

# ToolManager

---

En utvidelse til Amiga Workbench

Version 2.1  
16 Mai 1993

---

Opphavsrett © 1990-93 Stefan Becker

Tillatelse gis til å distribuere intakte kopier av denne manualen så fremt opphavsrettnotisen og denne tillatelsesnotisen følger med alle kopier.

Det er ingen som helst garanti at programmene som er beskrevet i dette dokumentet er 100% pålitelige. Du bruker dette materialet på egen risiko. Forfatteren **kan ikke** bli gjort skyldig i skader som er forårsaket av disse programmene.

Denne pakken kan distribueres fritt, men opphavsretten innehas av Stefan Becker. Dette betyr at du kan kopiere den fritt så lenge du ikke tar mer penger for den en et rimelig kopieringsbeløp. Dette beløpet **må ikke** overstige USD 5 or DEM 5.

**Denne grensen gjelder også tyske Public-Domain forhandlere!!**

Tillatelse gis til å inkludere pakken i Public-Domain samlinger, spesielt Fred Fishs Amiga Disk bibliotek (inkludert CD ROM versjoner). Distribusjonsfilen kan lastes opp på Bulletin Board systemer eller FTP servere. Hvis du ønsker å distribuere dette programmet, **må** du bruke de originale distribusjonsarkivene 'ToolManager2\_1bin.lha', 'ToolManager2\_1gfx.lha' og 'ToolManager2\_1src.lha'.

Ingen av disse programmene eller kildekoden (samt deler av den) kan bli inkludert eller brukt i kommersielle programmer uten forfatterens skriftlige tillatelse.

**Ingen** av disse programmene **eller** kildekoden (samt deler av den) kan bli brukt på maskiner som blir brukt til forskning, utvikling, bygging, testing eller produksjon av våpen eller andre militære områder. Dette gjelder også maskiner brukt til opplæring av personer for **alle** de nevnte formålene.

# 1 Viktige notiser

Velkommen til den vidunderlige ToolManager 2.1 :-)

- ToolManager og dets konsepter har forandret seg drastisk (se Tillegg B [Historie], side 30) siden versjon 1.5.
- Fra og med versjon 2.0 av ToolManager er dette programmet *GiftWare*. Hvis du liker og bruker programmet ofte, bør du sende en liten donasjon for å honorere arbeidet forfatteren har gjort med dette programmet. Jeg foreslår en donasjon på US \$10-\$20 eller 10-20 DM. Ikke send sjekker eller kontanter fra utlandet, da disse ofte koster mer å veksle enn de er verdt.  
Hvis du ikke sender donasjonen eller ikke har råd, trenger du ikke å føle deg slem for det. Vær bare snill og send meg et kort som sier at du bruker ToolManager. (Jeg liker fan mail :-). Se kapittel 2 [Forfatterens adresse], side 2.
- Brukere av ToolManager 1.X/2.0 kan starte med kapitlet for hurtiginstallasjon (se kapittel 3 [Rask installasjon], side 3). Noen saker har ikke forandret seg og resten er ganske lett å finne ut selv med prøv-og-feil metoden. For en detaljert beskrivelse av de nye konseptene kan du lese referanse delen av dette dokumentet (se kapittel 8 [Objekter], side 12).  
Du **må** fjerne en evt. kjørende ToolManager 1.X/2.0 ellers vil ikke den nye versjonen virke. Dette fordi den nye versjonen ikke kan lese de gamle 1.X konfigurasjons filene (Beklager).
- Førstegangsbrukere bør lese hele manualen for å forstå konseptet og formålet til programmet. Start med kapittel 4 [Introduksjon], side 4.
- ToolManager 2.1 bruker noen av AmigaOS Versjon V38s (og høyere) muligheter. Den støtter også nettverksmulighetene i det nye AmigaOS'et som forhåpentlig- vis vil være tilgjengelig snart for all Amigabrukere. Hvis du fremdeles bruker versjon 2.0 (heretter V37) trenger du ikke bekymre deg siden ToolManager ikke er avhengig av disse mulighetene. Alle ekstra muligheter er merket i dette dokumentet.

## 2 Hvor sender du feilrapporter, kommentarer & donasjoner

Forfatteren kan nåes på følgende adresser:

Postadresse:

Stefan Becker  
Holsteinstrasse 9  
D-W-5100 Aachen  
GERMANY

Vennligst bruk følgende adresse etter 1. juli 93:

Stefan Becker  
Holsteinstrasse 9  
D-W-52068 Aachen  
GERMANY

Internet elektronisk post:

`stefanb@pool.informatik.rwth-aachen.de`

## 3 Rask installasjon av ToolManager 2.1

Installasjon av ToolManager 2.1 består av fire deler:

`'Libs/toolmanager.library'` ⇒ `'LIBS:'`

Dette er hovedprogrammet til ToolManager. Det håndterer alle program, menyer, ikoner og dokker (se kapittel 10 [Bibliotek], side 23).

`'Prefs/ToolManager*'`  ⇒ `'SYS:Prefs'`

Dette er programmet for innstillinger (se kapittel 9 [Innstillinger], side 19).

`'WBStartup/ToolManager*'`  ⇒ `'SYS:WBStartup'`

Med dette verktøyet kan du starte og stoppe ToolManager. Hvis det ligger i WBStartup katalogen vil ToolManager alltid bli startet når maskinen din starter opp.

`'L/WBStart-Handler'` ⇒ `'L:'`

Dette programmer starter programmer med Workbench sin start metode. Det er en separat prosess, så du kan avslutte ToolManager selv om du har programmer kjørende, startet med WB metoden.

Etter å ha kopiert disse filene, bør du avslutte alle eldre versjoner av ToolManager og dobbelklikke ToolManager ikonet som er i `'WBStartup'` katalogen. Nå kan du starte innstillingsprogrammet og prøve deg frem. (Bruk "Test"-knappen istedet for "Bruk"-knappen mens du tester.) Du burde kunne finne ut av det meste med prøving og feiling, men nærmere informasjon finnes i ToolManagers objektbeskrivelser (se kapittel 8 [Objekter], side 12).

Distribusjonen inkluderer et eksempel på en konfigurasjonsfil, nemlig `'TM_Demo.prefs'`. Du kan hente den inn i innstillingsprogrammet med menyvalget **Hente**.

## 4 Hva er ToolManager?

ToolManager er et fleksibelt program for å holde styr på programmer i ditt arbeidsmiljø. Det kan starte Workbench og CLI programmer, ARexx skript og generere start-tast sekvenser. Det kan til og med sende kommandoer til en ToolManager som kjører på en annen maskin. Brukergrensesnittet består av menyer, ikoner eller dokk-vinduer. Hvis du liker en støyende maskin, kan du koble en lyd til hvilke som helst objekt Se avsnitt 8.3 [Lyd], side 14.

ToolManager kan legge til program til Workbenchs **Tools** meny. Hvis du velger et slikt menyvalg vil programmet som assosieres med det, startes. Alle valgte ikoner på Workbenchen vil bli brukt som argument til programmet. Dette er bare tilgjengelig hvis Workbench kjører. Se avsnitt 8.4 [Meny], side 14.

ToolManager kan legge til ikoner på Workbench vinduet. Når du dobbel-klikker et slikt ikon, vil programmet som det er tilknyttet startes. Hvis du dropper andre ikoner over dette ikonet, vil programmet starte med disse ikonene som argument. Dette er bare tilgjengelig hvis Workbench kjører. Se avsnitt 8.5 [Ikon], side 15.

ToolManager kan lage et såkalt dokkvindu av en samling programmer. Dette vinduet kan åpnes på hver offentlige skjerm. Hvert program er representert av et bilde eller en knapp. For å starte et program trenger du bare klikke på bildet eller knappen. Hvis dokkvinduet er åpnet på Workbench skjermen og Workbench kjører, kan du også droppe ikoner på bildet eller knappen for å starte programmet med argumenter. Se avsnitt 8.6 [Dokk], side 15.

I tillegg kan du legge til en start-tast til hvert program. Hvis du trykker denne start-tast, vil programmet starte. Merk deg at *ingen* argumenter kan bli sendt til programmet hvis du bruker denne start metoden. Se avsnitt 8.1 [Program], side 12.

## 5 Konseptene bak ToolManager

ToolManager 2.1 bruker en ny objekt-orientert vinkling for å gi et fleksibelt og ekspanderbart system. Denne vinklingen gjorde det mulig å forbedre ToolManager over 1.X versjoner, f.eks.: flere dokkvinduer åpne samtidig.

Et objekt er en samling av data som beskriver dets muligheter. Hvert objekt har et navn og en type. Du kan skape så mange objekter som du ønsker av hver type, men navnet på objektene må være ulike, fordi navnet blir brukt som referanse til objektet.

For tiden er det syv forskjellige objekt typer: Program, Bilde, Lyd, Meny, Ikon, Dokk og Aksess. De første tre er primitive objekt; det betyr at de ikke refererer til andre objekt. De supplerer data eller tjenester til de andre, komplekse objektene.

De siste fire objektene er komplekse objekt; det betyr at de refererer til primitive objekt og er avhengig av dem for med hensyn til data eller tjenester. Referansene gjøres med navn, og hvis ingen primitive objekt med dette navnet eksisterer så vil det komplekse objektet overse det. Merk deg at dette kan redusere funksjonaliteten til det komplekse objektet, f.eks et ikonobjekt *trenger* data fra et bildeobjekt, så hvis ikke dette objektet finnes vil det ikke bli laget noe ikon.

For en detaljert beskrivelse av alle objektene, see kapittel 8 [Objekter], side 12.

## 6 En tur gjennom ToolManagers muligheter

Har du ikke forstått et ord inntil nå? Forvirret av objekter, programmer og lenker? Ikke fortvil, hjelp er på vei.

Jeg vil lede deg gjennom et steg-for-steg eksempel om hvordan man skal stille inn ToolManager. Alt du trenger å gjøre er å installere ToolManager og kjøre innstillingsprogrammet. Etter hvert steg, trykk “Test”-knappen i hovedvinduet for å teste konfigurasjonen.

Som et eksempel bruker vi tekst fremviseren More som ligger i katalogen ‘SYS:Utilities’. Først må vi fortelle ToolManager hvilke program vi ønsker å bruke. Informasjon om programmer er lagret i programobjekter. Bare velg “Program” som objekttype i hovedvinduet til innstillingsprogrammet og trykk “Nytt”-knappen.

Etter å ha trykket denne knappen vil du se “Endre programobjekt”-vinduet. Åpne Utilities-katalogen på Workbench partisjonen, flytt More-ikonet ut av katalogen og dropp det på editorvinduet. Som du kan se har editoren satt navnet på objektet, kommandoen til programnavnet og den aktuelle katalogen til System:Utilities. Trykk ned “OK”-knappen for å bruke denne innstilling.

Du kan ikke gjøre så mye med et programobjekt alene, så i neste omgang ønsker vi å legge dette programmet i “Verktøy”-menyen til Workbench. Velg “Meny” som objekttype og trykk “Nytt” knappen. Nå vil du se “Endre menyobjekt”-vinduet. Forandre objektets navn til “Vis Tekst”.

ToolManager må vite hvilket program det skal starte når menyvalget gjøres, så vi lenker bare et programobjekt til menyobjektet. Trykk ned “Programobjekt”-knappen og velg objektet “More” fra fil-velgeren. Nå kan du trykke “OK”-knappen og “Test” knappen i hovedvinduet. Du kan nå se ditt menyobjekt i “Verktøy”-menyen. Velg en tekstfil på Workbench-skjermen (klikk et ikon) og velg ditt nye menyvalg. Programmet “More” skal nå starte og vise frem tekstfilen du valgte. Dette var vel lett?

Nå kan vi gå et skritt videre og lage et ikonobjekt på Workbench-skjermen. Til et ikon trenger vi billedata som er lagret i et bildeobjekt. Velg “Bilde” som objekttype og trykk “Nytt” knappen. “Endre bildeobjekt”-vinduet vil nå åpnes. Forandre navnet til “Bilde for More” og dropp More ikonet fra Utilities-katalogen i vinduet. Trykk ned “OK”-knappen for å bruke innstillingene.

I det neste steget vil vi skape ikonobjektet. Velg “Ikon” som objekttype og trykk “Nytt”-knappen. Forandre objektnavnet til “Vis Tekst”. Trykk ned “Programobjekt”-knappen og velg objektet “More” fra fil-velgeren. Trykk ned “Bildeobjekt”-knappen og velg objektet “Bilde for

More” fra filvelgeren. Sett X posisjonen til 100 og Y posisjonen til 50. Trykk ned “OK”- knappen og “Test”-knappen. Etter en kort pause vil et ikon dukke opp på Workbench-skjermen. På denne kan du nå droppe ikonene til tekstfilene dine for å vise dem.

Jeg er sikker på at du nå har en viss peiling hvordan du kan bruke ToolManager objekter og på hvilken måte du kan lenke dem sammen for å bygge ditt miljø. Nå kan du finne ut resten ved å prøve en ting av gangen. Du kan også ta en titt på filen ‘`TM_Demo.prefs`’ for å finne ut mer om div. innstillinger.

## 7 Beskrivelse av alle filene i distribusjonen

Den komplette ToolManager 2.1 distribusjonen består av flere kataloger som er forklart nedenfor. Merk at distribusjonen er delt opp i tre deler, så du har kanskje ikke alle katalogene.

### 7.1 Katalogen Docs

Denne katalogen inneholder dokumentasjonen til ToolManager. Dokumentasjonen er tilgjengelig i fire ulike format og flere språk. I tillegg er det en fil i AutoDoc format som beskriver ToolManagers delte bibliotekgrensesnitt.

Prefix `'TM_<language>'`

Disse filene inneholder denne filen for det spesifiserte språket. Følgende språk er tilgjengelig: Deutsch, English, Français, Norsk, Svenska.

Postfix `' .doc'`

Denne filen inneholder dokumentasjonen i ren ASCII tekst.

Postfix `' .dvi'`

Denne filen inneholder dokumentasjonen i T<sub>E</sub>Xs DVI format. For å få en utskrift av manualen kjører du denne filen gjennom en T<sub>E</sub>X printerdriver.

Postfix `' .guide'`

Denne filen inneholder dokumentasjonen i AmigaGuide format. Selv om det er ren ASCII med noen kommandoer, trenger du AmigaGuide for å utnytte hypertextlenkene i fila.

Postfix `' .tex'`

Denne filen inneholder dokumentasjonen i Texinfo format, som spesifisert av Free Software Foundation (FSF). Sammen med makropakken `'texinfo.tex'` kan du bruke T<sub>E</sub>X og `'texindex'` for å lage en fil i DVI format (se ovenfor).

`'toolmanager.doc'`

Denne filen inneholder en beskrivelse av ToolManagers delte bibliotek grensesnitt i AutoDoc format.

### 7.2 Katalogen Goodies

Denne katalogen inneholder ekstra programpakker som er nyttig for ToolManager.

‘GetPubName.lha’

Dette lille programmet skriver ut navnet på den offentlige skjermen som er i front, enten til stdout eller til en miljøvariabel. Den er laget av Michael ”Mick” Hohmann.

‘upd1\_20.lha’

Programmet `upd` er laget av Jonas Petersson. Det er et lite program som åpner en ARexx port og venter på kommandoer. Via ARexx kan du ordre `upd` til å spille av samplede filer. ToolManager bruker denne funksjonen til å implementere lydobjekter. Se avsnitt 8.3 [Lyd], side 14.

## 7.3 Katalogen Graphics

Denne katalogen inneholder en rik samling av bilder som du kan bruke. Bare last dem inn som bildeobjekter i ToolManager (se avsnitt 8.2 [Bilde], side 13).

Filene ble donert av forskjellige personer (se Tillegg C [Takk til], side 31). Hver av dem fikk en separat underkatalog i distribusjonen. Etersom filene ble laget av forskjellige personer har de også forskjellig stil, palett, størrelse, dybde og oppløsning. Det er ikke sikkert alle bildene ser bra ut på din maskin.

For å skjelne mellom de forskjellige bildeformatene som er understøttet av ToolManager, så har hver fil en beskrivende suffiks.

- ‘.anmb’ Dette er en IFF ANIM fil laget av et tegne/animasjons program. Det kan inneholde flere bilder. Selv om ToolManager kan laste inn komplette ANIM filer, må du bruke noe sånt som DPaints “AnimBrush” funksjon for å hente ut de interessante delene av animasjonen.
- ‘.brush’ Dette er en IFF ILBM fil laget av et tegneprogram. Den inneholder bare et bilde.
- ‘.info’ Dette er et normalt Amiga ikon laget av IconEdit (eller noe lignende). Den kan inneholde et eller to bilder.

## 7.4 Katalogen L

Denne katalogen inneholder bare en fil, nemlig ‘WBStart-Handler’. Du *må* kopiere denne filen til katalogen ‘L:’, ellers vil ikke ToolManager klare å starte noen programobjekt med WB (se avsnitt 8.1 [Program], side 12).

Den komplette pakken WBStart 1,2 finnes på Fish disk nr 757.

## 7.5 Katalogen Libs

Denne katalogen inneholder bare en fil, 'toolmanager.library'. Dette er hoved programmet til ToolManager og det må kopieres til 'LIBS:' katalogen.

## 7.6 Katalogen Locale

Denne katalogen inneholder alle filene for ToolManagers Locale support. Ettersom locale.library er nytt med V38, trenger du ikke kopiere disse filene hvis du bruker V37. Hvis du bruker V38, velg filene for ditt språk og kopier dem til de respektive plassene.

'Catalogs/<språk>/toolmanager.catalog'

Dette er oversettingsfilen for det spesifiserte språket. Kopier filen for ditt språk til katalogen 'LOCALE:Catalogs/<språk>'.

'Languages/<språk>.language'

Noen språk er ikke støttet av den standardiserte V38 Locale distribusjonen. Så noen av oversetterene har donert en '.language' fil, slik at ToolManager kan bruke deres oversettingsfiler. Kopier filen for ditt språk til katalogen 'LOCALE:Languages'. Ekstra språk er: Finnish (suomi), Eefeler Platt (eifel).

## 7.7 Katalogen Prefs

ToolManagers innstillingsprogram og dets ikon er i denne katalogen. Kopier begge filene til katalogen 'SYS:Prefs'. For nærmere informasjon om programmet se kapittel 9 [Innstillinger], side 19.

## 7.8 Katalogen Programmers

Denne katalogen inneholder alle filer som trengs for de forskjellige maskinspråkene og deres kompilatorer for å bruke ToolManagers delte bibliotekgrensesnitt.

Ta en titt på underkatalogen `examples` for noen eksempler hvordan du kan bruke dette grensesnittet. For en komplett grensesnitt beskrivelse, les fila `Docs/toolmanager.doc`.

Språk/kompilatorer som støttes ved dette tidspunkt: AmigaOberon, DICE C, M2Amiga Modula-2, MANX Aztec C og SAS C.

## 7.9 Katalogen Scripts

Denne katalogen inneholder en samling ARexx og Shell skript som kan brukes i ToolManagers programobjekt. Merk at de kan være spesielt tilpasset for forfatterens miljø, så du må kanskje forandre dem.

## 7.10 Katalogen Source

Denne katalogen inneholder komplett kildekode til ToolManager 2.1 og dets verktøy. Hvert program har sin egen underkatalog. Forfatteren gir denne kildekoden som et eksempel for programmering under OS 2.x/3.0.

`locale` underkatalogen er av interesse for oversettere. Hvis ditt språk ikke er understøttet i denne versjonen og du ønsker å oversette, ta en titt på fila `empty.ct`. Bare fyll inn de tomme linjene og send fila til meg. Kanskje blir den inkludert i den neste versjonen.

## 7.11 Katalogen WBStartup

Det er bare et program i denne katalogen: `ToolManager`. Dette verktøyet starter og stopper ToolManager 2.1. For det meste av tiden vil denne fila holde til i `SYS:WBStartup` katalogen, men den kan brukes fra Shell også.

## 8 ToolManager objektreferanse

Ette kapitell beskriver ToolManager objektene in detalj. Hvert objekt har en type og et navn. Navnet refererer til et objekt. Det er seks forskjellige typer objekt:

### 8.1 Programobjekter

Et programobjekt beskriver program eller hendelser som er startet av ToolManager. Tre ulike typer program støttes: CLI, Workbench and ARexx. Programobjekt tar følgende argumenter: (Standardverdier i parantes.)

#### Argument (Ja)

Dette flagget kontrollerer om argument skal sendes til programmet eller ikke. Om programmet ikke støtter eller trenger argument, kan du sette dette flagget til 'Nei'.

**Kommando** Navnet til programmt eller hendelsen som skal startes. Dette navnet kan være relativt til den aktuelle katalogen. Hvis typen er dokk, vil kommandoen beskrive navnet til dokkobjektet, som burde bli åpnet/lukket. For typen start-tast må denne strengen være Commodities Input Description String (se kapittel 11 [Start-taster], side 24). En fjernkommando (nettverksobjekt) er beskrevet som `objekt@maskein`. Denne forteller ToolManager som kjører på `machine` å aktivisere programobjektet `objekt`.

#### Aktuell katalog ('SYS:')

Navnet på programmets aktuelle katalog. Merk: ARexx program overser denne parameteren.

**Pause (0)** Etter å ha aktivert et programobjekt vil ToolManager vente **Pause** sek. før den starter programmet. Hvis denne verdien er negativ, vil programmet bli startet hvert **Pause** sek. For å stoppe et programobjekt som venter på å kjøre, er det bare å aktivere det på nytt. Merk: Hvis **Pause** er satt, vil programmet startes uten argumenter.

#### Programtype (CLI)

Spesifiserer typen til programmet eller hendelsen. Kan være en av: CLI, WB, ARexx, dokk, start-tast eller nettverk.

#### Start-tast

Du kan spesifisere en start-tast for hvert programobjekt. Hvis denne start-tast hendelsen genereres, vil programmet bli startet. Merk: Programmet vil bli startet uten argumenter.

#### Utdatafil ('NIL:')

Filnavn til utdatafilen. Kan bare brukes med CLI program.

**Sti** (sti fra ToolManager prosess)

Denne strengen setter programmets søkesti. Du kan spesifisere flere kataloger ved separere dem med et “;”. Fungerer bare i sammenheng med CLI-program.

**Prioritet** (0)

Setter prioriteten til den nye prosessen som kjører programmet.

**Offentlig skjerm** (standard offentlige skjerm)

Setter navnet på den allmenne skjermen som skal flyttes til front før programmet starter. Fungerer bare i sammenheng med **Til front** parameteren.

**Stakk** (4096)

Setter stakkstørrelsen til den nye prosessen som kjører programmet.

**Til front** (Nei)

Hvis du setter dette flagget til “Ja” så flyttes den offentlige skjermen spesifisert i **Offentlig skjerm** til front før programmet starter.

## 8.2 Bildeobjekter

Bildeobjekter spesifiserer bildedataene som brukes av ToolManager for ikoner eller dokker. Dette objektet tar bare et parameter:

**Filnavn** Spesifiserer navnet på fila som ToolManager skal lese bildet fra. ToolManager prøver å finne ut hvilken type fil det er:

1. ToolManager forsøker å lese den som en IFF fil. ToolManager kan for øyeblikket lese ILBM (et bilde) eller ANIM (to eller flere bilder) filer.
2. ToolManager forsøker å lese den som en ikon fil. Et ikon kan inneholde et eller to bilder.

Animasjoner støttes for øyeblikket bare av dokkobjekter. Ikonobjekter henter bare de to første bildene fra animasjonen for å lage et ikon. Hvis du ønsker å lage en animasjon for ToolManager, bør du følge disse retningslinjene:

**Bilde 1** Dette bør være et bilde som viser ikonets inaktive status.

**Bilde 2** Dette bør være et bilde som viser ikonets aktive status. Normalt er dette en reversert kopi av det første bildet.

**Bilde 3 to N-1**

Dette er bildene for selve animasjonen. Hvert bilde vises i ca. 1/3 sekund.

Bilde N Det siste bildet i animasjonen vises i et sekund. Etter dette vil det første bildet vises igjen.

### 8.3 Lydobjekter

Et lydobjekt kan brukes til å gjøre ToolManager støyende. ToolManager selv har ikke noe mulighet til å spille av lyder, den bruker ARexx til å aktivere en ekstern lydavspiller. Dette objektet tar to parametere:

**Kommando** Denne spesifiserer hvilken ARexx kommando som ToolManager sender for å aktivisere den eksterne lydavspilleren. For `upd` skulle dette bli noe sånt som `file samples:boing` som forteller `upd` til å spille av IFF samplingen `'samples:boing'`. Se avsnitt 7.2 [Goodies], side 8.

**ARexx Port**

Denne spesifiserer ARexx porten hvor ToolManager skal sende `command`. Standard er `PLAY` som er porten for programmet `upd`.

### 8.4 Menyobjekter

Menyobjekter kontrollerer alternativene i Workbenchs "Verktøy"-meny. Objektnavnet blir brukt som menytekst. For å aktivisere et slikt objekt er det bare å velge menyalternativet. Merk: Virker bare hvis Workbench kjører.

Dette objektet tar to parametre:

**Programobjekt**

Navnet på et programobjekt som skal aktiviseres når meny alternativet er valgt. Alle ikoner som er valgt på dette tidspunkt vil bli brukt som argumenter til programmet.

**Lydobjekt**

Navnet på et lydobjekt som skal aktiviseres når menyalternativet er valgt.

Merknad til ToolManager 1.X brukere: For å simulere den gamle verktøystypen "Dummy" er det bare å lage et menyobjekt og *ikke* spesifisere noe program- eller lydobjekt.

## 8.5 Ikonobjekter

Ikonobjekter beskriver applikasjonsikoner i Workbench-vinduet. Et slikt objekt kan aktiviseres ved å dobbeltklippe det eller å slippe noen ikoner på applikasjonsikonet. Ikonobjekter virker bare når Workbench'en er i gang.

Parametrene for denne objekttypen er følgende:

### Programobjekt

Dette er navnet til et programobjekt som skal startes når ikonet blir valgt. Alle ikoner som blir sluppet oppå applikasjonsikonet vil bli brukt som argumenter til programmet.

### Bildeobjekt

Dette er navnet til et bildeobjekt. Bildedataene til dette objektet blir brukt til å lage applikasjonsikonet.

### Venstre kant (standardverdi: 0)

Dette setter den venstre kanten til applikasjonsikonet.

### Vis navn (standardverdi: Ja)

Hvis denne parameteren er satt vil navnet til programobjektet bli brukt som navnet til applikasjonsikonet

### Lydobjekt

Dette er navnet til et lydobjekt som skal aktiviseres når ikonet blir valgt.

### Øverste kant (standardverdi: 0)

Dette setter den øverste kanten til applikasjonsikonet.

NB: Workbench er *vel*dig streng på plasseringen av ikoner. Hvis du spesifiserer koordinater som den ikke liker, vil den ignorere dem og plassere ikonet et annet sted på skjermen.

## 8.6 Dokkobjekter

Dokkobjekter beskriver vinduer. Disse vinduene kombinerer flere verktøy som er representert med bilder eller trykk-knapper. For å starte et slikt verktøy er det bare å trykke på bildet eller trykk-knappen. Du kan selvsagt også slippe ikoner på bildet eller trykk-knappen for å bruke dem som argumenter til verktøyet.

Hver dokk har flere parametre. Standardverdiene vises i parantes:

**Aktivisert (Ja)**

En dokk kan være aktiv (åpen) eller ikke (stengt).

**I bakgrunnen (Nei)**

Dette forteller dokk-vinduet at det ikke skal komme fremst av vinduene når det åpnes.

**Sentrert (Nei)**

Hvis denne parameteren er satt vil vinduet alltid bli sentrert til den nåværende mus-posisjonen når det åpnes.

**Kolonner (1)**

Denne parameteren setter antall verktøykolonner. Verktøy blir alltid sortert rekkevis, fra venstre kolonne mot kolonnen lengst til høyre.

**Skrifttype (Skjermens skrifttype)**

Hvis du har en dokk med parameteren **Text** satt, kan du velge skrifttypen til trykkknappene med denne parameteren.

**Fremst (Nei)**

Hvis du setter denne parameteren vil dokk-vinduet alltid åpnes på den offentlige skjermen som er lengst fram.

**Start-tast**

Du kan sette en start-tast for dokk-objektet. Hvis denne start-tasten trykkes vil aktiveringsstatusen til dokk-vinduet bli byttet om; dvs det vil bli åpnet eller lukket.

**Venstre kant (0)**

Denne parameteren setter venstrekanten til dokk-vinduet. Hvis parameteren **Sentrert** er satt, vil denne parameteren bli ignorert.

**Meny (Nei)**

Du kan ha en liten meny til hvert dokk-vindu. Denne menuen har to valg:

**Steng dokk** Stenger dokk-vinduet.

**Avslutt TM** Avslutter ToolManager

**Mønster (Nei)**

Dokk-vinduet justerer seg selv til det største bilde i det. Hvert bilde i dokken har samme størrelse, og mindre bilder sentreres samt at de får en blank ramme rundt dem. Hvis de ikke liker denne blanke rammen kan du sette denne parameteren. Dermed vil rammen bli fylt med et mønster istedet.

**Autosteng (Nei)**

Når denne parameteren er satt vil dokken automatisk bli stengt etter at et objekt i den er valgt. Dette er spesielt nyttig sammen med parametrene **Sentrert**, **Fremst** og en start-tast av typen **rawmouse** (se kapittel 11 [Start-taster], side 24).

**Offentlig skjerm** (Standard offentlig skjerm)

Spesifiserer hvilken offentlig skjerm som dokken skal åpnes på. Hvis dokken er ble åpnet v.h.a. en startt-tast vil den offentlige skjermen bli flyttet fremst etter at dokk-vinduet har blitt åpnet. Denne parameteren vil bli ignorert hvis parameteren **Fremst** er satt.

**Fast plass** (Nei)

Vanligvis lagrer dokk-vinduet sin siste posisjon når du stenger det og kommer på den plassen når du åpner dokken igjen. Hvis du vil at dokken alltid skal komme på samme posisjon må du sette denne parameteren.

**Tekst** (Nei)

Du kan velge mellom bilder og trykk-knapper i dokk-vinduer med denne parameteren. Trykk-knapper er spesielt nyttige når de blir brukt sammen med parameteren **Autosteng**.

**Tittel** Dette spesifiserer tittelen til dokk-vinduet. Hvis du skriver inn en tittel vil dokken bli et vanlig OS 2.0 vindu med dragbar, stengeknapp og dybdeknapp og ramme. Hvis du *ikke* spesifiserer en tittel vil du få et dokk-vindu med bare dragbar og *ingen* ramme.

**Øverste kant** (0)

Denne parameteren setter øverste kant til dokk-vinduet. Hvis parameteren **Centered** er satt vil denne parameteren bli ignorert.

**Loddrett** (Nei)

Hvis dokk-vinduet bare har en dragbar og ingen ramme, setter denne parameteren retningen til dragbar'en. Denne parameteren blir ignorert hvis du har skrevet inn en tittel til vinduet med parameteren **Tittel**.

## 8.7 Aksessobjekter

Aksessobjekter kontrollerer aksessrettighetene for nettverksforespørsler. Standard er at *alle* forespørsler blir nektet, slik at en ToolManager utenfor maskinen ikke skal ødelegge operasjonene på din maskin ved å aktivisere noen av dine programobjekter. Med aksessobjekter kan du la spesifiserte maskiner aktivisere noen av dine programobjekter.

Navnet til et aksessobjekt har en spesiell mening. Det blir sammenlignet med navnet til fjernmaskinen som sendte netverksforespørselen. ToolManager bruker følgende tre trinn for denne sammenligningen:

1. Sammenligne med det komplette vertsnamnet
2. Sammenligne med "realm"-navnet

### 3. Se etter aksessobjektet med navn **anyone**

Hvis et korresponderende objekt blir funnet så blir dette objektet brukt til aksessrettighetene til fjernmaskinen. Objektet med navn **anyone** blir brukt for alle nettverksforespørsler som ikke har et korresponderende aksessobjekt.

Aksessobjektets har bare en parameter:

#### Programobjekt

Denne parameteren kan bli brukt flere ganger. Den spesifiserer hvilke programobjekter som kan bli aktivisert fra fjernmaskinen. Hvis du ikke spesifiserer *noen* objektnavn kan fjernmaskinen aktivisere *alle* programobjekter på din maskin.

## 9 ToolManager innstillinger

Med programmet for innstillinger kan du handtere den komplette konfigurasjonen av ToolManager. Denne konfigurasjonen hentes automatisk når du starter ToolManager. Programmet for innstillinger startes automatisk ved å dobbelklikke på dets ikon. Du vil da få fram hovedvinduet.

Det fleste av knappene i innstillingsvinduene kan velges fra tastaturet ved å trykke en bestemt "kortvalgstast". Denne tasten er underlinjert (\_). Merk at hvis du holder på å skrive inn en tekststreng, må du først trykke tasten RETURN før du kan bruke kortvalgstaster.

### 9.1 Knappene i hovedvinduet

Hovedvinduet har flere objektgrupper:

#### Objekttype

Med denne knappen kan du velge hvilken objekttype som du ønsker å lage eller endre.

#### Objektliste

Denne listen viser alle objektene av en type. Du kan velge et objekt ved å klikke på navnet. Hvis du dobbeltklikker på navnet vil et endringsvindu bli åpnet.

#### Flytt objekt

Når et objekt er valgt kan du flytte det opp og ned i listen med disse knappene. Hvis du klikker på **Sorter**-knappen vil navnene i listen bli sortert alfabetisk.

#### Behandle objekt

Disse knappene behandler objekter. **Nytt** lager et nytt objekt av objekttypen som er valgt. Når du klikker på **Endre** vil det bli åpnet et endringsvindu for det valgte objektet. Med **Kopier** kan du lage en kopi av det valgte objektet. **Slett** sletter det valgte objektet.

#### Konfigurasjon

Du har flere valgmuligheter for å lagre konfigurasjonen. Med **Lagre** kan du lagre konfigurasjonen permanent på fila 'ENVARC:ToolManager.prefs'. For å kun bruke innstillingene midlertidig må du bruke knappen **Bruk**. Denne vil lagre konfigurasjonen på fila 'ENV:ToolManager.prefs'. Denne fila overlever ikke en reset av maskinen. For å teste konfigurasjonen uten å forlate programmet: Bruk **Test**-knappen. **Avbryt** går ut av programmet uten å lagre noe som helst.

## 9.2 Menyene til hovedvinduet

Hovedvinduet har flere menyvalg:

**Prosjekt** Med menyvalget **Hente** og **Lagre** som kan du hente og lagre konfigurasjonen. Menyvalget **Info** åpner et informasjonsvindu. Menyvalget **Avslutt** vil gå ut av programmet uten å lagre.

**Endre** Med disse menyvalgene gjenopprette gamle konfigurasjonen. **Sist lagret** henter inn den sist lagrede konfigurasjonen fra fila 'ENVARC:ToolManager.prefs'. Med menyvalget **Gjenopprett** kan du hente inn den konfigurasjonen som var gjeldende før du startet opp programmet. Denne hentes fra fila 'ENV:ToolManager.prefs'.

### Innstillinger

Med menyvalget **Lage ikoner** kan du velge om menyvalget **Lagre** som skal lage et ikon for konfigurasjonsfila eller ikke.

## 9.3 Vinduet for å lage objekter

Hvis du slipper et ikon oppå hovedvinduet vil vinduet "Lage objekter" åpnes. Her kan du velge hva slags type objekt som skal lages fra dette ikonet. Dette kan brukes til å legge til et program til din konfigurasjone på en enkel og rask måte.

Du kan bare lage et programobjekt eller et bildeobjekt fra ikonet hvis du velger en av de to første valgmulighetene. Men du kan også lage en komplett meny og/eller ikonobjekt hvis du velger en av de tre siste valgmulighetene.

## 9.4 Endringsvinduer

Hver objekttype har forskjellige endringsvinduer for å sette objektets parametre. For en detaljert liste over alle objektparametre; se kapittel 8 [Objekter], side 12.

Alle endringsvinduer har et felt for objektnavnet. Dette navnet er viktig siden det blir brukt som referanse til objektet. Merk at det ikke er innebygget noen kryssreferanse. Det betyr at hvis du forandrer navnet til et objekt som blir referert til fra et annet, vil denne referanse *ikke* bli oppdatert. Du må updatere denne referansen manuelt.

Trykk-knappene i endringsvindue åpner forskjellige typer vinduer. Du kan velge en ting ved å klikke på det og trykke OK, eller du kan dobbeltklikke den. For å gå ut av vinduet uten endringer, trykk Avbryt-knappen. Hvis du vil renske et felt som bare kan velges via et vindu, åpne vinduet og trykk OK-knappen *uten* å velge noe.

Endringsvinduet for objekttypene program og bilde har en ekstra mulighet: Du kan ganske enkelt slippe et ikon på dem for å sette parametrene fra dette ikonet.

## 9.5 Verktøytyper

Når du starter innstillingsprogrammet fra Workbench har du mulighet til å sette flere verktøytyper i programikonet eller i konfigurasjonsfilas ikon for å kontrollere programmet.

**USE** Hvis du setter denne verktøytypen i et ikon for en innstillingsfil vil innstillingsprogrammet installere denne fila som nåværende konfigurasjon.

**SAVE** Hvis du setter denne verktøytypen i et ikon for en innstillingsfil vil innstillingsprogrammet installere denne fila som nåværende og permanent konfigurasjon.

### PUBSCREEN

Denne verktøytypen forteller innstillingsprogrammet at det skal startes på den spesifiserte offentlige skjermen. Hvis du ikke setter denne verktøytypen vil nåværende offentlige skjerm bli brukt.

### CREATEICONS

Når denne verktøytypen er satt til YES vil editoren lage et ikon for hver innstillingsfil som blir laget med menyvalget **Lagre som**.

### DEFAULTFONT

Innstillingsprogrammet bruker vanligvis skrifttypen til den offentlige skjermen til å tegne trykk-knappene. Hvis denne verktøytypen er satt til YES vil den vanlige systemskrifttypen bli brukt istedet.

**XPOS** Denne spesifiserer startposisjonen i x-retning til innstillingsprogrammet.

**YPOS** Denne spesifiserer startposisjonen i y-retning til innstillingsprogrammet.

### MINLISTCOLUMNS

Denne spesifiserer det minste antall kolonner i listeoversiktene til innstillingsprogrammet.

### MINLISTROWS

Denne spesifiserer det minste antall kolonner i listeoversiktene til innstillingsprogrammet.

## 9.6 CLI argumenter

Når programmet for innstillinger startes fra et shell bruker det følgende syntaks for kommandolinjeargumenter:

```
FROM, EDIT/S, USE/S, SAVE/S, PUBSCREEN/K, DEFAULTFONT/S
```

**FROM** Denne parameteren spesifiserer navnet til innstillingsfila som innstillingsprogrammet skal hente inn.

**USE** Hvis du setter denne parameteren vil innstillingsprogrammet installere fila spesifisert ved FROM-parameteren som nåværende konfigurasjon.

**SAVE** Hvis du setter denne parameteren vil innstillingsprogrammet installere fila spesifisert ved FROM-parameteren som nåværende og permanent konfigurasjon.

### **PUBSCREEN**

Denne parameteren forteller innstillingsprogrammet at det skal startes på den spesifiserte offentlige skjermen. Hvis du ikke setter denne verktøytypen vil nåværende offentlige skjerm bli brukt.

### **DEFAULTFONT**

Innstillingsprogrammet bruker vanligvis skrifttypen til den offentlige skjermen til å tegne trykk-knappene. Hvis denne parameteren brukes, vil innstillingsprogrammet bruke den vanlige systemskrifttypen istedet.

## 10 ToolManagers bibliotekgrensesnitt

ToolManagers “handler” er bygget inn i et Amiga bibliotek (“shared library”). Dette biblioteket tilbyr flere forskjellige funksjoner for å lage og manipulere ToolManager objekter, slik at du kan bruke dem i dine egne programmer.

Det er for øyeblikket seks tilgjengelige funksjoner:

### `AllocTMHandle()`

For å lage ToolManager objekter må du først allokere en `TMHandle`. Denne handleren lagrer all informasjon om dine objekter og blir brukt som referanse til dem. Merk at informasjonen som er lagret i denne handleren *bare* er tilgjengelig fra programmet som lager den.

### `FreeTMHandle()`

Denne funksjonen frigjør en `TMHandle` og alle ToolManager objekter som er assosiert via den. Hver `AllocTMHandle()` må ha en `FreeTMHandle()`!

### `CreateTMObjectTags()`

### `CreateTMObjectTagList()`

Denne funksjonen lager et ToolManager objekt. Du må supportere et navn, objekttypen and forskjellige “tags” for objektparametrene. Navnet til objektet er viktig siden det blir brukt som referanse til objektet.

### `ChangeTMObjectTags()`

### `ChangeTMObjectTagList()`

Du kan modifisere parametrene til et ToolManager objekt med denne funksjonen. Objekttilstanden vil bli oppdatert til de nye parametrene. NB: Bildeobjekter kan ikke modifiseres (ennå).

### `DeleteTMObject()`

Med denne funksjonen kan du slette et ToolManager objekt. Hvis objektet er linket til andre objekter til de fortalt om endringen og deres tilstand vil bli endret.

### `QuitToolManager()`

Denne funksjonen forteller ToolManagers handler at den skal slutte så snart som mulig.

Det komplette biblioteksgrensesnittet er tilgjengelig i AutoDoc format. (se avsnitt 7.1 [Docs], side 8).

## 11 Hvordan definere en start-tast?

Dette kapitlet beskriver hvordan man definerer en start-tast som en inndata beskrivelsesstreng (“Input Description String”) som så blir parset av Commodities. Hver gang en start-tast blir aktivisert lager Commodities en hendelse som blir brukt av ToolManager til å starte programobjekter eller stenge/åpne dokkobjekter. En beskrivelsesstreng har følgende syntaks:

```
[<klasse>] {[<->][<kvalifikator>]} [-][opptrykk] [<tastkode>]
```

Store og små bokstaver skilles ikke i nøkkelordene.

**klasse** beskriver en InputEvent klasse. Denne parameteren er valgfri, og hvis den ikke oppgis brukes standardverdien **rawkey**. Se avsnitt 11.1 [InputEvent klasser], side 24.

**kvalifikatorer** er “signaler” som må/ikke må være satt når start-tasten blir aktivisert, ellers vil det ikke bli generert noen hendelse. For hver kvalifikator som må være satt må du ha med dette nøkkelordet. Alle andre som ikke oppgis blir sett på som ikke satt. Hvis du vil ignorere en kvalifikator; sett bare - foran dens nøkkelord. Se avsnitt 11.2 [Kvalifikatorer], side 25.

Normalt vil en start-tast hendelse bli generert når en tast trykkes ned. Hvis hendelsen skulle bli generert når tasten slippes opp må du legge til nøkkelordet **opptrykk**. Hvis det skal genereres en hendelse både når tasten trykkes ned og slippes opp, bruk: **-opptrykk**.

**tastkode** avhenger av klassen til InputEvent. Se avsnitt 11.3 [Tastkoder], side 26.

Merk: Velg start-startene dine *nøye* fordi Commodities har en høy prioritet i InputEvent handler kjeden. (Dvs: Den vil overta definisjoner som allerede brukes.)

### 11.1 InputEvent klasser

Commodities støtter de fleste InputEvent klasser som blir generert av input.device. Denne seksjonen beskriver de klassene som er mest nyttige å bruke som start-taster i ToolManager.

**rawkey** Dette er standardklassen og den dekker alle tastaturhendelser. For eksempel **rawkey a** eller **a** lager en hendelse hver gang tasten “a” trykkes. Du må spesifisere en tastkode for denne klassen. Se avsnitt 11.3.1 [rawkey tastkoder], side 26.

**rawmouse** Denne klassen beskriver alle hendelsene som kan lages med museknappene. Du må spesifisere en tastkode for denne klassen. Se avsnitt 11.3.2 [rawmouse tastkoder], side 27.

**diskinserted**

Hendelser i denne klassen genereres når en disk blir satt inn i en diskettstasjon. Denne klassen har ingen tastkoder.

**diskremoved**

Hendelser i denne klassen genereres når en disk blir satt inn i en diskettstasjon. Denne klassen har ingen tastkoder.

## 11.2 Kvalifikatorer

Noen nøkkelord ble lagt til i Commodities V38. Disse er merket med en \*.

**lshift, left\_shift \***

Venstre shift-tast.

**rshift, right\_shift \***

Høyre shift-tast.

**shift** Hvilken som helst shift-tast.

**capslock, caps\_lock \***

Tasten "Caps Lock".

**caps** Enten "Caps Lock" eller shift.

**control, ctrl \***

Kontroll-tasten ("Ctrl").

**lalt, left\_alt \***

Venstre alt-tast.

**ralt, right\_alt \***

Høyre alt-tast.

**alt** Hvilken som helst alt-tast.

**lcommand, lamiga \*, left\_amiga \*, left\_command \***

Venstre Amiga/kommando-tast.

**rcommand, ramiga \*, right\_amiga \*, right\_command \***

Høyre Amiga/kommando-tast.

**numericpad, numpad \*, num\_pad \*, numeric\_pad \***

Dette nøkkelordet *må* brukes for alle tastene på talltastaturet.

`leftbutton`, `lbutton *`, `left_button *`

Venstre museknapp. Se merknad under.

`midbutton`, `mbutton *`, `middlebutton *`, `middle_button *`

Midtre museknapp. Se merknad under.

`rbutton`, `rightbutton *`, `right_button *`

Høyre museknapp. Se merknad under.

`repeat` Denne kvalifikatoren er satt når tastaturrepetisjon er aktivt. Bare nyttig for `InputEvent`-klassen `rawkey`.

Merk: Commodities V37 har en feil som forhindrer at `leftbutton`, `midbutton` og `rbutton` kan brukes som kvalifikatorer. Denne feilen er rettet opp i V38.

## 11.3 Tastkoder

Hver `InputEvent` klasse har sine egne tastkoder:

### 11.3.1 Tastkoder for `InputEvent`-klassen `rawkey`

Noen nøkkelord ble lagt til i Commodities V38. Disse er merket med en `*`.

`a-z`, `0-9`, ...

ASCII tegn.

`f1`, `f2`, ..., `f10`, `f11 *`, `f12 *`

Funksjonstaster.

`up`, `cursor_up *`, `down`, `cursor_down *`

`left`, `cursor_left *`, `right`, `cursor_right *`

Kursortaster.

`esc`, `escape *`, `backspace`, `del`, `help`

`tab`, `comma`, `return`, `space`, `spacebar *`

Spesialtaster.

`enter`, `insert *`, `delete *`

`page_up *`, `page_down *`, `home *`, `end *`

Taster på talltastaturet. Sammen med hver av disse må kvalifikatoren `numericpad` brukes!

### 11.3.2 Tastkoder for InputEvent-klassen rawmouse

Disse nøkkelordene ble lagt til i V38 av Commodities. De finnes ikke i V37.

`mouse_leftpress`

Trykk venstre musetast.

`mouse_middlepress`

Trykk midtre musetast.

`mouse_rightpress`

Trykk høyre musetast.

Merk: For å kunne bruke en av disse tastkodene må du også sette den korresponderende kvalifikatoren. For eksempel:

```
rawmouse leftbutton mouse_leftpress
```

## 11.4 Eksempler på start-taster

`ralt t` Hold ned høyre Alt-tast og trykk "t"

`ralt lalt t`

Hold ned venstre og høyre Alt-tast og trykk "t"

`alt t` Hold ned hvilken som helst Alt-tast og trykk "t"

`rcommand f2`

Hold ned høyre Amiga-tast og trykk den andre funksjonstasten

`numericpad enter`

Trykk Enter på talltastaturet

`rawmouse midbutton leftbutton mouse_leftpress`

Hold ned midtre musetast og trykk venstre museknapp

`diskinserted`

Sett inn en disk i en diskettstasjon

## Tillegg A Spørsmål og svar om ToolManager

Her er svarene til de vanskeligste spørsmålene om ToolManager:

- Hvorfor kan ikke ToolManager lage flere “Tools”-menyer eller sub-menyer?  
Flere menyer eller sub-menyer er for tiden ikke støttet av systemprogramvaren. For å lage dem må du “hack’e” dem inn i systemet, noe som kan resultere i et ustabilt system. Jeg har ikke lyst til å lage ustabil programvare, så jeg vil ikke implementere slikt i ToolManager.
- WB programmer vil ikke starte, men alle andre programtypene virker.  
ToolManager bruker programmet `L:WBStart-Handler` for å starte WB-programmer. Det er to årsaker til at ToolManager ikke kan starte dette programmet:  
Fila ‘`L:WBStart-Handler`’ finnes ikke. Vennligst kopier den inn fra distribusjonsarkivet.  
Flagget “execute” (e) er ikke satt på fila. Bruk følgende kommando til å sette flagget:  
`protect L:WBStart-Handler +e`
- Hvordan lager jeg horisontale dokk-vinduer?  
Sett antall kolonner til antallet objekter i dokkobjektet.
- Hvordan lager jeg et utdatavindu for CLI programmer?  
Utdatavinduer lages med enheten `CON:.` Bruk følgende filnavn til å lage et vindu som åpner seg automatisk og har stengeknapp, og som ikke stenger seg med en gang programmet er ferdig:  
`CON:10/10/640/100/Utdata-vindu/AUTO/CLOSE/WAIT`  
`CON:-`enheten har mange opsjoner, se i din AmigaDOS manual for mer informasjon.
- Hvordan setter jeg inn argumenter i midten av en CLI/Arexx kommandolinje?  
Normalt legges alle argumenter til slutten av kommandolinja. For å sette inn argumenter hvor som helst i kommandolinja bruker ToolManager samme `[]` syntaks som det som brukes av kommandoen `alias` i AmigaShell. For eksempel:  
`Dir [] all`  
vil sette inn alle argumentene før nøkkelordet `all`.
- Hvordan sletter jeg en lenke fra et komplekst objekt til et enkelt objekt?  
Etter å ha valgt knappen “xxx objekt”, trykk bare “OK”-knappen *without* å velge et objekt. Dette betyr at du ikke velger noe objekt og derfor vil lenken bli slettet.
- Hvordan lager jeg sub-dokker?  
Du må bruke programobjekter av typen dokk. Legg slike objekter inn i hoveddokken og de vil dermed åpne/stenge andre dokker.
- ToolManager henger seg etter å ha startet en nettverkskommando.  
Det er et problem med netverksprogramvaren som mangler en lokal timeout forespørsel. Så hvis din maskin kalles `Vert1` og du har et programobjekt av typen nettnettverk, med kommandoen

Objekt0Vert1, vil ToolManager gå i deadlock når du aktiviseren den. Bruk kun navn til fjernmaskiner!

## Tillegg B ToolManagers historie

På ordre fra Stefan Becker skal dette kapitlet ikke oversettes!

### 2.1, Release date 16.05.1993

- New Exec object types: Dock, Hot Key, Network
- New Dock object flags: Backdrop, Sticky
- New object type: Access
- Network support
- Editor main window is now an AppWindow
- Gadget keyboard shortcuts in the preferences editor
- New tooltypes for the preferences editor
- Several bug fixes
- Enhanced documentation

### 2.0, Release date 26.09.1992, Fish Disk #752

- Complete new concept (object oriented)
- (Almost) Complete rewrite
- ToolManager is now split up into two parts
- Main handler is now embedded into a shared library
- Configuration is now handled by a Preferences program
- Configuration file format has changed again :-) It is an IFF File now and resides in ENV:
- Multiple Docks and multi-column Docks
- Docks with new window design
- Dock automatically detects largest image size
- Sound support
- Direct ARexx support for Exec objects
- ToolManager can be used without the Workbench. If the Workbench isn't running, it won't use any App\* features.
- Locale support
- Path from Workbench will be used for CLI tools
- Separate Handler Task for starting WB processes

### 1.0 upto 1.5

Please look into the english documentation for further information.

## Tillegg C Forfatteren vil takke...

ToolManager har gjennomgått flere utviklingsfaser siden den første implementasjonen midt i 1990. Dette utviklingen har ikke vært mulig hvis jeg ikke hadde fått en enorm tilbakemelding fra forskjellige brukere av ToolManager. Mange id'eer og muligheter har blitt lagt til på grunn av dette...

Derfor vil jeg takke:

For alfa/betatesting, id'eer og feilrapporter:

Amigaseksjonen av vår lokale computerklubb (Computerclub an der RWTH Aachen), Olaf 'Olsen' Barthel, Georg Hessmann (Gucky), Markus Illenseer (ill), Klaus Melchior, Rickard Olsson (Richie), Matthias Scheler (Tron), Ralph Schmidt (laire), Roger Westerlund (Budda), Juergen Weinelt, Brian Wright (SteveVai), Petra Zeidler (stargazer) og mange andre...

Matthew Dillon

Uten ditt **utmerkede** C utviklingssystem DICE og forskjellige andre verktøy ville ikke ToolManager eksistere!

For deres glimrende grafiske arbeid:

Andreas Harrenberg, Georg Hessmann, Michael "Mick" Hohmann, Markus Illenseer, Oliver Koenen, Klaus Melchior, Rickard Olsson, Jan Peter, Matthias Scheler, Brian Wright

For oversettelsene:

Tomi Blinnikka (suomi), Jorn Halonen (norsk), Dr. Peter Kittel (deutsch), Jasper Kehlet (dansk), Klaus Melchior (eifel), Rickard Olsson (svenska), Rullier Pascal (français), Marc Schaefer (français), Tor Rune Skoglund (norsk), Reinhard Spisser (italiano), Andrea Suatoni (italiano)

Alle gutter og jenter i West Chester:

For utviklingen av Amiga og det superbe operativsystemet.

Alle brukerne som sendte meg penger:

Jeg spurte ikke etter det i 1.X-versjonene, men det er hyggelig å se at noen setter pris på arbeidet mitt.

Alle brukerne som sendte meg et brev:

Jeg likte virkelig brevene deres!

og alle dem jeg glemte å nevne...

# Indeks

## A

Adresse .....	2
Aksessobjekter .....	17
AmigaGuide .....	8
ARexx skript .....	11
ASCII dokumentasjon .....	8

## B

Bibliotek dokumentasjon .....	8
Bibliotekgrensesnitt .....	23
Bildeobjekter .....	13

## C

Catalog filer .....	10
CLI argumenter .....	22

## D

DeleteTool .....	8
Diskinserted .....	24
Diskremoved .....	24
Distribusjonsfiler .....	8
Docs-katalogen .....	8
Dokkobjekter .....	15
Dokumentasjon .....	8
Donasjoner .....	2

## E

E-Mail .....	2
Eksempel .....	6
Eksempler på bilder .....	9
Eksempler på start-taster .....	27

## F

Feilrapporter .....	2
---------------------	---

## G

GetPubName .....	8
GiftWare .....	1
Goodies-katalogen .....	8

Graphics-katalogen .....	9
--------------------------	---

## H

Historie .....	30
----------------	----

## I

Ikonobjekter .....	15
Innstillinger .....	19
InputEvent klasser .....	24
Installasjon (rask) .....	3
Internet adresse .....	2
Introduksjon til start-taster .....	24
Introduksjon til ToolManager .....	4
Introduksjon til ToolManager objekter .....	5

## K

Kildekode .....	11
Kommentarer .....	2
Kompilatorstøtte .....	10
Konfigurasjon .....	19
Konsepter .....	5
Kvalifikatorer .....	25

## L

L-katalogen .....	9
Libs-katalogen .....	10
Liste: Kvalifikatorer .....	25
Liste: <b>rawkey</b> tastkoder .....	26
Liste: <b>rawmouse</b> tastkoder .....	27
Locale-katalogen .....	10
Lokalisering .....	10
Lyobjekter .....	14

## M

Menyobjekter .....	14
Muligheter .....	6

## O

Objekter .....	12
----------------	----

Opplæring .....	6	Språk .....	10
Oversettelser .....	10	Språkfiler .....	10
Oversettere .....	11	Start-taster .....	24
<b>P</b>		Svar .....	28
Postadresse .....	2	<b>T</b>	
Prefs-katalogen .....	10	Takk til .....	31
Programkonsepter .....	5	Tastkoder for <b>rawkey</b> .....	26
Programmers-katalogen .....	10	Tastkoder for <b>rawmouse</b> .....	27
Programobjekter .....	12	TEX .....	8
Programversjoner .....	30	Texinfo .....	8
<b>R</b>		Tildelte bilder .....	9
Rask installasjon .....	3	ToolManager objekter .....	12
Rawkey .....	24	ToolManagers bibliotekgrensesnitt .....	23
Rawmouse .....	24	Tur .....	6
Referanse: Bibliotekgrensesnitt .....	23	<b>U</b>	
Referanse: Distribusjonsfiler .....	8	UPD .....	8
Referanse: Innstillinger .....	19	<b>V</b>	
Referanse: Start-taster .....	24	V38 (og høyere) muligheter .....	1
Referanse: ToolManager objekter .....	12	Verktøytyper .....	21
<b>S</b>		Versjoner .....	30
Scripts-katalogen .....	11	Viktige notiser .....	1
Shell skript .....	11	<b>W</b>	
Skrevet dokumentasjon .....	8	WBStart 1.2 .....	9
Sound player .....	8	WBStart-Handler .....	9
Source-katalogen .....	11	WBStartup-katalogen .....	11
Spørsmål .....	28		

## Table of Contents

<b>1</b>	<b>Viktige notiser</b> .....	<b>1</b>
<b>2</b>	<b>Hvor sender du feilrapporter, kommentarer &amp; donasjoner</b> .....	<b>2</b>
<b>3</b>	<b>Rask installasjon av ToolManager 2.1</b> .....	<b>3</b>
<b>4</b>	<b>Hva er ToolManager?</b> .....	<b>4</b>
<b>5</b>	<b>Konseptene bak ToolManager</b> .....	<b>5</b>
<b>6</b>	<b>En tur gjennom ToolManagers muligheter</b> .....	<b>6</b>
<b>7</b>	<b>Beskrivelse av alle filene i distribusjonen</b> .....	<b>8</b>
	7.1 Katalogen Docs .....	8
	7.2 Katalogen Goodies .....	8
	7.3 Katalogen Graphics .....	9
	7.4 Katalogen L .....	9
	7.5 Katalogen Libs .....	10
	7.6 Katalogen Locale .....	10
	7.7 Katalogen Prefs .....	10
	7.8 Katalogen Programmers .....	10
	7.9 Katalogen Scripts .....	11
	7.10 Katalogen Source .....	11
	7.11 Katalogen WBStartup .....	11
<b>8</b>	<b>ToolManager objektreferanse</b> .....	<b>12</b>
	8.1 Programobjekter .....	12
	8.2 Bildeobjekter .....	13
	8.3 Lydobjekter .....	14
	8.4 Menyobjekter .....	14
	8.5 Ikonobjekter .....	15
	8.6 Dokkobjekter .....	15
	8.7 Aksessobjekter .....	17

<b>9</b>	<b>ToolManager innstillinger</b> .....	<b>19</b>
9.1	Knappene i hovedvinduet .....	19
9.2	Menyene til hovedvinduet .....	20
9.3	Vinduet for å lage objekter .....	20
9.4	Endringsvinduer .....	20
9.5	Verktøytyper .....	21
9.6	CLI argumenter .....	22
<b>10</b>	<b>ToolManagers bibliotekgrensesnitt</b> .....	<b>23</b>
<b>11</b>	<b>Hvordan definere en start-tast?</b> .....	<b>24</b>
11.1	InputEvent klasser .....	24
11.2	Kvalifikatorer .....	25
11.3	Tastkoder .....	26
11.3.1	Tastkoder for InputEvent-klassen <code>rawkey</code> .....	26
11.3.2	Tastkoder for InputEvent-klassen <code>rawmouse</code> .....	27
11.4	Eksempler på start-taster .....	27
<b>Tillegg A</b>	<b>Spørsmål og svar om ToolManager</b> .....	<b>28</b>
<b>Tillegg B</b>	<b>ToolManagers historie</b> .....	<b>30</b>
<b>Tillegg C</b>	<b>Forfatteren vil takke</b> .....	<b>31</b>
<b>Indeks</b> .....	<b>32</b>	