



A F S C H R I F T

van de akte van depot van door de  
vennootschap onder firma:  
1001 SOFTWARE DEVELOPMENT, geves-  
tigd te Alkmaar,  
gedeponeerde stukken.

d.d. 19 januari 1993.

=====

VAN LEERSUM & NAGTGLAS BOOT NOTARISSEN

TE

ALKMAAR





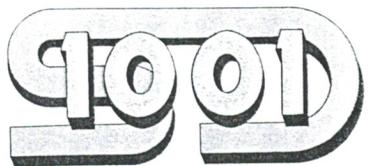
AFSCHRIFT  
Z10421DNO AKTE VAN DEPOT

Heden, de negentiende januari negentienhonderd -----  
drieënnegentig, verscheen voor mij, Mr Gerardus -----  
Wilhelmus Nagtglas Boot, notaris ter standplaats ---  
Alkmaar: -----

de Heer **Erik van Eykelen**, geboren te Alkmaar op een  
januari negentienhonderdachtenzestig, vennoot, -----  
wonende te Alkmaar, Ziederij 8 (postcode 1825 HR), -  
ongehuwd, -----  
ten deze handelende in zijn hoedanigheid van vennoot  
van de vennootschap onder firma: **1001 Software -----  
Development**, gevestigd te Alkmaar aan de Noorderkade  
2 G (postcode 1823 CJ), -----  
welke comparant verklaarde onder de minuten van mij,  
notaris, te deponeren: -----

- een uittredingsovereenkomst betreffende de vennoot  
Stefano Paulus van der Horst; -----
- een MediaLink brochure; -----
- schermafbeeldingen; en -----
- PageTalk en ScriptTalk language overview, -----  
alwelke stukken door de comparant en door de Heer --  
Johannes Mathijs Honig, wonende te Alkmaar, -----  
Koningsweg 6, en de Heer Stefano Paulus van der ----  
Horst, wonende te Alkmaar, Albatrosweg 84, zijn ----  
ondertekend respectievelijk gewaarmerkt. -----  
De comparant is mij, notaris, bekend. -----  
Deze akte is verleden in minuut te Alkmaar op heden,  
de datum in het hoofd dezer gemeld en is terstond na  
zakelijke opgave van haar inhoud aan de comparant en  
haar volledige voorlezing door de comparant en mij,  
notaris, getekend. -----  
w.g. E. van Eykelen; G.W. Nagtglas Boot. -----

Volgen aangehechte stukken,



# 1001 Software Development

## Uittredingsovereenkomst

### Partijen

De vennootschap onder firma 1001 Software Development gevestigd te Alkmaar heeft drie vennoten, te weten:

Erik van Eykelen  
Ziederij 8  
1825 HR Alkmaar  
072-615630

Joost Honig  
Koningsweg 6  
1811 LM Alkmaar  
072-125924

Stefan van der Horst  
Albatrosweg 84  
1826 KJ Alkmaar  
072-622578

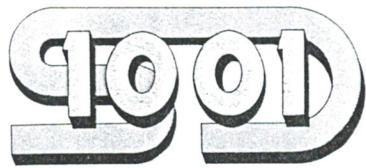
In deze overeenkomst zullen de vennoten respectievelijk Erik, Joost en Stefan worden genoemd.

1001 Software Development  
Noorderkade 2/g  
1823 CJ Alkmaar  
Tel. 072-114541  
Fax. 072-111497

### Bijlagen:

MediaLink brochure  
Scherm-afbeeldingen  
PageTalk en ScriptTalk language overview





# 1001 Software Development

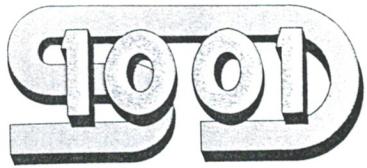
## Algemene omschrijving

Deze overeenkomst regelt de uittreding van Stefan uit de vennootschap onder firma 1001 Software Development, gevestigd te Alkmaar en ingeschreven bij de Kamer van Koophandel onder nummer 57143. De overblijvende vennoten Joost en Erik zullen de vennootschap voorzetten. De uittreding vindt in goed overleg plaats, Stefan stelt voorop dat Joost en Erik de firma kunnen voorzetten en zijn eisen zijn dan ook van dien aard dat de voorzetting plaats kan vinden.

## Onder voorzetting van de firma wordt verstaan:

- Het ontwikkelen van programmatuur. In concreto: het afmaken van het pakket MediaLink voor de Commodore Amiga en het spoedig (medio 1993) starten met de ontwikkeling van een versie van dit pakket voor de PC draaiende onder MS Windows en te zijner tijd voor de Apple Macintosh.
- De lopende betalingsverplichtingen kunnen voldoen. In concreto: de huur van de kantoorruimte aan de Noorderkade 2/g voor de komende twee jaar (zoals omschreven in het huurcontract), uitstaande rekeningen aan crediteuren van de firma, het betalen van de freelance medewerkers Pascal Eeftinck, Cees Lieshout en Mike Kennedy en de lonen en kosten van de twee overblijvende vennoten.
- De volgende (handels)namen kunnen gebruiken (zij worden eigendom van Joost en Erik):
  - *1001 Software Development*
  - *MediaLink* (een geregistreerde handelsnaam in de Benelux en VS).
  - *CAPS* (Computer Aided Presentation System) Classic
  - *CAPS XL* (eXtra Large)
  - *Xapp* (eXternal APPlication)
  - *ScriptTalk*
  - *PageTalk*
  - *Remote Access*
  - *AmigaTalk*

The image shows three handwritten signatures in black ink. From left to right: 1. A signature that appears to be 'Joost'. 2. A signature that appears to be 'Erik'. 3. A signature that appears to be 'Stefan'.



# 1001 Software Development

In deze overeenkomst zullen de volgende zaken worden vastgelegd:

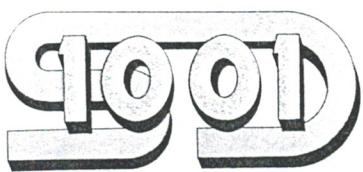
1. Rechten en plichten.
2. Meenemen van apparatuur.
3. Percentageregeling over de verwachte inkomsten van MediaLink.
4. Inleveren credit card, giro betaalpas, cheques, contant geld, sleutels.
5. Afdragen copyright over MediaLink, CAPS Classic en CAPS XL.
6. Afspraken over het werken voor andere firma's en de ontwikkeling van (mogelijk) concurrerende programmatuur.
7. Blokkeren van automatische afboekingen van de bank- en girorekeningen van de firma.
8. Uitschrijven van Stefan uit de vennootschap bij de KvK.
9. Vastlegging functionaliteit van MediaLink.
10. Opstellen van een functie-omschrijving van de Process Controller.
11. Afspraken over de ontwikkeling en verkoop van netwerk-producten voor de Amiga.

## **1. Rechten en plichten.**

Stefan heeft in principe als vennoot van de firma 1001 Software Development recht op 1/3 (één derde) van alle baten en goederen (minus de schulden) bij het verlaten van de firma. Omdat het uitnemen van deze zaken tot gevolg zou hebben dat de firma niet is voort te zetten, ziet hij hier van af. Deze clausule omschrijft welke rechten en plichten Stefan verwerft in ruil voor deze soepele opstelling:

### **Rechten:**

- De vennoten Joost en Erik nemen de huurschuld over en vrijwaren Stefan van claims van de verhuurder in geval van het opzeggen van het huurcontract door één van de partijen.
- Stefan wordt gevrijwaard van claims door cliënten van 1001 over o.a. geleverde goederen.
- Belastingaanslagen over de jaren 1989 t/m 1992 die voortkomen uit arbeid verricht door Stefan voor 1001 zullen door 1001 worden betaald.



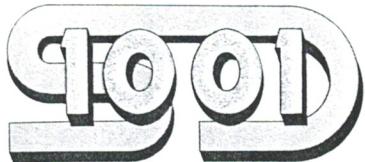
# 1001 Software Development

## Plichten:

- Stefan zal geen materiële en immateriële zaken aan de firma onttrekken anders dan omschreven in deze overeenkomst.
- Stefan zal zich houden aan afspraken zoals omschreven in deze overeenkomst. Overtreding van de regels kan leiden tot juridische stappen te nemen door 1001 en zullen bij een rechtbank te Alkmaar worden behandeld.
- Stefan zal geen acties ondernemen na ondertekening van deze overeenkomst die schade kunnen hebben voor de goede naam van de firma of die tot financiële schade kunnen leiden. Deze overeenkomst is in goede verstandhouding opgesteld en 1001 verwacht deze goede relatie te kunnen voortzetten.
- 1001 kan, in goed overleg, op redelijke tijdstippen en vooraf aangekondigd, beroep doen op de kennis van Stefan over MediaLink. MediaLink bevindt zich in het laatste stadium van ontwikkeling maar bevat nog fouten. Stefan zal, als 1001 daarom vraagt, werkzaamheden verrichten die tot doel hebben:
  - Fouten uit MediaLink te halen (zogenaamde 'bugs').
  - Die functies toe te voegen die MediaLink de functionaliteit geven zoals omschreven in de brochure van MediaLink (uitgezonderd de Remote Access Module) én onderdeel uitmaken van de Process Controler en daarmee samenhangende XaPPs.

De werkzaamheden worden niet betaald maar zullen worden gedragen door de inkomsten die Stefan heeft die voortvloeien uit de afspraak over het betalen van een percentage (zie clausule 3). 1001 kan beroep doen op Stefan voor de duur van 9 (negen) maanden.

- Stefan zal geen informatie verschaffen aan derden over de financiële situatie van 1001.
- Stefan zal na ondertekening van deze overeenkomst niet langer mogen handelen uit naam van 1001.



# 1001 Software Development

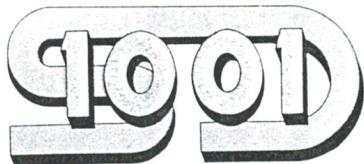
## 2. Meenemen van apparatuur.

De apparatuur die Stefan meeneemt:

- Amiga 2500, 1084 kleuren monitor, toetsenbord, muis.
- 2 Harddisks (20 Mb en 40 Mb).
- CBM 68020 accelerator card.
- Action Controler (eigen ontwikkeling 1001 SD).
- Oscilloscoop met probes.

## 3. Percentageregeling over de verwachte inkomsten van MediaLink.

1001 laat MediaLink distribueren via Activa International te Amsterdam. De verwachte prijs per pakket die Activa aan 1001 betaalt is circa Dfl. 230,00 voor de Europese versie en Dfl. 195,00 voor de Amerikaanse versie. Het percentage dat 1001 aan Stefan betaalt over dit bedrag zal 12% zijn met een minimum van Dfl 20,00. 1001 verplicht zich tot het betalen daarvan voor de eerste 500 verkochte MediaLink pakketten (Amiga versie). Voor elk navolgend verkochte pakket zal 1001 verplicht zijn een percentage van 7% te betalen aan Stefan. De betaling zal per twee maanden geschieden. 1001 kan niet garanderen dat het genoemde bedrag stabiel is, het zal de conjunctuur volgen en afhankelijk zijn van de prijs van o.a. Scala MM en andere concurrerende pakketten. Bij verandering van de productnaam MediaLink zal het bovengestelde automatisch betrekking krijgen op de nieuwe productnaam. Ter controle van het aantal verkochte pakketten heeft Stefan of een uit diens hoofde aangewezen boekhouder inzage in de verkoop-boekhouding.



# 1001 Software Development

## 4. Inleveren credit card, giro betaalpas, cheques, contant geld, sleutels.

Bij het ingaan van deze overeenkomst zullen de volgende betaalmiddelen moeten worden overgedragen aan Joost of Erik:

- MasterCard.
- Giropas op naam van Stefan voor het gironummer 2030959.
- Alle girocheques op naam van 1001 (gironummer 2030959).
- Contant geld (ook buitenlands geld).
- Alle andere waardepapieren waarmee geld kan worden opgenomen van de giro- en bankrekening (ABN Alkmaar, nummer 56.35.20.493).

Ook moeten de sleutels worden ingeleverd (voordeur, kantoordeur, postvak, parkeer-garage keytag).

## 5. Afdragen copyright over MediaLink, CAPS Classic en CAPS XL.

Stefan beschikt over volledig werkende versies van CAPS Classic, CAPS XL en ook over een zeer ver gevorderde versie van MediaLink. Deze pakketten zijn de producten die 1001 verkoopt en vormen daarmee de enige bron van inkomsten. Het is Stefan te allen tijde verboden deze pakketten (of de source coding) te verkopen, te verspreiden, te verhuren, in bruikleen te geven of in welke vorm dan ook te distribueren en daarmee de firma 1001 schade te berokkenen. Het maken van (nagenoeg) identieke software die qua functionaliteit hetzelfde doet als eerder genoemde pakketten is ook niet toegestaan, zie hiervoor clausule 6.

## 6. Afspraken over het werken voor andere firma's en de ontwikkeling van (mogelijk) concurrerende programmatuur.

Hierbij wordt afgesproken dat Stefan geen programmatuur gaat ontwikkelen die qua functionaliteit hetzelfde doet als CAPS Classic, CAPS XL en MediaLink. Aangezien 1001 de duidelijke intentie heeft ook onder MS Windows en op de Apple Macintosh programmatuur te gaan ontwikkelen, geldt dit verbod ook voor te ontwikkelen programmatuur voor deze systemen. 1001 kan en wil niet verbieden dat Stefan programmatuur gaat ontwikkelen voor zogenaamde 'multimedia' systemen maar wil voorkomen dat deze programmatuur een directe afspiegeling is van de drie eerder genoemde producten. Bovengesteld verbod geldt voor de tijdsduur van 48 (achtenveertig) maanden na ingaan van deze overeenkomst. Zie ook clausule 9.



# 1001 Software Development

## 7. Blokkeren van automatische afboekingen van de bank- en girorekeningen van de firma.

Bij het ingaan van deze overeenkomst moeten alle (automatische) afboekingen die gedaan worden om privé rekeningen van Stefan te betalen (huur, gas, water etc.) worden geblokkeerd. Stefan verklaart zich akkoord met onmiddellijke terugbetaling van ten onrechte afgeboekte bedragen mochten er onverhoop toch afboekingen worden verricht op zijn naam na het ingaan van deze overeenkomst.

## 8. Uitschrijven van Stefan uit de vennootschap bij de KvK.

Stefan zal zich binnen een week na ondertekening van deze overeenkomst laten uitschrijven uit de vennootschap 1001 Software Development. De vennootschap is ingeschreven bij de Kamer van Koophandel te Alkmaar onder nummer 57143.

## 9. Vastlegging functionaliteit van MediaLink.

Op de datum van ondertekening van deze overeenkomst beschouwt 1001 het pakket MediaLink als de drijvende kracht achter het huidige en toekomstige succes van de onderneming. Hiermee wordt bedoeld het huidige aanzien van het bedrijf in de 'Amiga wereld' en de kans op de succesvolle verkoop van het product. Er is 1001 alles aan gelegen MediaLink tot een succes te maken en het ook zo spoedig mogelijk op andere platforms uit te brengen (zie ook clausules 5 en 6). Om duidelijk te maken welke functies 1001 belangrijk acht voor het succes van MediaLink en wat zij beschouwt als 'geestelijk eigendom' volgt hier een omschrijving (zie ook de bijgevoegde scherm-afbeeldingen):

NB: De onderdelen afzonderlijk zijn niet te beschermen, wel de manier waarop deze samenwerken. 1001 wil de 'look en feel' van MediaLink beschermen maar realiseert zich dat dit een moeilijke zaak is. Onderstaande korte beschrijving, de MediaLink brochure en scherm-afbeeldingen moeten duidelijk maken wat met 'look and feel' bedoeld wordt.



# 1001 Software Development

Karakteristieke kenmerken van MediaLink:

- De splitsing van het pakket in een script- en een pagina-opmaaksysteem waar tussen snel heen en weer kan worden geschakeld.
- Het document-georiënteerde pagina-opmaaksysteem.
- De rangschikking van de pull down menus.
- De overal aanwezige mogelijkheid tot knippen en plakken van objecten (windows, text, script objecten).
- De 'look and feel' van het pakket (het gebruikte lettertype, de kleuren, de grote buttons).
- Het onderbrengen van de script-objecten in afzonderlijk modules (de zogenaamde xapps).
- Het 'favourite applications menu' waarin de gebruiker zelf applicaties kan zetten en het 'running applications menu' waarin een lijst van draaiende programma's staat.
- De schaal- en remap functies die in het pagina-opmaaksysteem zitten.
- De import-functie met grafische weergave van verkleinde schermen.
- Het frame-georiënteerde pagina-opmaaksysteem met verplaatsbare en schaalbare windows.
- De user levels die het uiterlijk van het programma aanpassen.
- De PageTalk en ScriptTalk talen waarin respectievelijk de documenten en de scripts worden weggeschreven (zie bijlage).
- De transities die in het script zitten.
- Het onderscheid tussen seriële en parallelle takken in het script.
- De wijze van presentatie van het script door de script editor (subtakken, root, scrollen, kleine icons, dubbel klikken, commentaarvelden, het opkomen van requesters door dubbelklikken op de verschillende plaatsen van een scriptregel).



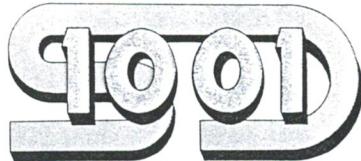
# 1001 Software Development

## 10. Opstellen van een functie omschrijving van de Process Controller.

Stefan zal binnen 2 (twee) maanden na ondertekening van dit contract een omschrijving geven van de Process Controller. Dit is het gedeelte dat de uitvoering van het script behandelt en de aanroep doet van de diverse script-objecten. 1001 heeft dit nodig om inzicht te krijgen in het functioneren van de programmatuur die Stefan ten behoeve van MediaLink heeft geschreven. De omschrijving dient dermate uitvoerig te zijn dat een programmeur aan de hand ervan binnen redelijke tijd de werking van de programmatuur kan doorgronden. In ieder geval zal per module moeten worden beschreven wat deze doet, welke weg de programmatuur doorloopt vanaf het starten van een script tot het stoppen, hoe de programmatuur los van MediaLink kan lopen, waar mogelijke 'zwakke' plekken zitten etc.

## 11. Afspraken over de ontwikkeling en verkoop van netwerk-producten voor de Amiga.

1001 heeft in 1992 netwerk-producten verkocht maar heeft de verkoop moeten staken omdat deze producten niet langer geleverd kunnen worden. Om toch te kunnen voldoen aan de vraag naar deze producten is de freelance programmeur Taco Schotanus gevraagd om eigen Amiga netwerk software te ontwikkelen. Stefan heeft veel tijd gestoken in onderzoek naar de hardware die deze nieuwe software moet ondersteunen. 1001 heeft de intentie om spoedig met deze eigen ontwikkeling op de markt te komen en wil niet dat Stefan binnen 24 (vierentwintig) maanden na ondertekening van dit contract met een concurrerend systeem voor de Amiga op de markt komt.



# 1001 Software Development

## Afsluiting en ondertekening:

De ondergetekenden verklaren zich akkoord met deze overeenkomst. De 'volledige bevoegdheid' van Stefan die hij heeft als vennoot vervalt hiermee en hij verklaart dat alle genoemde plichten gesteld door deze overeenkomst binnen 1 maand na de datum van ondertekening worden uitgevoerd (uitgezonderd de clausules die een andere termijn stellen). De rechten die Stefan heeft, eveneens door deze overeenkomst gesteld, zijn van kracht vanaf het moment van ondertekening.

De levensduur van de afspraken in deze overeenkomst is 5 (vijf) jaar. Daarna vervallen alle rechten en plichten van beide partijen.

Uittredende vennoot:

Stefan van der Horst

Overblijvende vennoten:

Joost Honig

Erik van Eykelen

Overeengekomen te Alkmaar d.d. 18 jan. 1993

Bekrachtigd door:

---

---

---

---

---



# MediaLink®

multimedia magic by 1001



If you want to create, combine and synchronize graphics, text and animations with audio and video, you'll need MediaLink 3.0.

Multimedia should be practical, it should work on inexpensive hardware and it must deliver solutions, not problems. MediaLink fits the bill: it's easy to use, inexpensive and doesn't require top-of-the-line hardware to perform well. MediaLink was developed by 1001 Software Development, a company with over six years of in-depth experience with the Amiga. We had our first Amiga based information system running in 1988 when nobody had ever heard of multimedia. MediaLink has evolved from several programs, of which many systems are being used all over the world to create 24 hours a day (non-stop) TV text channels. These systems are 'mission critical', which means that our software is stable: it stands the test of time and certainly meets the high standards of our users.

Designers of time-based applications, like stand-alone presentation systems or TV text channels, will love the three time code systems supported by MediaLink: standard timing with a precision of a tenth of a second, MIDI and SMPTE/EBU.

Interactive programs can be designed in the shortest possible time: our scripts are guaranteed to be shorter than similar scripts designed with the software of our competitors. And they're easier to read!

Video editors will love the large smoothly antialiased fonts, the line-by-line and page transitions, the realtime loading from harddisk of larger-than-memory animations and samples and the time base counter. The NewTek videotracker can be controlled, all the popular 24 bit cards, the Studio 16 audio card, laserdiscs and VCRs.

Synchronization of audio with pictures and animations is easy with the tempo control feature (tap the mouse buttons to make a rough synchronization, then fine tune it on a frame to frame basis by sliding some buttons). Timing is not performed relative to the previous event but on moments in time, relative to the start of the script. This means no out-of-sync problems when running a script several times.

MediaLink 3.0 covers six years of experience and adds many new ideas and Amiga features like 24 bit output, ARexx, parallel-processed script events (multitasking), sound module playing, plug-in objects and full support of Operating System 3.0. MediaLink fully adheres to Commodore's programming guidelines: menu driven, multitasking-friendly and localized for your country.

No compromises are made: we don't leave you out in the cold when you are running Kickstart 1.3 on an Amiga 500: you get fast bitmap scaling and remapping, breathtakingly smooth transmissions (even on unaccelerated machines!), overscan pictures and antialiased text from the Amiga 600 all the way up to the Amiga 4000.

MediaLink is menu-driven, with pull-down menus accessible by all screens and windows. Cutting and pasting, for example, gets a whole new meaning in MediaLink: everything you see on the screen can be put into the clipboard and pasted somewhere else. There's no difference between copying a window, a picture, a script object or a piece of text. And: you'll always find the menus at the same place, whether you're in the page lay-out system or in the script editor.

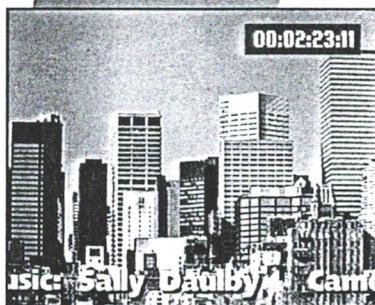
*M. E.*

## Commodore Info Main Menu

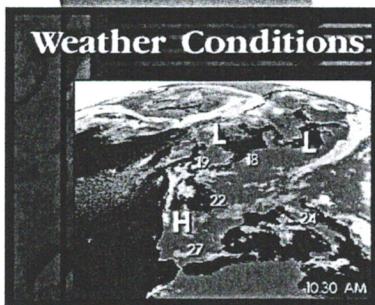
- 1** New Amiga models
- 2** CDTV titles
- 3** Multimedia software
- 4** Other products



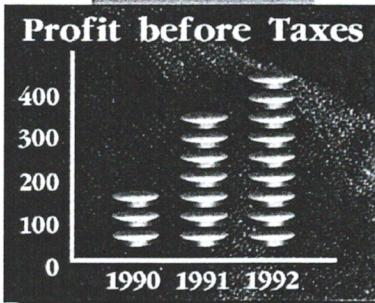
Interactive Information Systems



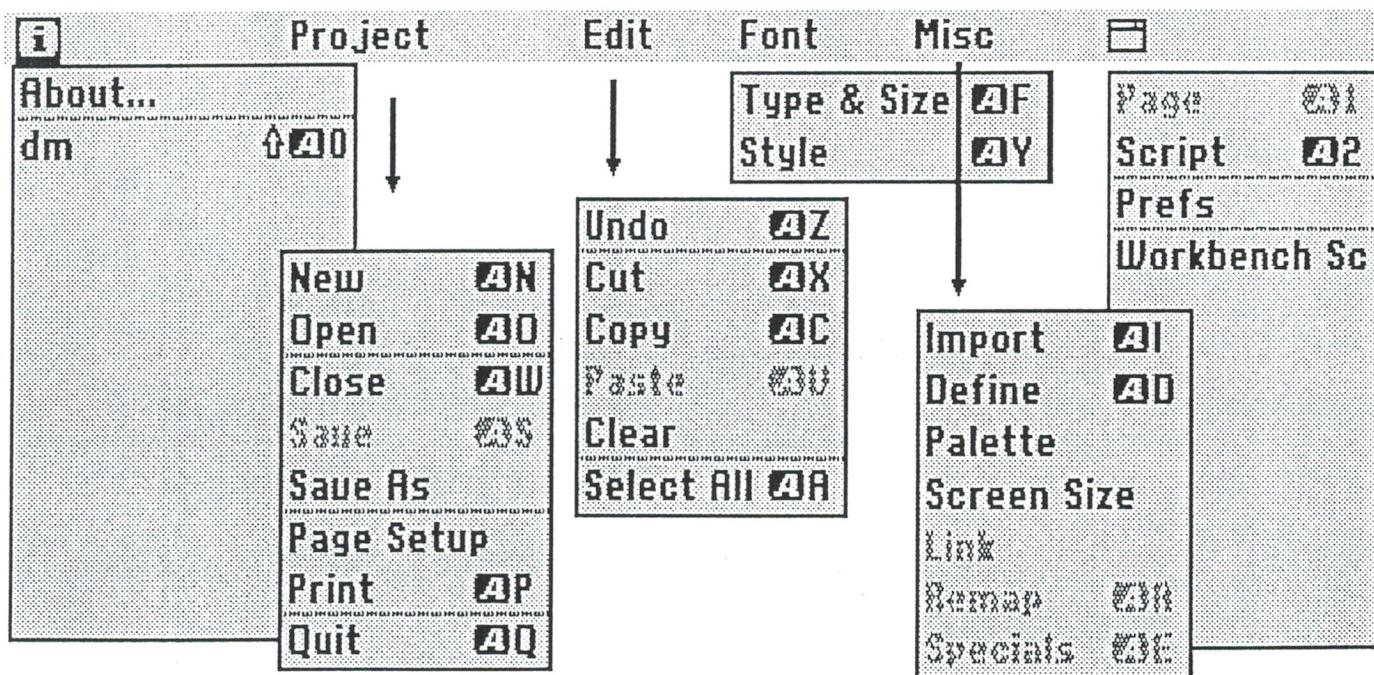
Video Titling

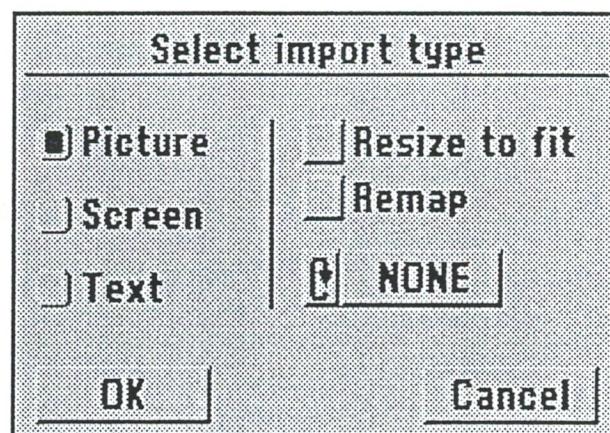


Information Channels



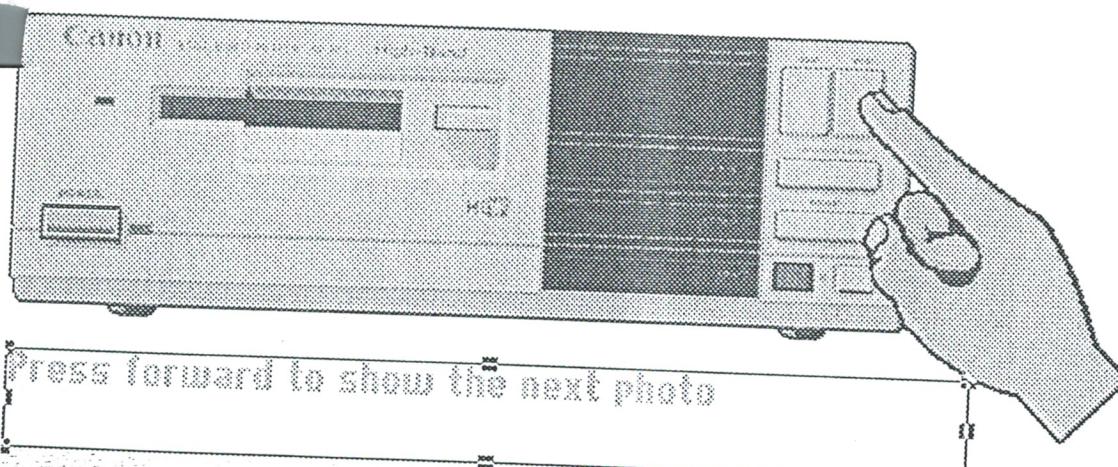
Presentations



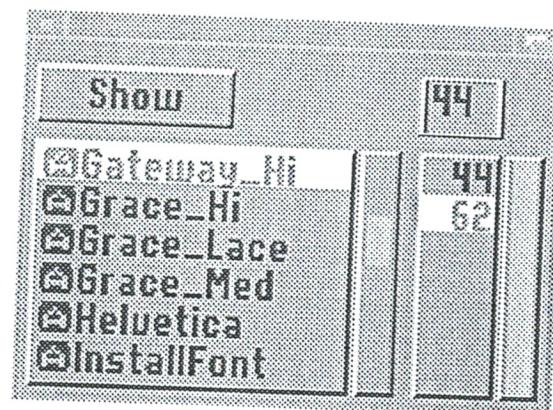


Bernardino

A



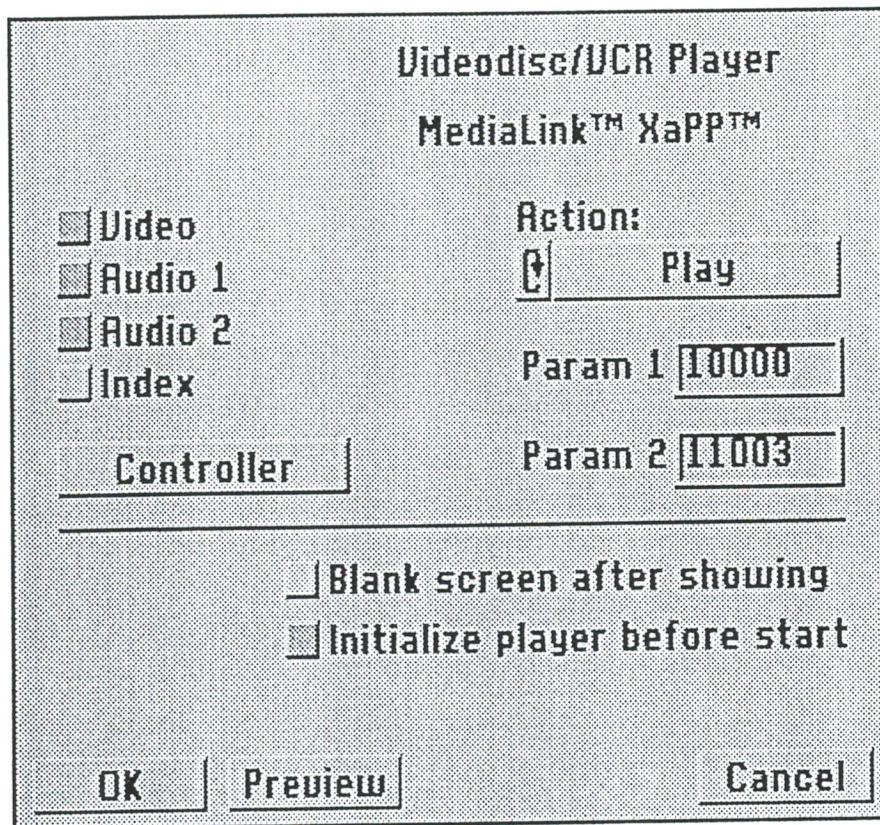
AaBbCcDd



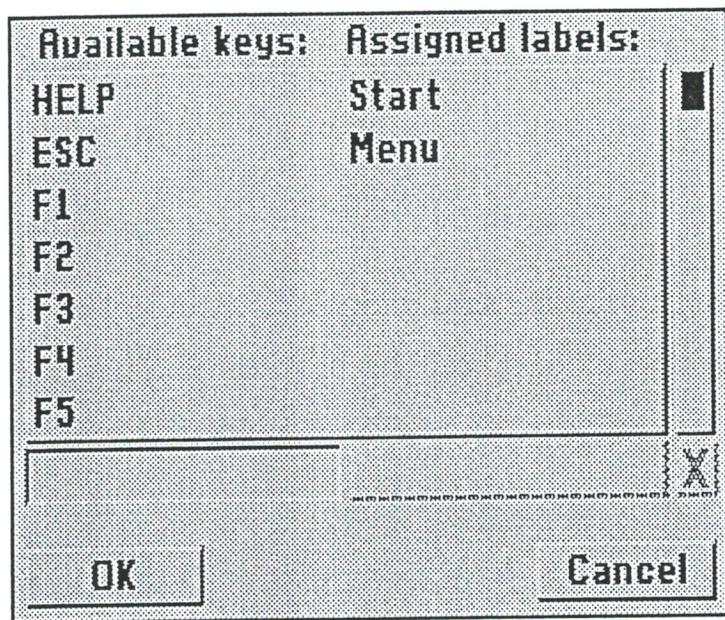
S E N

1

General Preferences	
User level:	<input type="radio"/> Multimedia
Startup screen:	<input type="radio"/> Script
Color set:	<input type="radio"/> 22
Video format:	<input type="radio"/> PAL
Thumbnail size:	<input type="radio"/> Small
Thumb colors:	<input type="radio"/> 4
Remap to:	<input type="radio"/> Grayscale
Script timing:	<input type="radio"/> Normal
Interlaced screens:	
<input type="checkbox"/> Thumbnails	<input type="button" value="Previous"/>
<input type="checkbox"/> Page	<input type="button" value="Next"/>
<input type="checkbox"/> Script	<input type="button" value="OK"/> <input type="button" value="Cancel"/>



S E M

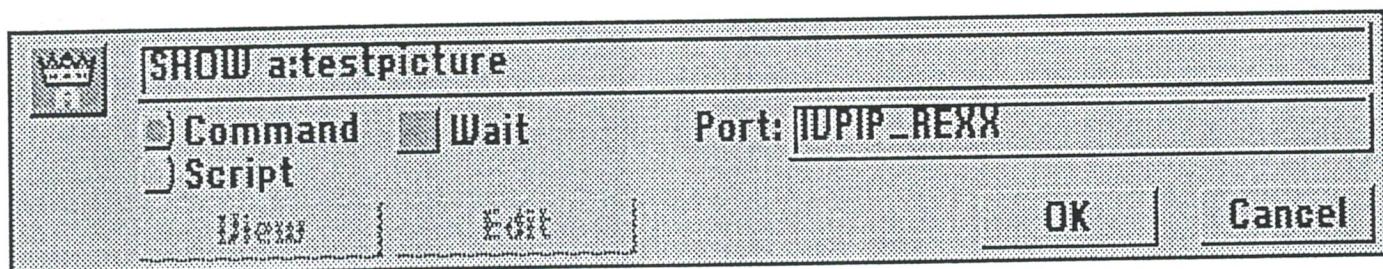


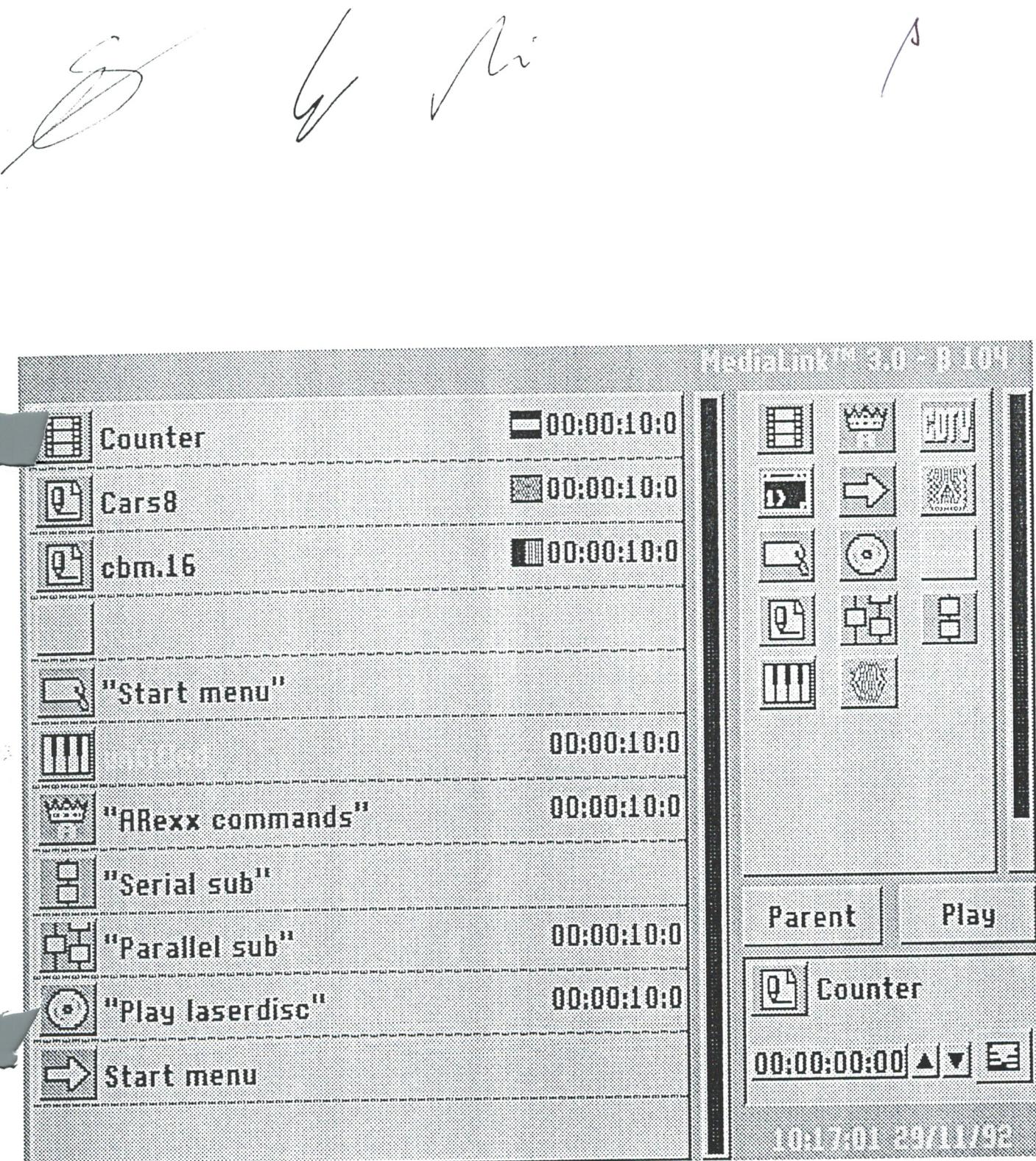
S Y N

Timing source:	Timing rate:
<input checked="" type="radio"/> Internal	<input checked="" type="radio"/> 25 FPS
<input type="radio"/> External	<input checked="" type="radio"/> 30 FPS
Timing type:	Offset: 00:00:00:00
<input checked="" type="radio"/> HH:MM:SS:T	<input checked="" type="radio"/> Send out
<input type="radio"/> MIDI Time Code	
<input type="radio"/> SMPTE/EBU	
<input type="radio"/> MediaLink Time Code	
<input type="text"/>	
Script timing:	<input checked="" type="radio"/> Normal
Preload:	<input checked="" type="radio"/> All objects
Play options:	<input checked="" type="radio"/> Auto + Manual
<input type="button" value="OK"/>	<input type="button" value="Cancel"/>

S E M

A





## ScriptTalk command summary

### SYNOPSIS

**ANIM "path", cycle, rotations, speed**

(in list)

### REGISTERS

nodeType = TALK\_ANIM  
objectPath/Name = path  
numericalArgs[0] = cycle (ARGUMENT\_OFF, ARGUMENT\_ON)  
numericalArgs[1] = rotations  
numericalArgs[2] = speed  
duration  
dayBits, startDays, startMinutes, startTicks, endDays, endMinutes, endTicks,  
startendMode  
effectNr

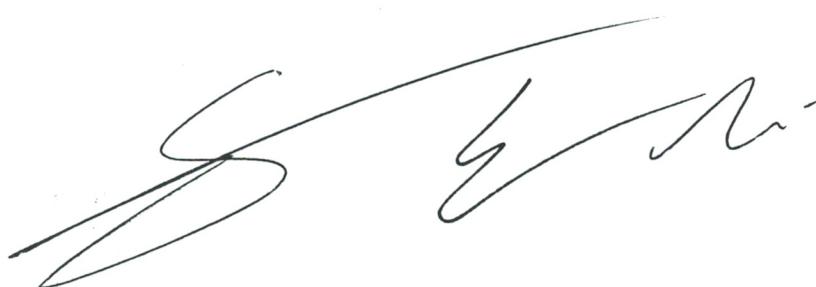
### POSITION

Between START and END.

### FUNCTION

Loads an animation or anim brush into memory and displays it on screen using the current effect number.

---

A handwritten signature in black ink, appearing to read "S. G. M." or "S. G. M." followed by a dash.

## SYNOPSIS

AREXX "port name", 'command string', "comment", mode, type  
AREXX "path", "comment", mode, type

(in list)  
(in list)

## REGISTERS

nodeType = TALK\_AREXX

if type is ARGUMENT\_COMMAND:

objectPath = command string  
extraData[0] = comment  
extraData[50] = portname

else if type is ARGUMENT\_SCRIPT:

objectPath/Name = path  
extraData = comment  
  
numericalArgs[0] = mode (ARGUMENT\_DEFER, ARGUMENT\_CONTINUE)  
numericalArgs[1] = type (ARGUMENT\_COMMAND, ARGUMENT\_SCRIPT)  
duration  
~~dayBits, startDays, startMinutes, startTicks, endDays, endMinutes, endTicks,~~  
~~startendMode,~~

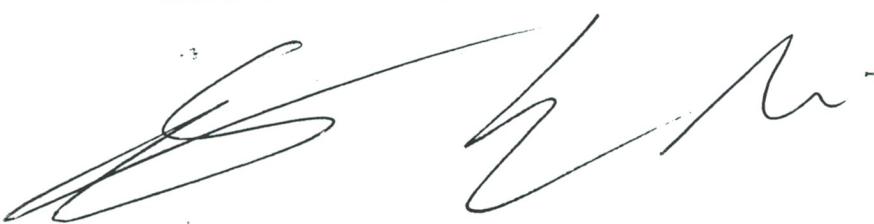
## POSITION

Between START and END.

## FUNCTION

Tries to find ARexx in the system and passes the path to an ARexx script executer. The mode parameter can be DEFER or CONTINUE indicating whether the SD should pause until completion of the command or continue with the next object. The duration parameter should always be taken into account regardless the mode.

---



/

## SYNOPSIS

BINARY "path"

(in list)

## REGISTERS

nodeType = TALK\_BINARY  
objectPath/Name = path

## POSITION

Between SCRIPPTALK and the first STARTSER.

## FUNCTION

Nothing more than a placeholder of an application or file. It can be used to transfer programs and data to the Script Driver and should not be executed by it.

---

A large, handwritten signature is written across the page, starting from the left edge and extending towards the right. The signature is fluid and cursive, appearing to be a combination of letters and numbers.

## SYNOPSIS

DOS 'command string', "comment", mode, type  
DOS "path", "comment", mode, type

(in list)  
(in list)

## REGISTERS

nodeType = TALK\_DOS

if type is ARGUMENT\_COMMAND:

objectPath = command string  
extraData[0] = comment

else if type is ARGUMENT\_SCRIPT:

objectPath/Name = path  
extraData = comment

numericalArgs[0] = mode (ARGUMENT\_DEFER, ARGUMENT\_CONTINUE)

numericalArgs[1] = type (ARGUMENT\_COMMAND, ARGUMENT\_SCRIPT)

duration

dayBits, startDays, startMinutes, startTicks, endDays, endMinutes, endTicks,

startendMode,

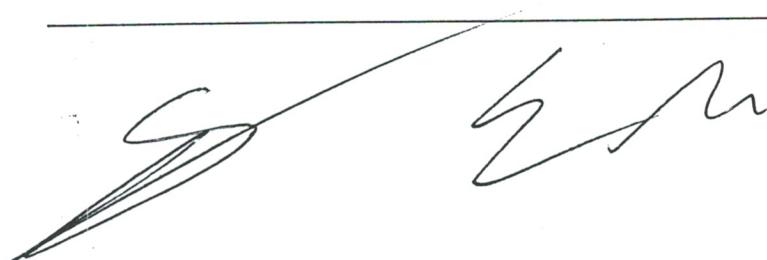
## POSITION

Between START and END.

## FUNCTION

Executes a DOS command or script. The mode parameter can be DEFER or CONTINUE indicating whether the SD should pause until completion of the command or continue with the next object. The duration parameter should always be taken into account regardless the mode.

---



## SYNOPSIS

**DURATION hh:mm:ss:t**

## POSITION

Inside a START/END or STARTPAR/ENDPAR block.

## FUNCTION

This function specifies for what period the current event should stay on screen. In case of a parallel event it defines how long the whole event may take.

---

## SYNOPSIS

**EFFECT n**

## POSITION

Between START and END.

## FUNCTION

The n parameter corresponds to an effect number.

---

## SYNOPSIS

**END**

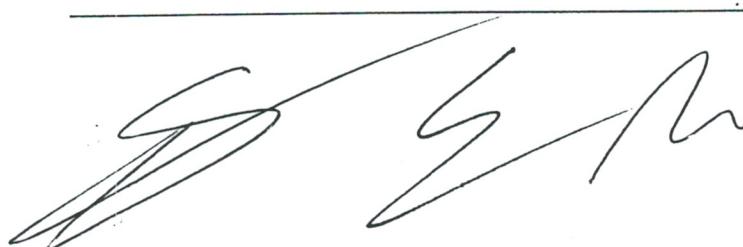
## POSITION

Inside a STARTSER/ENDSER or STARTPAR/ENDPAR block.

## FUNCTION

Indicates the end of a collection of commands which make up one event.

---



**SYNOPSIS**  
**ENDSER**

**POSITION**

See remarks in introduction.

**FUNCTION**

Ends a serial event.

---

**SYNOPSIS**  
**ENDPAR**

**POSITION**

See remarks in introduction.

**FUNCTION**

Ends a parallel event.

---

## SYNOPSIS

**GLOBALEVENT "c", "label"**  
**GLOBALEVENT key\_name, "label"**

## REGISTERS (in ScriptEventRecord)

keyCode	= c (or -1)
rawkeyCode	= key_name (or -1)
labelName	= label

## POSITION

Between SCRIPPTALK and the first STARTSER.

## FUNCTION

This command specifies to which key codes the SD should listen and to which labels it should jump when a key code enters the input event 'food processing chain'. Keys can be specified as a single character e.g. as "b" or by using a key name like HELP\_KEY or F1\_KEY. All the global events can be found in the ScriptInfoRecord which points to an array of ScriptEventRecords. In these SERs the fields 'keyCode', or 'rawkeyCode' are filled with the value of the assigned key. If the "c" representation is used then the keyCode field is filled with the ASCII value and when a key name is used the rawkeyCode field is filled with the raw key value. A maximum of MAX\_GLOBALLOCAL\_EVENTS can be stored in the array of SERs. The event handler should treat the typed keys in a case insensitive manner. Keys which can be defined on the main keyboard are: A...Z, 0...9, F1...F10, Help, Esc, cursor keys, Tab, Del, Backspace, Return and the Spacebar. On the numeric keypad: Enter, 0...9, (, ), /, \*, -, +, . (dot). Note: the return and enter key should be treated as one key, the same goes for the other keys on the numeric keypad which have an equivalent on the main keyboard (A600s don't have a keypad!).

(in list)

## SYNOPSIS

**GOTO "string"**

## REGISTERS

nodeType	= TALK_GOTO
objectName	= string
extraData	= pointer to ScriptNodeRecord

## POSITION

Between START and END.

## FUNCTION

Jumps to a label named 'string'. A goto can point to a TALK\_LABEL, TALK\_STARTSER or TALK\_STARTSER object.

---

## SYNOPSIS

**LABEL "string"**

(in list)

## REGISTERS

nodeType	= TALK_LABEL
objectName	= string

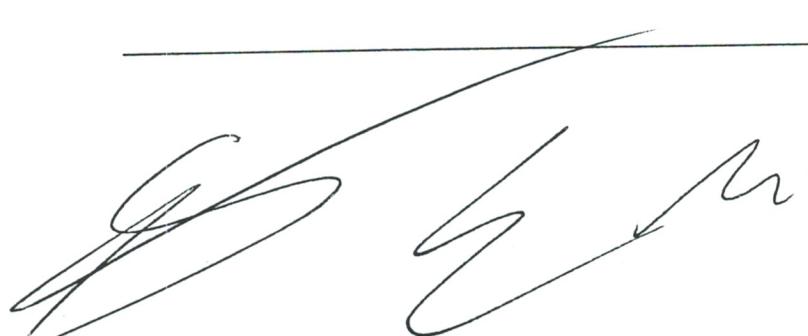
## POSITION

Between START and END.

## FUNCTION

This object holds a unique string where the SD looks for when a jump is carried out. For example, pressing a key defined as a GlobalEvent can carry out a jump to a label.

---



## SYNOPSIS

**MAIL "path"**

(in list)

## REGISTERS

nodeType = TALK\_MAIL  
objectPath/Name = path

## POSITION

Between SCRIPTTALK and the first STARTSER.

## FUNCTION

See BINARY for an explanation.

---

## SYNOPSIS

**PAGE "path", cycle**

(in list)

## REGISTERS

nodeType = TALK\_PAGE  
objectPath/Name = path  
numericalArgs[0] = cycle (ARGUMENT\_ON, ARGUMENT\_OFF)  
NYI { numericalArgs[1] = ARGUMENT\_PAGE or ARGUMENT\_ILBM }  
duration  
dayBits, startDays, startMinutes, startTicks, endDays, endMinutes, endTicks,  
startendMode  
effectNr

## POSITION

Between START and END.

## FUNCTION

Loads a page (structured document or IFF picture) into memory, renders it in an off-screen buffer and displays it on screen using the current effect number. Note that nodeType may hold two different node types.

---



## SYNOPSIS

**PROGRAM dayBits**

**PROGRAM dayBits, startDay, endDay, startTime, endTime, mode**

**PROGRAM startTime, endTime**

## POSITION

Inside a START/END, STARTSER/ENDSER or STARTPAR/ENDPAR block.

## FUNCTION

This function defines whether an object should be displayed or not. It can be applied to one object or to several by programming a sub script (serial or parallel events). The dayBits can be SU|MO|TU|WE|TH|FR|SA, the dates are written as dd-mmm-yyyy (e.g. 28-mar-1992), the times as hh:mm:ss and the mode can be CYCLICAL or CONTINUOUS.

---

## SYNOPSIS

**SCRIPTTALK version, revision**

## POSITION

First line of every ScriptTalk script. It must appear at the very top of the file.

## REGISTERS (in ScriptInfoRecord)

version	= version
revision	= revision

## FUNCTION

This function must be the very first line in every ScriptTalk script. Scripts without it aren't valid. The version and revision parameters can be found in the ScriptInfoRecord which accompanies every opened script. Note: unlike any other ScriptTalk command this command must be written in capitals and must appear at the very top at the first position of the script file.

---



## SYNOPSIS

**SOUND "path", type, volume [ , freq, loop ]**

(in list)

## REGISTERS

nodeType	= TALK_SOUND
objectPath/Name	= path
numericalArgs[0]	= type
numericalArgs[1]	= volume (0-64)
numericalArgs[2]	= frequency or -1
numericalArgs[3]	= loops or -1
duration	
dayBits, startDays, startMinutes, startTicks, endDays, endMinutes, endTicks,	
startendMode,	

## FUNCTION

This function loads a sample or a sound track and plays it. The freq and loop parameters are only necessary when playing raw sound samples.

---

## SYNOPSIS

**START**

## POSITION

Inside a STARTSER/ENDSER or STARTPAR/ENDPAR block.

## FUNCTION

Indicates the start of a collection of commands which make up one event.

---



(in list)

## SYNOPSIS

**STARTSER [ "string" ]**

## REGISTERS

nodeType = TALK\_STARTSER  
objectName = string  
dayBits, startDays, startMinutes, startTicks, endDays, endMinutes, endTicks,  
startendMode

## POSITION

See remarks in introduction.

## FUNCTION

Starts a serial event.

---

## SYNOPSIS

**STARTPAR [ "string" ]**

(in list)

## REGISTERS

nodeType = TALK\_STARTPAR  
objectName = string  
duration  
dayBits, startDays, startMinutes, startTicks, endDays, endMinutes, endTicks,  
startendMode

## POSITION

See remarks in introduction.

## FUNCTION

Starts a parallel event.

---



**SYNOPSIS****TIMECODE source, format, rate, mode, offset**

(in list)

**REGISTERS (in ScriptInfoRecord)**

timeCodeSource	=	TIMESOURCE_INTERNAL or TIMESOURCE_EXTERNAL
timeCodeFormat	=	TIMEFORMAT_HHMMSS, TIMEFORMAT_MIDI or TIMEFORMAT_SMPTE
timeCodeRate	=	TIMERATE_24FPS, TIMERATE_25FPS, TIMERATE_30FPS_DF, TIMERATE_30FPS
NYI { Offset	=	offset }
NYI { timeCodeOut	=	mode (TRUE or FALSE) }

**POSITION**

Between START and END.

**FUNCTION**

Defines the time code type and related definitions.

## SYNOPSIS

**USERAPPLIC "tool name", "comment", 'argument string', mode**

(in list)

## REGISTERS

nodeType	= TALK_USERAPPLIC
objectPath	= tool name
objectName	= comment
extraData	= argument string
numericalArgs[0]	= mode (1=DEFER, 2=CONTINUE)
duration	
dayBits, startDays, startMinutes, startTicks, endDays, endMinutes, endTicks,	
startendMode	

## POSITION

Between START and END.

## FUNCTION

This is only for ace users and for 1001 internal use. A USERAPPLIC is a two-part application for which we supply a framework in C. When a user wants to play e.g. a clip from a laserdisc he can write his own application using the framework which opens a window on the Script Editor screen. The mode parameter can be DEFER or CONTINUE indicating whether the SD should pause until completion of the command or continue with the next object. The duration parameter should always be taken into account regardless the mode.

A handwritten signature consisting of stylized letters 'G', 'E', and 'M'.

## PageTalk command summary

### SYNOPSIS

**CLIP "path", x, w, h, scale**

### POSITION

Inside an OBJECTSTART/OBJECTEND block.

### REGISTERS

nodeType	= TALK_CLIP
objectPath/Name	= path
numericalArgs[0]	= x (x co-ordinate)
numericalArgs[1]	= y (y co-ordinate)
numericalArgs[2]	= w (width)
numericalArgs[3]	= h (height)
numericalArgs[4]	= scale (0=off, 1=on)

### FUNCTION

This function loads an ILBM brush and places it into the last opened window. The x,y,w and h values indicate where (relative to the window) the clip should be placed and how wide and high the clip is.

---

### SYNOPSIS

**FONT "path", size**

### POSITION

Inside an OBJECTSTART/OBJECTEND block.

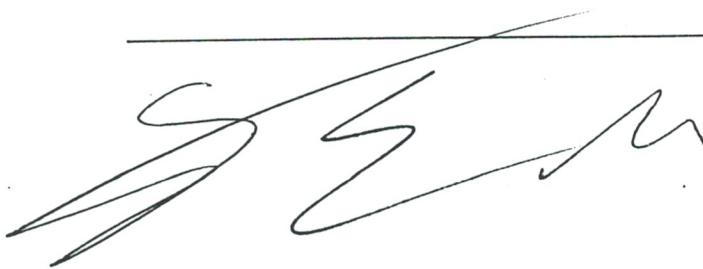
### REGISTERS

nodeType	= TALK_FONT
objectPath/Name	= path
numericalArgs[0]	= size

### FUNCTION

This function loads the font described by path (size included) and makes it the current font.

---



**SYNOPSIS**

**LOCALEVENT "c", "label", x, y, w, h, rendertype, cue**  
**LOCALEVENT key\_name, "label", x, y, w, h, highlighttype, cue**

**POSITION**

Inside an OBJECTSTART/OBJECTEND block.

**REGISTERS (in ScriptEventRecord)**

keyCode	= c (or -1)
rawkeyCode	= key_name (or -1)
labelName	= label
x	= x (x co-ordinate)
y	= y (y co-ordinate)
width	= w (width)
height	= h (height)
renderType	= rendertype (1=NONE, 2=COMPLEMENT, 3=BORDER)
audioCue	= cue (1=OFF, 2=ON)

**FUNCTION**

This command specifies to which key codes the SD should listen and to which labels it should jump when a key code enters the input event 'food processing chain'. Also, a button can be defined on which the mouse pointer can be placed. Pressing the left mouse button inside the button causes a jump to be carried out. Keys can be specified as a single character e.g. as "b" or by using a key name like HELP\_KEY or F1\_KEY. All the local events can be found in the ScriptInfoRecord which points to an array of ScriptEventRecords. In these SERs the fields 'keyCode', 'rawkeyCode' and 'labelName' are filled with the value of the assigned key. If both the keyCode and rawkeyCode fields are negative (-1) then no key has been assigned. If the "c" representation is used then the keyCode field is filled with the ASCII value and when a key name is used the rawkeyCode field is filled with the raw key value. A maximum of MAX\_GLOBALLOCAL\_EVENTS can be stored in the array of SERs. The event handler should treat the typed keys in a case insensitive manner. Notez bien: a key need not be defined, it may be negative (-1) as well. In this case only the mouse is monitored.

A

## SYNOPSIS

### PAGETALK version, revision

## POSITION

First line.

## REGISTERS (in ScriptInfoRecord)

version	= version
revision	= revision

## FUNCTION

This function must be the very first line in every PageTalk script. Scripts without it aren't valid. The version and revision parameters can be found in the ScriptInfoRecord which accompanies every opened script. Note: unlike any other PageTalk command this command must be written in capitals and must appear at the very top at the first position of the page description file.

---

## SYNOPSIS

### PALETTE hhh,...,hhh

## POSITION

Inside an OBJECTSTART/OBJECTEND block.

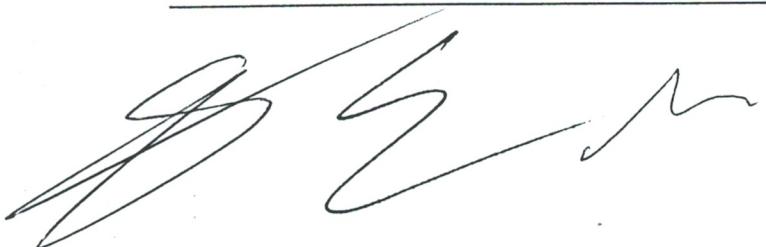
## REGISTERS

nodeType	= TALK_PALETTE
extraData	= colormap (array of bytes (r,g,b,...,r,g,b))
extraDataSize	= size of colormap

## FUNCTION

This function uses the array of hexadecimal color values for the color map of the currently opened screen.

---





## SYNOPSIS

**SCREEN width, height, numColors, mode**

## POSITION

Inside an OBJECTSTART/OBJECTEND block.

## REGISTERS

nodeType	= TALK_SCREEN
numericalArgs[0]	= width
numericalArgs[1]	= height
numericalArgs[2]	= numColors
numericalArgs[3]	= mode (1=lores, 2=lores ham, 3=lores ehb, 4=lores lace, 5=lores ham lace, 6=lores ehb lace, 7=hires, 8=hires lace, 9=super, 10=super lace, 11=productivity, 12=productivity lace)

## FUNCTION

This function opens a screen using the supplied parameters. The mode number is a number representing e.g. HIRES|LACE.

---

## SYNOPSIS

**TEXT "path"**

## POSITION

Inside an OBJECTSTART/OBJECTEND block.

## REGISTERS

nodeType	= TALK_TEXT
objectPath/Name	= path

## FUNCTION

This function loads, interprets and renders the text into the current window. The text file can be plain ASCII or a structured file (FTXT?) with additional information on color, style and justification.

---



## SYNOPSIS

WINDOW x,y,w,h,tbc,rbc,bbc,bbc,lbc,bw,ic,it

## POSITION

Inside an OBJECTSTART/OBJECTEND block.

## REGISTERS

nodeType	= TALK_WINDOW
numericalArgs[0]	= x (x co-ordinate)
numericalArgs[1]	= y (y co-ordinate)
numericalArgs[2]	= w (width)
numericalArgs[3]	= h (height)
numericalArgs[4]	= tbc (top border color register)
numericalArgs[5]	= rbc (right border color register)
numericalArgs[6]	= bbc (bottom border color register)
numericalArgs[7]	= lbc (left border color register)
numericalArgs[8]	= bw (border width in pixels)
numericalArgs[9]	= ic (interior color register)
numericalArgs[10]	= it (interior type, 1=FILLED, 2=PATTERN, 3=TRANSP)

## FUNCTION

This function defines a window with x,y,w and h as its dimensions. The other parameters are tpc (TopBorderColor), bw (BorderWidth), ic (InteriorColor) and it (InteriorType). The last parameter can be FILLED, PATTERN or TRANSP.



pagina 41.

✓

Uitgegeven voor eerste afschrift  
met daaraan gehechte stukken.



✓