

ShellScr

Kyzer/CSG

COLLABORATORS

	<i>TITLE :</i> ShellScr		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Kyzer/CSG	January 19, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	ShellScr	1
1.1	ShellScr v1.6 dokumentation	1
1.2	Introduktion	1
1.3	Användning av ShellScr	2
1.4	ShellScr tillval	3
1.5	PUBNAME tillval	3
1.6	MODEID tillval	4
1.7	DEPTH tillval	4
1.8	FONT tillval	5
1.9	AUTOSCROLL tillval	5
1.10	SHANGHAI tillval	5
1.11	TITLE tillval	6
1.12	NOTITLE tillval	6
1.13	CONSPEX (WINDOW) tillval	6
1.14	COMMANDFILE (FROM) tillval	7
1.15	STACKSIZE tillval	7
1.16	ShellScr historia	7
1.17	Framtida utveckling av ShellScr	8
1.18	Källkodsnoteringar	8
1.19	Beräkning av fönsterstorleken	9
1.20	Ordlista över viktiga termer	9
1.21	Tillägnat	10

Chapter 1

ShellScr

1.1 ShellScr v1.6 dokumentation

ShellScr v1.6

- Introduktion
- Användning
- ShellScr tillval
- ShellScr historia
- Ordlista över viktiga termer
- Tillägnat

ShellScr är ett program som öppnar en maximalt stor shell på en allmän skärm.
Copyright (C) 1997-1998 Kyzer/CSG

Detta program är ett gratisprogram; du kan ge det vidare och/eller ändra på det så länge du följer termerna i GNU General Public License så som de är publicerade av Free Software Foundation; endera version 2 av licensen eller valfri senare version.

Det här programmet distribueras i hopp om att det kommer att vara användbart, men utan NÅGON SOM HELST GARANTI; utan ens en underförstådd garanti om dess FÖRSÄLJNINGSDUGLIGHET eller ANVÄNDBARHET FÖR NÅGOT ÄNDAMÅL.
Läs GNU General Public License för mer information.

Du borde ha fått en kopia av GNU General Public License med detta program, om inte, skriv då till Free Software Foundation Inc.,
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

Kontakta upphovsmannen

1.2 Introduktion

Varför inte använda en skärm för din shell istället för ett fönster på Workbench:en där den kan ställa till med oreda?

ShellScr är ett program som öppnar en ny allmän skärm och på den en shell som täcker hela skärmen. Efter att alla fönster på skärmen har stängts,

inkluderat shell:en själv, så stängs skärmen.

ShellScr kan även öppna en shell med en annan fil än den vanliga S:Shell-Startup, vilket betyder att den också kan användas som ett alternativ till IconX.

Användning av ShellScr.

1.3 Användning av ShellScr

Krav:

ShellScr kräver Workbench 2 (v37) eller bättre för att kunna köras. Om du vill använda andra teckensnitt än topaz så måste du ha diskfont.library i LIBS: katalogen.

De valfria ASL dialogfönstren som kan användas för att välja teckensnitt och skärmläge kräver åtminstone Workbench 2.1 version av asl.library (v38) eller en ersättning/patch som ReqTools och ReqPatch.

Distribution:

Var vänlig och distribuera hela arkivet som du fick det, med källkod, exempelikoner och denna guide fil. Tack!

Installering:

Kopiera ShellScr programmet dit du vill ha det, till exempel 'C:' katalogen eller 'Workbench:Utilities' katalogen. Det kan också vara en bra ide att kopiera denna guide till ett säkert ställe för framtida behov.

De medföljande exempelikonerna är endast ämnade som demonstrationer, om du vill använda en av dem så skall du ändra på ikonens 'standardverktyg' (default tool) så att den är det ställe vart du kopierade ShellScr (ändra det från '/ShellScr' till exempelvis 'C:ShellScr', eller endast 'ShellScr' om du har Workbench 3.)

Katalogen 'src' innehåller Amiga E källkoden till ShellScr. Den behövs inte för att köra ShellScr. Programmerare hänvisas till att läsa källkodsnoteringar.

Att starta ShellScr:

Du kan starta så många kopior av ShellScr som du vill. Den är 'pure' och kan göras minnesresident.

För att starta så behöver du bara dubbelklicka på en ShellScr ikon eller skriva "ShellScr" i en redan öppen shell. En ny shell kommer sedan att öppnas på en egen skärm. Denna skärm är nu den förvalda skärmen, och många program kommer att öppna sina fönster på denna skärm istället för på Workbench skärmen.

Om du skift-väljer en fil eller en katalog på Workbench:en, och efter det startar

ShellScr så kommer ShellScr att starta shell:en i den valda katalogen. Det här kommer också att fungera om du väljer en fil eller katalog och startar ShellScr från denna katalog med "ToolsDaemon", "ToolManager" eller liknande 'dock' program.

Shell:en kan stängas när som helst genom att använda sig av tangentkombinationen CTRL-\ eller genom att skriva EndShell i shell:en. Kom dock ihåg att skärmen också kommer att stängas när alla fönster på den har blivit stängda. Vanligtvis är shell:en det enda fönstret på skärmen så skärmen stängs när shell:en stängs.

Läs ShellScr tillval för instruktioner om hur man ställer in skärmen och shell:en efter egen smak.

1.4 ShellScr tillval

Du kan ändra hur ShellScr fungerar med hjälp av följande tillval.

Du kan ställa in dem från shell:en, från 'verktygstyper' (tooltypes) i ShellScr programmet, från 'verktygstyperna' från en ikon vars 'standardverktyg' är ShellScr, från 'verktygstyperna' i en ikon eller katalog som är multivalda när du startar ShellScr eller från 'verktygstyperna' av en ikon som är vald när ↵ ShellScr startas som ett WB kommando i ToolsDaemon eller ToolManager eller något liknande 'dock' program.

Skärmtillval

PUBNAME
MODEID
DEPTH
FONT
AUTOSCROLL
SHANGHAI

TITLE
NOTITLE

Shell tillval

CONSPEC (WINDOW)
COMMANDFILE (FROM)

STACKSIZE

1.5 PUBNAME tillval

WB: PUBNAME
Shell: PUBNAME=NAMN

Det här är det namn som ShellScr ger till den allmänna skärm som den skapar.

Om inget anges så får skärmen namnet SHELL_XXXX, där XXXX är en unik ID kod för just den shell som öppnas.

Du avråds från att använda detta tillval om det inte är nödvändigt, eftersom du inte kan öppna två shell:ar med samma namn samtidigt. Var vänlig och ta en närmare titt på programmet MultiCX, som har tillval så att man kan skapa ett 'magiskt' allmänt skärnmamn för den främsta skärmen oberoende av vad det är för skärm.

Om du vill ändra på texten i titelbalken, var vänlig och använd TITLE tillvalet.

1.6 MODEID tillval

```
WB:      MODEID
Shell:   MODEID=ID
```

Det här är ModeID för det skärmläge du vill att skärmen skall använda. Den specificerar vilken monitor och upplösning ShellScr skärm skall använda.

Du kan specificera skärmläget på ett av dessa sätt:

- som ett decimalt nummer (kompatibelt med tidigare ShellScr modeid:er)
- som ett hexadecimal nummer, startande med ett '\$' eller '0x', t.ex. "0x29004"
- som ett skärmläges riktigt namn, t.ex. "PAL:High Res Laced".
- som ett frågetecken, "?", eller en blank sträng, "", i vilken fall en ASL skärmlägesdialogfönster kommer att öppnas och låta dig välja ett skärmläge.

Du kan få en lista på "riktiga namn" från 'Screenmode' inställningsprogrammet eller från ASL skärmlägesdialogfönstret. Kontrollen av riktiga namn skiljer på små och stora bokstäver vilket betyder att "PAL:High Res" är korrekt medan "pal:high res" inte kommer att accepteras.

Om du inte specificerar ett ModeID eller om du specificerar ett ogiltigt ett, så används samma skärmläge som den förvalda skärmen har (vanligen Workbench).

Läs också om DEPTH tillvalet, för att välja antalet färger för skärmen.

1.7 DEPTH tillval

```
WB:      DEPTH
Shell:   DEPTH/N
```

Detta är ett decimal nummer som anger hur många bitplan du vill använda för skärmen, mellan 1 och 4 eller mellan 1 och 8 för maskiner med AGA grafikkretsar eller grafikkort. Om du har ett modernt grafikkort så kan du välja 16 bitplan (hicolor) eller 24 bitplan (truecolor).

Ju fler bitplan du använder desto mer minne går åt.

Det förvalda värdet är alltid 2 bitplan (4 färger). MagicWB användare kan vilja ha 3 bitplans skärmar för deras 8 MWB färger, men de flesta kommer att

använda de normala 2 bitplanen för en shell!

Om du ber om ett skärmlägesdialogfönster med MODEID tillvalet, så kan du också ställa in antalet bitplan för skärmen därifrån.

Färgerna kommer att ärvas från den förvalda skärmen.

1.8 FONT tillval

WB: FONT
Shell: FONT/K

Det här tillvalet ger dig möjligheten att välja ett teckensnitt med fast bredd för skärmen. Det enda du behöver göra är att ge teckensnittets namn och storlek, "XEN/9" eller något liknande. Om du vill använda ett teckensnitt av storleken 8 så behöver du inte ens ange storleken utan det räcker med "XEN" eller något liknande.

Om du anger "" eller "?" som teckensnitt så kommer ett teckensnittsdialogfönster att öppnas varifrån du kan välja teckensnitt och storlek.

Shell fönster ärver vanligen ett teckensnitt med fast bredd från skärmen de öppnas på. Det finns ingen orsak att tvinga dig att använda teckensnitt med fast bredd men ShellScr kommer att varna dig om du valde ett proportionellt teckensnitt och du kommer att märka att shell:en troligen inte använder sig av det, endast skärmens titelbalk.

ShellScr är smart när den väljer sin förvalda teckensnitt. Om den förvalda skärmen använder sig av ett teckensnitt med fast bredd så kommer ShellScr att använda sig av det. Om inte så kommer den att använda sig av systemets förvalda teckensnitt, som väljs med 'Font' inställningsprogrammet.

1.9 AUTOSCROLL tillval

WB: AUTOSCROLL
Shell: AUTOSCROLL/S

Genom att välja detta tillval så aktiveras automatisk skrollning av skärmen. Det betyder att om du drar skärmen utanför rutan så kan du skrolla tillbaks skärmen i rutan genom att bara röra på musen.

1.10 SHANGHAI tillval

WB: SHANGHAI
Shell: SHANGHAI/S

Före Workbench 2 kom till Amiga:n, hade ett program två möjligheter när det öppnade ett fönster. Det kunde öppna det på sin egen skärm eller på Workbench skärmen.

Nu med Workbench 2, så kunde man öppna fönster på den förvalda skärmen men för kompabilitet så öppnas gamla program ännu på Workbench skärmen.

Amiga har ett speciellt globalt tillval för allmänna skärmar kallat "Shanghai mode". När det är aktiverat så öppnas även gamla program på den förvalda skärmen precis som modernare program.

Bara välj detta tillval för att aktivera Shanghai tillståndet.

1.11 TITLE tillval

```
WB:      TITLE
Shell: SCREENTITLE=TITLE
```

Det här är texten som kommer att visas i skärmens titelbalk. Det här tillvalet är här eftersom shell:en du öppnar inte kommer att ha en egen titelbalk.

Den förvalda texten i titelbalken är 'AmigaShell'.

För att ta bort titelbalken helt och hållet, läs mer om NOTITLE tillvalet.

1.12 NOTITLE tillval

```
WB:      NOTITLE
Shell: NOTITLE=HIDETITLE/S
```

Det här tillvalet specificerar att skärmen inte skall ha en titelbalk ↵
överhuvudtaget så att
du kan få hela skärmen till ditt förfogande. Beräkning av fönsterstorleken
tar detta med i beräkningarna.
Kom ihåg att den normala CON: fönstret ser mycket fult ut med detta tillval! Det ↵
är bäst
att använda det med VNC: eller liknande CON:-ersättare som verkligen använder hela ↵
skärmen.

1.13 CONSPEC (WINDOW) tillval

```
WB:      CONSPEC
Shell: CONSPEC=WINDOW
```

Det här är den mest tekniska inställningen av ShellScr. Du behöver inte använda detta tillval om du inte vill.

Det här specificerar hur konsollfönstret öppnas på en allmän skärm.
Den skall innehålla två "%" tecken som ersätts av ShellScr till vissa värden. Det första "%" ändras till koordinaterna och storleken på fönstret i formatet "v/u/b/h", t.ex. "0/3/640/253" på en 640x256 skärm.
Det andra "%" ändras till namnet på den allmänna skärmen som används.
Fönstret borde vara ett backdrop kantlöst knapplöst fönster utan titel.
För att ta ett exempel, det förvalda är "CON:%s//BACKDROP/NOBORDER/

SCREEN%s". Om du använder VINCEd, så är en bra inställning följande:

```
VNC:%s//SCREEN%s/BACKDROP/NOBORDER/NOCLOSE/NOSIZE/NODRAG/NODEPTH/  
NOPROPX/NOPROPY/NOBUTTONS/NOICONIFY/SHELL/MENU
```

Om du vill läsa mer om ämnet, beräkning av fönstrets storlek.

1.14 COMMANDFILE (FROM) tillval

```
WB:      COMMANDFILE  
Shell:   COMMANDFILE=FROM
```

Det här tillvalet låter dig köra ett skript när ShellScr startas.

Shell:en kör vanligen S:Shell-Startup, men det här tillvalet låter dig välja vilket skript som skall köras vid start. Precis som NewShell:s 'FROM' parameter (för att vara exakt, det är just så det görs). Till skillnad från några skillnader så är det också precis vad IconX gör.

1.15 STACKSIZE tillval

```
WB:      STACKSIZE  
Shell:   STACKSIZE=STACK/N
```

Normalt så ärver en ny shell sin stack storlek från ShellScr (vi kommer att ignorera det självklara kommandot 'Stack' i Amiga shell:en med vilket du kan ändra stackens storlek). Så för att få en stor stack för shell:en så måste du ge en stor stack åt ShellScr som endast behöver 1kB? Det behöver du inte! Du kan använda detta tillval för att ge en liten stack åt ShellScr medan shell:en får en större stack.

Du ger stackens storlek för den nya shell:en i bytes, som blir avrundad uppåt till närmaste 4 bytes. Den minsta storleken du kan ge är 1600 bytes. Den största storleken begränsas endast av tillgängligt minne. Den förvalda storleken för stacken åt en ny shell är 4096 bytes.

1.16 ShellScr historia

1.0: Första publiceringen

1.1: Lade till tillvalen PUBNAME, MODEID, DEPTH, TITLE och CONSPEC.
Använder System() istället för Execute().
Städade upp i källkoden.

1.2: Kollar ett antal flera saker.
Lade till NOTITLE tillvalet.
Bättre beräkning av fönsterstorleken
Väljer nu aktuell katalog och sökvägar på rätt sätt från Workbench.

1.3: Kollar nu ikonens verktygstyper.

Använder sig nu av den förvalda skärmens färger.
Städade upp i dokumentationen.

1.4: Skärmläges- och teckensnittsdialogfönster.

Lade till FONT tillvalet.
Lade till COMMANDFILE tillvalet.
Fixade det förvalda CONSPEC att fungera ordentligt.
Förklarar nu uppkomna fel mycket bättre, speciellt "kan ej öppna skärm".
Fixade tydning av argument. Kataloger och ikonlösa "ikoner"
krashade/hängde systemet/avlivade minnet;)/ osv.... inte mera.
Du kan nu göra ShellScr resident.
Visa 'Skärmen stängs' dialogrutan om ShellScr inte kan stänga sin skärm
genast.

1.4b: Försökte mig på ett antal bugfixar.

Fixade helt säkert args.m från att i framtiden orsaka Enforcer träffar.
Tog bort anrop till asl.library om det inte kunde öppnas.
Fixade skärmläges- och teckensnittavläsning litet .

1.5: Lade till STACKSIZE, AUTOSCROLL och SHANGHAI tillvalen.

Välj djup/autoscroll från skärmens dialogfönster.
Att avbryta skärmlägedialogsfönstret medför nu att programmet inte startar.
'Skärmen stängs' dialogfönstret har nu en 'Avbryt' knapp.
Sätter nu sig själv som den förvalda skärmen när den öppnas.
Lade till 'riktigt namn' skärmläge igenkänning.
Felmeddelande när NewShell kommandot inte kan köras.
Fixade fel vid teckensnittsval.
Ett antal småfixar och annat jag glömt.

Framtida utveckling

1.17 Framtida utveckling av ShellScr

Saker som jag kunde göra:

- teckensnitt specificerat i 'myfont.24'- och 'myfont/24'-format.
- lokalisering

Om du tycker 'Ja! Det vill jag ha!' om vilket som helst av dessa,
berätta då om det för mig. Om ingen ber om dem så antar jag att de
inte är eftertraktade.

Om du är starkt av åsikten att ShellScr innehåller onödiga tillval
som du inte gillar eller behöver så tar jag bort dem.
Gamla goda scrsh brukade vara lite över 1k, skall du veta. (Inte att
glömma 'kryllande av buggar och hardcoded' -Ed.)

Andra förslag från er är _också_ välkomna. Skicka dem till mig.

1.18 Källkodsnoteringar

Makefile kan endast köras på min dator och kommer inte att kompilera ShellScr. Eftersom alla kompillerade moduler följer med arkivet så utgör detta inget problem. Skriv bara 'ec opti ShellScr.e'.

Alla moduler och deras källkoder följer med. De är tillgängliga från mitt arkiv på Aminet, 'dev/e/kyz.lha'. Du kan använda dig av alla moduler endera som dom är eller ändrade så länge du nämner mig.

Assemblermodulerna använder sig av E globaler, tack vare 'include' filen skapad av eglobs.e

Jag frigör inte alla strängar och listor jag skapar, det lämnar jag till 'startup' koden. Det gör det kompillerade programmet mindre och källkoden enklare, och gör nästan upp för minnet som programmet slösar.

OK. Jag medger det. Jag hade fel om teckensnittspekaren i skärmstrukturen. När den förvalda skärmen vi kopierade från stängdes förlorade vi pekaren och hejdå med all teckensnittsdata. Jag gör nu en ordentlig kopia av strukturen när jag kopierar den. Jag kommer nu även ihåg att _kopiera_ namndelen från ASL teckensnittsdialogfönstret, istället för att ta pekaren till den just före jag frigör den. Oops. Men ändå - två buggar mindre, eller hur?

1.19 Beräkning av fönsterstorleken

Beräkning av fönsterstorlek görs av ShellScr på följande sätt:

Bredden av fönstret är alltid bredden av hela skärmen.

Om du är nöjd med det förvalda CON: fönstret och inte specificerar ett alternativ CONSPEC, då blir höjden på fönstret höjden av skärmen, på grund av att CON: ser bäst ut med detta tillval så täcks dess oanvändbara titelbalk med skärmens titelbalk.

Om du använder en skräddarsydd CONSPEC, så gör inte ShellScr antaganden och använder height_of_the_screen - height_of_the_titlebar som höjden. Om du använder NOTITLE tillvalet så visas ingen titelbalk och du får hela skärmen till shell:ens förfogande.

1.20 Ordlista över viktiga termer

Bildskärm

Den bild som visas av din Amiga på i en TV eller monitor.

Skärm

Det grafiska område som visas på bildrutan.

Skärmar kan dras till eller utanför skärmen, arrangeras framför eller bakom varandra. Du kan ha så många skärmar öppna som minnet tillåter.

Workbench skärmen

Det här är den fundamentala skärmen för Amiga. Det är den första skärmen som användaren ser när datorn startas och är vanligen den skärm som program använder för att öppna sina fönster på då ingen annan skärm är specificerad. Workbench skärmen är en allmän skärm och om inget annat definierats den förvalda skärmen.

Skräddarsydd skärm

Det här är en skärm som programmet skapar åt sig själv. Inget annat program kan (legalt) öppna ett fönster på det.

Allmän skärm

En skärm som vilket program som helst kan använda. En allmän skärm har ett namn (Workbench:ens skärm har namnet 'Workbench') och program kan använda skärmen genom att ge skärmens exakta namn när det öppnar ett fönster. Om inget namn ges så öppnas fönstret på den förvalda skärmen.

Förvald skärm

Den här skärmen är vanligen Workbench skärmen, men det kan ändras av program eller av användaren med m.h.a. program. Alla allmänna skärmar kan designeras som förvald skärm.

Den förvalda skärmen används för att öppna fönster på då ingen annan skärm är vald av programmet.

Det finns dock ett undantag till denna regel och det är gamla program som alltid öppnar sina fönster på Workbench skärmen. Denna kompatibilitet kan stängas av med Shanghai mode.

Stack

Ett område av minnet som tilldelas varje program för att förvara tidsmässigt kortvarigt data.

UNIX program är ökända för att använda stora mängder stack.

1.21 Tillägnat

ShellScr adapterades från assemblerprogrammet 'scrsh' skrivet av Kyzer/CSG.

ShellScr programmerades av Kyzer/CSG i Amiga E 3.3a, baserat på källkoden scrsh.asm av Kyzer/CSG, RKM källkoden clonescreen.c och källkoden till dospath.library av Stefan Becker. Källkoden till ShellScr liknar ej mer någon av dem.

De bästa lyckönskningar går till följande personer:

(DE) Walter Haidinger och deJoker som önskade tillvalet COMMANDFILE.

(AU) Frank Bunton, som önskade bättre färger.

(DE) Thomas Richter, för att ha 'fixat' WBLoad, och för ViNCeD!

(NL) Wouter van Oortmerssen, för E programmeringsspråket.

(DE) Frank Wille, för PhxAss.

(US) Free Software Foundation, för GNU licensen och GNU make.

Hälsningar går till:

IconX är © Amiga International (är det skrivet här...)

MultiCX är ©~Martin Berndt.

PubChange är © Steve Koren.

ReqTools och ReqPatch är © Nico François, Magnus Holmgren och Dave Jones.

ToolManager är © Stefan Becker

ToolsDaemon är © Nico François

ViNCed och VNC: är © Thomas Richter.

XEN är © Martin Huttenloher

Kontakta:

Kyzer/CSG,

49 Fairview Road,

AB22 8ZG, Scotland.

eller e-post: kyzer@4u.net

<http://zap.to/kyz>
