsfonts können alle auf einem Server verfügbaren Schriften aufgelistet werden. Zum Hinzufügen oder Löschen von Schriftarten dient der Befehl xset.

Ressourcendatei `app-defaults` länderspezifisch anpassen

Der voreingestellte Speicherort der Datei `app-defaults` für die Desktop-Clients ist `/usr/dt/app-defaults/sprache`. Wenn beispielsweise `LANG` auf `Swedish_locale` gesetzt ist, sucht die Anwendung in `/usr/dt/app-defaults/Swedish_locale` nach ihrer `app-defaults`-Datei. Wenn `LANG` dagegen nicht gesetzt ist, wird *sprache* ignoriert, und die Anwendungen suchen ihre `app-defaults`-Datei in `/usr/app-defaults/C`.

Über die Umgebungsvariable `XFILESEARCHPATH` kann der Speicherort von `app-defaults` geändert werden. Wenn `app-defaults` beispielsweise nach `/users` verlagert werden soll, setzen Sie `XFILESEARCHPATH` auf `/usr/app-defaults/sprache/klassenname`.

Wenn Sie `XFILESEARCHPATH` in *HomeVerzeichnis*/`.dtprofile` einstellen, gilt dieser Wert für alle ausgeführten Desktop- und X-Clients. Nicht-Client-Anwendungen finden ihre Ressourcendateien nur dann, wenn Sie diese in das Verzeichnis kopieren bzw. mit dem Verzeichnis verknüpfen, das mit `XFILESEARCHPATH` festgelegt ist.

Aktionen und Datentypen länderspezifisch anpassen

Hinweis - Dateien des Verzeichnisses `/usr/dt/appconfig` können erst dann verändert werden, nachdem sie in das Verzeichnis `/etc/dt/appconfig` kopiert wurden.

Der Suchpfad für Aktions- und Datentypdefinitionsdateien enthält länderspezifische Verzeichnisse:

- Persönlich: *HomeVerzeichnis* `/dt/types`
- Systemweit: `/etc/dt/appconfig/types/sprache`
- Integriert: `/usr/dt/appconfig/types/sprache`

Der Suchpfad für die Konfigurationsdateien des Anwendungsmanagers lautet:

- Persönlich: *HomeVerzeichnis* `/dt/appmanager`
- Systemweit: `/etc/dt/appconfig/appmanager/sprache`
- Integriert: `/usr/dt/appconfig/appmanager/sprache`

Datei- und Verzeichnisnamen in diesem Verzeichnis werden länderspezifisch angepaßt.

Symbole und Bitmaps länderspezifisch anpassen

Zur länderspezifischen Anpassung bearbeiten Sie ein Symbol im Symboleditor und speichern es in:

```
/etc/dt/appconfig/icons/sprache
```

Wenn Sie das veränderte Symbol in einem anderen Verzeichnis ablegen, müssen Sie die Umgebungsvariable `XMICONSEARCHPATH` so anpassen, daß das betreffende Verzeichnis berücksichtigt wird. Die Variable `XMICONBMSEARCHPATH` definiert den Suchpfad für Symbole.

Namen von Hintergrund-Ablagen länderspezifisch anpassen

Die länderspezifische Anpassung von Hintergrund-Ablagen erfolgt über Beschreibungsdateien (`desc.sprache` und `desc.backdrops`). Es gibt kein eigenes angepaßtes Verzeichnis, wie etwa `/usr/dt/backdrops/sprache` für die zu den Hintergrund-Ablagen gehörigen Dateien. Alle Sprachtreiber verwenden dieselben Hintergrund-Ablagedateien, aber jeweils andere Dateien des Typs `desc.sprache`, in denen die übersetzten Namen der Hintergrund-Ablagen enthalten sind.

Die Beschreibungsdatei enthält Ressourcenspezifikationen für die übersetzten Namen von Hintergrund-Ablagen. Ein Beispiel:

```
Backdrops*Corduroy.desc:    Velours
Backdrops*DarkPaper.desc:  PapierKraft
Backdrops*Foreground.desc: AvantPlan
```

Die Datei `desc.sprache` dient zum Abrufen der Beschreibung der Hintergrund-Ablagen für die Zwecke eines bestimmten Sprachtreibers (*sprache*), damit die Hintergrund-Ablage im Style Manager angezeigt werden kann. Wenn eine Beschreibung vorliegt, wird diese im Style Manager in der Auswahlliste für Hintergrund-Ablagen angezeigt. Andernfalls wird der Name der Hintergrund-Ablage verwendet.

Benutzer können in der Datei *HomeVerzeichnis* `/.dt/backdrops/desc.backdrops` eigene Beschreibungen von Hintergrund-Ablagen definieren. Aus dieser Datei werden die Beschreibungen aller benutzerdefinierten Hintergrund-Ablagen abgerufen, unabhängig vom aktuellen Sprachtreiber.

Der Suchpfad für die Beschreibungsdateien lautet:

- Persönlich: *HomeVerzeichnis* `/.dt/backdrops/desc.backdrops`
- Systemweit: `/etc/dt/backdrops/desc.sprache`
- Integriert: `/usr/dt/backdrops/desc.sprache`

Palettennamen länderspezifisch anpassen

Die länderspezifische Anpassung von Paletten erfolgt über Beschreibungsdateien (`desc.sprache` und `desc.palettes`). Ein eigenes angepaßtes Verzeichnisse gibt es nicht (wie z.B. `/usr/dt/palettes/sprache`). Alle Sprachtreiber verwenden dieselben Palettendateien, aber jeweils eine eigene Datei `desc.palettes`, welche die übersetzten Namen der Paletten enthält.

Die Beschreibungsdatei enthält Ressourcenspezifikationen für die übersetzten Palettennamen. Ein Beispiel:

```
Palettes*Cardamon.desc:    Cardamone
Palettes*Cinnamon.desc:   Cannelle
Palettes*Clove.desc:       Brun
```

Die Datei `desc.sprache` dient zum Abrufen der Palettenbeschreibung für die Zwecke eines bestimmten Sprachtreibers (*sprache*), damit die Palette in der Auswahlliste des Style-Managers angezeigt werden kann. Wenn eine Beschreibung vorliegt, wird diese in die Liste aufgenommen, andernfalls der Palettenname.

Benutzer können in der Datei *HomeVerzeichnis* `/.dt/palettes/desc.palettes` eigene Palettenbeschreibungen definieren. Aus dieser Datei werden die Beschreibungen aller vom Benutzer hinzugefügten Paletten abgerufen, unabhängig vom aktuellen Sprachtreiber.

Der Suchpfad für die Beschreibungsdateien lautet folgendermaßen:

- Persönlich: *HomeVerzeichnis* `/.dt/palettes/desc.palettes`
- Systemweit: `/etc/dt/palettes/desc.sprache`
- Integriert: `/usr/dt/palettes/desc.sprache`

Hilfebände länderspezifisch anpassen

Wenn Sie einen Hilfeband länderspezifisch angepaßt haben, müssen Sie ihn in einem der folgenden Verzeichnisse speichern. Der erste gefundene Hilfeband wird verwendet. Die Verzeichnisse sind hier in der Reihenfolge genannt, in der sie durchsucht werden:

- Persönlich: *HomeVerzeichnis* / .dt/help
- Systemweit: /etc/dt/appconfig/help/ *sprache*
- Integriert: /usr/dt/appconfig/help/ *sprache*

Meldungskataloge länderspezifisch anpassen

Wenn Sie einen Meldungskatalog länderspezifisch angepaßt haben, speichern Sie ihn in folgendem Verzeichnis:

`/usr/dt/lib/nls/msg/sprache.`

Dort befinden sich auch die Dateien des Typs `*.cat`.

Fernausführung länderspezifischer Desktop-Anwendungen

Länderspezifisch angepaßte Desktop-Anwendungen können über einen beliebigen entfernten Host aufgerufen werden, wenn dieser Host über eine ähnliche länderspezifisch konfigurierte Desktopinstallation verfügt. Die Werte der NLS-bezogenen Umgebungsvariablen auf dem Host, der die Anwendung aufruft, werden beim Start der Anwendung an den entfernten Host übergeben. Allerdings enthalten die Umgebungsvariablen keine Hostinformationen.

Tastaturbelegung zurücksetzen

Wenn auf dem Bildschirm unerwünschte Zeichen zu sehen sind oder wenn die Tastatur nicht wie gewohnt funktioniert, muß eventuell die Tastaturbelegung zurückgesetzt bzw. installiert werden, oder Sie müssen die Eingabemethode wechseln.

Die Eingabemethode wird festgelegt durch die Umgebungsvariablen `LC_CTYPE`, `LANG` und `LC_ALL` oder durch die Sprache, die mit der Option `--lang` definiert ist.

Angenommen, ein Benutzer möchte ein Terminal innerhalb einer POSIX-Shell mit dem Sprachtreiber `C` öffnen:

```
LANG=C dtterm
```

Dieses neue Terminal verwendet den `C`-Sprachtreiber einschließlich der `C`-Eingabemethode und den zugehörigen Schriftarten. Wenn Sie mit einer sprachspezifischen Tastatur arbeiten, akzeptiert die Eingabemethode eventuell keine Sonderzeichen. Wenn der Sprachtreiber `C` in Verbindung mit einer sprachspezifischen Tastatur eingesetzt wird, muß der Benutzer erst die Umgebungsvariable `LC_CTYPE` (oder `LANG` bzw. `LC_ALL`) geeignet setzen, bevor das Terminal aufgerufen wird.

Im folgenden Beispiel soll der Sprachtreiber `C` mit einer deutschen Tastatur verwendet werden:

```
LANG=C LC_CTYPE=DeDE dtterm
```

Wenn der `X`-Server zurückgesetzt und die Tastaturtabellen initialisiert wurden, können Sie die gewünschte Tastaturbelegung am Server mit dem Befehl `xmodmap` wiederherstellen.

Hilfeseite dtconfig(1)

NAME

dtconfig - Desktop-Konfigurationsprogramm

SYNTAX

dtconfig [-d |-e |-kill |-reset |-p |-inetd |-inetd.ow]

BESCHREIBUNG

Desktop-Konfigurationsprogramm. Integriert CDE und das Betriebssystem der zugrundeliegenden Plattform. Für den Einsatz von dtconfig werden die Rechte des Benutzers "root" benötigt.

OPTIONEN

-d Deaktiviert die Autostart-Funktion des Desktop. Nach dem Start wird die zeichenorientierte Login-Prozedur der jeweiligen

(fortgesetzt)

Plattform verwendet.

-e Aktiviert die Autostart-Funktion des Desktop. Es erfolgt ein Desktop-Login.

-kill

Löscht den Desktop-Login-Prozeß und alle zugehörigen Benutzersitzungen aus dem Speicher. Die Steuerung wird anschließend an die zeichenorientierte Konsole des Systems übergeben.

-reset

Der Desktop-Login-Prozeß wird angewiesen, die Konfigurationsdatei erneut zu lesen, um die enthaltenen Änderungen zu aktivieren.

-p Druckoperationen werden auf allen der Plattform bekannten Druckern ausgeführt, wenn keine spezifischen "Drucken"-Aktionen in der Aktionsdatenbank der Plattform existieren. Diese Aktion wird beim Start automatisch ausgeführt, wenn die Autostart-Funktion für das Desktop aktiviert ist.

-inetd

Fügt den Dämon /usr/dt/bin in die Datei /etc/inetd.conf ein.
Spezifische Hintergrunddämonen für CDE sind

(fortgesetzt)

rpc.ttdbserverd (ToolTalk), rpc.cmsd (Kalender Manager) und dtspcd (Subprozeß-Steuerung). Die Option `-inetd` wird automatisch bei einer Solaris CDE-Paketinstallation verwendet. Sie ist auch für das Setup von CDE-Dämonen unabhängig von der Solaris CDE-Installation hilfreich (einschließlich System-Setup, wenn `/usr/dt` einfach von einem Remote-Server eingehängt wurde und letzterer das Verzeichnis `/usr/dt` exportiert).

`-inetd.ow`

Schaltet die Startzeilen der Dämonen ToolTalk und Kalender Manager (`rpc.ttdbserverd` und `rpc.cmsd`) in `/etc/inetd.conf` auf den älteren Bereich `/usr/openwin/bin` zurück. Diese Option wird automatisch von Deinstallationskripten für das Solaris-CDE-Paket verwendet, wenn dies erforderlich ist. Es ist auch hilfreich, wenn `/usr/dt` manuell entfernt oder ausgehängt wird.

RÜCKGABEWERTE

0 Erfolgreicher Abschluß
>0 Fehlerbedingung

DATEIEN

(fortgesetzt)

(Fortsetzung)

`/usr/dt/bin/dtconfig` Position des Dienstprogramms `dtconfig`

SIEHE AUCH

`dtlogin (1)`, `dtprintinfo (1)`

Index

Numbers

workspaceCount (Ressource), 303

A

Abgelegte Datei, Aktion akzeptiert, 213

Ablagezone

 Aktionssymbol, 213

 Steuerung im vorderen Steuerbereich, 286

actionIcon-Ressource, 207

ACTIONS Feld, 240

activeColorSetId Ressource, 330

Aktion erstellen, 181

 Aktionsbefehlssyntax, 186

 Aktionsname, 185

 Dateiargument angeben, 186

 Dateieingabe, 188

 Datentyp erstellen, 172, 188

 Datentypname, 190

 Einschränkungen, 182

 erstellte Konfigurationsdatei, 182

 Funktionsweise, 181

 Hauptfenster, 185

 starten, 184

 Symbol angeben, 196

 Verwendung, 183

Aktionen

 abgelegte Dateien akzeptieren, 213

 abgelegte Dateien akzeptieren oder
 anfordern, 214

 andere Aktionen ausführen, 228

 andere Aktionen starten, 227

Argumente, 182, 212

 Argumente einschränken, 221

Argumente, Nicht-Datei, 215

 Ausführungszeichenfolge, 211

 austauschbare Argumente, 217

 bearbeiten, 208

Bedienfeld, 166

Beispiel, 203, 204

 Beschriftungen, 207

 beschränken durch Argumentanzahl, 222

 Bezeichnungen, 230

 COMMAND, 200

Datei anfordern, 214

 Dateiargumente, 171

Datentypeneinschränkungen, 176

Datentypen, 174

Datentypen zuordnen, 240

 durch Datentyp beschränken, 222

 Einführung, 163, 164

 Einschränkungen, 182

 entfernte Anwendungen ausführen, 224

 erforderlich für Registrierung, 69

Fensterunterstützung, 187, 219

 für anderen Benutzer ausführen, 228

 Gründe für manuelles Erstellen, 200

- integriert mit dtappintegrate, 82
- keine Bildschirmausgabe, 187
- Konfigurationsdateien, 201
- lokalisiert, 229
- manuell erstellen, 199, 201, 202
- MAP, 200
- mehrere abgelegte Dateien
 - akzeptieren, 218
- Name, 182, 206
- neu laden, 205
- nicht austauschbare Argumente, 217
- Nicht-Dateiargumente, 215
- ohne Argumente, 212
 - ohne Daten, 212
- Parameter, *siehe* Argumente, 182
 - Reihenfolge, 209
- Server, 131
 - Shell bereitstellen, 216
- Standardsymbol, 207
- Standardterminal, 220
- Suchpfad, *siehe* Datenbanksuchpfad, 145
 - Symbol zuordnen, 257
- Symbole, 196, 207
 - Symbole für Anwendungen erstellen, 168
- Symbolrepräsentation, 205
 - Terminaloptionen, 220
- Terminalunterstützung, 220
- TT_MSG, 201
- Typen, 200, 202
- Umgebungsvariablen, 227
 - unterschiedliche Funktionsweisen für
 - Doppelklicken oder Ablegen, 223
- Variablen in Definitionen, 226
- verwendet in Menüs, 166
 - Verwendungen, 172
- Zeichenfolgevariablen, 226
- zuordnen, 176
- ändern, 208
 - übergebene Dateien akzeptieren, 171
- Aktionen , *siehe* Aktionsdatei
 - Dateien,,
 - Aktionen anfordern, 214
 - Aktionen bearbeiten, 208
 - Aktionen neu laden, 205
 - Aktionen, *siehe* Datenbanksuchpfad
 - Suchpfad,,
 - Aktions-Server, , *siehe* Datenbank-Server,
- Aktionsdatei, 182, 205
 - Definition, 168
 - erstellen, 79, 205
 - Inhalt, 169
- Aktionsdefinitionsdatei, mit Aktion
 - erstellt, 182
- Aktionsnamenfeld in Aktion erstellen, 185
- Aktionssuchpfad,, *siehe* Datenbanksuchpfad,
- Aktionssymbol, 168, 205, 206
 - erstellen, 205
- Aktionssymbolsteuerung, in Aktion
 - erstellen, 185
- Aktionssystembol
 - Desktop, 73
- Aktuelle Sitzung, 24
- ALTERNATE_ICON-Feld, 287
- Animation für vorderen Steuerbereich, 290
- ANIMATION-Definition, 290
- Anmelde-Server
 - konfigurieren, 123
- Anmeldeabrechnungen, 120
- Anmeldeanzeige
 - anpassen, 11
 - anzeigeabhängige Funktionsweise, 14
 - auf Netzwerkanzeige anzeigen, 6
- Begrüßung, 12
 - Begrüßungsmeldung ändern, 12
- Erscheinungsbild ändern, 11
 - Funktionsweise ändern, 13
- Inhalt des Menüs Sprachen ändern, 15
- Lokalisierung, 13
- Ressourcen, 11
- Schriftarten, 12
- Standardsprache ändern, 15
- X-Server-Umgebung, 15
- Zugriff auf X-Server, 14
- Anmeldebildschirm
 - Failsafe—Sitzung (Option), 99
 - verlassen, 97
- Anmeldedateien, 39
- Anmeldefehlerprotokoll, 40
- Anmeldemanager, 2
 - anpassen, 2
 - Befehle ausführen, 16
 - Definition, 2
- Fehler, 9
 - Konfigurationsdateien, 20

- Ressourcen, 11, 13
 - verwalten, 19
- Anmeldeprüfung, 39
- Anmeldeserver
 - Anmeldeanzeige anzeigen, 2
 - Anzeigen zuordnen, 2
 - beenden, 9
 - Befehlszeilenanmeldung, 5
 - Benutzer authentisieren, 2
 - Benutzerpfad, 18
 - deaktivieren, 10
 - Fehlerbehebung, 9
 - in Befehlszeile starten, 2
 - konfigurieren, 3
 - ohne Grafikanzeige, 5
 - ohne lokale Anzeige ausführen, 5
 - Prozeß-ID, 3
 - Prozeß-ID löschen, 10
 - Sitzung starten, 2
 - starten, 2
 - System-Shell, 19
 - Umgebungen, 17, 18
 - Zeichenkonsole, 6
 - Zeitzone ändern, 19
 - Zugriff steuern, 7
 - Übersicht, 2
- Anpassungen beim E-Mail-Druck, 107
- AnswerBook—Dokumentation, hinzufügen
 - über das Netz, 109
- Anwendungen
 - Datentypen, 61
 - dem Anwendungsmanager
 - hinzufügen, 47
 - Desktop-fähig, 49
 - entfernen, 54
 - erfassen im Session Manager, 28
 - erforderliche Aktionen, 69
 - erforderliche Datentypen, 69
 - existierender Anwendungsgruppe
 - hinzufügen, 50
 - Funktionen durch die
 - Anwendungsregistrierung, 60
 - hinzufügen, 48
 - im Anwendungsmanager
 - zusammenstellen, 44
 - neu laden, 55
 - ohne Registrierungspaket hinzufügen, 49
 - Registrierung, 48
 - Stammverzeichnis, 66
 - Start bei Anmeldung, 25, 31
 - Symbole erstellen, 168
 - Verzeichnis app_root, 66
 - zusammenstellen, 46
 - Anwendungen Erneut Laden Aktion, 55
 - Anwendungen, *siehe* Registrierung
 - registrieren,,
 - Anwendungs-Server
 - Client, 130
 - Client konfigurieren, 130
 - Definition, 115
 - konfigurieren, 124, 130
 - Verfügbarkeit von Anwendungen, 63
 - verwalten, 129
 - Anwendungsfenster, Symbol zuordnen, 259
 - Anwendungsgruppe
 - benennen, 52
 - Anwendungsgruppen
 - Aktionen, 77
 - anpassen, 52
 - Beispiel zur Erstellung, 90
 - benennen, 51
 - Datentypen, 77
 - Definition, 43
 - erstellen im Registrierungspaket, 74
 - Inhalt, 78
 - integriert mit dtappintegrate, 83
 - Namen, 75
 - persönlich, 51
 - Readme-Dateien, 80
 - Standard, 46
 - Symbole, 73, 76
 - systemweit, 51
 - verwalten, 51
 - Verzeichnisse, 74
 - zusammenstellen, 44
 - Anwendungsmanager
 - aktualisieren, 55
 - allgemeine Verwaltung, 54, 56
 - Anwendungen erfassen, 28
 - Anwendungen hinzufügen, 47
 - Anwendungen integrieren, 60
 - Anwendungen zusammenstellen, 44
 - Beschreibung, 43
 - symbolische Verbindungen, 45
 - Verzeichnisposition, 44

- Vorrangsregeln, 45
- Anwendungspaket, , *siehe*
 - Registrierungspaket,
- Anwendungsregistrierung, , *siehe*
 - Registrierung,
- Anwendungsressourcen,, *siehe* Ressourcen,
- Anwendungsstammverzeichnis, 66
- Anwendungssuchpfad
 - Anwendungen suchen und zusammenstellen, 44
- Anwendungs-Server hinzufügen, 52
 - assemblieren, 151
 - Definition, 149
 - persönlich, 54
 - Position, 161
 - Rangfolge ändern, 151
 - Standard, 52, 149
 - Syntax, 150
 - systemweit, 53
 - Umgebungsvariable, 150
 - ändern, 53
- Anwendungssymbol, 202
 - doppelklicken, 171
 - erstellen, 168, 205
 - Verwendung von Aktion erstellen, 182
 - übergebene Dateien, 171
- Anwendungssysmbol
 - Desktop, 73
- app-defaults (Datei)
 - sprachabhängig, 341
- /appconfig, 94
- /app-defaults, 94
- Apple Macintosh Application
 - Environment, 110
- app_root, , *siehe*
 - Anwendungsstammverzeichnis,
- Arbeitsbereich (Menü)
 - erstellen, 308
 - Syntax, 306
 - ändern, 307
- Arbeitsbereiche
 - anpassen, 303
 - Anzahl, 303
 - Namen, 304
 - Standardanzahl ändern, 292
- Arbeitsbereiche, *siehe* Hintergrund-Ablagen
- Hintergrund-Ablagen,
- Argumente

- Aktionen, 212
 - anfordern, 214
 - Anzahl für Aktionen, 222
 - austauschbar für Aktionen, 217
 - einschränken für Aktionen, 221
 - mehrere für Aktionen, 216
 - nicht austauschbar, für Aktionen, 217
 - Nicht-Datei, 215
 - von Aktionen, 171
- Arg_1 Syntax, 171
- ARG_CLASS-Feld, 210
- ARG_COUNT-Feld, 210, 222, 223
- ARG_MODE-Feld, 210
- Arg_n syntax, 213
- ARG_TYPE Feld, 240
 - drucken, 144
- ARG_TYPE-Feld, 210, 222
- auf Name basierende Datentypen, 243
- auf Pfad basierender Datentyp, 245
- auf Position basierender Datentyp, 245
- ausführbare Datei, Dateitypkriterien, 247
- Ausführungs-Host
 - Aktionen erstellen, 224
 - angegeben im Feld EXEC_HOST, 225
- Ausführungszeichenfolge, 211
 - abgelegte Dateien, 213
 - absoluter Pfad, 212
 - Allgemeine Merkmale, 211
 - ausführbare Datei angeben, 212
 - Datei anfordern, 214
 - mehrere Dateiargumente, 216
 - ohne Argumente, 212
 - Shell-Syntax, 212
 - Zeichenfolge anfordern, 215
- Ausgabevariable, 146
- Ausgewählt Menü, 167, 174, 175
- Auswahl, 102, 103
- authorize-Ressource, 15
- Automounter, 126

B

- %B, 148
- background, Ressource, 329
- Basisdateiname, 208, 239
- Bedienfeld
 - Element im Registrierungspaket, 80

- Symbol anzeigen, 258
- Verwendung von Aktionen, 166
- Befehlszeile für Aktionen, 211
- Befehlszeilenanmeldung, 5
- Begrüßungsmeldung
 - anpassen, 27
 - anzeigen, 25, 27
 - Standard, 12
 - ändern, 12
- Benutzer, ändern für Aktion, 228
- Benutzer-ID, 120
- Berechtigungsnummer, angeben mit Aktion
 - erstellen, 192
- Beschriftungen
 - Aktionen, 207
 - Steuerungen im vorderen Steuerbereich, 295
- Bezeichnungen
 - Aktionen, 230
- Bilder, *siehe* Symbole,
- Bildschirmfarben
 - maximale Anzahl, 332
- Bildschirmspezifische Ressourcen, 33
- Bildschirmspezifische Sitzungen, 35
- /bin, 94
- Bitmaps, 254
 - Dateien finden, 254
 - Namenskonventionen, 255
 - Suchpfad, 256
 - Wie Dateien gefunden werden, 256
- blank-Steuerungstyp, 284
- .bm Dateinamenerweiterung, 254
- bottomShadowColor, Ressource, 329
- BOX-Definition, 268
 - Syntax, 270
- BROADCAST, indirekter XDMCP-Modus, 8
- broadcast (Flag), 106
- buttonBindings (Ressource), 311

C

- C (Sprachtreiber), 105
- CDE-MIN Dateien, 124
- CDE-TT Dateien, 124
- CDE-Verzeichnis aushängen, 100
- Chooser, 102
- CHOOSEER-Zeichenfolge, 8
- Client-Server-Konfiguration, *siehe* Netzwerk,

- client-Steuerungstyp, 284
- Clients
 - Definition, 114
 - eines Server konfigurieren, 123
 - Fenster im vorderen Steuerbereich, 288
- CLIENT_GEOMETRY-Feld, 289
- CLIENT_NAME-Feld, 288, 289
- clock-Steuerungstyp, 284
- colorUse (Ressource), 31
- colorUse, Ressource, 332, 333
- COMMAND
 - Ausführungszeichenfolge, 211
- COMMAND-Aktion, 200
 - Beispiel, 203
 - erforderliche Felder, 211
- /config, 94
- CONTAINER_NAME-Feld, 270 to 272, 276
- CONTAINER_TYPE-Feld, 270
- CONTENT, Feld, 248
- CONTENT-Feld, 243
- CONTROL-Definition, Syntax, 270
- CONTROL_BEHAVIOR-Feld, 295
- COPY_TO_ACTION-Feld, 242
- cpp (Anweisung), 34
- current.old (Verzeichnis), 36

D

- %DatabaseHost%, 225
- DataBaseHost-Schlüsselwort, 225
- DATA_ATTRIBUTES
 - definieren, 239
 - Definition, 234
 - Syntax, 236
- DATA_CRITERIA
 - definieren, 242
 - Definition, 234
 - mehrere, 250
 - paarig mit DATA_ATTRIBUTES, 234
 - Syntax, 236
- date-Steuerungstyp, 284
- Datei, Datentypkriterien, 247
- Datei-Server, 115, 116
- Dateiargument
 - angegeben mit Aktion erstellen, 186
 - verwendet in Aktionen, 171

- Dateieingabe, angegeben mit Aktion erstellen, 188
- Dateien
 - anhängen, 120
 - auf der Grundlage des Datentyps verstecken, 241
 - Einhängepunkt, 125
 - ferne Daten, 125
 - ferner Zugriff, 120
 - Namenkonsistenz, 122
 - Netzwerk, 124
 - Zugriff auf verteilte Dateisysteme, 120
- Dateien, Anmeldung, 39
- Dateinhalt, 192
- Dateimanager, Symbole anzeigen, 259
- Dateinamen zuordnen, 125
- Dateinamen, Symbol, 254
- Dateinamendatenbank-Server, 124
- Dateinamenkonsistenz, 122
- Dateitypen
 - auf Name basierend, 243
 - auf Pfad basierend, 245
 - auf Position basierend, 245
 - Hilfe, 239
 - mehrere Kriterien, 249
 - Moduskriterium, 246
 - Positionen, 250
 - Symbole, 239
 - verdeckt, 241
- Dateitypen,, *siehe* Datentypen,
- Datenbank
 - Aktionen neu laden, 205
 - neu laden, 205
- Datenbank-Host, 133, 224
- Datenbank-Server, 118, 133
 - Client, 132
 - erstellen, 132
 - konfigurieren, 124, 131
- Datenbank-Suchpfad
 - Auswirkung auf EXEC_HOST, 134
- Datenbanksuchpfad, 153, 201
 - assemblieren, 156
 - Auswirkung des
 - Anwendungssuchpfads, 152
 - bezogen auf Anwendungssuchpfad, 154
 - Siehe auch Suchpfad, 153
 - Standard, 154
 - Syntax, 155
 - Umgebungsvariable, 154
- Datentypen
 - Aktionen, 174
 - Aktionen beschränken, 222
 - Aktionen neu laden, 205
 - Aktionen zuordnen, 240
 - Attribute, 239
 - ausführbar, 247
 - Bedingungen für manuelle Erstellung, 183
 - Beispiel, 237
 - Berechtigungs muster, 192, 193
 - definieren, 235
 - differenzieren, 242
 - drucken, 178
 - Einführung, 163, 172
 - Einschränkungen, 183
 - erforderlich für Registrierung, 69
 - Erfordernisse für manuelle Erstellung, 234
 - erstellen, 61
 - Inhalt, 192, 248
 - integriert mit dtappintegrate, 82
 - Konfigurationsdateien, 235
 - Kriterien, 242
 - Kriterien definieren, 242
 - manuell erstellen, 233
 - mit Aktion erstellt, 172, 188
 - Name, 192
 - neu laden, 205
 - schreibgeschützt, 247
 - Server, 131
 - Suchpfad, *siehe* Datenbanksuchpfad, 153
 - Symbole, 73, 196
 - Symbole zuordnen, 257
 - Variablen in Definitionen, 226
 - Verhalten bei Doppelklicken, 177
- Datentypenliste in Aktion erstellen, 189
- DELETE-Feld, 267
- DESCRIPTION, Feld, 239
- DESCRIPTION-Feld, 207
- Desktop-fähige Anwendung, 49
- Desktop-Start
 - mehrere Bildschirme, 100
 - Probleme, 39
- Desktop-Suchpfade, 25
- Desktop-Umgebungsdatei, 110
- Desktops im Netzwerkbetrieb, 102
- Desktops, Netzwerkbetrieb, 102

- Desktop_Tools Anwendungsgruppe
 - ändern, 52
- /dev/console, 96
- DISPLAY-Variable, vom Anmeldemanager
 - gesetzt, 18
- %DisplayHost%, 225
- DisplayHost-Schlüsselwort, 225
- DISPLAY_displayname (Makro), 34
- DROP_ACTION-Feld, 286
- DROP_ANIMATION-Feld, 290
- Druck von E-Mails anpassen, 107
- Drucken
 - Datentypen, 144
 - konfigurieren für Datentyp, 178
 - Konzepte, 143
 - Standard, 142
 - Test, 122
 - Verwaltung, 139
- Drucker
 - Bezeichnungen, 142
 - Einheitennamen, 122
 - ferner Zugriff, 122
 - Hilfe, 142
 - hinzufügen, 140
 - Jobaktualisierungsintervall, 141
 - löschen, 140
 - Standard, 142
 - Symbolabbilder, 141
- Druckjob-Aktualisierungsintervall, 141
- Druckmanager, 140
 - Jobaktualisierungsintervall, 141
- dt-Dateien, 201
- dtaction
 - Benutzer ändern, 228
 - Syntax, 227
- dtappgather, 28
- dtappgather Desktop-Dienstprogramm, 45
- dtappintegrate, 81
 - Anwendung entfernen, 54
 - Funktionen, 82
 - Syntax, 82
- DTAPPSEARCHPATH Variable
 - assemblieren, 150, 151
 - Definition, 147
- DtButtonBindings, 309
- dtchooser-Datei, 20
- dtconfig-Befehl, 10
- DTDATABASESEARCHPATH Variable
 - assemblieren, 155, 156
 - Definition, 147
- DTDATABASESEARCHPATH-Variable
 - Verwendung, 209
- Dterrors-Datei, 9
- dtgreet-Datei, 20
- DTHELPSEARCHPATH Variable
 - assemblieren, 161
 - Definition, 147
- dtlogin, 95
- Dtlogin*language-Ressource, 15
- dtlogin, , *siehe* Anmeldemanager,
- dtlp, 108
- dtmailpr, 108
- DTMOUNTPOINT Variable
 - Prozesse, 126
 - Wert setzen, 126
- Dtpid-Datei, 3
- .dtprofile, 39 to 41
- .dtprofile (Datei)
 - anlegen, 26
 - LANG einstellen, 339
 - Umgebungsvariablen einstellen, 32
 - ausführen, 26
 - ausführen, 25
 - Syntax, 33
- DtRootMenu, 307
- dtsearchpath, 27, 209
- dtsearchpath Variable, 147
- dtsmcmd (Befehl), 34
- DTSOURCEPROFILE (Variable), 28
- dtspcd, 124, 126, 127
 - Identifikationsverzeichnis, 121
 - Identifikationsüberprüfungsverzeichnis, 127
 - konfigurieren, 127
- DTSPSYSAPPHOSTS Variable, 131
 - Definition, 147
 - Syntax, 150
 - ändern, 53
- DTSPSYSDATABASEHOSTS Variable
 - Definition, 147, 155
 - Syntax, 155
- DTSPSYSDATABASEHOSTS Variable
 - Auswirkung auf EXEC_HOST, 134
- DTSPSYSHELP Variable, 160
 - Definition, 147
 - Syntax, 160

DTSPSYSICON Variable
 Definition, 147
 Syntax, 158

DTSPUSERAPPHOSTS Variable, 131
 Definition, 147
 Syntax, 150
 ändern, 54

DTSPUSERDATABASEHOSTS Variable
 Definition, 147, 155
 Syntax, 155

DTSPUSERHELP Variable
 Definition, 147
 Syntax, 160

DTSPUSERICON Variable
 Definition, 147
 Syntax, 158

dtstart_appgather (Variable), 28
 dtstart_searchpath (Variable), 28
 dtstart_ttsession (Variable), 29
 dtterm, 42
 dtwm, , *siehe* Workspace Manager,
 dtwm.fp file, 264
 dtwmfp.session-Datei, 266
 dtwmrc (Datei), 300
 bearbeiten, 301
 Dtwmrc bearbeiten (Aktion), 301
 dynamicColor (Ressource), 31
 dynamicColor Ressource, 332, 335

E

E-Mail-Druck, anpassen, 107
 Eingabemethode, Internationalisierung, 345
 Eingabevariable, 146
 Einhängen
 installiertes CDE, 99
 Einhängepunkt für ferne Dateien, 125
 Entfernte Ausführung
 durch Aktionen, 224
 /etc/dt, 94
 /etc/rmmount.conf, 111
 /examples, 94
 Execution Host
 angeben, 133
 konfigurieren, 135
 EXEC_HOST Feld, 134
 Standardwert, 134
 EXEC_HOST,, *siehe* Ausführungs-Host,

EXEC_HOST-Feld, 225
 Standardwert, 225
 EXEC_STRING, 108
 EXEC_STRING-Feld, , *siehe*
 Ausführungszeile,

F

Failsafe-Sitzung, Option, 40
 FailSafe-Sitzungen, 17
 Failsafe—Sitzung (Option)
 Anmeldebildschirm, 99

Farbe
 Farbskalen, 329
 maximale Anzahl, 332
 Schatten mit der Ressource
 shadowPixmap erstellen, 334
 steuern mit der Ressource
 dynamicColor, 335
 Verwendung mit Symbolen, 260

Farben
 aktiver Fensterrahmen , 330
 Anwendungsfenster, 330
 benutzte Anzahl, 332
 Paletten, 328
 Standard, 330
 Texteingabebereiche, 330

Farbpaletten, 328
 Farbressourcen, ändern für Registrierung, 65
 Farbserver, 25
 Ressourcen, 31
 starten, 31

Farbskalen, 328
 Standard, 330

Fehlerbehebung beim Desktop-Start, 39
 Fehlerprotokoll
 Anmeldung, 40
 Session Manager, 40
 Start, 40
 Verzeichnisse, 40
 Xsession, 40

Fenster (Menü)
 Definition, 306
 neu, 308

Fenstermanager
 ändern, 31

Fenstermenü

Index-358

Solaris Common Desktop Environment: Benutzerhandbuch für Fortgeschrittene und Systemverwalter ♦ September 1997

- Syntax, 306
- Fensterunterstützung für Aktionen, 187
- Fernausführung
 - Unterstützung der Landessprache, 344
- ferne Ausführung
 - Anwendungs-Server konfigurieren, 129
- file-Steuerungstyp, 284, 287
- FILE_NAME-Feld, 286, 287
- Flags, -broadcast, 106
- Font-Ressource, 323
- FontSet-Ressource, 323
- foreground Ressource, 329
- foregroundColor (Ressource), 31
- foregroundColor Ressource, 335
- foreign-Anzeigetyp, 5
- fp_dynamic-Verzeichnis, 265

G

- gemeinsam benutzte Dateien, 120
- getty, 6, 25
- GID, 120
- Grafikanzeige, Anmeldeserver, 4
- Gruppe suchen Dialogfenster, 196
- Gruppen von Anwendungen
 - benennen, 51
- Gruppen-ID, 120

H

- %H, 148
- Help Developer's Kit, 72
- HELP_STRING-Feld, 291
- HELP_TOPIC-Feld, 291
- HELP_VOLUME-Feld, 291
- Hilfe, 72
 - Aktionsdatei, 207
 - angegeben mit Aktion erstellen, 187
 - Datentyp, 239
 - Druckersymbole, 142
 - Partielle Integration, 72
 - Volle Integration, 72
 - Vorderer Steuerbereich, 291
- Hilfe-Server, 118
 - Client, 132
 - erstellen, 132
 - konfigurieren, 124, 131
- Hilfeband

- Integrationsstufen, 72
- Position im Registrierungs paket, 72
- Stammdatei, 72
- Hilfebände
 - länderspezifisch anpassen, 344
- Hilfedateien
 - integriert mit dtappintegrate, 83
 - Registrierungs paket, 72
- Hilfeseite dtconfig(1), 347
- Hilfesuchpfad, 159
 - assemblieren, 161
 - Auswirkung des
 - Anwendungssuchpfads, 152
 - bezogen auf Anwendungssuchpfad, 160
 - siehe auch Suchpfad, 145
 - Standard, 159
 - Syntax, 160
 - Umgebungsvariable, 160
- Hintergrund-Ablagen, 300
 - Grafiken verwenden, 305
 - hinzufügen, 304
- Hintergrund-Ablagen
 - Datei-Speicherorte, 305
- Home-Sitzung, 24
- HOME-Variable, 18
- Home-Verzeichnis, 95
 - gemeinsame Nutzung, 121
 - Netzwerk, 121
- home.old (Verzeichnis), 36

I

- ICON Feld
 - für Datentyp, 239
 - gültige Werte, 239, 241
- ICON-Feld
 - vorderer Steuerbereich, 277
 - zulässige Werte, 207
- icon-Steuerungstyp, 287
- Identifikationsverzeichnis, 121
- Identifikationsüberprüfungsverzeichnis, 127
- identifizierende Merkmale
 - Dialogfenster, 191
- inactiveFrameColorId, Ressource, 330
- /include, 94
- include (Anweisung), in Workspace
 - Manager-Dateien , 302

- indirect Option, 7
- inetd.conf Datei, 126
- Installation
 - Verzeichnisstruktur, 94
- Internationalisierung
 - app-defaults, 341
 - Fehlerbehebung, 345
 - LANG (Variable), 337
 - NLS—Umgebungsvariablen, 340
 - Schriftarten, 340
 - Sprache einstellen, 338
- IS_TEXT-Feld, 242

K

- Kalender-Dämon, 128
- keyBindings (Ressource), 313
- Konfigurationsdateien, 95
 - Aktion, 201
 - Anmeldemanager, 20
 - Datentypen, 235
 - Registrierungspaket, 62
 - Session Manager, 37
 - Verzeichnisposition, 94
 - Vorderer Steuerbereich, 264
 - Window Manager, 300
 - Workspace Manager, 300

L

- %L, 148
- LABEL-Aktionsfeld, 207
- LANG (Variable), 337
 - in .dtprofile, 339
- LANG Variable
 - Auswirkung auf Datentypen, 250
- LANG-Variable
 - Vom Anmeldemanager gesetzt, 17
- /lib, 94
- LINK_TO_ACTION-Feld, 242
- local-Anzeigetyp, 5
- %LocalHost%, 225
- localTerminal-Ressource, 221
- LOCKED-Feld, 267
- .login, 41
- .login (Datei), 25
- Login Server
 - abbrechen, 97

- Start beim Hochfahren aktivieren, 96
- Start beim Hochfahren deaktivieren, 96
- Start von Befehlszeile, 97
- .login (Datei)
 - kein Zugriff durch Login Manager, 33
 - ausführen, 28
- LOGNAME-Variable, 18
- Lokalisierung
 - Aktionen, 229
 - Aktionsbezeichnungen, 230
 - Anmeldeanzeige, 13
- LOW_COLOR, 334
- lp Befehl, 122
- lp Druck-Spooler, 122
- LPDEST Variable, 143
- Länderspezifische Anpassung, 343
 - Meldungskataloge, 344
 - Palettennamen, 343
 - Symbole, 343, 342

M

- %M, 148
- mail-Steuerungstyp, 287
- mailx Befehl, 122
- /man, 94
- MAP-Aktionen
 - Beispiel, 204
 - Definition, 200
- Maustastenbelegung, 309
 - hinzufügen, 310
 - neue Sets definieren, 311
 - Syntax, 309
- MEDIA-Feld, 242
- Mehrere Anzeigen
 - Anmeldemanager, 14
- Mehrere Bildschirme, 100
- Meldungskatalog, 340
- Menüs
 - Verwendung von Aktionen, 166
 - Workspace Manager, 306
- MIME_TYPE_MEDIA-Feld, 242
- mkfontdir (Befehl), Dateien kompilieren, 340
- MODE Feld
 - Syntax, 246
- MODE-Feld, 243
- MONITOR_TYPE-Feld, 287

Index-360

Solaris Common Desktop Environment: Benutzerhandbuch für Fortgeschrittene und Systemverwalter ♦ September 1997

MOVE_TO_ACTION-Feld, 242

N

Name des Datentypentextfelds, 190

Namensmuster von Datentypen, 192

NAME_PATTERN-Feld, 243

National Language Support (NLS)
Einstellungen, 337

Netzwerk

allgemeine Schritte für die
Konfiguration, 119

Basiskonfiguration, 119

Clients und Server konfigurieren, 123

erforderliche Dateien, 124

Servicetypen, 114

Übersicht, 114

NFS, 120

NICHT Operator im MODE Feld, 247

NLS-Umgebungsvariablen, 340

NLS—Fernausführung, 344

NoBackdrop (Dialogfeldoption), 305

NoPrint Aktion, 178

NO_STDIO- Fensterunterstützung, 220

NUMBER_OF_ROWS-Feld, 292

NumFonts, Ressource, 323

O

ODER Operator im MODE Feld, 247

-once (Option), 106

OpenWindows

Umgebung, 96

Optionen (Menü)

Sprache, 104

OSF/Motif Window Manager, , *siehe*
Workspace Manager,

P

Paletten, 328

länderspezifische Namen, 343

/palettes, 94

PANEL-Definition, 268

Syntax, 269

PANEL_GEOMETRY-Feld, 294

PATH-Variable, 212

Vom Anmeldemanager gesetzt, 18

PATH_PATTERN Feld

Syntax, 245

PATH_PATTERN-Feld, 243

PERM_TERMINAL- Fensterunterstützung, 220

Persönliche Aktion und Datentypen

erstellen, 237

Persönliche Anwendungsgruppe, 51

Persönliche Anwendungsgruppen, 54

Persönliche Gruppe von Anwendungen, 51

Pfad

Benutzer, 18

System, 19

Pixmaps

Dateien finden, 254

Namenskonventionen, 255

Suchpfad, 256

Platzhalterzeichen *, 243

Platzhalterzeichen ?, 243

Platzhalterzeichen in Datentypen, 243

.pm Dateinamenerweiterung, 254

Positionen

Datentyp, 250

Positionen, Symbole, 42

POSITION_HINTS-Feld, 275

Post ausdrucken, 107

primaryColorSetId Ressource, 330, 332

Probleme, Desktop-Start, 39

.profile, 41

.profile (Datei), 25

.profile (Datei)

kein Zugriff durch Login Manager, 33

ausführen, 28

Protokoll, XDM, 103

Prüfung, Anmeldung, 39

PUSH_ACTION-Feld, 285

PUSH_ANIMATION-Feld, 290

Q

-query Option, 6

R

Readme-Dateien, 80

Registrierung, 78

allgemeine Schritte, 63

Anwendungsgruppe, 74

Anwendungsstammverzeichnis, 66

- Beispiel, 84, 93
 - Definition, 48, 63
 - dtappintegrate, 81
 - erforderliche Aktionen, 69
 - erforderliche Datentypen, 69
 - Farbe ändern, 65
 - Funktionen, 60
 - Hilfdateien, 72
 - Ressourcen ändern, 64
 - Schriftart ändern, 65
 - Symbole, 73
 - Zweck, 62
 - Änderung der Ressourcen, 64
 - Übersicht, 60
 - Registrierungspaket, 48, 78
 - Anwendungsgruppeninhalt, 78
 - Anwendungssymbol, 78
 - Bedienfeld-Element, 80
 - Beispiel zur Erstellung, 84
 - Definition, 62
 - Readme-Dateien, 80
 - Verzeichnisse, 66
 - Zweck, 60
 - RESOURCE_MANAGER (Eigenschaft), 30, 33
 - Ressourcen
 - bildschirmspezifische, 33
 - colorUse, 333
 - einstellen, 33, 315
 - foregroundColor, 335
 - laden, 29
 - shadowPixmaps, 334
 - Sitzung, 25
 - sprachabhängige, 341
 - Standarddesktop, 30
 - Standardwerte, 316
 - Ressourcen neu laden (Aktion), 30
 - RGB Farbwert, 329
 - rgb.txt Datei, 329
 - Root-Fenster, 305
 - Root-Menü, , *siehe* Arbeitsbereich (Menü),
 - rpc.cmsd, 128
 - rpc.ttdbserver, 124, 126
- S**
- Schreibgeschützte Datentypkriterien, 247
 - Schriftarten
 - Anzahl im Style-Manager, 323
 - Benutzer, im Style-Manager, 323
 - Bitmap, 322
 - Eigenschaftszeichenfolge angeben, 324
 - mit Befehl mkfontdir finden, 340
 - primäres Verzeichnis, 340
 - Ressourcen für die Registrierung ändern, 65
 - System, im Style-Manager, 323
 - X Logical Font Description, 324
 - xlsfonts (Befehl), 340
 - über Aliasdatei finden, 340
 - über Verzeichnisdatei finden, 340
 - Schriftartenpfad, 104
 - .sdl-Dateien, 72
 - sdtcm_convert (Skript), 108
 - secondaryColorSetId Ressource, 330, 332
 - selectColor Ressource, 329
 - sendmail, Funktion, 122
 - Server
 - Aktionen, 131
 - Anmeldung, 123
 - Anwendung, 115, 124
 - Datei, 115
 - Datenbank, 124
 - Datentypen, 131
 - Definition, 114
 - Hilfe, 118, 124
 - konfigurieren, 123
 - Session, 123
 - Server, 115
 - Symbol, 118, 124
 - Typen, 118
 - Session Manager, 25
 - Anwendungen erfassen, 28
 - Anwendungen starten, 31
 - Anwendungsstart konfigurieren, 32
 - Befehle beim Abmelden ausführen, 36
 - Begrüßungsmeldung, 25
 - Client, 29
 - Dateien, 37
 - Einführung, 23
 - Fehlerbehebung, 37
 - Fehlerprotokoll, 37, 40
 - Ressourcen laden, 29
 - Sitzungen speichern, 36
 - starten, 25
 - Suchpfade einrichten, 27

- systemweite Anpassung, 26
- Verzeichnisse, 37
- Workspace Manager starten, 31
- zusätzliche Befehle ausführen, 36
- Session-Server, *siehe* Anmelde-Server,
- sessionetc (Datei), 36
- sessionexit (Datei), 36
- %SessionHost%, 225
- SessionHost-Schlüsselwort, 225
- shadowPixmaps (Ressource), 31
- shadowPixmaps Ressource, 334
- /share, 94
- Shell
 - Aktionen, 216
 - individuelle Anpassung, 26
 - .profile und .login ausführen, 28
 - Siehe auch Umgebungsvariablen, .profile, .login, .dtprofile, 33
 - Syntax in Ausführungszeichenfolge, 212
 - systemweite Anpassung, 26
 - Verwendung in Aktion, 216
- SHELL-Variable, vom Anmeldemanager gesetzt, 18
- Sitzung
 - aktuelle, 24
 - Backup, 36
 - Befehle beim Abmelden ausführen, 36
 - Befehle beim Start ausführen, 36
 - bildschirmpezifische, 24, 35
 - Definition, 23
 - erste Anmeldung, 24, 34
 - Failsafe, 17
 - Fehlerprotokoll, 37
 - Home, 24
 - Ressourcen, 25
 - Skript beim Beenden ausführen, 17
 - Standard, 24
 - starten, 25
 - wiederherstellen, 26
- Sitzungsverzeichnis, 35
- Solaris CDE
 - Desktop abbrechen, 97
- SPC, 127
 - Sicherheit, 127
- Sprache (Menüeintrag), 104
- Sprache einstellen über Datei Xconfig, 338
- Sprachen-Menü, anpassen, 15
- Sprachtreiber für Anmeldung, 104
- Sprachtreiber, C, 105
- Standarddateien
 - Desktopanwendungen, 316
- Startfehlerprotokoll, 40
- startlog (Datei), 37
- Steuerung
 - Ablagezone, 286
 - animieren, 290
 - Client, 288
 - Datei öffnen, 285
 - Datei überwachen, 287
 - definieren, 283
 - Einzelinstanz, 288
 - Elementhilfe, 291
 - entfernen aus vorderem Steuerbereich, 274
 - Erscheinungsbild, 277
 - ersetzen, 276
 - erstellen, 284
 - Hauptsteuerbereich hinzufügen, 273
 - Klicken vs. Doppelklicken, 295
 - Position wechseln, 275
 - Schalter, 288
 - sperrern, 267
 - Symbol, 277
 - Typen, 283
 - wiederherstellen, 267
 - ändern, 275
- Steuerung 'Symbol installieren', entfernen, 282
- Steuerungen
 - beschriften, 295
 - Wechselbereich, 292
- Steuerungen beschriften, 295
- Steuerungsdaemon für Unterprozesse., *siehe* dtspcd,
- stty, 41
- Style-Manager
 - Farben angeben, 329
 - Farbintegration, 65
 - Schriftartenintegration, 65
- SUBPANEL-Definition, 269
- Suchpfad
 - Anwendungen, 44
- Suchpfade
 - Aktionen, 201
 - aktueller Wert, 147
 - Anwendungen, 149

- Ausgabevariable, 146
 - Definitionen des vorderen Steuerbereichs, 264
- Desktop, 145
- Eingabevariable, 146
- Hilfe, 159
- Position, 161
- Session Manager, 27
- Symbole, 256
 - Umgebungsvariable, 147, 166
 - Wert setzen, 147
- SWITCH-Definition, 271
- Symbol-Server, 118
 - Client, 132
 - erstellen, 132
 - Konfigurieren, 124
 - konfigurieren, 131
- Symbole
 - Aktionen, 196, 207
 - Aktionssymbole, 205
 - Anwendung starten, 73
 - Anwendungen, 78
 - Anwendungsgruppen, 73, 76
 - Bedienfeld, 258
 - Dateien finden, 254
 - Dateiformat, 254
 - Dateimanager zum Anzeigen verwenden, 259
 - Dateiname ohne Pfad, 239
 - Dateinamen, 254
 - Datentyp, 73
 - Datentypen, 196, 239
 - Druckerabbilder, 141
 - einem Anwendungsfenster zuordnen, 259
 - einer Aktion oder einem Datentyp zuordnen, 257
 - Empfehlungen zur Erstellung, 260
 - erforderlich für Registrierung, 73
 - Farbpalette, 260
 - für Anwendungen, 168
 - Gruppe suchen Dialogfenster, 196
 - Größenkonventionen, 255
 - integriert mit dtappintegrate, 83
 - länderspezifische, 343, 342
 - mit dem Dateimanager anzeigen, 259
 - Namenskonvention, 254
 - nicht-englische, 343, 342
 - Vorderer Steuerbereich, 277
 - Zuordnungen, 256
 - Symbole mit dem Dateimanager anzeigen, 259
 - Symbole, *siehe* Symbolserver
 - Server,,
 - Symbole, *siehe* Symbolsuchpfad
 - Suchpfad,,
 - Symbolgrößen, 255
 - Symbolische Verbindungen
 - Dateinamenkonsistenz, 122
 - Symbolische Verbundungen
 - Datentypkriterium, 243
 - Symbolische Verknüpfungen
 - Registrierung, 82
 - Symbolpositionen, 42
 - Symbolsuchpfad, 156
 - assemblieren, 158
 - Auswirkung des
 - Anwendungssuchpfads, 152
 - bezogen auf Anwendungssuchpfad, 157
 - Standard, 156
 - Syntax, 158
 - Umgebungsvariable, 157
 - sys.dtpofile (Datei), 26
 - sys.dtwmrc (Datei), 300, 301
 - sys.resources (Datei), 30, 33
 - sys.resources Datei, 316
 - sys.session (Datei), 32, 34
 - systemPath-Ressource, 19
- T**
 - Tastaturbelegung
 - neue Sets definieren, 313
 - Standard, 312
 - Syntax, 312
 - Tastaturbelegung zurücksetzen, 345
 - Tastenbelegung, Maus, *siehe*
 - Maustastenbelegung,
 - TERMINAL- Fensterunterstützung, 220
 - Terminal-Emulation, 42
 - Aktionen, 219
 - Terminal-Emulator
 - automatisches Schließen, 187
 - Befehlszeilenoptionen für Aktionen, 220
 - manuelles Schließen, 187
 - Standard-Emulator für Aktionen, 220
 - ändern, 55

- Text -Editor, ändern, 55
- textColorSetId Ressource, 330
- timeZone-Ressource, 19
- title (Ressource), 304
- ToolTalk
 - Anwendungen, Aktionen für, 230
 - Nachrichtendämon, 25, 29
- ToolTalk Meldungs-Server, *siehe* tsession,
- ToolTalk, *siehe* rpc.ttdbserver
 - Datenbank-Server,,
- ToolTalk, *siehe* TT_MSG-Aktion
 - Aktion,
- topShadowColor Ressource, 329
- tset, 41
- ttsession, 128
 - starten, 29
- tty-Einstellungen, 42
- ttyModes, 42
- TT_MSG-Aktion
 - erstellen, 230
 - Schlüsselwörter, 231
- TYPE-Feld, 283
- TZ-Variable, 18, 19

U

- UID, 120
- Umgebungsdatei, Desktop, 110
- Umgebungsvariable
 - Anwendungssuchpfad, 150
 - Bitmapsuchpfad, 157
 - Datenbanksuchpfad, 154
 - ferne Ausführung, 128
 - Hilfesuchpfad, 160
 - Pixmapsuchpfad, 157
 - Suchpfad, 147, 166
 - Symbolsuchpfad, 157
- Umgebungsvariablen
 - Aktionsdefinitionen, 227
 - Anmeldemanager, 19
 - einstellen, 32
 - in .dtprofile, 26
 - exportieren, 32
 - in .dtprofile, 26
 - .login und .profile ausführen, 28
 - persönliche, 32
 - Standard, 26
 - systemweite, 32

- UND Operator im MODE Feld, 247
- UNIX Tastenbelegung, 315
- Unterprozeßsteuerungs-Service, , *siehe* SPC,
- user-prefs.dt Datei, 57
- USER-Variable, 18
- userPath-Ressource, 18
- /usr/dt, 94

V

- /var/dt, 95
- Variablen
 - Aktionsdefinitionen, 226
 - Umgebung, *Siehe*
 - Umgebungsvariablen, 32
- Verbindung, Datentypkriterien, 247
- Verwendung von XTerminals, 103
- Verzeichnis
 - Datentypkriterien, 247
- Vorderen Steuerbereich
 - wiederherstellen-Aktion, 267
- Vorderer Steuerbereich, 295
 - Animation, 290
 - anpassen, 263
 - Aufbau der Definition, 268
 - Bildschirmposition, 294
 - Client, 288
 - dynamische Anpassungen, 265, 266
 - Hilfe, 291
 - Komponenten, 268
 - Konfigurationsdateien, 264
 - Menü, 306
 - Namenskonvention für Dateien, 264
 - neu erstellen, 295
 - persönliche Anpassungen
 - kontrollieren, 267
 - Steuerung als Ablagezone, 286
 - Suchpfad, 264
 - Syntax, 269
 - Verwaltung mit Workspace Manager, 300
 - Vorrang bei Konfiguration, 265
 - Zeilen hinzufügen, 296
 - ändern, 272
- Vorderer Steuerbereich, *siehe* Steuerung
 - Steuerung,,
- Vorrang
 - Aktionsdatenbank, 209

Konfiguration des vorderen Steuerbereichs, 265

W

Wechselbereich

anpassen, 292

Anzahl der Arbeitsbereiche, 292

Anzahl der Zeilen, 292

Definition, 269

Steuerungen hinzufügen, 292

Syntax der Definition, 271

Wechseln, *siehe* Arbeitsbereiche,

Window Manager, 300

windowMenu (Ressource), 306

WINDOW_TYPE- Feld, 220

wmStartupCommand (Ressource), 31

WM_CLASS Eigenschaft, 259

Workspace (Menü)

Definition, 306

Workspace Manager

Definition, 300

Funktionen, 307

individuelle Anpassung, 301

Konfigurationsdateien, 300

Maustastenbelegung ändern, 309

Menüs, 306

neu starten, 302

starten, 25, 31

systemweite Anpassung, 301

vorderen Steuerbereich verwalten, 300

wechseln OSF/Motif, 314

weitere Dateien einbinden, 301

Workstations, als XTerminals, 105

writeXrdbColors (Ressource), 31

wscon console log file, 40

X

X -Berechtigung, 122

X -Stationen, 123

X Ressourcen, *siehe* Ressourcen,

X-Berechtigung, 122

X-Netzwerk

X -Berechtigung, 122

X-Server

Umgebung ändern, 15

Zugriffsrechte ändern, 14

X-Stations

Session-Services, 116

X-Terminals, 105

CHOOSER-Zeichenfolge, 8

direkter XDMCP-Modus, 8

indirekter XDMCP-Modus, 8

mögliche Konfiguration des

Anmeldeservers, 3

ohne XDMCP-Unterstützung, 7

Xaccess-Liste, 8

XDMCP, 6

X11-Server, 96

X400_TYPE-Feld, 242

Xaccess-Datei, 7

XAUTHORITY-Variable, vom

Anmeldemanager gesetzt, 17

Xconfig, 95, 100

Xconfig-Datei

Ressourcen festlegen, 13

ändern, 3

?Xconfig (Datei)

Sprache einstellen, 338

.Xdefaults, 42

.Xdefaults Datei, 316

.Xdefaults (Datei), 30

XDM, 96, 103, 107

Protokoll, 103

XDMCP, 2

Abfragemodus, 6

Definition, 6

direkte Anforderungen, 6

Direktzugriff, 8

Indirekte Anforderungen, 6

indirekte Anforderungen, 20

indirekter Zugriff, 8

Xerrors-Datei, 9

Xfailsafe-Datei, 17, 18, 20

xlsfonts (Befehl)

Installation, 340

Schriftarten auf Server auflisten, 340

xlsfonts Befehl, 323

XMICONBMSEARCHPATH Variable

assemblieren, 158

Definition, 147

Verwendung, 158

XMICONSEARCHPATH Variable

assemblieren, 158

- XMICONSEARCHPATH variable
 - definition, 147
- XMICONSEARCHPATH Variable
 - Verwendung, 158
- XmText*FontList-Ressource, 323
- Xpid-Datei, 3
- Xreset-Datei, 17
- Xresources-Datei, 11, 12
- Xserver, 106
 - abbrechen, 97
- Xservers (Datei), 96
- Xservers-Datei
 - lokale Anzeigen verwalten, 20
 - Server starten, 4
 - Standard, 5
 - Syntax, 4
- Xsession
 - Skript, 39
- Xsession (Datei), 26
 - Session Manager starten, 25
 - systemweite Anpassung, 26
- Xsession-Datei
 - ausführen durch Anmeldeserver, 16
 - PATH festlegen, 18
- Xsession.d (Verzeichnis), 25, 32
 - anpassen, 26
 - Skripts, 26
- Xsetup-Datei, 16
- Xstartup-Datei, 16
- xterm, 42
 - starten, 99
- XTerminal
 - verwenden, 103

- XTerminals, 102, 105
 - Workstations, 105
- XUSERFILESEARCHPATH (Variable), 341

Z

- Zeichenfolge-Aktionsargumente, 215
- Zeichenfolgevariablen in
 - Aktionsdefinitionen, 226
- Zeichenkonsole, 6
- Zeitzone, ändern, 19
- Zentrale Konfigurationsdateien, 95
- Zuordnungsaktionen, 176
- Zusätzlicher Steuerbereich
 - anpassen integrierter Bereiche, 279
 - Container, 271
 - erstellen, 278
 - gelöschten Bereich wiederherstellen, 267
 - Hauptsteuerbereich zuordnen, 278
 - neu, 279
 - Syntax, 271
 - systemweite Anpassungen, 279
 - Ändern der Selbstschließfunktion, 282
 - ändern, 278
- Zusätzlicher Steuerbereich
 - Definition, 271
 - Syntax, 271

Special Characters

- Öffnen Aktion, 176
- Überwachungstypen für Steuerungen, 287