

HTMLread

Gunter Hinrichsen

COLLABORATORS

	<i>TITLE :</i> HTMLread		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Gunter Hinrichsen	November 7, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	HTMLread	1
1.1	Willkommen bei HTMLread.guide	1
1.2	EMail is watching you!	1
1.3	Erste Hilfe	2
1.4	Wie benutze ich clearCache?	3
1.5	Wie benutze ich DownloadPrefs?	4
1.6	Wie benutze ich DownloadPrefs?	5
1.7	Wie benutze ich DownloadPrefs?	5
1.8	Wie benutze ich HTMLreadPrefs?	5
1.9	Wie benutze ich DownloadPrefs?	5
1.10	Wie benutze ich DownloadPrefs?	6
1.11	Wie benutze ich DownloadPrefs?	6
1.12	Wie benutze ich DownloadPrefs?	6
1.13	Wie benutze ich DownloadPrefs?	6
1.14	Wie benutze ich DownloadPrefs?	6
1.15	Wie benutze ich DownloadPrefs?	6
1.16	Wie benutze ich DownloadPrefs?	6
1.17	Wie benutze ich DownloadPrefs?	7
1.18	Wie benutze ich DownloadPrefs?	7
1.19	Wie benutze ich DownloadPrefs?	7
1.20	Wie benutze ich DownloadPrefs?	7
1.21	Wie benutze ich DownloadPrefs?	7
1.22	Wie benutze ich DownloadPrefs?	7
1.23	Wie benutze ich DownloadPrefs?	7
1.24	Wie benutze ich DownloadPrefs?	8
1.25	Erklärung, was macht HTMLread eigentlich?	8
1.26	Was braucht HTMLread?	8
1.27	Wie Installiere ich HTMLread?	8
1.28	Wie benutze ich HTMLreadPrefs?	10
1.29	Wie benutze ich HTMLreadPrefs?	11

1.30	Wie benutze ich HTMLreadPrefs?	11
1.31	Wie benutze ich HTMLreadPrefs?	11
1.32	Wie benutze ich HTMLreadPrefs?	12
1.33	Wie benutze ich HTMLreadPrefs?	12
1.34	Wie benutze ich HTMLreadPrefs?	12
1.35	Wie benutze ich HTMLreadPrefs?	12
1.36	Wie benutze ich HTMLreadPrefs?	13
1.37	Wie benutze ich HTMLreadPrefs?	13
1.38	Wie benutze ich HTMLreadPrefs?	14
1.39	Wie benutze ich HTMLreadPrefs?	14
1.40	Wie benutze ich HTMLreadPrefs?	14
1.41	Wie benutze ich HTMLreadPrefs?	15
1.42	Wie benutze ich HTMLreadPrefs?	15
1.43	Wie benutze ich HTMLreadPrefs?	15
1.44	Wie benutze ich HTMLreadPrefs?	16
1.45	Wie benutze ich HTMLreadPrefs?	16
1.46	Wie benutze ich HTMLreadPrefs?	16
1.47	Wie benutze ich HTMLreadPrefs?	17
1.48	Wie benutze ich HTMLreadPrefs?	17
1.49	Wie benutze ich HTMLreadPrefs?	17
1.50	Wie benutze ich HTMLreadPrefs?	17
1.51	Wie funktioniert HTMLread.rexx?	18
1.52	Der Optimizer!	18
1.53	Welche Fehlermeldungen gibt es?	18
1.54	DISCLAIMER	19
1.55	Wer hatt es gemacht?	20
1.56	Welche Zukunft erwartet HTMLread noch?	20
1.57	Was ist passiert?	21
1.58	Welche Problem gibt es mit HTMLread?	27
1.59	Vielen Dank!	27
1.60	Wie konfiguriere ich das GenesisPlugin?	27

Chapter 1

HTMLread

1.1 Willkommen bei HTMLread.guide

HTMLRead 1.1

<http://www.htmlread.50g.com>

Written by Gunter Hinrichsen

Was macht HTMLread und wer braucht es?
Was brauche ich für HTMLread?
Wie installiere ich HTMLread?

Wie funktioniert HTMLread?
Fehlermeldungen
DISCLAIMER
Wer hatts gemacht?
Zukunft?
Geschichte...
Bekannte Probleme
Dank und Grüße!

Hilfe HTMLread läuft nicht!

Zusätzliche Programme

HTMLreadPrefs
Das GenesisPlugin
clearCache
DownloadPrefs
Der Optimizer (entfernt)

Spionage durch HTML-E-mails

Wie es funktioniert und wie schützt HTMLread davor?

1.2 EMail is watching you!

Spionage durch HTML-E-mails

HTMLE-mails bieten nicht nur ein schöneres Aussehen der E-mails man kann durch geschickte Programmierung auch den Leser solcher E-mails ausspionieren. So kann man feststellen wann und ob eine E-mail geöffnet/gelesen wurde oder eben nicht. Für soetwas gibt es sogar richtige Dienste im Internet. Ein Kandidat ist zum Beispiel <http://www.itraceyou.com>. Der Trick wie soetwas geht ist ganz einfach. In den normalen HTMLcode der E-mail wird ein unsichtbares Bild eingebaut (1x1 Pixel gross). Wenn nun die E-mail geöffnet wird versucht HTMLread oder z.B. ein PC-Mailer nun genau dieses Bild zu laden und gibt damit preis ob und wann diese E-mail gelesen wurde. Es kann aber auch z.B. ein JavaScript ausgeführt werden das Daten weitergeben kann.

Den einzigen Schutz den man gegen solche Schnüffelfrei hat ist dieses Bild eben nicht zu laden und JavaScript im Browser auszuschalten. Aus diesem Grund besitzt HTMLread nun eine Funktion Bilder die von bestimmten URLs kommen nicht zu laden.

Die URLs sind in einer normalen TextDatei gespeichert. Sie befindet sich im Ordner »YAM:« und heißt »badurls.htmlread«.

Das Format der Datei ist einfach. Man muss nur die URL ohne das <http://www.> eintragen. Bei <http://www.itraceyou.com> heißt der Eintrag:
itraceyou.com

Nun wird HTMLread nicht mehr Bilder, die von dieser Adresse kommen, laden und der böse "Agent" kann nicht mehr feststellen ob du die E-mail gelesen hast oder nicht.

Der Nachteil ist das die URLs bekannt sein müssen, damit HTMLread davor schützen kann. Darum rufe ich alle auf, mir sofort die URLs von solchen Leuten/Diensten mitzuteilen. Ich werde dann auf meiner HomePage <http://www.htmlread.50g.com> eine geupdatete BadUrl-Liste veröffentlichen.

1.3 Erste Hilfe

Hilfe HTMLread läuft nicht!

Hier eine CheckListe arbeitete jeden Punkt ab wenn es trotzdem nicht klappen sollte eine E-mail an mich.

Problem: Es wird eine HTML-E-mail angewählt aber nicht passiert:

- Starte HTMLreadPrefs
- Setze den Punkt »Show HTML-MAIL« auf »ALL«
- Überprüfe in der MUI-Konfig für die Programme YAM und deinen HTMLviewer ob der ARexxPort angeschaltet ist!
- Überprüfe die YAM-Konfig siehe dazu Wie installiere ich HTMLread? (insbesondere Punkt f)
- Überprüfe ob sich im C: Verzeichniss HTMLread befindet

1.4 Wie benutze ich clearCache?

clearCache

Yam lässt es leider nicht das man beim Löschen einer Nachricht eine Aktion starten kann und so bleiben evt. Bilder dieser Nachricht im CacheVerzeichniss und verschwenden unnötig Festplattenplatz. clearCache ist ein Skript das alle Nachrichten im Cache einliest und überprüft ob diese EMail's überhaupt noch existieren und wenn nicht diese sinnlos gewordenen Bilder löscht. Dazu gibt es drei Möglichkeiten:

Bild: clearCache

Purge ALL: Löscht sämtliche Bilder aus dem Cache. Achtung es wird der Befehle delete cachePfad/#? ALL verwendet. Seien sie sich absolut sicher das der CachePfad für sich allein steht wie MAIL:HTMLread/. Wenn z.B. nur MAIL: stehen würde wird alles auf MAIL: gelöscht und nicht nur die Bilder. (!!Verwendung dieser Option nur auf eigene Gefahr!!)

Fast check: Fast check setzt vorhandene Index-Files voraus die normalerweise automatisch mit angelegt worden sind. In diesem Index-Files stehen Daten die das Überprüfen wesentlich beschleunigen. Das funktioniert aber nur dann wenn die EMail's noch in dem Folder sind wo sie beim öffnen waren. Also Wenn man eine Mail im Eingang geöffnet hatt und sie im Eingang geblieben ist funktioniert »Fast check«. Wenn man sie aber zwischendurch in Archiv verschoben hatt meint clearCache das sie nicht mehr vorhanden ist und will die Bilder löschen. In diesem Fall muss man »Comlett check« benutzen, das Index-File wird dadurch ebenfalls ge-updatet so das man beim nächstenmal wieder »Fast check« benutzen kann.

Bild: Fast check

Complet check: Comlet check ist die sicherste Art zu überprüfen ob eine EMail noch vorhanden ist. Die EMail wird über ihrer MSGID erkannt, egal an welchem Ort sie liegt. Ausserdem werden die Index-Files die für »Fast check« gebraucht werden gegenfalls geupdatet. Allerdings ist Complet check auch die langsamste Methode da sämtliche Nachrichten in YAM überprüft werden müssen.

Bild: Complet check

Sollte eine Nachricht gefunden werden, die schon gelöscht worden ist doch ihre Bilder sind noch im Cache, erscheint ein Requester wo man gefragt wird was zu tun ist:

Bild: Complet check

YES : Löscht die Bilder dieser Nachricht

ALL : Löscht absofort ohne vorher zufragen

ALL in this Folder : Löscht absofort alle Bilder der Nachrichten die nicht gefunden wurden ohne vorher zufragen für diesen aktuellen Folder. (Nur in »Fast check« vorhanden)

Break : Breche ab

NO : Lösche die Bilder nicht.

1.5 Wie benutze ich DownloadPrefs?

DownloadPrefs

Wenn man Bilder in die HRDownload.liste gespeichert hatt, will man natürlich auch diese Liste verwalten. Dieses macht das DownloadPrefs.

Bild: DownloadPrefs

```

Das Menü von DownloadPrefs
+-----+
|+-----+|
|| Folder: ( / ):      ||
||                     ||
||                     ||
<<
<
>
>>
END      ||
        ||+-----+|| | |
        ||| STATUS...:      (Number: / )      |||
        ||| FROM.....:      |||
        ||| SUBJECT...:      |||
        ||| MAILPART.:      |||
        ||| DATE.....:      |||
        |||                     TOP
<<
<
>
>>
END      |||
        |||                     DELETE
READ
WAIT
LOAD      |||
        ||+-----+||
        ||+-----+||
        ||                     SAVE
QUIT      |

```



```
+-----+
```

1.6 Wie benutze ich DownloadPrefs?

SAVE

Speichert die HRDownload.liste.

1.7 Wie benutze ich DownloadPrefs?

QUIT

Beendet DownloadPrefs.

1.8 Wie benutze ich HTMLreadPrefs?

Das Menü von DownloadPrefs:

```
+-----...
| Project      |
|-----...
| About        |
+-----+
| Open         |
| Last Saved   |
| Save As      |
|-----|
| Quit         |
+-----+
```

Project:

About : Gibt die Version des Programmes an.
Open : Lädt die HRDownload.liste mit einem andern Namen.
Last Saved : Lädt die letzte HRDownload.liste.
Save As : Speichert die HRDownload.liste mit einem anderen Namen.
Quit : Beendet DownloadPrefs.

1.9 Wie benutze ich DownloadPrefs?

LOAD

Setzt den STATUS der Nachricht auf LOAD.
Die Bilder aus dieser Nachricht werden beim nächstenmal geladen

1.10 Wie benutze ich DownloadPrefs?

WAIT

Setzt den STATUS der Nachricht auf WAIT.
Die Bilder aus dieser Nachricht werden beim nächstenmal nicht geladen
sondern übersprungen.

1.11 Wie benutze ich DownloadPrefs?

READ

DownloadPrefs gibt YAM den Auftrag die aktuelle Mail anzuzeigen.

1.12 Wie benutze ich DownloadPrefs?

DELETE

Damit entfernt man die aktuelle Mail aus der Liste.

1.13 Wie benutze ich DownloadPrefs?

END

Springe ans Ende.

1.14 Wie benutze ich DownloadPrefs?

>>

Springe fünf vor.

1.15 Wie benutze ich DownloadPrefs?

>

Ein vor.

1.16 Wie benutze ich DownloadPrefs?

<

Ein zurück.

1.17 Wie benutze ich DownloadPrefs?

<<

Springe fünf zurück.

1.18 Wie benutze ich DownloadPrefs?

TOP

Springe an den Anfang.

1.19 Wie benutze ich DownloadPrefs?

DATE

Hier wird angezeigt von wann die Mail war.

1.20 Wie benutze ich DownloadPrefs?

MAILPART

Hier wird der MailPart angezeigt für den die Bilder geladen werden sollen.

1.21 Wie benutze ich DownloadPrefs?

SUBJECT

Hier wird der Betreff der Mail angezeigt.

1.22 Wie benutze ich DownloadPrefs?

FROM

Hier wird der Absender von dem diese Mail stammt angezeigt

1.23 Wie benutze ich DownloadPrefs?

STATUS

Hier wird der Status der Mail angezeigt:

LOAD : Beim nächstenmal werden die Bilder aus dieser Mail geladen

WAIT : Beim nächstenmal werden die Bilder aus dieser Mail übersprungen

In Klammern wird angezeigt welche Nummer die Mail hatt und wieviele Mails es insgesamt in diesem Folder gibt.

1.24 Wie benutze ich DownloadPrefs?

Folder

Hier wird der aktuelle Folder angezeigt aus dem die aktuelle Mail stammt. Die Mails sind nach Folder geordnet.

In den Klammern steht welche Nummer der aktuelle Folder hatt und wieviele Folder es insgesamt gibt.

1.25 Erklärung, was macht HTMLread eigentlich?

Was macht HTMLread.rexx und wer braucht es?

HTMLread.rexx ist ein ein ARExx-Skript das ein lästiges Problem von YAM behebt. YAM ist nämlich nicht in der Lage HTML-Mails anzuzeigen, die oft von Netscape oder dem IE versandt werden. Es ist wegen den HTML-Tags nur sehr mühsam eine solche HTML-MAIL zulesen. Allerdings ist es möglich eine solche MAIL zu erkennen und dann mit einen HTML-Anzeiger zu lesen. Genau dieses automatisiert dieses ARExx-Skript. Das heißt wenn man beim lesen auf solch eine MAIL stößt erkennt das HTMLread.rexx blitzschnell und startet den HTML-Anzeiger. Jeder der hin und wieder mal solche HTML-MAILs bekommt und sie auch ansehen will kann dieses Skript also gebrauchen.

1.26 Was braucht HTMLread?

Was brauche ich für HTMLread.rexx?

- HTMLread selber braucht nur YAM 2.1
- Der DownloadManager braucht rexxreqtools.library und MicroRexx.
- HTMLreadPrefs braucht BetterString.mcc (und natürlich MUI)

Im Archiv ist alles vorhanden und wird mit dem Installer mitinstalliert. (außer MUI)

1.27 Wie Installiere ich HTMLread?

Wie installiere ich HTMLread.rexx?

- a) Benutze den Installer
- b) Starte YAM oder drücke hier: run YAM:YAM
- c) Starte das KonfigurationsFenster
- d) Wähle aus der Liste links ARexx
- e) Wähle nun aus der Liste rechts:
Beim Lesen einer Nachricht
- f) Wähle nun ARexx und trage bei Skript ein:
YAM:Rexx/checkHTML.rx

Achtung! Wenn HTMLread nicht auf EMailS reagieren sollte die HTMLcode haben kann es sein das in checkHTML.rx ein falscher Pfad für HTMLread.rx eingetragen ist. Oder die ENV-Variable HTMLread falsch gesetzt ist. Um sie richtig zusetzen benutzen Sie die beiden Knöpfe:

Setze ENV:HTMLread Öffnet FileRequester und setzt ENV:HTMLread
Kopiere nach ENVARC: Kopiert ENV:HTMLread nach ENVARC:

Wenn die Zeile 4 »HTMLread=""« ist wird die ENV-Variable benutzt andernfalls der dort eingetragende Pfad.

z.B. »HTMLread="YAM:Rexx/HTMLread.rx"«

Man sollte der ENV-Variable den Vorzug geben. (also »HTMLread=""«)

Starte Editor:

AmigaOS 3.5 / 3.9 SYS:Tools/EditPad YAM:Rexx/checkHTML.rx

AmigaOS 2.x / 3.x C:ed YAM:Rexx/checkHTML.rx

Bild: checkHTML.rx

Eine dritte Möglichkeit ist es in der YAM-Konfig anstatt YAM:Rexx/checkHTML.rx auch YAM:Rexx/HTMLread.rx einzutragen. Das verlangsamt allerdings das checken auf HTMLcode.

- g) HTMLread benutzt normalerweise ein KonfigFile (ENV:HTMLread.config)
Nur wenn ein anderes benutzt werden soll muß der Name des anderen Konfigfile in der Zeile Skript hinter YAM:Rexx/checkHTML.rx angehängt werden. (Mit einem Space zwischen beiden natürlich)
- h) Sorge dafür das die beiden Knöpfe: »Öffne Konsolenfenster« und »Warte auf Beendigung« NICHT angeklickt sind.

Bild: YAM-Window

- i) Wähle nun aus der Liste rechts:
»Nach dem holen von Post«
- j) Wähle nun ARexx und trage bei Skript ein:
YAM:Rexx/HRdownload.rx
- k) Sorge dafür das der Knopf: »Öffne Konsolenfenster« nicht angeklickt ist!
Sorge dafür das der Knopf »Warte auf Beendigung« angeklickt!

Bild: YAM-Window

- l) Wer will kann genau das gleiche (Punkt i) bis Punkt k)) in dem Punkt »Nach dem Versenden von Post« eintragen.
 - m) Speichere nun ab.
-

n) Starte nun den HTMLreadPrefs. Du findest ihn unter YAM:HTMLread/

1.28 Wie benutze ich HTMLreadPrefs?

HTMLreadPrefs

Es gibt zwei Möglichkeiten HTMLread zusagen welchen HTML-Anzeiger er benutzen soll, einmal von Hand das Konfigfile zu ändern, oder die wesentlich einfachere den HTMLreadPrefs zu benutzen.

HTMLreadPrefs benutzt MUI mit der BetterString.mcc. BetterString kann aus dem AmiNet geladen werden oder von der Homepage:
http://www.p2p-it.dk/duff/MCC_BetterString.LZX

Main:

Bild: HTMLreadPrefs

Das Menü von HTMLreadPrefs

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
		Viewer							
		+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+							
			@	Viewer	Viewername				
		+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+							
		Path Viewer	:	Path Viewer		#			
		Startarguements:	:	StartArgs		/			
		ARexxPort	:	ARexxPort		/			
		PublicScreen	:	PublicScreen		/			
		REXX Open-File	:	Open-File		/			
		REXX NewWindow	:	NewWindow		/			
		Show HTML	:	@ Show HTML					
		+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+							
		Save				Use	Quit		
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									

HTTP-Mail:

Bild: HTMLreadPrefs

Das Menü von HTMLreadPrefs

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
		HTTP-Mail							
		+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+							
		LoadWebImage	:	@ LoadWebImage					
		Make Imagelist	:	@ Make Imagelist					
		SaveWebImage	:	@ SaveWebImage					
		Quiet	:	@ Quiet					
		Cachepath	:	Cachepath		#			
		Downloadlist	:	Downloadlist		#			

```

| | Downloadmanager: Downloadmanager # | |
| | OnOffline      : OnOffline      # | |
| +-----+-----+-----+-----+ |
|                               Save  Use   Quit                               |
| +-----+-----+-----+-----+ |

```

1.29 Wie benutze ich HTMLreadPrefs?

Quiet

Mit Quiet veranlasst man den Downloadmanager nicht immer seine Statuswindows zuöffnen. Die Downloadinformationen stehen dann in der Statuszeile von YAM. Die Statuswindows werden nur dann geöffnet wenn man gerade eine Nachricht liest, nicht aber wenn man gerade Nachrichten empfangen hatt und der Downloadmanager gestartet wird. Wenn Quiet auf NO gesetzt ist werden die Statuswindows immer geöffnet.

1.30 Wie benutze ich HTMLreadPrefs?

Default

Wenn angewählt werden die gespeicherten Defaultwerte genommen. Die Checkmarks sind abgeschaltet wenn als Viewer »other« gewählt ist.

1.31 Wie benutze ich HTMLreadPrefs?

OnOffline

Das OnOfflinePlugin regelt das Onlinegehen, wenn man dieses wünscht, das heisst wenn man Offline ist und sich entschieden hatt die Bilder die im Web sind jetzt zu laden. (Siehe Make Imagelist)
Diese Plugin muss angegeben werden und es muss vom Typ OnOfflinePlugin sein.

Wenn du Miami(Dx) benutzt nimm MiamiOnOffPlugin. Diese Plugin überprüft ob MiamiDx vorhanden ist und falls ja versucht es mit MiamiDx Online zugehen. Falls MiamiDx nicht vorhanden ist wird das normale Miami genommen.
Das MiamiPlugin sollte sich im Pfad YAM:Rexx/ befinden und heisst »miamionoff.rx«.

Wenn du Genesis benutzt nimm das GenesisOnOffPlugin. Du findest es YAM:Rexx/ und es heisst »GenesisOnOff.rx«. Du musst jetzt nur noch das Interface einstellen. Gehe dazu zu Punkt Genesisplugin .

1.32 Wie benutze ich HTMLreadPrefs?

Downloadlist

Hier stellt man die Liste ein in der die Bilder gespeichert werden wenn man Offline ist. Man stellt dieses unter Make Imagelist ein. Es ist darauf zu achten das es ein Platz ist der nicht durch ein Reset verloren geht, wie zum Beispiel ENV: denn es liegt normalerweise in der Ram-Disk. ENVARC: das auf der Festplatte liegt ist zum Beispiel ein guter Ort für die Liste.

1.33 Wie benutze ich HTMLreadPrefs?

Downloadmanager

Der Downloadmanager ist das wichtigste Plugin von HTMLread. Es muss angegeben sein und vom Typ HTMLreadDownloadplugin sein. HTMLreadPrefs überprüft dieses und gibt evt. eine Fehlermeldung aus wenn es nicht richtig gesetzt ist.

Mit diesem DownloadManager lädt HTMLread die Bilder aus dem Web. Es sollte im Verzeichniss YAM:Rexx/ sein und es heisst »Hrdownload.rx«.

Zur Zeit wird zum Bilderladen YAM genutzt es ist aber geplant das es ein Plugin geben soll das dieses mit HTMLresume macht oder anderen Programmen.

Bild: HRDownload

1.34 Wie benutze ich HTMLreadPrefs?

Cachepath

Hier wird eingestellt wo die Bilder zwischen gespeichert werden sollen. Der Cachepfad muss in einem eigenen Verzeichniss liegen, es ist also verboten in nach MAIL: zulegen, da es sonst zu Datenverluste kommen kann!

Es ist also erlaubt ihn z.B. nach MAIL:HTMLread/ zulegen. Das Verzeichniss muss schon existieren. Also zum Beispiel vorher mit »mkdir mail:HTMLread« erstellen.

Die Einträge im in diesem Cachpfad kann man mit clearCaches verwalten.

1.35 Wie benutze ich HTMLreadPrefs?

SaveWebImage

Wenn ein HTML-EMail Bilder hatt die im Web sind können diese geladen werden. Man kann hier nun einstellen ob diese Bilder in einem Cache gespeichert werden sollen, damit man nicht immer Online sein muss um

die EMail mit den Bildern zu lesen. HTMLread findet selbstständig heraus ob ein Bild im Cache ist und setzt es automatisch für das Bild das im Web liegt ein. So kann man diese Nachricht auch Offline mit allen seinen Bildern lesen.

Die Bilder werden im Pfad der bei Cachepath angegeben ist zwischen- gespeichert. Mit dem Skript clearCache kann man diesen Cache verwalten.

Ask first: Fragt zuerst nach ob die Bilder gespeichert werden sollen

NO : Speichert die Bilder nicht im Cache sondern nur nach t: das sich normalerweise in der Ram-Disk befindet und die nach einem Reset gelöscht wird.
(Sinnvoll wenn es vertrauliche Bilder sind die nicht offen auf der Festplatte liegen sollen.)

YES : Speichert die Bilder zwischen

1.36 Wie benutze ich HTMLreadPrefs?

LoadWebImage

Es gibt HTML-EMails die Bilder haben, die nicht selbst in der EMail abgelegt sind sondern irgendwo im Web. Mit LoadWebImage kann man nun entscheiden ob HTMLread diese Bilder berücksichtigen soll, also aus dem Web laden soll oder nicht.

Mit Make Imagelist kann man einstellen wie HTMLread auf solche Bilder reagieren soll.

Wenn man die Bilder schon geladen hatt und sie im Cache gespeichert sind werden sie automatisch von HTMLread eingefügt.
(Selbstverständlich auch dann wenn man inzwischen LoadWebImage auf NO gestellt hatt.)

YES = Berücksichtige Bilder aus dem Web

NO = Lasse Bilder aus dem Web weg

1.37 Wie benutze ich HTMLreadPrefs?

Make Imagelist

Wenn HTMLread nun auf eine HTML-EMail stösst die Bilder enthält die im Web abgelegt sind und man Offline ist kann HTMLread verschieden darauf reagieren:

Ask first: - Man wird gefragt ob man Online gehen will
- oder ob man die Bilder in eine Liste speichern will die dann abgearbeitet wird wenn man dann Online ist
- oder ob man die Bilder einfach weglassen will.
Bild: Ask-Requester

YES : Bilder ohne zufragen automatisch in die Liste speichern

NO : Bilder grundsätzlich weglassen wenn man Offline ist, also nur Bilder laden wenn man sowieso Online ist.

1.38 Wie benutze ich HTMLreadPrefs?

Viewer:

Mit Viewer können sie einen Viewer aus der Liste mit vorgefertigten HTML-Anzeigern auswählen. Wenn ihr Anzeiger nicht in der Liste erscheint wählen sie »other«.

Nachdem man einen Anzeiger gewählt hat erscheint ein FileRequester wo man den Pfad und das Programm(unser HTML-Anzeiger) angeben muß.

Bild: FileRequester

1.39 Wie benutze ich HTMLreadPrefs?

Viewername:

Wenn man »other« gewählt hat kann man neben dem CycleGadget den Namen seines eigenen Viewers eintragen.

Bild: viewername

1.40 Wie benutze ich HTMLreadPrefs?

StartARG:

Bild: StartARG

Hier werden die Argumente festgelegt, die beim Starten des Anzeigers übergeben werden sollen.

Dieses brauchen Sie nur anzugeben wenn Sie:

- a) als Viewer »other« gewählt haben oder
- b) wenn Sie mit mit den voreingestellten Startargumenten nicht zufrieden sind.

Wenn sie einen Viewer aus der Liste ausgewählt haben können Sie dieses Feld frei lassen. HTMLread fügt automatisch die richtigen Argumente hinzu.

Wenn Sie aber doch ihre eigenen haben wollen gibt es zwei Variablen die HTMLread bereitstellt:

%ys gibt den Screen an den sie unter Punkt PublicScreen gewählt haben.

%hm gibt das HTML-Mail-File an.

Zum Beispiel:

StartArg: "PUBSCREEN=%ys FILE=%hm"

1.41 Wie benutze ich HTMLreadPrefs?

PublicScreen:

Bild: PublicScreen

Hier wird festgelegt welchen Screen %ys werden soll. %ys wird von einigen voreingestellten HTML-Anzeigern benutzt. Oder kann auch im Punkt StartArgs vom User selbst benutzt werden.

Man braucht dieses Feld nicht unbedingt anzugeben. Dann wird %ys automatisch »Workbench«. Wenn man »YAM« eingibt wird %ys der Screen auf dem Yam gerade geöffnet ist. Diese Einstellung wird empfohlen.

Ansonsten kann man jeden anderen PublicScreen eingeben.

Zum Beispiel:

PublicScreen: "YAM"

1.42 Wie benutze ich HTMLreadPrefs?

Path+Viewer:

Bild: Path+Viewer

Hier steht der Pfad und das Programm des gewählten HTML-Anzeiger. Dieser Punkt wird automatisch aufgerufen wenn man einen Anzeiger unter dem Punkt "Viewer" wählt. Dieser Punkt muß richtig gesetzt sein, da HTMLread dort ihren HTML-Anzeiger sucht und startet.

Wenn man Path+Viewer anwählt erscheint ein FileRequester.

Zum Beispiel:

Path+Viewer: "SYS:Utilities/MultiView"

1.43 Wie benutze ich HTMLreadPrefs?

ARexxPort:

Bild: ARexxPort

Soll ihr HTML-Anzeiger mit ARexx gesteuert werden? Wenn ja dann muß hier der ARexxPortName angegeben werden.

Wenn ein vorgefertigter HTML-Anzeiger aus der Liste unter Punkt Viewer ausgewählt wurde kann dieses Feld freigelassen werden, da HTMLread den richtigen ARexxport kennt.

Wenn der HTML-Anzeiger mehrere mögliche ARexxPorts kennt z.B AWeb (AWEB.1, AWEB.2, AWEB.3...) kann HTMLread den richtige selbständig herausfinden. Dazu muss man einen Punkt ».« ans Ende des Namens setzen. Also »AWEB.«

Zum Beispiel:

ARexxport: "AWEB." Für mehrer mögliche Ports

ARexxport: "VOYAGER" Für einen festen

1.44 Wie benutze ich HTMLreadPrefs?

REXX:OpenFile:

Bild: REXX:OpenFile

Hier stehen die ARexx-Befehle die an den HTML-Anzeiger gesendet werden wenn dieser von HTMLread gestartet werden mußte. Es können auch mehrere Befehle sein, sie müssen dann mit einem »;« voneinander getrennt sein. Außerdem kennt HTMLread noch dieselben Variablen wie unter Punkt StartArgs.

Wenn ein ein vorgefertigter HTML-Anzeiger aus der Liste unter Punkt Viewer ausgewählt wurde kann dieses Feld freigelassen werden, da HTMLread die richtigen ARexxBefehle kennt.

Zum Beispiel:

REXX:OpenFile: "OPENURL FILE://localhost/%hm;SCREENTOFONT"

1.45 Wie benutze ich HTMLreadPrefs?

REXX:NewWindow:

Bild: REXX:NewWindow

Hier stehen die ARexx-Befehle die an den HTML-Anzeiger gesendet werden wenn dieser schon läuft. Es können auch mehrere Befehle sein, sie müssen dann mit einem »;« voneinander getrennt sein. Außerdem kennt HTMLread noch dieselben Variablen wie unter Punkt StartArgs.

Wenn ein ein vorgefertigter HTML-Anzeiger aus der Liste unter Punkt Viewer ausgewählt wurde kann dieses Feld freigelassen werden, da HTMLread die richtigen ARexxBefehle kennt.

Zum Beispiel:

REXX:NewWindow: "OPENURL NEWWIN file://localhost/%hm;SHOW;SCREENTOFONT"

1.46 Wie benutze ich HTMLreadPrefs?

Show HTML-Mail:

Hier kann man einstellen wie sich HTMLread verhält wenn es eine HTMLmail erkannt hatt:

ALL	= Zeigt alle HTML-Dokumente an ohne zufragen, auch Attachments.
Ask befor	= Kann alle HTML-Dokumente anzeigen fragt aber vorher nach. Bild: Ask-Message
Only first mailpart	= Überprüft nur das allererste Dokument in einer EMail und startet gegebenenfalls den Anzeiger. Es werden so nur HTMLmails angezeigt die keinen ASC-Text zusätzlich zum HTMLcode haben.
Only 1st & 2nd mailpart	= Überprüft nur das erste und zweite Dokument in einer EMail und startet gegebenenfalls den

Anzeiger automatisch.
Wenn eine EMail aus einen ASC-Text und zusätzlich dazu einen HTMLcode hatt wird so der HTMLcode der EMail angezeigt.

1.47 Wie benutze ich HTMLreadPrefs?

Save:

Speichert die Einstellungen.

1.48 Wie benutze ich HTMLreadPrefs?

Use:

Nur nach ENV: speichern und nicht nach ENVARC:

1.49 Wie benutze ich HTMLreadPrefs?

Quit:

Verlässt HTMLreadPrefs!

1.50 Wie benutze ich HTMLreadPrefs?

Das Menü von HTMLreadPrefs:

```
+-----+-----+
| Project |
|-----+-----+
| Open   |
|-----+-----+
| Save As|
|-----+-----+
| About  |
|-----+-----+
| Quit   |
|-----+-----+
```

Project:

Open : Lädt die HTMLread-Konfigdatei mit einem andern Namen.
Save As : Speichert die HTMLread-Konfigdatei mit einem anderen Namen.
About : Gibt die Version des Programmes an.
Quit : Beendet HTMLreadPrefs.

1.51 Wie funktioniert HTMLread.rexx?

Wie funktioniert HTMLread.rexx?

Immer wenn eine Nachricht gelesen wird führt YAM nun HTMLread.rexx aus. HTMLread überprüft nun ob der MIME-Type TEXT/HTML vorhanden ist, da so ein HTML-Dokument gekennzeichnet ist. Wenn JA dann überprüft HTMLread ob Bilddateien mit in der Nachricht sind. Wenn JA werden diese gespeichert und der HTMLcode so umgeändert das die Anzeiger ihn problemlos mit den Bildern interpretieren können. Danach wird der HTML-Anzeiger gestartet. Es dürfen auch mehrere HTML-Dokumente in einer Mail sein.

1.52 Der Optimizer!

Der Optimizer

(wurde entfernt)

1.53 Welche Fehlermeldungen gibt es?

Fehlermeldungen

HTMLread scheint überhaupt nichts zumachen!

Siehe Hilfe HTMLread läuft nicht!
Siehe Wie installiere ich HTMLread? Punkt f).

No HTMLread-configfile!

Das ausgewählte KonfigFile wird nicht als HTMLreadKonfigfile erkannt.
Es ist sehr wichtig das die erste Zeile in diesem File so heißt:
HTMLread.config !!Do not edit the first line!!

Problem to open the configfile!

Das Konfigfile existiert einfach nicht. Es gibt zwei Möglichkeiten das zubeheben: a) Speichere mit HTMLreadPrefs ein KonfigFile mit diesem Namen und Pfad ab oder
b) sollte bei YAM/Konfig/ARexx/Beim Lesen einer Nachricht ein falsches Konfigfile angegeben sein, sollte man es berichtigen oder löschen. Wenn keins angegeben ist nimmt HTMLread automatisch ENV:HTMLread.config, dann gehe zu Punkt a)

Unknown keyword found in configfile!

Im KonfigFile wurde ein Schlüsselwort gefunden das nicht bekannt ist.
HTMLread akzeptiert nur folgende Schlüsselwörter:

viewer , path , startarg , pubscreen , arexxport ,
arexxopenfile , arexxnewwindow , showhtml , loadwebimage ,
downloadmanager, savewebimage , downloadlist, makeimagelist,
onoffline, webimagepath.

Die Argumente zu diesen Schlüsselwörtern müssen in "" stehen!.

Zur Fehlerbehebung: a) Lösche diese File und speichere mit HTMLreadPrefs

- ein neues!
- b) Lösche mit einem Editor diese Zeile aus dem File.

The »Viewer« or »path« is not correct!

Eines der beiden Schlüsselwörter ist im KonfigFile leer. Löschen sie das KonfigFile und speichern sie mit HTMLreadPrefs ein neues ab oder starten sie einen Editor und berichtigen Sie es!

I need this keyword unconditional!

Ein Schlüsselwort fehlt im Konfigfile das unbedingt gebraucht wird damit HTMLread funktionieren kann. HTMLread beendet sich danach.

Zur Fehlerbehebung: Lösche diese File und speichere mit HTMLreadPrefs ein neues oder starten sie einen Editor und berichtigen Sie es!

Warning, Keyword not found!

Ein Schlüsselwort fehlt im Konfigfile das NICHT unbedingt gebraucht wird. HTMLread ersetzt dieses automatisch mit einem voreingestellten und fährt dann fort.

Zur Fehlerbehebung: Lösche diese File und speichere mit HTMLreadPrefs ein neues oder starten sie einen Editor und berichtigen Sie es!

Program does not exists!

HTMLread wollte ihren HTML-Anzeiger starten. Doch das ging nicht da das Programm das sie festgelegt haben nicht zu finden ist. Starten sie HTMLreadPrefs und laden sie das Konfigfile, nun wählen sie unter dem Punkt Path+Viewer den RICHTIGEN Pfad und Anzeiger aus.

I can not find the ARexxPort!

Der Anzeiger ist gestartet worden, aber sein ARexxPort den sie unter Punkt ARexxPort angeben haben taucht nicht auf. Überprüfen Sie die Richtigkeit Ihrer Konfig! Ist der ARexxPort wirklich richtig? Ist es das richtige Programm das ich starte?

The ARexx-comand-returncode is not 0 !

Es wurde ein ARexxBefehl zum HTML-Anzeiger gesendet, allerdings ist der Return-Code nicht 0. Überprüfen Sie ob der ARexxBefehl richtig ist oder ob es sich beim HTML-Anzeiger um eine DEMO oder SE Version handelt, die nicht alle ARexxBefehle ausführt.

Siehe: arexxopenfile, arexxnewwindow

1.54 DISCLAIMER

DISCLAIMER

Haftungsausschluß

Durch die Verwendung dieses Programms akzeptiert der Benutzer die volle

Verantwortung für alle Schäden, die durch seine Benutzung oder das Unvermögen seiner Benutzung auftreten können. Der Entwickler dieser Software kann nicht verantwortlich gemacht werden.

Die Dokumentation und alle ARexxskripte sind PublicDomain.
Die Programme HTMLread und HTMLreadPrefs sind Freeware.

Achtung Teile des Archives und benutzter Programme sind ©,
bzw. Eingetragende Warenzeichen ihre Eigentümer!

YAM:

YAM ist © bei Marcel Beck

Bezugsquelle: - <http://www.yam.ch>
- AmiNet

RexxReqTools

RexxReqTools ist © bei Rafael D'Halleweyn

Bezugsquelle: - AmiNet

MicroRexx

MicroRexx ist © bei FR-SoftWorks

Bezugsquelle: - AmiNet

BetterString

BetterString is © byAllan Odgaard

- AmiNet
- <http://www.DIKU.dk/students/duff/>

1.55 Wer hatt es gemacht?

Wer hatts gemacht?

Meine Adresse: Gunter Hinrichsen
Koesterwaei 6
25920 Risum-Lindholm
Germany

EMAIL: frontier.user@gmx.net

Anregungen und Bugreports sind immer willkommen!

Was in ein BugReport drinnstehen sollte:

- HTMLread Version
- YAM Version
- Dein System (Amiga, Speicher, CPU, AmigaOS, Patches..)
- Die HTMLread-Konfig (viewer,....)
- und eine genaue beschreibung des Bugs!

1.56 Welche Zukunft erwartet HTMLread noch?

Zukunft

Was noch zu machen ist:

- mehr HTML-Anzeiger

- sonst noch Ideen?

1.57 Was ist passiert?

Geschichte...

V0.1 : erste Beta (nur für Betatester) (07-Jul-00)

V0.2beta : zweite Beta (nur für Betatester) (15-Jul-00)

- HTMLread V.2
 - HTMLread komplett überarbeitet
 - externes Konfigfile

- HTMLreadPrefs V.1 (neu)

- Optimizer V.1 (neu)

- Installer V.1 (neu)

- HTMLreadPrefs V.11 (17-Jul-00)

- Bugfix: Fonteinstellung geändert

- HTMLreadPrefs V.12 (18-Jul-00)

- Use hinzugefügt
 - Rechtschreibfehler behoben

- HTMLreadPrefs V.13 (20-Jul-00)

- Menü hinzugefügt:
 - About = Über HTMLreadPrefs
 - Load = Lädt die Konfig mit Filerequester
 - Save = Speichert die Konfig mit Filerequester
 - Quit = Beendet HTMLreadPrefs
 - Funktion der Knöpfe verändert:
 - Save = Speichert die Konfig ohne Filerequester
 - Load = Lädt die Konfig ohne Filerquester
 - Use = speichert die Konfig ohne sie nach ENVARC: zu kopieren und verlässt dann HTMLreadPrefs
 - Quit = Beendet HTMLreadPrefs ohne zu speichern
 - Windowstatus verändert, Window nichtmehr vergrößern/kleinerbar
 - Filerequestertitelzeile ausführlicher gemacht

- Optimizer V.12
 - Bug behoben = Konfigfilename wird nicht mehr vom Script fehlerhaft verändert
 - Einige Erklärungsfenster besser verständlich gemacht
 - Filerequestertitelzeile ausführlicher gemacht
 - Bei configfile: Option mit Standardkonfigfile und Cancel hinzugefügt

V0.3 : dritte Beta (nur für Betatester) (22-Jul-00)

- HTMLread V.3
 - HTMLread unterstützt nun auch HTML-Mails mit Bildern
 - einige Bugfix
 - HTMLread funktioniert nun auch bei mehreren YAM-Lesefenstern richtig
 - Probleme mit gelockten Files werden umgangen...
 - AWebBug gleicher Filenamen anderer Inhalt wird umgangen
 - AWeb OS 3.5: Zwischen mehreren HTML-Files in einer MAIL wird gewartet, bis bestätigt wurde und dann erst die nächste geladen!
- HTMLreadPrefs V.14
 - Bei Stringrequestern braucht man nicht mehr die "" anzugeben
- Optimizer V.13
 - Das neue Script hatt jetzt auch das Schutzbit: "S"
- Installer V.3
 - Wenn HTMLread schon vorhanden ist wird ein Backup gemacht
 - Man kann nun den Ort für HTMLread und HTMLread.rexx wählen.
 - Hilfe etwas überarbeitet
- UnInstaller V.1 (neu)

- HTMLreadPrefs V.15 (23-Jul-00)
 - interne einige Umstellungen gemacht
 - kein ständiges öffnen und schliessen des Fensters mehr
 - Einstellungen werden nun nichtmehr nach 15 Zeichen gekürzt
 - neues Menu zum einstellen von HTMLreadPrefs:
 - Font (stellt den verwendeten Font ein
 - Snapshot stellt die Koordinaten ein wo HTMLreadPrefs erscheint
 - MicroRexx stellt den Pfad wo MicroRexx gestartet wird ein
 - Load lädt die Einstellungen
 - Save speichert die Einstellungen

- HTMLreadPrefs V.16 (24-Jul-00)
 - Etwas bessere Fontunterstützung
 - Kann nun auch von Yam aus gestartet werden
 - MicroRexx-Pfad wird nicht mehr ignoriert
 - bessere Fehlerbehandlung beim Starten von MicroRexx

-
- HTMLreadPrefs V.17 (26-Jul-00)
 - Kann jetzt den Systemfont benutzen
-

- V0.3a : vierte Beta (nur für Betatester) (27-Jul-00)
- HTMLread V.3a
 - Altlasten aus V.2 rausgeschmissen
 - bessere Kompatibilität mit HTML-Mails mit Bildern
 - HTMLreadPrefs V.17a
 - SystemFonterkennung verbessert
 - funktioniert nun auch mit Kick2.0 Files
-

- V0.3b : fünfte Beta (nur für Betatester) (31-Jul-00)
- HTMLread V.3b
 - HTMLread V.3b [(un)optimiert]
 - kleine interne Änderung
 - Installer V.3
 - Texte überarbeitet
-

- HTMLreadPrefs V.18 (03-Aug-00)
 - Der Knopf "Load" ist jetzt ins Menu gewandert und heisst "Last Saved"
 - HTMLreadPrefs schlägt jetzt "richtige" Pfade vor:
Zum Beispiel ist MultiView wohl bei 99% aller User im Verzeichniss SYS:Utilities/MultiView zu finden. HTMLreadPrefs sieht nun nach ob MultiView in diesem Verzeichnis ist und wenn ja erscheint im File-Requester gleich die richtige Angabe. Es wird gesucht nach:
 - SimpleHTML => "C:SimpleHTML"
 - MultiView => "SYS:Utilities/MultiView"
 - OpenURL => "C:OpenURL"
 - HTMLviewDemo => "MUI:Demos/HTMLview-Demo"
 - ARexx-Command-Text ergänzt
 - kleinere interne Änderung
-

- HTMLreadPrefs V.18a (07-Aug-00)
 - Bug behoben: Abbruch bei Viewer "other" verändert nicht mehr die Werte
 - Bug behoben: Speichern der HTMLreadPrefs-Prefs funktioniert nun auch mit den Patch HappyEnv
 - Bug behoben: Die PrefsFile wird nun nichtmehr immer eine Zeile länger beim speichern.
 - Untere Knopfleiste ist nun zentriert
-

V0.3c : sechste Beta (nur für Betatester) (18-Aug-00)

- HTMLread V.3c
 - Altlasten aus V.2 entfernt
 - ShowHTML-Option mit eingebaut
 - HTMLreadPrefs V.19
 - Kann nun auch den Voyager Pfad herausfinden
 - Bug im Vorschlagssystem entfernt
 - ShowHTML-Option mit eingebaut
 - Optimizer V.14
 - ShowHTML-Option mit eingebaut
-

V0.3d : Public Release (02-Sep-00)

- HTMLread V.3d
 - unterstützt nun Option showhtml=ASK
 - HTMLreadPrefs V.20
 - unterstützt nun Option showhtml=ASK
 - Bug behoben, erzeugt nun nichtmehr die Pfade NOREQ und CLONE im aktuellen Verzeichniss.
 - Optimizer V.15
 - unterstützt nun Option showhtml=ASK
 - configfile/ENVARC:HTMLread.config in ENV(ARC):HTMLread.config umgeändert, es werden nun beide Files in ENV: und ENVARC: optimiert und nichtmehr nur in ENVARC:
-

V0.4beta : achte Beta (nur für Betatester) (30.Sep.00)

- HTMLread V.4
 - unterstützt nun Bilder die im Web abgelegt sind
- HTMLreadPrefs V.30
 - unterstützt nun Option LoadWebImage
 - unterstützt nun Option Downloadmanager
 - unterstützt nun Option SaveWebImage
 - unterstützt nun Option Downloadlist
 - unterstützt nun Option Make Imagelist
 - unterstützt nun Option OnOffline
 - unterstützt nun Option Cachepath
- clearCaches V.1 (neu)
 - Verwaltet den Bilder-Cache
- DownloadPrefs V.1 (neu)
 - verwaltet die HRDownload.liste
- optimizer entfernt
- checkHTML.rx V.1 (neu)
 - Schnelleres checken einer Mail auf HTMLcode.

- Miami OnOfflinePlugin V.1 (neu)
- HRdownload-Plugin V.1 (neu)

V0.4 : Public Release (04-Okt-00)

- HTMLreadPrefs V.30b
 - Die Gruppe wird nichtmehr überschrieben
 - Grafikausgabe optimiert
- DownloadPrefs V.1b
 - Wenn man mit HTMLreadPrefs nicht die HTMLreadPrefs gespeichert hatt, kann man nicht DownloadListPrefs starten. Das ist nun behoben.
 - Grafikausgabe optimiert
- checkHTML.rx V.1a
 - Seit V.4 heist HTMLread.rexx nun HTMLread.rx, aber checkHTML versucht noch HTMLread.rexx zustarten, das wurde nun richtigerweise auf HTMLread.rx gesetzt.

V0.45 : Public Release (25-Nov-00)

- HTMLread (bin) (neu)
 - Wesentlich verbesserter HTMLpharser
 - kann nun HTML-EMails bin <base>-Tag behandeln
 - background-Argument kann nun überall stehen
 - viel viel schneller
 - eine Bugs entfernt
- HTMLread (rexx) (V.45)
 - an HTMLread.bin angepasst

V0.46 : Public Release (01-Dec-00)

- HTMLread (bin) (V.46)
 - kein FileName-Bug behoben

V0.47 : Public Release (03-Dec-00)

- HTMLread (bin) (V.47)
 - BildNamenBug entfernt
- HTMLread (rexx) (V.47)
 - Fehlermeldungen werden jetzt richtig angezeigt
 - Verbesserter Schutz gegen falsches Einstellen
 - AutoCreate von Cachepath
- HTMLreadPrefs V.30c
 - Cachepath wurde zweimal angezeigt, behoben
 - Prüft auf gefährlichen Cachepath
- DownloadPrefs V.1b
 - kann nun leere Listen speichern
- clearCaches V.1a

- Prüft auf gefährlichen Cachepath

V0.48 : Public Release (17-Dec-00)

- HTMLread (bin) (V.48b)
 - Image in '' wurden ignorierte, behoben
 - schlechter HTMLcode konnte HTMLread zum Absturz bringen
 - eigener Code durch OS-Funktionen ersetzt
 - kopierfunktion hinzugefügt
 - StatusImages werden nun wieder unterstützt
(SimpleHTML, HTMLdatatype können nicht richt skalieren)
- HTMLread (rexx) (V.48a)
 - kennt nun die Option Quiet
 - habe kopierfkt entfernt
 - einige Sicherheitscheck verändert
- DownloadPrefs V.1b
 - kennt nun die Option Quiet
 - versteck HTTP-Knöpfe nicht mehr
 - Namen des CachePath verändert
- HRdownload-Plugin (V.4a)
 - öffnet nicht jedesmal ein Fenster
 - kennt nun die Option Quiet
- StatusBilder (V.2)
 - hübschere Statusbilder

V1.0 : Public Release (03-Feb-01) !! Stabile Version !!

- HTMLread (bin) (V1.00)
 - einige interne Veränderungen
 - bessere <base> Tag Unterstützung kann nun auch HTMLImage laden
 - HTMLread 1.0 (Rexx) (V1.00)
 - MultiARexx Port support (z.B AWEB.1 AWEB.2)
 - Es wird nun nur noch eine gelesene Nachricht in T: gehalten
 - MultiFenster Unterstützung etwas verbessert
 - Automatischer Abbruch
 - checkHTML.rx (V1.00)
 - Unterstützung der envVariable HTMLread für den HTMLreadPfad
 - Hrdownload.rx (V1.00)
 - Pfade angepasst
 - HTMLreadPrefs.rx (V1.00)
 - RexxPort erklärung ergänzt
 - clearCache.rx (V1.00)
 - nur Version angepasst
 - downloadlistPrefs.rx (V1.00)
 - nur Version angepasst
-

V1.1 : Public Release (??-Mar-01) !! Stabile Version !!

- HTMLread (bin) (V1.1)
 - HTMLparser komplett neu geschrieben!
 - einige interne Veränderungen
 - bessere <base> Tag Unterstützung auch bei NichtBildern
- HTMLread 1.1 (Rexx) (V1.1)
 - besserer Unterstützung für Space im Path & Filenamen
 - unnötigen Code entfernt
- HTMLreadPrefs V1.0
 - komplett neu in C geschrieben
 - nun mit MUI-Interface
- StatusBilder (V.3)
 - unbekannter CID hinzugefügt

1.58 Welche Problem gibt es mit HTMLread?

Bekannte Probleme

- Ich habe eine HTMLmail und MultiView oder HTMLview-Demo wird auch gestartet aber sie zeigen nichts an!

Grund:

Das ist kein Fehler von HTMLread, sondern das HTMLdatatype kann den HTMLcode nicht interpretieren und bricht ab, läßt aber das MultiView Fenster offen. HTMLview-Demo zeigt dann nur den Begrüssungstext an.

Abhilfe:

Benutze einen Browser, diese können am besten HTMLcode interpretieren.

1.59 Vielen Dank!

Dank und Grüße

Im folgenden möchte ich allen meinen Betatestern und allen Menschen die mich Unterstützt haben Danken...: (alphabetisch sortiert)

Christian Aichinger, Frank Bußmann, Gundula Hinrichsen, Jean Holzammer, Markus Lunk, Martina Jacobs, Michael Scholz, OPI, Richi, Roberto Lang, volki, Zdenko Klanfar...
und alle die ich vergessen habe...

1.60 Wie konfiguriere ich das GenesisPlugin?

Wie konfiguriere ich das GenesisPlugin

a) Starte einen editor und lade GenesiOnOff.rx

(oder benutze einen der Knöpfe:

- OS 2.0 - 3.1 C:ED YAM:Rexx/GENESISonoff.rx
- OS 3.5 - 3.9 SYS:Tools/EditPad YAM:Rexx/GENESISonoff.rx)

b) in Zeile 4 findest du folgenden Text:

```
interface = "Youre Interface"      /* Please enter between the " youre ↵  
Interface */
```

Hier musst du nun dein Interface eintragen das im GenesisFenster unter Interface angezeigt wird.

Picture: The Genesis Window (OS 3.9)

In meinem Fall heisst Zeile 4 nun:

```
interface = "Callino"      /* Please enter between the " youre Interface */
```

(Achtung ändere auf keine Fall die erste Zeile!!)

c) Speichere nun ab.

(Wenn du den ED (OS 2.0-3.1) benutzt drücke dazu ESC dann x und dann Return)

d) Starte nun HTMLreadPrefs

(Du findest HTMLreadPrefs in YAM:HTMLread/)

e) Wähle die Seite HTTP-Mails.

f) Wähle nun unter Punkt OnOffline GENESISonoff.rx .

Picture: The Genesis Window

g) Speichere nun.

Fertig.
