

ProNET V3.4

Benutzerdokumentation
29. Mai 1999

Michael Krause

1 Einleitung

1.1 Was ist *ProNET*?

ProNET ist ein einfaches Netzwerksystem. Die bedeutendste Anwendung ist die gemeinsame Nutzung von Gerten wie z.B. Festplatten und CD-ROMs in allen vernetzten Rechnern.

ProNET ist ein "einfaches" Netzwerksystem, weil es keinen Standard wie z.B. TCP/IP darstellt und höchstwahrscheinlich auch nie einer werden wird – es soll auch gar keine Konkurrenz zu ihnen sein. Dieses Programmpaket ist rein hobbyartig entstanden und ist deswegen auch kein absolut professionelles Produkt.

Die Software basiert auf zweiseitigen Verbindungen, d.h. man kann immer nur zwei Computer auf einer Schnittstellenebene verbinden – jede Ebene hat genau zwei Enden. Dies entspricht dem sogenannten Client-Server-Modell:

```

/-----\           /-----\
|         | ---- ANFRAGE ---->> |         |
| Client  |                       | Server  |
|         | <<-- ANTWORT ----- |         |
\-----/           \-----/

```

Da *ProNET* jedoch ein modulares Treiberkonzept benutzt und keine festen Schnittstellen vorgibt, kann man ohne weiteres mehr als zwei Amigas vernetzen, indem man einfach eine zweite Verbindung *ffnet*.

Dieses Programm ist brigens nur auf Amiga-Computern lauffhig. Es wird auch nie Versionen fr andere Rechnerplattformen geben, weil das Dateisystem wesentlich darauf basiert, wie der Amiga generell mit Dateien umgeht.

Betrachtet man diese Tatsachen genau, ist *ProNET* genau das richtige fr Sie, wenn Sie

1. nicht viel Geld ausgeben wollen
2. eine perfekte Integration in Ihre Arbeitsumgebung wnschen
3. hauptschlich daran interessiert sind, zwei Amigas zu verbinden.

All dies mag Sie ein wenig an *ParNet* erinnern, und tatschlich ist *ProNET* als Nachfolger zu dieser alten und in seinen Fhigkeiten begrenzten Software zu sehen. Es folgen ein paar Beispiele dessen, was gegenüber Matt Dillons Software verbessert wurde:

- Modulares Treiberkonzept: parallele, serielle und MultiFaceCard3-Module sind mitgeliefert.
- Neues Netz-Gert fr jedes Zielgert (kein NET: mehr)
- Erkennt deswegen auch Diskwechsel
- Untersttzt die meisten der V40-DosPackets
- 'cd's in Netzwerkverzeichnis funktionieren!

Bitte beachten Sie noch, da es fr dieses Paket kein Installationsskript gibt – Wenn Sie es benutzen wollen, sollten Sie sich ein wenig mit der Shell und mit Texteditoren auskennen.

1.2 Systemanforderungen

ProNET benötigt grundstzlich Version 2 des AmigaOS und luft auf jedem beliebigen Prozessor. Einige wenige Ausnahmen gibt es jedoch: `pronet-server`, `pronet.device`, `pronet-talk` und smtliche Treibermodule laufen mit lteren Kickstart-Versionen. Sie wurden jedoch nur mit 1.3 getestet.

2 Installation

Bevor wir mit dem Kopieren beginnen, möchte ich Ihnen noch einmal erklären, was ‘Devices’ oder zu deutsch ‘Gerte’ sind. Wenn Sie finden, da Sie hierber genug wissen, können Sie diese Sektion gerne berspringen und gleich mit dem Installieren beginnen. Wenn Sie den Installationsvorgang nicht verstehen, kann Ihnen dies hier vielleicht weiterhelfen!

Weil AmigaDOS mehr als nur ein Speichermedium verwalten kann, bekommt jedes seinen eigenen Namen. Die prominentesten Beispiele wren da ‘DF0:’ fr das eingebaute Diskettenlaufwerk und ‘HD0:’ fr die erste Festplattenpartition. Immer, wenn Sie auf eine Datei auf so einem Medium zugreifen, geben Sie diesen Namen an, es sei denn natrlich, dieses Medium ist das Wurzelverzeichnis des aktuellen Verzeichnisses.

All diese Namen beschreiben einen Teil von AmigaDOS, den man *AmigaDOS-Device* oder *Gert* nennt. ‘CDO:’, ‘HD7:’, ‘PRT:’ und ‘RAM:’ beschreiben alle verschiedene AmigaDOS-Gerte. Wie Sie vielleicht am Beispiel ‘RAM:’ bemerkt haben, ist das AmigaDOS-Device nur ein *Symbol* fr ein Speichermedium, denn die Ram Disk ist schlielich kein reeller Teil der Hardware.

Dieses Device ist nicht mit den *Exec-Devices* zu verwechseln, obwohl man beide normalerweise nur *Device* nennt. Das erstere ist ein Symbol, das letztere ein Programm, das auf ‘.device’ endet und sich im ROM oder im ‘DEVS:’-Verzeichnis befindet. Exec-Devices sind Treiber fr Hardware wie z.B. Festplatten- oder Diskettenlaufwerke. In diesem Fall mssen Sie nicht unbedingt wissen, wozu sie gut sind oder wie sie funktionieren.

2.1 Installieren des ‘pronet.device’

Das `pronet.device` ist der Kern des gesamten *ProNET*-Pakets. Es enthlt alle Programmteile, die den Datentransfer zwischen mehreren Amigas verwalten. Es enthlt *keine* eigentlichen bertragungsmodule, wie es der Fall in Version 1 war, weil es sich als sehr schwierig herausstellte, neue bertragungsmodule fr andere Schnittstellen zu schreiben.

2.1.1 Kopieren der Dateien

Dieses Device mu ins ‘DEVS:’-Verzeichnis aller am Netz beteiligten Maschinen kopiert werden.

Die *bertragungsmodule* sind externe Programmmodule, die im ‘DEVS:ProNET’-Verzeichnis gelagert werden. Bitte kopieren Sie alle mitgelieferten Module dorthin, natrlich können Sie spter die nicht benutzen immer noch lschen. Wichtig: Nicht direkt nach ‘DEVS:’ kopieren, sondern in das eigene Verzeichnis!

Sobald Sie Ihr Keyfile empfangen, legen Sie es bitte als ‘.key’ in das gleiche Verzeichnis.

Bis jetzt gibt es die Module ‘`internal-parallel`’ fr den internen Parallelport (was brigens die einzige Mglichkeit in *ProNET* Version 1 war), ‘`mfc-parallel`’ fr den Parallelport einer mglicherweise vorhandenen MultiFaceCard3, und ‘`serial`’ fr serielle Schnittstellen, die ber `serial.device`-kompatible Treiber gesteuert werden. Das wren z.B. ‘`BaudBandit.device`’, ‘`duart.device`’ etc.

2.1.2 Installieren der Treiber

Bitte erstellen Sie nun in dem gleichen Verzeichnis die Datei 'DEVS:ProNET/.config'. Diese wird Konfigurationsdaten für alle Schnittstellen beinhalten, die Sie für das Netzwerk benötigen.

Diese Datei enthält beliebig viele Zeilen der Art:

```
[treibername] [treiberdaten]
```

Solch eine Zeile beschreibt eine sogenannte *Unit*, die einer durch einen *ProNET*-Treiber gesteuerten Schnittstelle entspricht. Je mehr Schnittstellen und Treiber Sie besitzen, desto mehr Verbindungen können Sie zu anderen Rechnern aufmachen. Sie werden die Möglichkeit haben, jede Unit zur Verbindung mit anderen Amigas zu benutzen. Wenn Sie z.B. eine Multiport-Karte mit 3 seriellen Ports haben, könnten Sie sich später mit 5 Amigas vernetzen, indem Sie die Multiport-Karte zusammen mit internem Parallel- und Seriellport benutzen.

Die Units sind von 0 bis n durchnummeriert. Unit 0 wird durch die allererste Zeile in der Datei beschrieben, Unit 1 durch die zweite usw. `treibername` ersetzen Sie bitte durch den Namen des externen Treibermoduls, `treiberdaten` sind Angaben, die der jeweilige Treiber zum korrekten Funktionieren benötigt. Diese können sich natürlich von Treiber zu Treiber unterscheiden.

2.1.2.1 'internal-parallel'

Als erstes werden wir den Treiber für den eingebauten Parallelport einrichten. Wir erstellen folgende Zeile:

```
internal-parallel 0 5
```

Die Zahl 5 wird vom Treiber benötigt und gibt die Priorität des Transferprozesses an. Wenn Sie nicht wissen, wovon ich rede, lassen Sie diese Zahl einfach stehen! Kleinere Zahlen, wie zum Beispiel 0, bewirken, da der Treiber nicht einfach die gesamte Rechenzeit für sich beansprucht, wenn Daten übertragen werden.

Die Zahl 0 ist die sogenannte *Rechnernummer* und ist von großer Bedeutung für das korrekte Funktionieren des Treibers. Zwei Rechner, die über den `internal-` oder `mfc-parallel-` Treiber vernetzt werden, müssen zwei verschiedene Nummern hier stehen haben: Eine Seite '0', die andere '1'. Wenn Sie beiden Seiten die gleiche Nummer geben, wird *ProNET* nicht funktionieren.

Dieser Treiber benötigt ein ParNet-kompatibles Kabel am eingebauten Parallelport.

2.1.2.2 'mfc-parallel'

Dieser Treiber wird für Sie nützlich sein, wenn Sie die MultiFaceCard3 von BSC besitzen. Hiermit können Sie den entsprechenden Parallelport benutzen, um Verbindungen zu anderen Rechnern aufzubauen. Dies funktioniert sowohl zu einem normalen internen Parallelport als auch zu einer zweiten MultiFaceCard3.

Bitte beachten Sie, da Sie nur MultiFaceCards benutzen können, die mit einer 6821 PIA arbeiten – der Treiber wurde nur mit der MultiFaceCard3 getestet.

Erstellen Sie jetzt folgende Zeile::

```
mfc-parallel 0 1 5
```

0 ist die Nummer des MFC-Parallelports; falls Sie mehrere Karten dieses Typs besitzen, können Sie auch andere Nummern einsetzen.

Die Zahl 5 wird vom Treiber benötigt und gibt die Priorität des Transferprozesses an. Wenn Sie nicht wissen, wovon ich rede, lassen Sie diese Zahl einfach stehen! Kleinere Zahlen, wie zum Beispiel 0, bewirken, daß der Treiber nicht einfach die gesamte Rechenzeit für sich beansprucht, wenn Daten übertragen werden.

Die Zahl 0 ist die sogenannte *Rechnernummer* und ist von großer Bedeutung für das korrekte Funktionieren des Treibers. Zwei Rechner, die über den internal- oder mfc-parallel-Treiber vernetzt werden, müssen zwei verschiedene Nummern hier stehen haben: Eine Seite '0', die andere '1'. Wenn Sie beiden Seiten die gleiche Nummer geben, wird *ProNET* nicht funktionieren.

Dieser Treiber benötigt ein ParNet-kompatibles Kabel am entsprechenden Parallelport.

2.1.2.3 'serial'

Wenn Sie Daten über serielle Kabel übertragen möchten, erstellen Sie jetzt folgende Zeile:

```
serial BaudBandit.device 0 19200
```

`BaudBandit.device` ist der Name des `serial.device`, das Sie benutzen wollen, 0 ist die entsprechende Unit, die nur bei Multiport-Karten von Bedeutung ist. Bitte beachten Sie, daß das `sn1.device` auf einem A500 nicht mit diesem Treiber zusammenarbeitet. Wenn Sie ganz sichergehen wollen, benutzen Sie am besten das original `serial.device`

19200 ist die Baudrate, die auf beiden Seiten bereinstimmen muß. Probieren Sie, wie hoch Sie gehen können. 19200 ist eigentlich ein sehr sicherer Wert, der auch auf unbeschleunigten 500ern keine Fehler liefert. Typische höhere Werte wären z.B. 38400, 57600, 64000, 76800 oder 115200. Wenn Sie doch einen zu hohen Wert benutzen, kann es sein, daß das Netzwerk einfach aufhört zu arbeiten, besonders wenn der Prozessor sehr beansprucht wird. Es kann aber auch genauso gut sein, daß alles eine halbe Stunde in Ordnung ist und der Treiber dann "zusammenbricht".

Ohne weitere Argumente benötigt dieser Treiber ein Null-Modem-Kabel mit 7 Drähten, um korrekt zu arbeiten. Falls Sie in die Verlegenheit kommen sollten, nur ein 3-adriges zu haben, z.B. wenn Sie sich mit einem CD32 vernetzen, sollten Sie das Argument `3WIRE` ans Ende der Zeile setzen, natürlich auf beiden Seiten der Verbindung. 3-Wire-Verbindungen sind allerdings generell instabiler als 7-adrige, besonders wenn Ihre CPU stark ausgelastet ist. Hier hilft wieder nur ein Herumspielen mit der Baudrate.

Weiterhin müssen Sie im folgenden darauf achten, daß beim Start des Netzwerks immer zuerst der Server gestartet wird, danach können Sie auf dem anderen Rechner '`pronet-start`' anwenden. Die anderen Treiber warten solange, bis der Server gestartet ist, beim serial-Treiber ist das nicht möglich!

2.2 Herstellen der Kabel

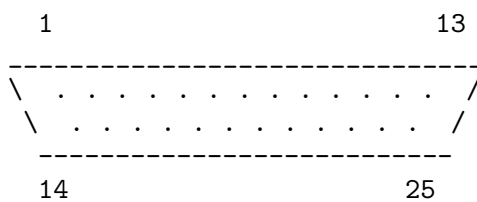
Nur damit es klar ist: Was auch immer Sie sich hier zusammenleihen, ich übernehme keine Verantwortung für irgendwelche Schäden.

2.2.1 Das *ParNet*-Kabel

Sie lten jetzt ein Kabel, das die beiden Parallelports zweier Amigas verbindet. Verbinden Sie jeweils D7-D0, SEL, POUT und BUSY und schlieen Sie ACK und SEL auf beiden Seiten kurz:

(2-9)	D7-D0	-----	D7-D0	
(12)	POUT	-----	POUT	
(11)	BUSY	-----	BUSY	PARALLEL PORT
(13)	SEL	---+-----+--	SEL	
(10)	ACK	-/ \-	ACK	
(18-22)	GND	-----	GND	(18-22)

Dies ist ein DB-25-Steckverbinder, wie man von auen draufschaugt:



Achtung A1000-Benutzer: Der Parallelport ist ein Stecker, also brauchen Sie eine Buchse als Verbinder. Alle anderen benutzen einen Stecker. Schauen Sie sich den Port einfach mal an.

Das einfachste wre es, ein fertiges 25er-Kabel zu kaufen, die ungtigen Verbindungen zu kappen und SEL und ACK auf beiden Seiten zu verbinden. **Verbinden Sie die Rechner niemals mit einem unmodifizierten Kabel! Prfen Sie das Kabel lieber dreimal vorher, als sich hinterher ber eine kaputte Kiste zu rgern!**

Je lnger Sie das Kabel machen, desto weniger stabil luft das Netzwerk. Und nicht nur das, Sie knnen sich leicht die CIA-Chips zerhauen. 5 Meter sind absolut sicher, ich kenne auch Leute, die 10 Meter ohne Probleme benutzen, aber ich persnlich wrde nicht hher als fnf Meter gehen.

2.2.1.1 Wechselwirkungen mit dem seriellen Port

Achtung: Die RI-(Ring Indicator-)Leitung am eingebauten *seriellen* Anschlu Ihres Amigas benutzt die SEL-Leitung, um einen Transistor zu speisen. Daraus ergeben sich aber leider einige Probleme, da diese Leitung am Netzwerk beteiligt ist.

Stellen Sie also sicher, da entweder:

- kein serielles Kabel angeschlossen ist, oder
- da es die RI-Leitung nicht benutzt, oder
- Ihr Modem die RI-Leitung nicht benutzt, oder
- da Sie nicht angerufen werden ;-)

Befolgen Sie diesen Ratschlag nicht, strzt der 'internal-parallel'-Treiber einfach ab, sobald ein Anruf reinkommt.

2.2.2 Das Null-Modem-Kabel

Sie sollten versuchen, das Kabel möglichst kurz zu halten. 5 Meter sind noch in Ordnung, aber das Netzwerk wird bei noch größeren Längen leicht instabil.

2.2.2.1 7-Draht-Schema

Zwei Schnittstellen müssen auf folgende Weise verbunden werden, damit Sie ein 7-adriges Null-Modem-Kabel erhalten:

```
Shield Earth -> Shield Earth
System Earth -> System Earth
TXD          ->          RXD
RXD          ->          TXD
RTS          ->          CTS
CTS          ->          RTS
DSR          ->          DTR
DTR          ->          DSR
```

Niemals Shield earth und System earth verbinden!

2.2.2.2 3-Draht-Schema

Um ein 3-Wire-Kabel zu erhalten, lassen Sie die DSR/DTR- und RTS/CTS-Verbindungen einfach weg, so dass sich folgender Aufbau ergibt:

```
Shield Earth -> Shield Earth
System Earth -> System Earth
TXD          ->          RXD
RXD          ->          TXD
```

Hier ebenfalls **niemals** Shield earth und System earth verbinden!

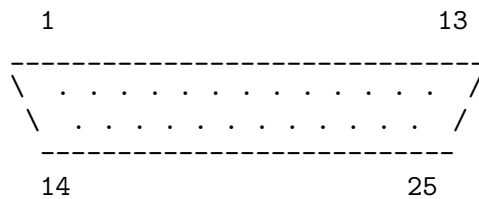
Wenn Sie Standard-Ports besitzen, sollten Sie lieber 7-Draht-Kabel verwenden, da sie einfach sicherer sind.

2.2.2.3 25-polige Steckverbinder

...sind die normalen seriellen Amiga-Ports.

```
Shield Earth (GND) - Pin 1
System Earth (GND) - Pin 7
TXD              - Pin 2
RXD              - Pin 3
RTS              - Pin 4
CTS              - Pin 5
DSR              - Pin 6
DTR              - Pin 20
```

Dies ist ein DB-25-Steckverbinder, wie man ihn von außen sieht:

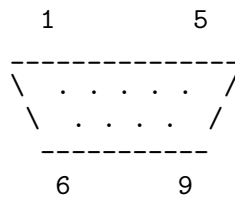


2.2.2.4 9-polige Steckverbinder

...werden häufig bei den restlichen Computersystemen benutzt und finden sich auch manchmal auf I/O-Extendern wie der *MultiFaceCard3* wieder.

Shield Earth (GND)	- (mit dem Rahmen verbinden)
System Earth (GND)	- Pin 5
TXD	- Pin 3
RXD	- Pin 2
RTS	- Pin 7
CTS	- Pin 8
DSR	- Pin 6
DTR	- Pin 4

Dies ist ein DB-9-Steckverbinder, wie man ihn von außen sieht:



2.3 Installieren des Dateisystems

Ganz einfach: Nehmen wir mal an, Ihr Arbeitsplatz ist ein 4000er. Sie wollen das CD-ROM-Laufwerk eines CDTV importieren, das unter dem Tisch steht, weil Sie keine Lust haben, Geld für ein extra SCSI-CD-ROM-Laufwerk auszugeben. Bei dieser Konfiguration ist es offensichtlich, da das CDTV der *Server* sein wird, während der 4000er als *Client* fungiert.

Kopieren Sie einfach die Datei 'pronet-server' in das 'C:'-Verzeichnis oder an eine andere Stelle des Suchpfades auf dem Server, das ist der Rechner, von dem Sie Geräte importieren wollen. Danach kopieren Sie den 'pronet-handler' in das 'L:'-Verzeichnis des Hauptrechners, dem Client.

Das war's!

2.4 Installieren der Hilfsprogramme

Kopieren Sie 'pronet-run', 'pronet-page', 'pronet-start', 'pronet-stop' und 'pronet-talk' in die 'C:'-Verzeichnisse Ihrer Amigas.

2.5 Updaten von V3

Falls Sie bereits *ProNET* V3 benutzen, brauchen Sie nicht komplett von vorne mit der Installation beginnen, denn ich weise Sie hier auf die wichtigsten Unterschiede hin:

- Die Dateien `pronet.key` und `pronet.config` liegen jetzt in dem `ProNET`-Directory unter den Namen `.key` bzw. `.config`.
- In der Datei `DEVS:ProNET/.config` müssen Sie die Nummern und die Doppelpunkte am Anfang der Zeilen entfernen. Jede Zeile beginnt also sofort mit dem Namen des Treibers. Kommentare sind nicht mehr erlaubt, denn die Units entsprechen jetzt der Zeilennummer: Die erste Zeile definiert Unit 0, die zweite Zeile Unit 1 usw.
- Server-Reboot ist jetzt teilweise implementiert, d.h. Sie können den Server ohne den Client neu starten. Das funktioniert allerdings nicht immer und nicht sehr gut. Sie müssen sehen, wie weit Sie damit zurechtkommen.
- Es ist nicht möglich, eine V3.1-Installation und eine V3-Installation zu vernetzen, da das Übertragungsprotokoll geändert wurde.

Genaue Beschreibungen der Änderungen finden sich in der beiliegenden Datei `ChangeLog`.

3 Anwendung

3.1 Starten des Servers ('pronet-server')

Auf jedem Amiga, den Sie als Server benutzen wollen, müssen Sie zuerst 'pronet-server' starten. Als Vorgabe wartet es auf Client-Anfragen an der *ProNET*-Unit 0, aber Sie können die Unit ändern, indem Sie sie beim Start als Argument übergeben. Daraus resultiert, dass Sie natürlich mehrere Verbindungen von außen zu einem zentralen Server-Rechner herstellen können – dann müssen Sie lediglich den 'pronet-server' für jede Unit einzeln starten.

Wenn Sie vergessen, den Server zu starten, werden die Client-Programme auf den anderen Rechnern einfach in eine Endlosschleife gehen – wenn also später mal nichts funktioniert, prüfen Sie als erstes, ob der Server überhaupt gestartet ist.

Sie können das Programm jederzeit durch ein 'CTRL-C'-Signal unterbrechen, das empfiehlt sich besonders, bevor Sie den Serverrechner neustarten, weil er dann den angeschlossenen Clients Bescheid sagt. Ich empfehle Ihnen, den Server mittels `run <>nil:` zu starten, so dass er Ihre Shell nicht blockiert. Wenn Sie ihn abbrechen wollen, müssen Sie zu den Shell-Kommandos 'status' und 'break' greifen.

3.2 Importieren von Geräten ('pronet-start')

Das Netzwerk zu starten ist keine große Angelegenheit. Dazu dient der 'pronet-start'-Befehl, den man folgendermaßen anwendet:

```
pronet-start LOCALNAME/A,REMOTENAME/A,UNIT/N,FLAGS/N,UNIQUE/S
```

LOCALNAME ist der Name des Gerätes, das Sie erzeugen wollen, während REMOTENAME den Namen des entsprechenden Gerätes auf dem Server darstellt. Es ist wichtig, dass Sie hier nicht den Namen des Datenträgers (wie z.B. 'Work') angeben, sondern den Gerätenamen (z.B. 'hd0'). Bitte beachten Sie auch, dass Sie den Doppelpunkt weglassen müssen!

Für das genannte Beispiel könnten Sie also folgendes eingeben:

```
pronet-start cd1 cd0
    ^^^
```

So heißt das Gerät auf dem A4000.

```
    ^^^
```

So heißt es auf dem CDTV.

Die restlichen Argumente sind nicht zwingend erforderlich:

UNIT ist die *ProNET*-Unit, die Sie benutzen wollen (Vorgabe ist 0). FLAGS hat gegenwärtig keine Auswirkungen (und ist ebenfalls mit 0 vorgegeben). Der UNIQUE-Schalter hilft Ihnen bei folgender Situation:

Wenn Sie zwei Amigas vernetzen, die beide eine Festplattenpartition besitzen, die 'Workbench' heißt, werden Sie nach dem Netzwerkstart zwei Icons mit dem gleichen Namen 'Workbench' entdecken. Das ist belästigend und sollte vermieden werden, um einem seltsamen Verhalten von AmigaDOS vorzubeugen. Die Lösung ist eben dieser UNIQUE-Schalter. Er fügt an jeden importierten Datenträgernamen eine Zahl an, die der jeweiligen *ProNET*-Unit entspricht.

'pronet-start' zeigt eine Fehlermeldung an, falls etwas schiefgeht, z.B. wenn es das angegebene Gerät auf dem Server nicht gibt oder wenn die angegebene Unit nicht definiert

ist. Treten keine Fehler auf, ttigt das Programm keine Ausgaben. Starten Sie es nie mit ‘run’, das Programm kehrt sofort zur Shell zurck.

So, und jetzt wiederholen Sie diesen Schritt fr jedes andere Gert, das Sie noch importieren wollen. Viel Spa!

3.3 ‘pronet-stop’

Wenn Sie meinen, da Sie ein bestimmtes Netzwerkert nicht mehr bentigen, knnen Sie es entfernen. Rufen Sie einfach ‘pronet-stop’ mit dem entsprechenden Gertenamen auf, und schon wird die entsprechende Verbindung geschlossen. Sie knnen auch das UNIT-Keyword benutzen, dann werden alle Gerte auf der angegebenen Unit heruntergefahren.

Warum sollten Sie das tun? Nun, falls Sie den Server ausschalten oder ihn zurcksetzen, ohne den Client zu stoppen, knnen ganz seltsame Dinge auf dem Client passieren: Es knnte sein, da er komplett hngt, oder da Sie auf keine Gerte mehr zugreifen knnen – das hngt damit zusammen, da der Client denkt, da der Server immer noch luft und deswegen fleiig Anfragen an ihn schickt.

Klar: Wenn Sie sowieso beide Rechner abschalten wollen, knnen Sie sich diese Prozedur sparen.

3.4 ‘pronet-flush’

Sie knnen dieses Programm (ohne weitere Argumente) benutzen, um das ‘pronet.device’ aus dem Speicher zu entfernen. Dadurch werden alle Ressourcen freigegeben, die von den *ProNET*-Treibern belegt wurden. ‘pronet-server’ und ‘pronet-stop’ haben diese Funktion bereits eingebaut. Somit werden Sie kaum eine Gelegenheit finden, bei der Sie dieses Programm nutzbringend einsetzen knnen, aber ich habe es der Vollstndigkeit halber trotzdem geschrieben.

3.5 ‘pronet-page’

Mit Hilfe dieses Programms knnen Sie Nachrichten an die Benutzer anderer Netzwerkrechner schicken. Rufen Sie es mit der *ProNET*-Unit und der entsprechenden Nachricht auf, z.B.:

```
pronet-page 0 Guten Morgen!
```

Die Nachricht wird in einer Warnmeldung am oberen Bildschirmrand angezeigt.

3.6 ‘pronet-run’

Mit diesem Programm knnen Sie Programme auf anderen Netzwerkrechnern starten, ohne deren Maus oder Tastatur zu benutzen. Rufen Sie es mit der *ProNET*-Unit und dem auszufhrenden Kommando auf. Der Befehl *mu* dabei in Anführungszeichen stehen, z.B.:

```
pronet-run 0 "dir hd0: all"
pronet-run 13 "list >con:0/0/640/200/test"
```

Dabei ist zu beachten, da die Ausgabe in das Shell-Fenster erfolgt, von dem der jeweilige *pronet-server* gestartet wurde. Es ist also zu empfehlen, die Ausgabe wie im zweiten Beispiel explizit umzuleiten, falls jenes Fenster nicht mehr existiert.

3.7 'pronet-talk'

Mit diesem Programm können sich die Benutzer zweier Netzwerkrechner miteinander unterhalten. Rufen Sie das Programm auf *beiden* Rechnern mit der richtigen *ProNET*-Unit auf. Als Vorgabe benutzt es die Unit 0.

4 Weitere Anwendungen

In diesem Kapitel möchte ich Ihnen ein paar Tips geben, wie Sie das letzte aus ProNET herausholen. Es gibt da ein paar Dinge, die nicht ganz offensichtlich sind, aber trotzdem nützlich sein könnten.

4.1 Wiederverbindung

Schauen wir uns noch einmal das Beispiel von vorhin an. Manchmal werden Sie in die Verlegenheit kommen, den A4000 neuzustarten, weil irgendein Programm mal wieder abgestürzt ist (...nein, nicht wirklich, weil.. wir benutzen ja einen Amiga ;-)). Wenn Sie das Netzwerk nun wieder hochfahren wollen, müssen Sie das CDTV nicht ebenfalls neu starten – Es reicht, wenn Sie die Gerte auf dem A4000 einfach wieder neu starten, mit Hilfe von `'pronet-start'`, das ist alles.

4.2 Verbindungen ueber Kreuz

Die meisten Konfigurationen werden so ähnlich wie meine Beispielkonfiguration aussehen, d.h. ein Hauptrechner (der Arbeitsplatz) und ein Server, an dem nicht gearbeitet wird, der aber seine Gerte zur Verfügung stellt. Wie auch immer, wenn Sie sich mit einem Freund vernetzen, und beide arbeiten an den Rechnern, können Sie das Netzwerk auch über Kreuz starten, d.h. beide Rechner sind Client und Server zugleich. Sie können beide die Gerte des jeweils anderen Systems ansprechen.

4.3 Netzwerk ueber die Telefonleitung

Stellen Sie sich vor, Sie wollen einem Freund, der weit weg wohnt, die Installation einer bestimmten Software erklären. Sie würden ihn anrufen und ihm genau sagen, was er zu tun hat – aber das wird schiefgehen, denn er wird Sie nicht verstehen (Diese Annahme ist von Murphys Gesetz abgeleitet und hat sich in vielen Situationen als wahr erwiesen ;-)).

Sie könnten auch eine Modemverbindung herstellen, ein Netzwerk zwischen Ihren Amigas einrichten, für ihn ein paar Shellscripts einrichten etc. Klingt das interessant? Dann lesen Sie weiter!

4.3.1 Vorbereitung

Beide Seiten müssen eine *ProNET*-Unit definieren, die den `'serial'`-Treiber verwendet. Die entsprechende Schnittstelle wird jedoch kein Nullmodem-Kabel verwenden, sondern ein richtiges Modem! Wie gewohnt, setzen Sie die Geschwindigkeit des seriellen Ports viermal so hoch wie die Leitungsgeschwindigkeit des Modems. Diese Geschwindigkeit muss nicht unbedingt auf beiden Seiten bereinstimmen.

Beide Modems müssen so konfiguriert sein, da sie das DTR-Signal ignorieren, was meistens durch den Befehl `AT&D0` geschieht.

4.3.2 Verbindung

Laden Sie jetzt Ihr Terminalprogramm. Entscheiden Sie sich, welcher von Ihnen der Anrufer ist, dieser ruft jetzt den anderen mittels ATD an. Sobald der Angerufene mit ATA geantwortet hat und Sie beide die CONNECT-Nachricht erhalten haben, verlassen Sie das Terminalprogramm.

Jetzt fahren Sie einfach so fort, als wren Sie durch ein Nullmodem-Kabel verbunden: Starten Sie den Server und richten Sie die Gerte ein, natrlich alles mit der vorhin definierten Unit, die das Modem anspricht. Das ist alles – nicht gerade sehr schnell, aber es funktioniert.

5 Interna

5.1 Erstellen eigener *ProNET*-Treiber

Lesen Sie die beiden autodoc-Dateien, die diesem Paket beiliegen. Eins ist für die generelle Benutzung des 'pronet.device', das andere beschreibt, wie man neue Treiber schreibt. Alles leider nur auf Englisch ;)

5.2 Unterstützte DosPacket-Typen

ProNET unterstützt die meisten neuen DosPackets, die in V36 und V39 eingeführt wurden: Vielleicht wurden Sie schon von *ParNet* durch so einen '2.0 Pkt ACT_PARENT_FH'-Requester genervt – Sie werden ihn nie wiedersehen! Hier ist die Liste der von *ProNET* unterstützten DosPackets:

```

ACTION_FINDINPUT
ACTION_FINDUPDATE      V33
ACTION_FINDOUTPUT
ACTION_END
ACTION_READ
ACTION_WRITE
ACTION_SEEK
ACTION_CURRENT_VOLUME
ACTION_SET_FILE_SIZE   V36
ACTION_LOCK_RECORD     V36
ACTION_FREE_RECORD     V36

ACTION_LOCATE_OBJECT
ACTION_FREE_LOCK
ACTION_COPY_DIR
ACTION_PARENT
ACTION_SAME_LOCK       V36
ACTION_CREATE_DIR
ACTION_CHANGE_MODE     V36
ACTION_FH_FROM_LOCK    V36
ACTION_COPY_DIR_FH     V36
ACTION_PARENT_FH       V36
ACTION_EXAMINE_OBJECT
ACTION_EXAMINE_NEXT
ACTION_EXAMINE_FH     V36

ACTION_DELETE_OBJECT
ACTION_RENAME_OBJECT
ACTION_MAKE_LINK       V36
ACTION_READ_LINK       V36
ACTION_SET_COMMENT
ACTION_SET_DATE
ACTION_SET_PROTECT

```

ACTION_INFO
ACTION_RENAME_DISK
ACTION_INHIBIT
ACTION_FORMAT V36
ACTION_SERIALIZE_DISK V39
ACTION_MORE_CACHE
ACTION_WRITE_PROTECT
ACTION_IS_FILESYSTEM V36

ACTION_NIL
ACTION_FLUSH
ACTION_DISK_INFO

Diese Typen werden noch nicht untersttzt und werden vielleicht spter implementiert:

ACTION_EXAMINE_ALL V36
ACTION_EXAMINE_ALL_END V39
ACTION_SET_OWNER V39
ACTION_ADD_NOTIFY V36
ACTION_REMOVE_NOTIFY V36

6 Verschiedenes

6.1 Support

Wegen der Kosten herkömmlicher Post, ist es mir nicht möglich, jede Frage auf diesem Wege zu beantworten. Besitzen Sie eine e-mail-Adresse, steigen die Chancen auf eine Antwort ins Unermessliche ;-) Bitte versuchen Sie auch zunächst, eine Antwort in dieser Anleitung zu finden, besonders im FAQ-Kapitel.

ProNET wird seit V3.2 nicht mehr weiterentwickelt. Die einzige Möglichkeit, die ich für eine V3.5 sehe, wäre wenn mir jemand über einen richtig bösen Bug im Programm berichtet. Da das aber schon seit über einem Jahr nicht mehr geschehen ist, nehme ich an, da ProNET mittlerweile recht stabil läuft.

Und so können Sie mich erreichen:

Michael Krause
 Mannesallee 24
 D-21107 Hamburg

m.krause@tu-harburg.de
 rawstyle@ms.demo.org
 michael@tnx.de

<http://www.tu-harburg.de/~semk2104/>

Falls all diese Adressen irgendwann nicht mehr funktionieren sollten, probieren Sie in einer beliebigen Suchmaschine für's WWW (oder welches Medium dann auch immer aktuell sein sollte) doch mal aus, nach meinem Namen und/oder "ProNET" zu suchen. So sollte man mich eigentlich immer finden.

6.2 Credits & Danksagungen

Gesamte *ProNET*-Software von

Michael Krause

ParNet (Inspiration) und Entwurf des Parallelkabels von

Matt Dillon

Leute, die mir bei der Software geholfen haben, indem Sie mich auf Bugs hingewiesen haben, die ich alleine nie gefunden hätte (DANKE!):

Thomas Schwarz, Wolfgang Gutberlet, Mat Bettinson, Robert W. Adams, Nik Soggia, Anti-"Es-funktioniert-nicht"-brain

Schließlich möchte ich noch meinen Dank allen denen aussprechen, die mir Mails geschrieben haben, die mehr als nur 'ProNET funktioniert nicht' enthielten und mich so ermutigten, an diesem Paket weiterzuarbeiten. Diese Version wäre ohne Eure Hilfe niemals entstanden!

6.3 Geschichte

Dieser Abschnitt behandelt nur die wichtigsten Änderungen. Ein ausführlicherer Änderungsbericht ist die 'ChangeLog'-Datei.

6.3.1 Version 3.4

Freigegeben am 29-Mai-99. Keinerlei Änderungen an den Programmen selbst, es wurden lediglich die Shareware-Requester entfernt sowie das Programm unter eine neue Lizenz gestellt – der komplette Sourcecode liegt jetzt bei.

6.3.2 Version 3.3

Freigegeben am 17-Mai-98. Keinerlei Änderungen an den Programmen selbst, es wurde lediglich die Anleitung auf den neuesten Stand gebracht (Adressen).

6.3.3 Version 3.2

Freigegeben am 27-Jan-97, mit `pronet.device` **38.0**, `pronet-handler` **38.0**, `pronet-server` **38.0**, `pronet-start` **37.0**, `pronet-stop` **38.0**, `pronet-flush` **36.0**, `pronet-talk` **37.1**, `pronet-run` **37.0**, `pronet-page` **37.0**, `internal-parallel` **37.0**, `mfc-parallel` **37.0**, `serial` **37.0**.

- `pronet-stop` entfernt die DeviceNode komplett; ein Gert kann also unter dem gleichen Namen nochmals gestartet werden.
- Piktogramme erscheinen schneller auf der Workbench.
- `pronet-stop` UNIT-Keyword zum gemeinsamen Abmelden aller Gerte auf einer Unit.
- Kleinere Fehler beseitigt.

6.3.4 Version 3.1

Freigegeben am 30-Nov-96, mit `pronet.device` **37.2**, `pronet-handler` **37.1**, `pronet-server` **37.0**, `pronet-start` **37.0**, `pronet-stop` **36.1**, `pronet-flush` **36.0**, `pronet-talk` **37.1**, `pronet-run` **37.0**, `pronet-page` **37.0**, `internal-parallel` **37.0**, `mfc-parallel` **37.0**, `serial` **37.0**.

- Major Bugfix: Crash beim Starten von ProNET.
- Major Bugfix: Disksalv und Workbench/NewDrawer funktioniert jetzt richtig.
- Server-Neustart ist jetzt teilweise möglich, befindet sich aber immer noch in der Entwicklungsphase.
- bertragungsprotokoll und Deviceschnittstelle geändert -> inkompatibel zu ProNET V3!
- ...und ein paar andere Kleinigkeiten.

6.3.5 Version 3

Freigegeben am 23-Aug-96, mit `pronet.device` **36.6**, `pronet-handler` **36.4**, `pronet-server` **36.4**, `pronet-start` **36.3**, `pronet-stop` **36.1**, `pronet-flush` **36.0**, `pronet-talk` **36.0**, `pronet-run` **36.1**, `pronet-page` **36.0**, `internal-parallel` **36.1**, `mfc-parallel` **36.0**, `serial` **36.2**.

- Kein Herumfummeln mehr mit der MountList und der ProNET.config, 'pronet-start' erledigt alles!

- MultiFaceCard3-Treiber geschrieben.
- Serial-Treiber funktioniert jetzt richtig.
- Netzwerk kann wieder heruntergefahren werden, ohne den Rechner neuzustarten.
- Komplette neue Dokumentation.
- Serial-Treiber arbeitet auf Wunsch auch mit 3-Draht-Kabeln.
- Kommandozeilenprüfung in 'pronet-page' and 'pronet-run'.
- 'pronet-talk' benutzt jetzt immer Port 1994.
- Neuer 'pronet.device' Fehlercode 'PNDERR_UNIT_NOT_DEFINED', Fehler in der Device-Open-Routine behoben.

6.3.6 Version 2.1

(Bugfix) Freigegeben am 08-Jul-95, mit pronet-talk **34.2**, pronet-server **35.0**.

- pronet-talk UND pronet-server akzeptierten das UNIT-Argument nicht...

6.3.7 Version 2

Freigegeben am 24-Jun-95, mit pronet.device **35.4**, pronet-handler **34.2**, pronet-server **34.9**, pronet-talk **34.1**, pronet-run **34.1**, pronet-page **34.0**.

- Wiederverbindung.
- Device friert den Rechner nicht mehr ein.
- Priorität des Transferprozesses kann eingestellt werden.
- Diskwechsel-Erkennung verbessert.
- Komplette neue Dokumentation
- READ-Actions werden gesplittet übertragen und verbrauchen nicht mehr so viel Speicher.
- EXAMINE-Packets bewirkten Speicherverlust auf dem Server.
- Komplette neues pronet.device, mit modularem Treiberkonzept.
- "+"-Erweiterung >> Unitnummer
- C include file
- Alles, was ich sonst noch vergessen habe.
- Viele Bugfixes, ProNET läuft stabiler.

6.3.8 Version 1

Freigegeben am 01-Nov-94 (Erste Release), mit pronet.device **33.9**, pronet-handler **33.5**, pronet-server **33.5**, pronet-talk **33.0**, pronet-run **33.0**, pronet-page **33.0**.

6.3.9 Version 0

Mir kam die Idee für *ProNET* im Dezember 1993, nachdem ich von ParNet so enttäuscht war. Mit dem pronet.device fing ich im April 1994 an.

7 FAQ - Frequently Asked Questions

- Ist es möglich, das Netzwerk zu unterbrechen, um kurz etwas zu drucken und dann über eine Switchbox wieder aufs Netz zurückzuschalten?
Nein, das ist nicht möglich, und das wird es auch vermutlich nie sein, weil der Parallel-Treiber davon ausgeht, da die Hardware-Register für den Parallelport nicht verändert werden.
- Gibt es eine Möglichkeit, ein Programm auf einem Rechner zu starten und seine Ausgaben in einem CON:-Window auf einem anderen zu sehen?
Nein, aber das möchte ich in der nächsten Version einbauen.
- Ist es möglich, den seriellen Port eines anderen Netzrechners vom Hauptrechner aus mitzubeneutzen, um z.B. ein zweites Modem oder ein MIDI-Interface anzuschließen?
Nein, auch das wird es erst in einer späteren Version geben. Danke an Giovanni Gigante für den Vorschlag!
- Wird der 'pronet-handler' jemals auf Kickstart 1.3 laufen?
Nein.
- Würde eine spezielle 020/030/040-Version des 'pronet.device' die Geschwindigkeit erhöhen?
Nein.
- Ich habe einen Bug entdeckt: Wenn ich den Hauptcomputer resette und alle Devices neu mounte, kann ich ohne Probleme weiterarbeiten. Wenn ich allerdings den Server resette und den 'pronet-server' dann neu starte, läuft nichts mehr.
Das ist kein Bug, weil der 'pronet-server' einige wichtige Datenstrukturen für den Hauptrechner erzeugt, und diese werden gelöscht, sobald Sie den Server resettet. Eine Folge davon ist auch, da in 'ber-Kreuz-Netzwerken' kein Rechner einen Reset machen darf. Das ist ein essentieller Unterschied zwischen *ProNET* und professionellen Netzwerksystemen.
- Kann man das pronet.device als Treiber für SANA-II Netzwerksoftware benutzen?
Nein, weil mein Device nicht nach den SANA-II-Regeln gebaut ist. Allerdings möchte ich eine Art Aufsatz für das pronet.device programmieren, so dass man alle ProNET-Treiber dann z.B. mit AmiTCP benutzen kann.
- Warum habe ich das Device nicht gleich SANA-II-kompatibel gemacht?
Weil ich noch nicht von diesem Standard gehört habe, als ich mit *ProNET* angefangen habe.
- Müß ich 'pronet-start' für jedes Gerät einzeln starten oder kann ich irgendwie die Aufrufe für mehrere Geräte zusammenfassen?
Nein. Erstellen Sie sich ein Shell-Script, wenn Sie zu faul sind ;-)
- Ich benutze ein CDTV bzw. A500+A570 als Server. Das CD-ROM-Laufwerk scheint Diskwechsel nicht zu erkennen. Ist das ein Bug in *ProNET*?
Nein, es handelt sich hierbei um einen Bug im CDTV-Dateisystem. Verwenden Sie einfach den Shell-Befehl `diskchange cd0:` auf dem Client oder auf dem Server.

- Ich benutze die Workbench, um groe Files auf den Server zu kopieren. Manchmal beschwert sich mein Rechner ber zu wenig Speicher.

Versuchen Sie, zum Kopieren die Shell zu benutzen oder ein DOPus-hnliches Tool. Es handelt sich dabei um ein Problem, welches bei groen ACTION_WRITEs auftritt, weil ein entsprechender Speicherblock auf dem Server alloziert werden mu.

Hmm... bitte entschuldigen Sie, da nicht allzuvielen Fragen dabei waren, die ich mit einem einfachen 'Ja' beantworten konnte :) Obwohl, Moment.. hier ist doch noch eine:

- Ist *ProNET* besser als ParNet?
<Was wohl?> :-)

8 Lizenz

Dieses Programm untersteht der GNU General Public License. Die genauen Bestimmungen stehen in der Datei COPYING.

Table of Contents

1	Einleitung	1
1.1	Was ist <i>ProNET</i> ?	1
1.2	Systemanforderungen	2
2	Installation	3
2.1	Installieren des ‘ <i>pronet.device</i> ’	3
2.1.1	Kopieren der Dateien	3
2.1.2	Installieren der Treiber	4
2.1.2.1	‘ <i>internal-parallel</i> ’	4
2.1.2.2	‘ <i>mfc-parallel</i> ’	4
2.1.2.3	‘ <i>serial</i> ’	5
2.2	Herstellen der Kabel	5
2.2.1	Das <i>ParNet</i> -Kabel	6
2.2.1.1	Wechselwirkungen mit dem seriellen Port	6
2.2.2	Das Null-Modem-Kabel	7
2.2.2.1	7-Draht-Schema	7
2.2.2.2	3-Draht-Schema	7
2.2.2.3	25-polige Steckverbinder	7
2.2.2.4	9-polige Steckverbinder	8
2.3	Installieren des Dateisystems	8
2.4	Installieren der Hilfsprogramme	8
2.5	Updaten von V3	9
3	Anwendung	11
3.1	Starten des Servers (‘ <i>pronet-server</i> ’)	11
3.2	Importieren von Gereten (‘ <i>pronet-start</i> ’)	11
3.3	‘ <i>pronet-stop</i> ’	12
3.4	‘ <i>pronet-flush</i> ’	12
3.5	‘ <i>pronet-page</i> ’	12
3.6	‘ <i>pronet-run</i> ’	12
3.7	‘ <i>pronet-talk</i> ’	13
4	Weitere Anwendungen	15
4.1	Wiederverbindung	15
4.2	Verbindungen ueber Kreuz	15
4.3	Netzwerk ueber die Telefonleitung	15
4.3.1	Vorbereitung	15
4.3.2	Verbindung	16

5	Interna.....	17
5.1	Erstellen eigener <i>ProNET</i> -Treiber	17
5.2	Unterstützte DosPacket-Typen	17
6	Verschiedenes	19
6.1	Support	19
6.2	Credits & Danksagungen	19
6.3	Geschichte	20
6.3.1	Version 3.4	20
6.3.2	Version 3.3	20
6.3.3	Version 3.2	20
6.3.4	Version 3.1	20
6.3.5	Version 3	20
6.3.6	Version 2.1	21
6.3.7	Version 2	21
6.3.8	Version 1	21
6.3.9	Version 0	21
7	FAQ - Frequently Asked Questions.....	23
8	Lizenz	25