

Zeitschrift: Visionen : Magazin des Vereins der Informatik Studierenden an der ETH Zürich
Herausgeber: Verein der Informatik Studierenden an der ETH Zürich
Band: - (1993)
Heft: 1

Heft

Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. [Siehe Rechtliche Hinweise.](#)

Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. [Voir Informations légales.](#)

Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. [See Legal notice.](#)

Download PDF: 06.01.2025

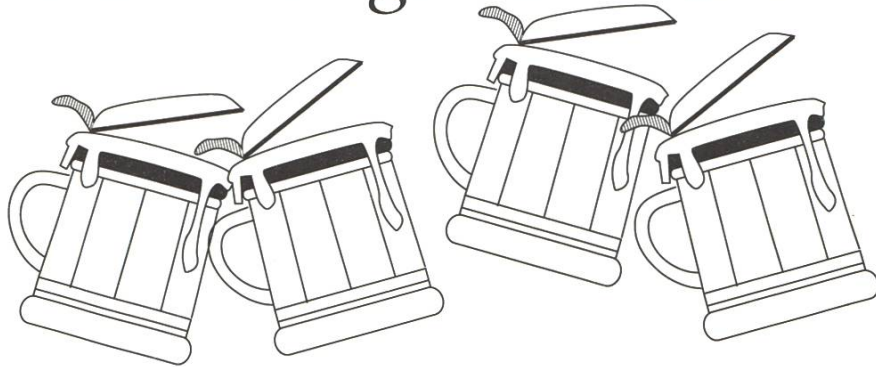
ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>

Visionen

1

Januar
93

MV mal ganz anders:



VIS Apéro im StuZ

17. Februar um 18:15

***Fachgruppe Oberon gegründet
Spielzeuge der Informatiker, Teil 3
Chris Flu's Kochecke: Poulet-
geschnetzeltes an Yngwersauce***

Adressen

Aktuarin: Grete Danielsen
Dohlenweg 26
8050 Zürich, Tel 01 / 302 48 97
e-mail: gcdaniel@iic.ethz.ch

Exkursionen: Christian Franz
Sonneggstr. 61
8006 Zürich, Tel. 01/ 261 26 96
e-mail: cfranz@iic.ethz.ch

Feste & Kultur: Frank Möhle
Dielsdorferstrasse 7
8155 Niederhasli, Tel. 01 / 851 03 21
e-mail: fmoehle@iic.ethz.ch

Präsident: Florian Schlotke
Roswiesenstr. 161
8051 Zürich, Tel. 01/ 321 46 23
email: fschlotk@iic.ethz.ch

Quästor: Daniel Kluge
Irringersteig 3
8006 Zürich, Tel. 01/ 252 04 14
e-mail: dankluge@iic.ethz.ch

Redaktor: George Fankhauser
Schaffhauserstr. 298
8050 Zürich, Tel. 01/ 312 10 32
e-mail: gfankhau@iic.ethz.ch

Verleger: Boris Nordenström
Hardstrasse 324
8005 Zürich, 01 / 273 24 80
e-mail: banordens@iic.ethz.ch

Visinfo(Infosystem): Maxim Samo
Forchstrasse 245
8032 Zürich, Tel. 01/ 381 17 50
e-mail: samo@nessie.cs.id.ethz.ch

Vordiplome: Marcel Waldvogel
Hägetstalerstr. 37
8610 Uster, Tel. 01/ 941 61 94
e-mail: mwaldvog@iic.ethz.ch

Impressum

Herausgeber:
Verein der Informatikstudierenden an
der ETH Zürich.

Verleger: Boris Nordenström
Redaktor: George Fankhauser

Adresse Verlag & Redaktion:
VIS
Verein der Informatikstudierenden
Haldeneggsteig 4, IFW B29
ETH Zentrum
8092 Zürich

Tel: 01 254 72 12 (Mo-Fr, 1215-1300)
e-mail: vis@iic.ethz.ch

Postscheckkonto 80-32779-3
Präsenzzeit: Mo..Fr: 1215..1300

Auflage: 1400
Inseratenpreis/Seite 500.-
Jahresabonnement 15.-

Redaktions- und Anzeigeschluss für
die nächste Ausgabe:

26. Februar 1993

Visionen

© 1992 by Verein der Informatikstudierenden

Tschau Zame

Die MV steht wieder einmal vor der Tür...

Nein, halt!, dageblieben. Diesmal wollen wir nicht, wie üblich vor gähnendem Publikum eine langweilige Traktandenliste herunterleiern. Ich habe vielmehr die grosse Ehre, alle VIS-Mitglieder ganz herzlich am Mittwoch, den 17. Februar um 18:15 Uhr im StuZ, zum grossen **VIS - Semesterend-Plauder und Kennenlern-Mitgliederversammlungs-Apéro** zu begrüssen. Die Traktandenliste wird sich auf das wesentliche beschränken: Mitteilungen, Finanzen und Wahlen. Wichtig sind vor allem die Wahlen. Aus dem VIS-Vorstand werden sich wohl zwei bis drei Leute verabschieden (so ist unter anderen auch für mich die Zeit gekommen...). Nachfolgespräche wurden zwar schon geführt, es ist aber trotzdem jede und jeder dazu eingeladen, sich um einen Posten zu bewerben und so den ersten Schritt in Richtung einer glanzvollen Karriere zu starten. Der VIS als Kaderschmiede: Ex-VIS-Mitarbeiter finden sich heute in gehobenen Positionen, wie Abteilungssekretär, Doktoranden oder Wirtschaftsvertreter an der Kontaktparty wieder.

Doch im Ernst, kommt einfach mal auf einen netten Abend im StuZ vorbei. Es gilt das VIS-Büro von diversen Getränken und Knabbergebäckmischungen zu befreien. Abgesehen davon hat mir Frank so etwas von einem *Buffet* geflüstert...

Ich hätte hier eigentlich noch gerne verschiedene Dinge angekündigt und bekanntgemacht, doch macht mir der zweiwöchige Erscheinungsdelay der VISIONEN zu schaffen, so dass die Informationen entweder beim Erscheinen schon veraltet sind oder zum Redaktionsschluss noch nicht definitiv vorlagen. Ich bitte Euch deshalb, doch ab und zu einen Blick auf die Schaukästen im IFW und HG zu werfen, um die aktuellsten Infos zu erhalten.

Ansonsten schärft die Messer und macht euch auf eine neue Runde von Chris Flu's sagenhafter Kochecke gefasst...

Florian Schlotke

Kontaktparty 1993

Es ist uns ein Bedürfnis, allen Firmenteilnehmern am Kontakttreffen vom 25. Januar 1993 nochmals unseren Dank auszusprechen.

Wir freuen uns auf weitere gute Zusammenarbeit und hoffen, Sie an der Kontaktparty 1994 wieder begrüßen zu dürfen.

Abteilungssekretariat IIIC und
Organisationskomitee des VIS

H. Hilgarth Ch. Franz

Neues aus der Stabstelle Software

Ein 1/4" Streamer Tape Drive QIC-150 ist für Sun Benutzer an die Workstation a15 in D33 angeschlossen worden. Dies ist ein Versuch. Wir nehmen an, falls Interesse seitens der Studenten besteht, dass der VIS fuer den Einkauf der Streamer Tape Cartridges besorgt sein wird. Preis ca. 30 sFr. pro Stück.

R. Wakefield

Chris Flu's Kochecke

Folge 8:

The return of the killer knife

oder: ein Lob dem Handwerk

oder: Pouletgeschmetzeltes an Ingwersauce

Ok, sorry, mea culpa, mea culpa, mea maxima culpa, ich fühle mich schuldig, mehrere hundert InformatikstudentInnen seien Anfang Jahr verhungert, verhungert! Man/frau stelle sich das vor: so mit Wasserbäuchen, Haar- und Zahnausfall, scheusslich - und alles, weil in den letzten Visionen keine Kochecke erschienen ist (ich verschweige wohl besser den Umstand, dass mir für diese Dezimierungsaktion von der Abteilungsleitung eine nicht zu unterschätzende Summe überwiesen worden ist (das spart ihr nämlich den Umweg über die Prüfungen...)). Dem muss selbstverständlich Einhalt geboten werden, soll der Fortbestand einer gesunden Studentenschaft unser aller Abteilung gesichert sein. Waaaaas? Ich soll mich für dieses Debakel entschuldigen? Ähm, tia, nannu, nasowas, sss..., hm, so, eben, tirili, also hatte halt mal eben so eine Identitätskrise (eigentlich ist mir das ja schon ein wenig peinlich, schliesslich erscheint dieses Käseblatt (sorry George) immerhin in einer Auflage

von 1600 Exemplaren), genauer gesagt: vor einer Woche bin ich aus der Nervenheilstation entlassen worden, wohin mich eine Reihe höchst unglücklicher Vorfälle Ende Dezember gebracht hat.

Mein von Natur aus eher labiles Gemüt war ja durch die Geschichte mit dem Kühlschrank meiner Eltern (siehe Visionen 10/92) schon arg angeschlagen - aber das war nur der Anfang...

In meinem Freundes- und Bekanntenkreis ist es allgemein bekannt, äh befreundet, äh... dass mich wenigstens drei Dinge von einem seriösen Studium abhalten: Wein, Weib und Gesang (sprich: meine Liebe zum Kochen, meine Liebe zu (na das ist dann also wirklich meine Privatsache, immerhin erscheint usw...) und die Liebe zu meiner Trompete (es ist nicht jedermann/frau mit einem Nachtigallenstimmchen gesegnet)). Es scheint deshalb nicht ganz abwegig, mich im weihnachtlichen MitGeschenkenUmSichWerfTaumel küchenmaterialmäßig für die nächsten fünfzehn Jahre inklusive den vier in der Zwischenzeit zu erwartenden Sintfluten einzudecken, was dann auch etwelche taten mit dem Ergebnis, dass sich in meinem Schlafzimmer nun die Suppenkellen, Fleischermesser, Schleifsteine, Fondue-réchauds, Woks, selbstreinigenden Schüsseln, Polentapfannen, Zitronenpressen, Wallhölzer, und was es halt sonst noch so an analogen Küchengeräten gibt, stapeln. Nun, so schlimm

ist das ja nicht, beim Einschalten des Kühlschranks fällt nun einfach zusätzlich noch die Pastamaschine auf das teflonbeschichtete Kuchenblech und die kunstvoll aufgeschichteten Espressomaschinen ergiessen sich über meine Bettdecke, alles harmlos gegen mein elektrisches Brotmesser. Ihr lacht jetzt noch, ha, das Lachen wird Euch früh genug vergehen; die Geister, die ich gerufen usw...

Ich will ja nichts gegen Dani sagen, er ist echt ein guter Freund und ich habe mich ja auch riesig gefreut, als er am 24. Dezember abends um acht hereinplatzte und mich so gerade noch der Pflicht enthob meine Stimme zu einem jubelnden "Stille Nacht" im trauten Kreise der Familie zu erheben. Er müsse gleich wieder gehen, hat er gesagt, die daheim warteten, inzwischen werde ich die Vermutung nicht los, ich will ja niemanden verdächtigen, aber immerhin wäre es möglich, dass dieses ominöse Päckchen, welches er mir überreichte, bereits unter seinem Weihnachtsbaum... ein paar Tage später sei sein Haus irrtümlicherweise abgerissen worden, tja, ich weiss von nichts. Unter dem kunstvoll drapierten Geschenkpapier kam unvermutet ein elektrisches Brotmesser hervor (wenn Du das nicht bereits geahnt hast, wirst Du spätestens beim zweiten Vordiplom durchfliegen), wohlgemerkt ohne Bedienungsanleitung, die ja im allgemeinen sowieso nicht gelesen wird, speziell wenn es sich um Hardware handelt (spätestens nach dem ersten Headcrash sollte man sich versichern, ob man den SCSI Bus nicht

irrtümlicherweise am Netz angeschlossen hat (erkennbar an den 50 Hz-Einschlägen des Kopfs auf der Platte)). Drei Knöpfe, einen Netzschalter und ein beleuchtetes LC-Display besass das Ding. Üblicherweise deutet dies bei Geräten dieser Klasse auf zwei Geschwindigkeitsstufen und einen sogenannten Ökogang hin, der zwar tonnenweise Energie verbraucht aber so schwach ist, dass von der Funktionalität der Maschine nichts mehr vorhanden bleibt. Kurzerhand die Maschine eingeschaltet, ein Brot zur Hand, im Display erscheint die Meldung "Schwarzbrot, Alter: 10 Stunden",

hm, ist ja noch ganz hübsch, wie wär's denn nun mit einer Peperoni? Auch dieses Problem wird tadellos gemeistert: "Peperoni, Alter: 5 Tage"; ein Frischei: "Ei, Alter: 3 Monate". Ein klassischer Fall eines Moore-Automaten denkt sich der/die hardware-interessierte InformatikerIn nun; weit gefehlt: Irrtümlicherweise gerate ich auf die Ökotaste und siehe da, statt des erwarteten Feld- Wald- und Wiesenmenüs (ich habe eigentlich nichts gegen den Sparmodus, aber ein manuelles Brotmesser ist immer noch wesentlich sparsamer als das sparsamste Elektromesser) gerate ich in den, W sei's gelobt, Programmiermodus. Erfahrungswerte können hier eingegeben werden, womöglich ein neuronales Netz dahinter...mal ausprobieren...ha, da gibt's ja sogar einen Learnmodus...Peperoni daher... Zwiebel...

Pouletfleisch, da springt mir das Ding plötzlich aus der Hand, trennt den Wasserhahn ab, ein riesige Fontäne spritzt mir ins Gesicht, durch die Wassermassen kann ich gerade noch erkennen, dass die Maschine begonnen hat, das Küchenmobiliar in ofengerechte Form zu zersägen. Das Wasser reicht mir bereits bis zu den Knien, watend versuche ich die Stromzufuhr zu unterbrechen, aber da greift mich *mein* Brotmesser hinterlistig an und ich kann mich nur noch tauchend dem Sägeblatt entziehen. Als ich im Wohnzimmer nach Luft schnappe, hat dieses Monstrum bereits die Designermöbel und den heissgeliebten Picasso zerstört, wühlt in den Innereien der Stereoanlage und macht sich an den Fensterrahmen zu schaffen. Das Wasser reicht inzwischen bis zur Taille, die Schallplattensammlung ist im Eimer, auch das 17-bändige Konversationslexikon meiner Grossmutter selig aus dem 18. Jahrhundert. Ich stelle erleichtert fest, dass das Wasser abzufließen beginnt - aber nur, weil das Messer in seinem überbordenden Lerneifer die Aussenmauer vom Boden bis zur Decke abgetragen hat. Letztere beginnt sich bereits langsam zu senken, gerade noch kann ich meine Eltern auf die Strasse befördern, dann stürzt das Haus ein. Ein paar wenige Sekunden höre ich noch ein sägendes Geräusch, das leiser und leiser wird, dann ist alles ruhig. Unter den Trümmern soll man/frau übrigens ein Loch im Boden gefunden haben, es wurde damals auf etwa 5 Kilometer Tiefe geschätzt (übrigens ein weiteres Indiz dafür, dass sich

unter unseren Füßen keine bedeutenden Erdölvorkommen befinden...), wie tief es heute ist lässt sich schwer sagen. Laut meinen Berechnungen dürfte Ende Februar die Erde durchquert sein...

Dass ich nach diesem Erlebnis, mit einer Axt bewaffnet, sämtliche Haushaltsgeschäfte der Stadt konsultierte, dürfte wohl niemanden erstaunen (besonders wenn man bedenkt, dass ich dereinst der glückliche Erbe oben beschriebener Liegenschaft hätte sein sollen). Über die Folgen dieser Aktion habe ich bereits orientiert. Es versteht sich von selbst, dass ich für die Herstellung des heutigen Gerichts von der Verwendung jeglicher Automaten abrate, denn *jeder Moore ist ein potentieller Mealy*.

A propos Meal: das heutige Rezept ist eine Erfindung, die ich ganz bescheiden mir zuschiebe, es ist (wie immer) kinderleicht, (wie immer) schnell und (wie ...) wahnsinnig lecker.

Wir benötigen (für 2-3 StudentInnen):

200 g geschnetzeltes Pouletfleisch
1 El Honig
1/2 Zitrone
(oder 2 El Zitronensaftkonzentrat)
1 dl Weisswein
1 El Sojasauce
2 El frischer Ingwer (wer Lust hat, kann auch ne ganze Knolle reinpellen)
1 Zwiebel
2 Knoblauchzehen
1/2 rote Peperoni
1/2 gelbe Peperoni
2 Teelöffel Maizena
Salz zum abschmecken

Man/frau nehme eine Schüssel und gebe in diese hinein das Pouletfleisch, den Honig, den Saft von di Zitronä (ausdrücken, verdammt nochmal, wie oft muss ich das noch sagen), den Weisswein, die Sojasauce, den Ingwer (frischen Ingwer kriegste in jedem Supermarkt: Haut wegschneiden und fein hacken) und die Knoblauchzehen (wieder mal gepresst, nur so zur Abwechslung). Gut umrühren und eine Stunde (oder länger) stehen lassen: sowas wird im Volksmund "marinieren" genannt (hat nichts mit Meer, Nieren oder Pissen zu tun).

Inzwischen haben wir genügend Zeit um die Peperoni in feine Streifen zu schneiden (Kerne wegschneiden und dem Meerschweinchen oder der WG-Sau verfüttern) und die Zwiebel zu hacken (ich lasse die sonst üblichen Bemerkungen zu diesem Thema ausnahmsweise beiseite).

Wer nun nichts mehr zu tun hat, kann sich dem ZüriTip-Kreuzworträtsel, RTL Plus oder der letzten Insy-Übung zuwenden. Jetzt geht's erst richtig los: das Fleisch wird aus der Marinade genommen, in wenig Öl (ziemlich heiss werden lassen, bevor ihr das Fleisch hineinschmeisst) angebraten, Zwiebel dazu, weiterdünsten bis diese leicht glasig ist (siehe Visionen 12/91, die Kleinen können dieses Heft selbstverständlich zu einem Unkostenbeitrag von 15.- beim Verlag beziehen), Peperoni dazu, Marinade dazu, mit Salz abschmecken, 10 Minuten kochen las-

sen, das Maizena in wenig Wasser auflösen, ebenfalls begeben und nochmals kurz aufkochen - fertig.

Weter de yelaach Nok di trückerey hafden Für örgentwelke drukfelär Örgendwelscher ard on länen allä änlichkeyt Mid die lebendige lüüüte oder irchendwelche anterä bersonen oufs höftigste in irgöndwölcher ard ap. ond Sowiso hassend wör ale Dises chleygetruckde langsm.

Guäten Appätiit!

Chris Flu IIIIC/7



Masus.
42, öh... 93

Einladung
zum grossen
Mitgliederversammlungs-
Semesterend-

APÉRO

des VIS im Wintersemester 92/93
am Montag, 17. Febr. um 18¹⁵ im StuZ-Saal

Traktanden:

1. Begrüssung
2. Wahl der StimmenzählerInnen
3. Wahl der / des ProtokollführerIn
4. Änderung und Genehmigung des letzten Protokolls
5. Änderung und Genehmigung der Traktandenliste
6. Rechnung und Budget, Entlastung des Vorstandes
7. Mitteilungen des Vorstandes, der AK/UK, des DC, der Kommissionen und der Mitglieder
8. Bestätigung der Kommissionen
9. Wahl des Präsidenten/ der Präsidentin
10. Wahl des Vorstands
11. Wahlen AK/UK und DC
12. Resolutionen
13. Varia

Sollten noch Traktanden hinzukommen, so werden diese in den Schaukästen veröffentlicht. Anträge können jederzeit beim VIS abgegeben werden.

Fachgruppe Oberon der SI

Am 12. Januar wurde die Fachgruppe "Oberon" oder der Schweizer Informatiker Gesellschaft gegründet (Im internationalen Verkehr verwenden wir den Namen "Oberon User Group"). Sie bezweckt die aktive Förderung des Einsatzes von Oberon System und Sprache in Ausbildung und Industrie. Der Beitritt ist empfohlen, erwünscht und ab sofort möglich. Beitrittsformulare können bestellt werden bei:

*Schweizer Informatiker Gesellschaft
Schwandenholtzstrasse 286
CH-8046 Zürich
Telefon: +41-1-3717342
Fax +41-1-3712300;
e-mail Internet: si@ifi.unizh.ch*

Bei Fragen und Problemen steht Dir Pascal Faivre (eMail-INTERNET: 100021.3471@compuserve.com) jederzeit gerne zur Verfügung.

Vorerst werden für Mitglieder der Fachgruppe folgende Dienstleistungen geboten:

Unterstützung durch Spezialisten zur Beantwortung von Fragen aller Art:
Permanent:

- USENET newsgroup comp.lang.oberon (ab April 93)
- eMail:
>INTERNET:100021.3471@compuserve.com (prov. Adresse), Benutzer sollen unter "Subject" den Vermerk *USERSUPPORT* anbringen.
- Montag und Mittwoch 1200-1300 (Während der Hochschulsesemester)
- Präsenzzeit: Zürich ETH, Gebäude IFW, Raum A44
- Telefonische Hotline: ++41-1-2547213
- FAQ-List (Frequently Asked Questions List) (ab April 93)
- Software- und ProjektKatalog (ab April 93)
- Vertrieb von Compilern, Systemen, Systemerweiterungen

Bis in zwei, drei Monaten werden wir über eine Pressemappe verfügen und auch regelmässig redaktionelle Beiträge in den SI-Informationen verbreiten.

Mitglieder des Komitees sind:

- Pascal Faivre (Vorsitz)
- Markus Dätwyler
- Patrick Saladin
- Christian Schäpper
- Dr. Gerhard Schwärzler
- Josef Templ

Pascal Faivre

M MIGROS-GENOSSENSCHAFTS-BUND

INFORMATIK

Die Informatik des Migros-Genossenschafts-Bund plant und entwickelt für die Zukunft!

- Moderne Tele-Kommunikationsnetze für die ganze Migros-Gemeinschaft
- Optimale Logistik- und Lagerbewirtschaftungssysteme
- Effiziente Datenbanken
- Ausgereifte Rechnerverbund-Lösungen
- Experten-Systeme

Sind Sie der ausgewiesene Spezialist,

dann können Sie aus dem Vollen schöpfen und bei der Mitarbeit in einem dieser Projekte einen massgeblichen Beitrag leisten.

Kleine Teams und ein freundschaftliches Arbeitsklima tragen das ihre dazu bei!

Wir freuen uns auf Ihren Anruf, Sie werden alles Wichtige über Ihre zukünftige Laufbahn erfahren.

Unsere Adresse:

Migros-Genossenschafts-Bund
Informatik
Limmatstrasse 152
8005 Zürich
Tel: 01 277 21 11

Tücken in C++

Der Release 3.0 von C++ hat viel Neues gebracht: Multiple Inheritance, Nested Types, Templates und Exception Handling. Es gibt allerdings noch keinen Compiler, der diesen Standard erfüllt. Doch auch ohne diese Konzepte gibt es reichlich Tücken in C++. Die hier präsentierten Programme sind immer vollständig und können ohne Fehlermeldung kompiliert werden. Alle aber haben einen oder mehrere Fehler, die sich erst zur Laufzeit zeigen oder überhaupt nur an Seiteneffekten zu erkennen sind. Zu den gemeineren Fehlern gehört Speicher, der nicht mehr freigegeben wird, Pointer, die in ungültige Speicherbereiche zeigen, Speicher, der mehrfach freigegeben wird und ähnliches. Alle Fehler können mit kleinen Änderungen behoben werden. Die fünf Programme wurden auf einer SparcStation mit Sun C++ 2.1 kompiliert und getestet.

Programm 1

```
#include <string.h>

class Array {
private:
    int size;
    int lBnd, hBnd;
    int *data;
public:
    Array(int lb, int hb);
    ~Array();
};

Array::Array(int lb, int hb)
: size(hb-lb+1), lBnd(lb), hBnd(hb),
data(new int[size])
{}
Array::~Array()
```

```
{ delete [] data;
}

class NamedArray: public Array {
private:
    const char* const arrayName;
public:
    NamedArray(int lb, int hb, const char
*s);
    ~NamedArray();
};

NamedArray::NamedArray(int lb, int hb,
const char *s)
: Array(lb, hb), arrayName(new char[32])
{ strcpy(arrayName, s);
}
NamedArray::~NamedArray()
{ delete [] arrayName;
}

main()
{
    Array *pa = new NamedArray(3, 20,
"Trouble");
    delete pa;
}
```

Programm 2

```
class SimpleArray {
private:
    int *data;
    int size;
    int lBnd;
public:
    SimpleArray(int lb, int hb);
    ~SimpleArray();
    int& operator[](int i) { return
data[i-lBnd]; }
};

SimpleArray::SimpleArray(int lb, int hb)
: lBnd(lb), size(hb-lBnd+1), data(new
int[size])
{}
SimpleArray::~SimpleArray()
{ delete [] data;
}

main()
{
    SimpleArray pa(100,20000);
    pa[19000] = 15;
}
```

Programm 3

```
class String {
public:
    String();
    ~String();
private:
    char *data;
};

String::String()
{ data = new char[1]; *data = '\0';
}
String::~String()
{ delete data;
}
```

```

main()
{
  String *stringArray = new String[10];
  delete stringArray;
}

```

Programm 4

```

#include <string.h>

class String {
private:
  char *str;
public:
  String(char *s);
  ~String() { delete [] str; }
  String& operator=(const String& rhs);
  operator const char*() const { return
str; }
};
String::String(char *s)
{ str = new char[ strlen(s) ];
  strcpy(str, s);
}
String& String::operator=(const String&
rhs)
{ delete [] str;
  str = new char[ strlen(rhs) ];
  strcpy(str, rhs);
  return *this;
}

main()
{
  String h("Hello");
  String e = h;
}

```

Programm 5

Jeder, der schon C oder C++ programmiert hat, weiss, dass 0 nicht nur ein int-Wert ist, sondern auch der NIL-Wert für alle Arten von Pointern. Wie also übersetzt der C++-Compiler das folgende Beispiel? Ruft er in main() den Constructor Mouse(char*) oder den Constructor Mouse(int) auf, oder gibt er wegen Zweideutigkeit eine Fehlermeldung aus?

```

#include <iostream.h>
#include <string.h>

class Mouse {
private:
  char *name;
  int children;
public:
  Mouse(char *s);
  Mouse(int cds);
  ~Mouse();
};

```

```

Mouse::Mouse(char *s)
{ cout << "Mouse::Mouse(char*) called."
<< endl;
  children = 0; name = s;
}
Mouse::Mouse(int cds)
{ cout << "Mouse::Mouse(int) called." <<
endl;
  children = cds; name = 0;
}
Mouse::~~Mouse()
{ delete [] name;
}

main()
{
  Mouse *m = new Mouse(0);
  delete m;
}

```

Die Lösungen zu diesen fünf Aufgaben werden in der nächsten Ausgabe der Visionen zu finden sein. Dort wird auch der Code abgedruckt, den ich benutzte, um einige Programmfehler nachzuweisen.

Manuel Bleichenbacher IIC/5

Wer sich über C++ genauer informieren möchte, dem seien die folgenden Bücher empfohlen:

Stanley B. Lippman,

C++ Primer, 2nd Edition, Addison-Wesley, 1991, eine gute Einführung in C++.

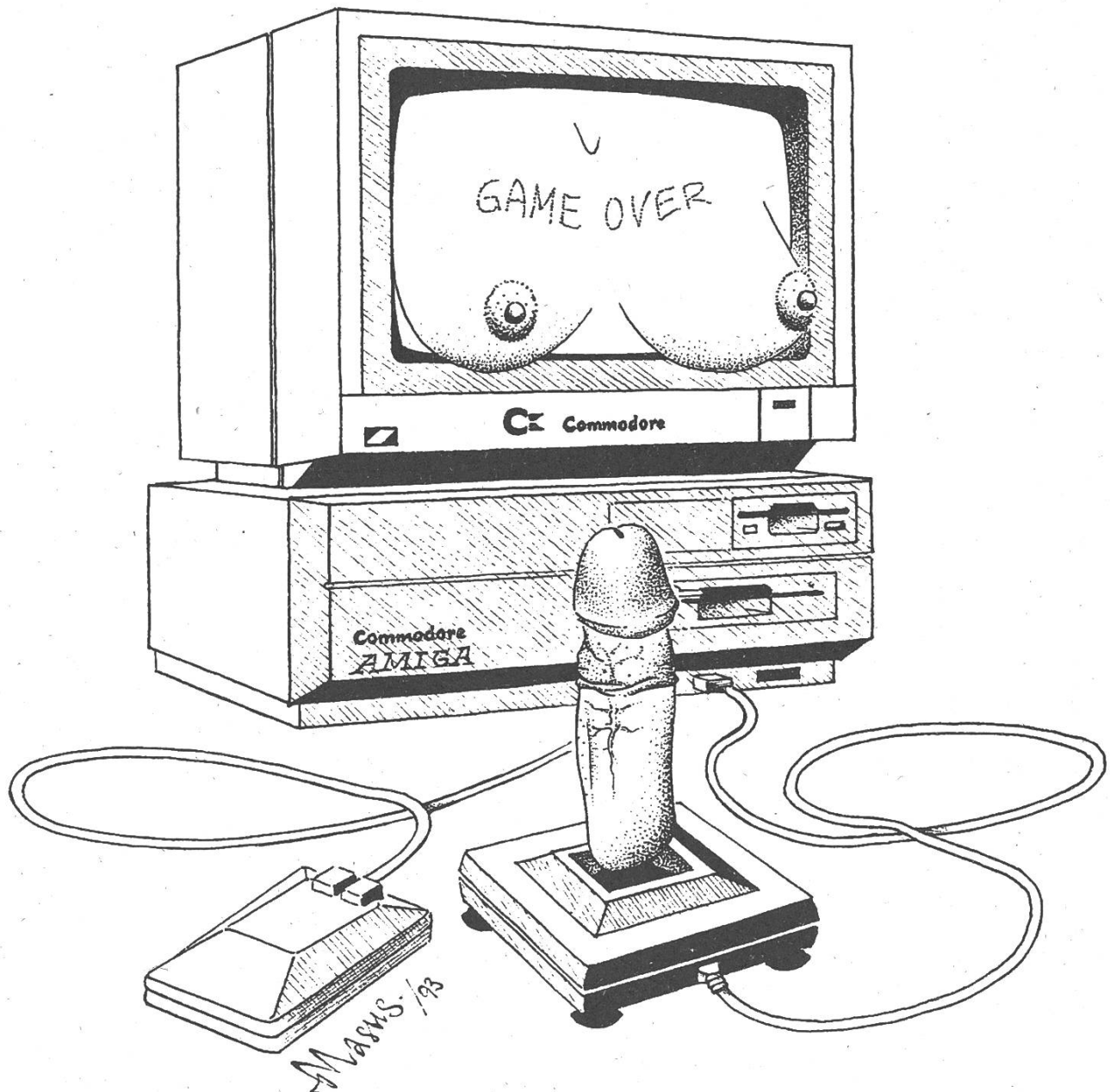
Bjarne Stroustrup,

The C++ Programming Language, 2nd Edition, Addison-Wesley, 1991, ein Buch, das ins Detail geht.

Margaret Ellis and Bjarne Stroustrup,

The Annotated C++ Reference Manual, Addison-Wesley, 1990. Hier steht alles drin über C++, allerdings nur als Nachschlagewerk brauchbar.

Spielzeuge der Informatiker, 3. Folge



Die Freundin

Achtung: Dieses Bild ist sexistisch und könnte insbesondere weibliche oder schwule Gefühle verletzen. Der Zeichner bittet um Nachsicht...

Praktikum bei Landis & Gyr Building Control

Die Firma

Building Control ist ein Unternehmensbereich der Landis & Gyr AG, einer weltweit tätigen Firma mit Hauptsitz in Zug. Es werden Lösungen für das technische Infrastruktur-Management von Wohn- und Geschäftsbauten angeboten, mit dem Ziel, den Gesamtenergieverbrauch und den Schadstoffausstoß zu minimieren, die Wirtschaftlichkeit der Bauten zu verbessern und deren störungsfreien Betrieb zu gewährleisten. Building Control beschäftigt weltweit etwa 8000 Mitarbeiter, davon rund 1800 in der Schweiz.

Innerhalb des Bereiches findet sich auch die Abteilung "System Products" und der Fachbereich "Engineering Tool" (ETO), wo ich während der 13 Wochen Praktikum arbeitete. Im ETO werden vor allem graphische Systeme und technische Datenbanken entwickelt, mit deren Hilfe Konfigurationswerkzeuge und Bedienstationen für neue Gebäude-Automatisierungssysteme hergestellt werden.

Die Arbeit

Die Ingenieure von ETO entwickelten in Modula-2 ein Graphiksystem (GBS), welches auf dem Presentation Manager (PM) von OS/2 aufsetzt und ihnen ermöglicht, neuen Anforderungen möglichst schnell und präzise

gerecht zu werden.

Meine erste Aufgabe bestand darin, einen GBS-Anschluss ans PM-Clipboard zu schreiben. Die Schnittstelle sollte es ermöglichen, Daten vom Format TEXT, BITMAP und META-FILE ins Clipboard zu schreiben, bzw. von dort zu lesen. Ausserdem wurde gewünscht, den dynamischen Datenaustausch zwischen Excel und anderen Applikationen mittels Clipboard zu unterstützen. Ein Testprogramm sollte Beispiele aller Formate zur Verfügung stellen und durch Anzeigen auf dem Bildschirm die jeweiligen Operationen testen. Die nicht gerade wenigen Funktionen des PM-API und das etwas sehr kärglich dokumentierte GBS zehrten zwar anfänglich ziemlich an den Informatiker-Nerven, doch kam Zeit, kam auch Rat.

Als zweites galt es, einen GBS-Anschluss für Dialog-Ressourcen zu erstellen. Solche Ressourcen definieren eine oder mehrere Dialogboxen, welche ihrerseits verschiedene Dialogelemente wie Pushbuttons, Radio-buttons, Checkboxes, Entryfields, Listboxes, Comboboxes, Spinbuttons, Scrollbars etc. enthalten können. Die Dialogboxen werden entweder mit interaktiven Tools definiert, oder direkt in der dafür vorgesehenen Resource-Sprache beschrieben. Der Compiler übersetzt diese Definitionen in eine Form, die vom Programm zur Laufzeit verwendet werden kann. Die zu erstellende Schnittstelle musste beliebig viele Dialogboxen (modale und nicht modale gemischt) und alle

möglichen Dialogelemente unterstützen. Der Nutzen der Schnittstelle besteht nun darin, dass in Zukunft Dialogelemente sehr bequem mit dem Editor gezeichnet werden können, und sich die Semantik mit genau einer einfachen callback-Prozedur beschreiben lässt.

Die beiden Arbeiten erlaubten einen vertieften Eindruck in den Presentation Manager von OS/2 und in das Graphiksystem der Firma. Es war interessant, mich in diese Strukturen hineinzuwühlen und ein Produkt zu erzeugen, das hoffentlich bestens in das ETO-Konzept hineinpasst. Grossen Ärger – gleichzeitig aber auch Beustigung – bereitete mir die Erkenntnis, dass auch die Cracks von IBM ganz schlechte Tage zu haben scheinen. Aber nicht nur die. Jedenfalls gelang es, noch rechtzeitig die Dokumentationen abzuschliessen und die beiden Arbeiten als kleine Präsentation den Arbeitskollegen vorzuführen.

Das Umfeld

Zur technischen Seite lässt sich sagen, dass in diesem Fachbereich durchwegs mit vernetzten PC's der 386/-486-Klasse gearbeitet wird. Auch der Rest der Hardware entspricht so ziemlich der Luxus-Klasse. Zum Beispiel stand mir ein modernster 17-Zoll Bildschirm zur Verfügung, dessen Qualität sich bei langen Programmierzeiten halt eben schon zeigt. Programmiert wird vorallem mit Logitech Modula-2, mit Struktur zum Erfolg!

Die Mitarbeiter vom ETO nahmen mich ohne Umstände in ihr Grüppchen auf und liessen mich voll am Firmenleben teilnehmen. Meine Betreuer verhielten sich eher zurückhaltend und gaben mir relativ viel Freiraum bei der Arbeit, standen aber immer sofort zur Verfügung, wenn es irgendwo brannte. Ich fühlte mich schon bald sehr wohl.

Und noch etwas

Die intensivste Erfahrung, die ich während des Praktikums machte, ist eine bedrückende. Die zur Zeit schlechte Wirtschaftslage beeinflusste das Arbeitsklima nachhaltig, war oft Gesprächsthema Nummer eins; klar, wenn massive Entlassungen ins Haus stehen. Und wenn der ETH-Absolvent nun denkt, so was könne ihm bei seiner Ausbildung nicht passieren, so hat er sich leider getäuscht. Verständlich, dass solche Bedingungen nicht zum Jubeln Anlass geben. Für mich, als nicht direkt Betroffener, war das Teilnehmen an solchen Gesprächen ungemein interessant und lehrreich, denn sie ermöglichten mir sehr schnell einen Einblick in die geschäftlichen Sorgen und Nöte der Mitarbeiter und die Problematik eines Grossunternehmens.

Für die freundliche Aufnahme möchte ich allen ganz herzlich danken. Ich wünsche Euch einen friedlichen Umzug ins neue Gebäude. Denkt an die neue Mensa!

Hanspeter Lüthi, IIC/7

Should Software Be Copy Protected

Part III

Tentative Conclusion

In researching the topic, this researcher has reached the tentative conclusion that some form of copy protection is necessary to protect the rights of the software publisher. The type of protection scheme advocated should be one of the passive forms previously described. This would provide software developers some protection from indiscriminate software pirates and also allow users unhindered use of the software they have legitimately purchased.

Although not completely passive, it appears that the new OIDS-type protection scheme would be a good choice because it provides maximum protection for the publisher while minimizing the problems encountered by users. Recall that this type of protection allows full use of the software once it has been properly registered.

Alternatively, a developer could use the code lookup form of protection, but it should be implemented in such a way that users will not be annoyed. This means requiring a code lookup only when the program is run (not repeatedly while the program is being used), and having the codes printed (embedded) as part of the user's manual, not on a separate sheet of paper that can be easily lost.

Registration screens would be the least favored method because it has been shown that they provide little or no real protection. Instead they make an appeal to the user's integrity and act only as a deterrent.

Coupled with this protection, the software industry should attempt to make the registration of software a value-added incentive such as providing free upgrades to future products or extended support services. Once a software product is registered, assuming it has a unique serial number, there is a way of tracking pirated software to a

specific source. At this point the laws regarding illegal copying of software should be more strictly enforced.

In support of this conclusion, this researcher argues that even though the effects of software piracy may be exaggerated, the fact remains that software developers lose a great deal of revenue to software piracy. Furthermore, the majority of the pirating is performed by everyday users. It is felt that the above course of action would reduce the losses suffered by the software industry. Most users simply do not have the knowledge, time or inclination required to defeat protection.

Some users may argue against this proposed solution because it still advocates some form of copy protection. However, since most of the problems users have encountered from copy protection occurred when one of the "active" forms was used, this researcher feels that they would encounter few problems from the "passive" form proposed.

Values

In prescribing the above course of action, this researcher believes that a compromise solution should be reached among the parties involved. Fairness is valued above all other competing values. This view is founded on the principle of equality which states, "Choose that option which, more than any other available option, maximizes benefits to the greatest number of people and/or minimizes potential harm to any one person or group." It is believed that the proposed solution protects the interests of both parties involved to the greatest extent possible.

For example, software developers would benefit because more people will buy their products instead of stealing them. This allows developers to earn what they deserve for their work and stay in business. Legitimate users would benefit because they will not be bothered by risky copy protection schemes, and more importantly will continue to receive support for the software they have purchased from the software developers who might otherwise go out of business.

A secondary value held by this researcher is

Informatik-Projekte zu einem festen Preis?

Normalfall

Cancel

- Das Software-Haus macht eine oberflächliche Schätzung des Projektaufwands
- Aufgrund dieser Schätzung vereinbaren Auftraggeber und Software-Haus einen Stunden- oder Tagesstarif für die eingesetzten Informatiker
- Je länger das Projekt dauert, desto grösser die Einnahmen des Software-Hauses

Festpreisprojekt

OK

- Das Software-Haus schätzt den Projektaufwand aufgrund seiner grossen Erfahrung genau ab und unterbreitet ein Festpreisangebot
- Der Auftraggeber erteilt den Auftrag
- Die Verantwortung für das Einhalten der Kosten- und Terminbudgets liegt beim Software-Haus

Economation AG wickelt seit 20 Jahren erfolgreiche Informatik-Projekte (darunter viele zu einem festen Preis) für namhafte Kunden ab.

An unseren beiden Geschäftsstellen beschäftigen wir insgesamt 35 Mitarbeiter (80% mit Hochschulabschluss).

Wir betreiben eine eigene Entwicklungsinfrastruktur, die Hardware und Betriebssysteme unterschiedlicher Hersteller miteinander verbindet (Schwerpunkt UNIX).

Interessiert? Nehmen Sie mit uns Kontakt auf!

economation⁺

AG für Computertechnik und Automation

Stockerstrasse 46

CH-8039 Zürich

Tel. 01/201 25 52

Fax 01/201 25 56

Freie Strasse 3

CH-4001 Basel

Tel. 061/261 66 01

Fax 061/261 87 32

the belief that unauthorized copying or use of software is still stealing, no matter what euphemistic name it is given. Most people would not think of going into a store and shoplifting some item, no matter what the price of the item is, yet those same people will often rationalize their participation in software piracy.

Interviews

In order to test the tentative solution, this researcher interviewed several individuals from each side of the copy protection issue. Those persons interviewed were: John Calhoun, a software developer, whose company, Soft Dorothy Software, has produced several shareware and commercial games; Wayne Baker, marketing coordinator at Origin Systems Inc., publisher of several computer-based fantasy role-playing games; Robert Allgood, CAD manager at Motorola Inc., whose organization oversees a large computer network and supports over 75 engineers; Steve Holden, a Macintosh computer consultant at Science Applications International Corp.; Glenda Keyworth, manager of the Academic Computing Lab and RCM instructor at St. Edward's University; Robert Dorsett, a contract programmer; Povl Pedersen, a student studying economics and assistant at a university computer lab; Dave Smith a part-time software developer for a small software company whose next product may use the OIDS-style protection technique; and a person who works as a programmer for the research services department at a major university library who chose to remain anonymous.

Using the questions shown in Appendix B, this researcher found that the opinions regarding the issue are not clearly divided among software developers and software users.

Should Software be Copy protected?

Surprisingly, most of those interviewed feel that software should not be copy protected. Glenda Keyworth, Robert Dorsett, John Calhoun, Robert Allgood, and the anonymous

user believe that more problems are created by copy protection than are solved by it. Steve Holden feels that copy protection is an individual decision and every developer has the right to copy protect their software if they choose to. Wayne Baker believes that there needs to be some means to prevent people from pirating but that does not necessarily mean copy protection. He feels that there may be better alternatives.

Dave Smith was the only strong supporter of software copy protection. Mr. Smith feels that users who have paid for software should not subsidize those who have stolen it. He believes that if the amount of pirating were reduced then software prices would fall.

Does copy protection serve its intended purpose?

Ironically, given the above opinions, almost all those interviewed feel that software copy protection does serve its intended purpose, that is, to reduce the amount of software pirating. John Calhoun believes that most copy protection is very effective, since "99% of the piracy that goes on out there is done by 'amateurs' - users who pirate because they can. When you introduce a copy protection scheme, these amateurs whine and bawl about it, but are nonetheless incapable of pirating the software." Povl Pedersen, Wayne Baker, Robert Allgood, Glenda Keyworth and the anonymous respondent agree with Mr. Calhoun, but question the degree of success it achieves and feel the effect is probably negligible.

Robert Dorsett believes that software copy protection does "absolutely nothing" to reduce the amount of pirating. He thinks that the "amateur" users comprise a small minority of the persons pirating software. Instead he thinks most users possess the knowledge required to remove or bypass any copy protection scheme.

What other measures could reduce software pirating?

Five of the nine persons interviewed feel that lowering the price of software would have the largest effect in reducing the amount of soft-

WG mit Drucker gesucht.

BIRD/ERIC



So unkompliziert, wie Apple Computer zu bedienen sind, so mühelos kann die Verbindung zu weiteren Geräten hergestellt werden. Zum Beispiel zum Drucker.

Zur Kontaktnahme genügt bekanntlich ein Kabel. Übrigens auch dann, wenn es sich um Anschlüsse an Produkte anderer Hersteller handelt.

Wer eine Apple-WG findet, hat also ausgesprochen Glück gehabt. Und unter uns: Es gibt immer mehr davon.

Generalvertretung für die Schweiz
und Liechtenstein:

Industrade AG

Apple Computer Division
Hertistrasse 31, 8304 Wallisellen
Telefon 01/832 81 11



Apple Computer

ware pirating. In addition, they feel that the documentation should be improved and additional services should be provided to the legitimate user free of charge. This view is exemplified by the anonymous respondent who stated, "Reduce the price of software, make the documentation good enough that people will be willing to buy the software just for the manuals. Free technical support and free and timely upgrades to registered users would also help." John Calhoun disagrees with this view, however, and states, "From a publisher/developer stand point, I don't think there are any other measures. Low price and high quality do not reduce piracy too much in my opinion." Glenda Keyworth stated that price is a big factor at the university, and that many students just cannot afford the popular software. She feels the educational discounts now offered by many software vendors help to reduce software piracy at the university.

Wayne Baker's employer, Origin Systems, takes a different approach with their adventure game software. Origin adds items to the software package that cannot be obtained without buying it. These items include: cloth maps, rune stones, posters, and copies of the game autographed by Lord British, a game character created by the founder of the company. It is believed that these items serve to make purchasing the games more appealing and make them more popular among game players.

Dave Smith and Robert Dorsett would prefer stronger legal action. They feel that aggressive enforcement of the copyright laws and stronger punishment would help reduce the amount of pirating.

Do the passive protection schemes offer sufficient protection?

Four of the nine persons interviewed believe that the many passive protection schemes (registration screens, authorization codes and code lookup) do very little to reduce the amount of software pirating, but may have some limited effect. Steve Holden stated that "it definitely makes people think." Povl Pedersen concurred that ". . . not all people

want their personalized copy floating around the world." Several of the respondents stated, however, that registration screens are the least effective form of copy protection because they do not require a user to enter "real" information during the registration process. Glenda Keyworth stated that she remembers seeing one program at the university registered to "Bart Simpson."

John Calhoun believes that the passive protection schemes may be more effective than most people think in preventing piracy. He compares software pirating to a disease and that illegal copies of software spread in much the same way. For example, if one person gives a copy of a program to two friends, who each in turn give two copies to friends or colleagues at work, who in turn pass the copy on to two more people, the number of illegal copies grow exponentially. If copy protection can prevent just one person from obtaining an illegal copy, the results could be significant.

What are the opinions regarding the proposed solution?

When asked to evaluate the proposed solution, Steve Holden, Dave Smith, Robert Allgood, and John Calhoun stated that it sounded like a reasonable solution. Steve Holden stated, "I think it is very future oriented and a decent attempt at providing an easy way for the user and the authors to protect themselves." John Calhoun stated that the OIDS-type of copy protection is the "best scheme" he has ever seen.

However, several of those interviewed did raise questions or foresee problems with the proposed solution. The first potential problem identified regarded maintenance and customer support. Several of the respondents felt that the solution may be too costly in terms of the manpower required to answer the mail and return the authorization codes in a timely manner. John Calhoun questioned what would happen if the developer went out of business with people still buying the software off store shelves. Would the user ever get an authorization code for the software they had purchased? Wayne Baker

and the anonymous respondent also questioned what would happen if the authorization code were lost in the mail.

Second, most of the interviewees stated that they felt most people would not like waiting to receive the authorization code necessary to fully use the software they had purchased. Glenda Keyworth stated that for games this may be all right but not for "critical" software, such as word processors, accounting, or software development software. Wayne Baker concurred that most users would find it frustrating having to wait to use their software. He noted that most of the games produced by Origin are so large that they are distributed in compressed format on several floppy disks and therefore cannot be played without being first installed on a hard disk drive. If their software were protected using the OIDS-type of copy protection, then it could not be used until the authorization code arrived.

Finally, the anonymous respondent questioned the fundamental reasoning behind the protection scheme, since once the software is registered it is totally unprotected.

"The only effect of OIDS-style protection is to provide the publisher with a clearer legal recourse in the event of piracy. I don't think this will have much effect on private users for two reasons. First, who's going to ever find out? I think the majority of software pirates are 'casual' pirates who copy from friends. These friends freely offer the software to be copied and don't expect the publisher to ever find out. Second, what company is going to go to the expense of suing anyone for copyright infringement unless they can get enough money out of it to make it worthwhile. And the private individual is rarely worth it. I know if a company sued me and won, I'd declare bankruptcy."

Conclusion

Based on the information provided by the interview respondents, this researcher has decided to retain the proposed solution with one alteration--the OIDS-type copy protection is no longer advocated. There are two reasons behind this decision.

First, the proposed use of OIDS-type copy protection would place businesses and academic institutions in a precarious situation. Recall that this type of copy protection requires proper registration in order "unlock" the software. If someone were to illegally copy the program, the software developer has the necessary information to seek legal action against the copyright infringer. If someone copies software registered to a business or university, is the person who obtained the copy guilty of copyright infringement or is the organization the software is registered to? To protect themselves from this, many businesses and universities have formal policy regarding employees or students illegally copying software. Usually it results in job dismissal or suspension. Many institutions using software do not have such a policy.

Second, since the OIDS-type of copy protection does not allow the software to be installed on a hard disk drive until it is registered, this limitation may be too restricting. Today, most software is so complex and large that it is usually distributed on several floppy disks and cannot be run without first being installed on a hard disk drive. This may not be a problem for some software but is a severe problem for work-oriented software such as word processors, graphics programs, spreadsheets and software development software, which require a large amount of storage space. In addition, many games with their graphical images cannot run without first being installed on a hard disk drive. The latest game produced by Origin Systems requires 21 megabytes of disk space, which is approximately equivalent to storing a 4,375 page document. Currently, a floppy disk employing the latest technology can only hold approximately 1.44 megabytes of information, equivalent to a 300 page document.

With the above constraints in mind, this researcher continues to propose that software be sold with one of the passive forms of copy protection described in the tentative conclusion or with no copy protection at all. As stated above, OIDS-type copy protection is

no longer advocated. In lieu of this, the software developer/publisher should seek other methods to attract people to legitimately purchase their software. This can be accomplished by lowering the price of the software in some cases; providing cheaper site licenses to businesses, allowing them to purchase fewer individual copies of software; and by providing value added incentives such as free technical support, free upgrades, or free tie-in products or other items which cannot be obtained by the user without legitimately purchasing the software. Finally, all software developers should provide a money back guarantee for the software they sell. This would reduce the occurrences of those users "trying" software for an extended period before they decide to buy it. In reaching the above decision, this researcher still believes that a compromise solution should be reached among the parties involved. As previously stated, fairness is valued above all other competing values. It is believed that the solution advocated protects the interests of both parties involved to the greatest extent possible.

Rodney Jacks
Professors Winters and Zacchaeus
RCM 60.02
12/4/91

Appendix A

The following message was posted to the Internet, on Sept. 5, 1991, to solicit responses from members of all sides to the question, "Should Personal Computer Software be Copy Protected?"

I have decided to do a research paper for school (Research and Critical Missions) concerning software copy protection. The main question posed in the research will be "Should Computer Software Be Copy Protected?" I am seeking rational opinions on the subject from software developers and users (including those persons who do not purchase, for whatever reasons, all the software they use). I am looking for any position on the subject and the reasons you choose this position.

If you wish to contribute, please send me E-Mail at the address below.

Thanks for your time,

Appendix B

Interview Questions

1. Do you think personal computer software should be copy protected? Why or why not?
2. Do you think copy protection solves its intended purpose (to reduce the amount of software pirating)?
3. What other measures do you think should be taken to reduce the number of occurrences of software pirating?
4. Do you think the many "passive" protection schemes (authorization screens, code lookup, authorization codes) offer sufficient protection to the software publisher?
5. What is your opinion regarding the new OIDS-style copy protection? Do you think it will work in the long term? What problems do you foresee from it?

Falls unzustellbar bitte zurück an:

*Verein der Informatikstudierenden
IFW B29
ETH-Zentrum*

CH-8092 Zürich

Inhalt

<i>Adressen</i>	<i>S. 2</i>
<i>Tschau Zame</i>	<i>S. 3</i>
<i>Chris Flu's Kochecke</i>	<i>S. 4</i>
<i>MV WS93</i>	<i>S. 9</i>
<i>Fachgruppe Oberon</i>	<i>S. 10</i>
<i>Tücken in C++</i>	<i>S. 11</i>
<i>Spielzeuge, 3</i>	<i>S. 14</i>
<i>Praktikumsbericht</i>	
<i>Landis&Gyr</i>	<i>S. 15</i>
<i>Copy Protection Part III</i>	<i>S. 17</i>