

MILL-beta

COLLABORATORS

	<i>TITLE :</i> MILL-beta		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 13, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	MILL-beta	1
1.1	WELCOME!!!	1
1.2	legal	2
1.3	install	2
1.4	mill	3
1.5	notifier	4
1.6	integrity	5
1.7	watchdog	6

Chapter 1

MILL-beta

1.1 WELCOME!!!

----- ←

MILL (01.04.03)

(C)2002-2003 Zbigniew Trzcionkowski <zeeball@interia.pl>

MILL ist ein neues modulares Antivirusprojekt, wobei jede Komponente unabhängig als eigenständiges Programm arbeitet. ←

Rechtliches
- es ist kostenlos

Installation
- ist sehr einfach

Das Projekt besteht derzeit aus:

MILL
- Dateiprüfung (XVS+XFD+XAD+intern)
und Ausführung der anderen Komponenten

Notifier
- Werkzeug zur Echtzeiterkennung von Änderungen in ausgewählten Dateien ←

Integrity
- Werkzeug zur periodischen Erkennung von Änderungen in ausgewählten Verzeichnissen/Partitionen. Man sieht, was auf einer Festplatte passiert ist. ←

WatchDog
- Killt Viren im Speicher (XVS+intern)

Für den Inhalt dieses Archivs wird keinerlei
Garantie übernommen!

Alle Eigenschaften des Pakets:

- fontsensitive Oberfläche
- Codeschutz im Speicher mittels MMU
(wenn verfügbar, benötigt die mmu.library von Thomas Richter.).
- vollständige Lokalisierung (benötigt locale.library)
- Commodities-Unterstützung (benötigt commodities.library)
- merkt sich die Einstellungen selbständig
- Ikonifizierung auf WB/Hilfsmittelenü

1.2 legal

Das MILL-Paket ist Freeware.

MILL benutzt folgende freie Bibliotheken:

xvs.library, (c) Georg Hoermann

xfdmaster.library, (c) Georg Hoermann und Dirk Stoecker

mmu.library, (c) Thomas Richter

und folgende SHAREWARE Bibliothek:

xadmaster.library, (c) Dirk Stoecker

XAD ist als Teil des OS3.9 oder als eigenständige SHAREWARE Bibliothek verfügbar.
Falls noch nicht erfolgt: bitte erachte, dein OS aufzurüsten oder diese Bibliothek ↔
zu
registrieren - es ist das einzige, was für MILL bezahlt werden müsste!

1.3 install

Kopiere einfach das ganze MILL-Verzeichnis irgendwohin. Außerdem kopiere die
notwendigen Dinge nach catalogs und media.

Falls einige Komponenten von MILL von Start an auf deinem Rechner laufen sollen,
so kopiere Projekt-Piktogramme nach WBStartup. Und setze das richtige
Standardprogramm mitsamt Pfad ein. Alle anderen Merkmale sollten im
Piktogramm des eigentlichen Programms gesetzt werden.

Natürlich können die MILL-Komponenten auch aus dem CLI/Shell oder AmigaDos

Skripts wie S:startup-sequence oder S:user-startup gestartet werden.

MILL und Integrity öffnen standardmäßig ihre Oberflächen, so muss CX_POPUP=NO als Argument gesetzt werden, wenn dies nicht geschehen soll.

Für das Erstellen einer Boot-Diskette mit MILL liegt ein Skript namens _makedisk bei. Das Skript kopiert alle notwendigen Dinge in ein Verzeichnis. Man muss diese Sachen jetzt auf eine formatierte bootfähige Diskette kopieren. Die amigaguide.library sollte von OS 1.3 oder 2.0 stammen und keine Datatypes verwenden. Diese Bibliothek wird von Integrity benötigt und zur Anzeige von Meldungen verwendet.

(*) diese Dinge sollte heutzutage eigentlich jeder verstehen...

Es gibt mehrere Arten von Piktogrammen, wie z.B. Projekt- oder Programmpiktogramme ↔

Das Standardprogramm eines Projektpiktogramms zeigt auf das auszuführende Programm ↔

Die Merkmale und Standardprogramm eines Piktogramms können durch eine Anwahl des Piktogramms und Rechte Amiga + 'i' geändert werden.

1.4 mill

MILL

(c) 2002-2003 von Zbigniew Trzcionkowski <zeeball@interia.pl>

FREEWARE

MILL ist das Basisprogramm des MILL Antivirusprojekts.

Die Hauptaufgabe besteht in der Dateiprüfung mittels XVS, XFD und XADMASTER sowie eingebauter Routinen. Die Oberfläche von MILL ermöglicht das Starten der anderen Komponenten des Pakets.

Weiterhin hat MILL einen Abschottungsmechanismus zur automatischen Virenprüfung. Nach erfolgter Einstellung werden Dateien vom Eingabeverzeichnis ins Ausgabeverzeichnis gespielt und infizierte Dateien zurückbehalten. Falls das Eingabeverzeichnis gleich dem Ausgabeverzeichnis ist bzw. das Ausgabeverzeichnis nicht verfügbar ist, werden die Dateien geprüft und die sauberen gelöscht!

MILL wurde in Assembler geschrieben, benutzt 30KB Speicher und den Speicher für jede Datei *2 oder mehr...
MILL benötigt OS37+.

MILL kann einen Warnton spielen, wenn ein Virus gefunden wurde (benötigt datatypes.library).

MILL kann Dateien innerhalb von Archiven wie LHA, LZX, ZOO, DMS usw. prüfen..

Dazu wird die `xadmaster.library` von Dirk Stoecker benötigt.

MILL kann aus der SHELL oder von Workbench aufgerufen werden und unterstützt folgende Merkmale:

CX_POPUP - YES/NO (standardmäßig öffnet MILL seine Oberfläche)

CX_PRIORITY - -128/+127 (Standard 0)

DONOTWAIT - Nur als Merkmal. MILL erzeugt keinen neuen Task, die Standardmethode, der Workbench mitzuteilen, dass MILL nichts zurückgibt. ←

1.5 notifier

----- ←

Notifier

(c) 2002-2003 von Zbigniew Trzcionkowski <zeeball@interia.pl>

FREEWARE

Notifier dient der ständigen Überwachung ausgewählter Dateien. Es nutzt die Überwachungsmechanismen des AmigaDos und daher ist die Aktivität sehr schnell und transparent. Nach dem Start installiert es sich im Speicher und überwacht die gewählten Dateien. Es meldet, welche Dateien verschwinden/geändert oder erzeugt werden. Man kann zwar auch Verzeichnisse auswählen, diese können aber nicht ordnungsgemäß überwacht werden... Man kann aber `.Int` Dateien für Verzeichnisse anlegen mit Integrity und diese mit Notifier überwachen!

Notifier wurde in Assembler geschrieben, nutzt 6KB Speicher plus benötigte Komponenten plus einen halben KB für jede überwachte Datei... Notifier benötigt OS37+.

Notifier kann aus der SHELL oder von Workbench aufgerufen werden und bietet ← folgende

SHELL Argumente/Piktogrammerkmale:

CX_POPUP - YES/NO (standardmäßig öffnet Notifier keine Oberfläche)

CX_PRIORITY - -128/+127 (Standard 0)

DONOTWAIT - Nur als Merkmal. MILL erzeugt keinen neuen Task,
die Standardmethode, der Workbench mitzuteilen, dass Notifier ←
nichts zurückgibt.

Schalter Schnellstartmodus. Einträge aus der gespeicherten Datei werden
standardmäßig auf Existenz geprüft (um Informationen über neu angelegte
Dateien zu bekommen). Die Geschwindigkeit auf langsamen Maschinen wird
entscheidend erhöht, nichtexistierende Dateien werden als verändert
gemeldet.

Der Schließknopf des Fensters verbirgt die Oberfläche. Beenden kann man Notifier
über den entsprechenden Menüpunkt oder mit Exchange.

Die Dateien s:startup-sequence, s:user-startup und libs:xvs.library
werden standardmäßig durch WatchDog überwacht.

1.6 integrity

Integrity2

(c) 2002-2003 von Zbigniew Trzcionkowski <zeeball@interia.pl>

FREEWARE

Integrity dient dem periodischen Überwachen von Änderungen an ausgewählten
Partitionen/Verzeichnissen. Zuerst muss man die Ausgabedatei und den zu
überwachenden Pfad angeben. Es werden Informationen über alle Dateien und
deren Prüfsummen gesammelt. Wenn eine neue Ausgabedatei begonnen wird (z.B.
eine Woche später) kann man beide Dateien vergleichen und neue, gelöschte und
geänderte Dateien feststellen. Ein zusätzliches Passwort wird empfohlen.

Integrity wurde in Assembler geschrieben, benutzt 18KB Speicher plus
alle benötigten Komponenten plus Speicher für zu überwachende Dateien usw...
Integrity benötigt OS37+

Integrity kann aus der SHELL oder von Workbench aufgerufen werden und
stellt folgende Merkmale bereit:

CX_POPUP - YES/NO (standardmäßig öffnet Integrity seine Oberfläche)

CX_PRIORITY - -128/+127 (Standard 0)

LOGSPATH - Standard ist PROGDIR:logs/. Hier kann man seinen
Pfad für .Int Dateien angeben. Man sollte dies einstellen, bevor ←
man
Integrity auf CD brennt :P

DONOTWAIT - Nur als Merkmal. MILL erzeugt keinen neuen Task,
die Standardmethode, der Workbench mitzuteilen, dass Integrity ↔
nichts zurückgibt.

1.7 watchdog

WatchDog2

(c) 2002-2003 von Zbigniew Trzcionkowski <zeeball@interia.pl>

FREEWARE

WatchDog ist ein Speicherüberwachungsprogramm, welches XVS und auch interne
Routinen verwendet.

WatchDog wurde in Assembler geschrieben, nutzt 10KB Speicher.
WatchDog benötigt OS37+

WD2 kann aus der SHELL oder von Workbench aufgerufen werden und unterstützt ↔
folgende

SHELL Argumente/Piktogrammmerkmale:

CX_POPUP - YES/NO (standardmäßig öffnet WatchDog keine Oberfläche)

CX_PRIORITY - -128/+127 (Standard 0)

DONOTWAIT - Nur als Merkmal. MILL erzeugt keinen neuen Task,
die Standardmethode, der Workbench mitzuteilen, dass WatchDog ↔
nichts zurückgibt.

Im Einstellungsmenü kann folgendes definiert werden:

Resetvektoren prüfen, xvs.library benutzen, den Anfang des Systemspeichers
prüfen, Laufwerksspeicher auf HitchHiker5 prüfen, alten BCPL-Kram prüfen
und Änderungen an s:startup-sequence, s:user-startup und libs:xvs.library
melden

Das Fensterschließsymbol verbirgt die Oberfläche.

Beenden kann man WatchDog über den entsprechenden Menüpunkt oder mit Exchange.
