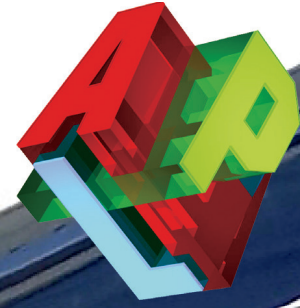


APL - Journal

A Programming Language



1-2/2019

Jacob Brickman

Behavior of APL primitives in a system with depth-1 scalars (APL+1)

Dieter Kilsch

Quaternionen und Bilderkennung

Martin Barghoorn

Weil das Gute nur kurze Zeit das Neue ist

IM BLICKPUNKT

Stefan Ebelt

Wie man alte Computer wieder flott macht

Liebe Mitglieder von APL Germany,
liebe APL-Freunde,

die aktuelle Ausgabe des APL-Journals erscheint in einer für uns alle besonderen Zeit. Das hat auch den späten Zeitpunkt des Fertigstellens verursacht. Ich wünsche Ihnen viele Freude bei der Lektüre der Artikel und vor allem, dass Sie gesund und guter Dinge diese Krise ertragen und überwinden.

Die ersten drei Beiträge basieren auf Vorträgen der Autoren auf den Tagungen von APL Germany und der GSE Working Group APL im vergangenen Jahr. Jacob Brickman setzt sich mit der Tiefe eines Skalars auseinander und zeigt die Vorzüge bei Tiefe 1. In meinem Artikel führe ich in die Theorie der Quaternionen ein und stelle eine Anwendung in der Bilderkennung vor. Die Beiträge von Martin Barghoorn und Stefan Ebelt erklären, wie man Computer länger am Leben erhält.

Die beiden Tagungen fanden im Frühjahr bei der Allianz AG in Stuttgart und im Herbst an der Technischen Hochschule in Bingen am Rhein statt. Auf der Mitgliederversammlung in Stuttgart wurde unsere Satzung endgültig verabschiedet. Sie ist inzwischen vom Amtsgericht Düsseldorf genehmigt und damit rechtskräftig. Veröffentlicht ist sie auf der Internetseite des Vereins.

Der in Stuttgart beschlossene Übergang zur elektronischen Veröffentlichung des Journals wirkt sich positiv auf die finanzielle Lage des Vereins aus, wir haben das vergangene Jahr trotz eines leichten Rückgangs der Mitgliederzahlen mit einem Überschuss beendet.

Die diesjährige Mitgliederversammlung sollte auf der Tagung, die für den 4./5. Mai in Berlin geplant war, stattfinden. Sie musste abgesagt werden. Die notwendigen Beschlüsse werden elektronisch gefasst.

Wir hoffen, dass die Corona-Krise sich so weit stabilisiert, dass im Herbst eine Tagung stattfinden kann. Martin Barghoorn würde sie wieder in Berlin organisieren.

Interessant zu beobachten ist in dieser Zeit, wie intensiv IT in unser Leben eingedrungen ist. Ohne die Möglichkeiten von home office, E-Mail, elektronischen Skripten, Video-Konferenzen, ... wären die Auswirkungen der Schließungen von Betrieben, Schulen, Hochschulen und der Kontaktsperre viel drastischer zu spüren!

Ich wünsche Ihnen ein gesundes Überwinden dieser Krise!
Dieter Kilsch

INHALT

Editorial	1
Behavior of APL primitives in a system with depth-1 scalars (APL+1)	3
Quaternionen und Bilderkennung	14
Weil das Gute nur kurze Zeit das Neue ist	23
Wie man alte Computer wieder flott macht	31
Nachrichten	46
Impressum	48

Behavior of APL primitives in a system with depth-1 scalars (APL+1)

Examples for the Minnowbrook 2017 talk “The Mathematical Foundations of Arrays”

Jacob Brickman

In the talk, an array of rank r was defined as the pair $((sh), I)$, where (sh) is the shape tuple (n_1, \dots, n_r) , and I is the index function, defined on the index set $S_{n_1} \times \dots \times S_{n_r}$, where $S_k = \{1, \dots, k\}$. For scalars, $r = 0$, so the index set becomes the singleton set $\{E_\emptyset\}$, where $E_\emptyset : \emptyset \rightarrow \emptyset$ is the empty function to the empty set. The definition agrees with APL, except in some scalar cases. In all dialects, for any X , $\subset X \leftrightarrow (X \ X) [1]$, and the definition confirms this. Some of the consequences of the definition are the existence of depth-1 scalars, indexing a simple vector with a scalar ($v[1]$) returns a depth-1 scalar, enclosing a simple scalar returns a depth-1 scalar ($2 \neq \subset 2$), reduction of a simple vector returns a depth-1 scalar, non-collapsing scalar towers and consistent depth behavior. The set member function (\in) is misleading when used with arrays because arrays are not sets – they can have repeated values and order is significant, none of which hold for sets. The definition also implies that $2 \in 2$ should return 0, not 1, and that $(\subset 2 \ 3) \in \subset 2 \ 3$ should return 1, not 0. A different symbol should be used, and the talk introduced Ξ (square epsilon), where, for any X , $X \Xi X$ is false and $X \Xi \subset X$ is true.

APL+1 is different from Sharp APL, where $(2 \ 2) [1] \neq \subset 2$. The box function \subset is unrelated to the array definition. It produces scalars (boxes), which are equivalent to single-member C structs, so \subset creates new datatypes, not new arrays. From the point of view of array definition, boxes are user-defined depth-0 scalars, so Sharp APL has a flat array system extended laterally with boxes. It is possible to define \subset in Sharp APL in the same way, and in that system we would still have $\subset \subset 2 \leftrightarrow \subset 2$, so Sharp APL has the same behavior as APL2 on an even larger class of simple scalars, because of the common origin of the dialects. If in Sharp APL we had $(2 \ 2) [1] = \subset 2$, then \subset would be the same as \subset in APL+1. Pervasive functions work the same way in APL+1, because pervasiveness is a property of the function, not the data.

TAPL = Traditional APL (without depth-1 scalars); both APLs = TAPL and APL+1

Primitives affected (APL2 definition): $, \subset \supset \uparrow \underline{\in} \in / \ \backslash + [] \sim \leftarrow \sqcap \rho \phi \boxtimes \perp \cdot$

```

      ]display v←2 3      ⍝ both APLs
.→---.
|2 3|
'|~--'|

      ]display v[1]      ⍝ depth-0 scalar; TAPL
2

      ]display v[1]      ⍝ depth-1 scalar; APL+1
.~.
|2|
'|~'|

      ]display v[,1]     ⍝ both APLs
.→.
|2|
'|~'|

      ]display ((2 3)(4 5))[1]  ⍝ both APLs
.-----
|.→---.|
||2 3||
|'|~--'|
|'ε-----'|
    
```

```

    ldisplay 2      A depth-0 scalar; both APLs
2
    ldisplay c2    A depth-1 scalar; same as v[1]; APL+1
.~.
|2|
|~|
    ldisplay c<2  A depth-2 scalar tower; APL+1
.---.
|.~.|
||2||
|'~'|
|ε--|
    ldisplay c,2  A both APLs
.---.
|.→.|
||2||
|'~'|
|ε--|
    ldisplay v[c1] A depth-1 scalar; same as v[1]; APL+1
.~.
|2|
|~|
    ldisplay ↑c2  A get depth-0 scalar from depth-1 scalar; APL+1
2
    ldisplay ↑c<2 A get depth-1 scalar from depth-2 scalar; APL+1
.~.
|2|
|~|
    ldisplay 2↑c2 A overtake depth-1 scalar; APL+1
.→---.
|2 0|
|~---|
    ldisplay 2↑c<2 A overtake depth-2 scalar tower; APL+1
.→-----.
|.~.~.~.|
||2||0||
|'~'|~'|
|ε-----|
    ldisplay ε2   A both APLs
.→.
|2|
|~|
    ldisplay ε<2  A APL+1
.→.
|2|
|~|
    ldisplay ε<<2 A APL+1
.→.
|2|
|~|
    ldisplay (c10)⋄2 A both APLs
2
    ldisplay (c10)⋄c2 A get depth-0 scalar from depth-1 scalar; APL+1
2

```

```

        |display 2+c2      A APL+1
.-.
|4|
'|~'|

        |display 2+c2      A APL+1
.-.-.
|.-.|.
||4||
'|~'|
'|ε--'|

        |display (c2)+c2      A APL+1
.-.-.
|.-.|.
||4||
'|~'|
'|ε--'|

        |display (cccc2)+c2  A APL+1
.-.-.-.-.
|.-----|.
||.---.|| | | | |
|||.---.||||
||||4||||
||||'~'||||
||'|ε--'|||
||'|ε----'|
'|ε-----'|

        |display (ccc,2)+c,2  A both APLs
.-.-.-.-.
|.→----|.
||.---.|| | | | |
|||.→.||||
||||4||||
||||'~'||||
||'|ε--'|||
||'|ε----'|
'|ε-----'|

        |display (ccc,2)+cc,2  A both APLs
.-.-.-.-.
|.→----|.
||.→--.|| | | | |
|||.→.||||
||||4||||
||||'~'||||
||'|ε--'|||
||'|ε----'|
'|ε-----'|

        |display 2,3          A both APLs
.→--.
|2 3|
'|~--'|

        |display (c2),c3      A joining depth-1 scalars; APL+1
.→--.
|2 3|
'|~--'|

```

```

        ]display (c,2),c,3      A both APLs
.→-----
|.→...→.|
||2||3||
|'~'|~'|
|'ε-----'

        ]display (,2)(,3)      A both APLs
.→-----
|.→...→.|
||2||3||
|'~'|~'|
|'ε-----'

        ]display (c2)(c3)      A APL+1
.→-----
|.→...→.|
||2||3||
|'~'|~'|
|'ε-----'

        ]display c`2 3          A APL+1, same as c[ 10 ]
.→-----
|.→...→.|
||2||3||
|'~'|~'|
|'ε-----'

        ]display c`2 3          A TAPL, same as c[ 10 ]
.→--
|2 3|
|'~--'

        ]display +/c2           A APL+1
.-.
|2|
|'~'

        ]display +/v            A APL+1
.-.
|5|
|'~'

        ]display +/v            A TAPL
5

        ]display +/(1 2)(3 4)(5 6)(7 8)  A both APLs
.-----
|.→-----|
||16 20||
|'~-----'
|'ε-----'

        ]display +\ (1 2)(3 4)(5 6)(7 8)  A both APLs
.-----
|.→-----|
||1 2||4 6||9 12||16 20||
|'~-----'
|'ε-----'

```



```

        ldisplay +\,"2 3 4           A both APLs
.→-----
|.→...→...→.|
||2||5||9||
|'~'|~'|~'|
|'ε-----'

        ldisplay +\c","2 3 4       A both APLs
.→-----
|.---...---...---.|
||.→.||.→.||.→.|| | | | |
|||2|||5|||9|||
||'~'|~'|~'|~'|
|'ε--'ε--'ε--'
|'ε-----'

        ldisplay +\c"2 3 4         A APL+1
.→-----
|.---...---...---.|
||2||5||9||
|'~'|~'|~'|
|'ε-----'

        ldisplay 1/2               A both APLs
.→.
|2|
|'~'|

        ldisplay 1/c2              A APL+1
.→.
|2|
|'~'|

        ldisplay 0/2               A both APLs
.θ.
|0|
|'~'|

        ldisplay 0/c2              A APL+1
.θ.
|0|
|'~'|

        ldisplay 0/c,2            A both APLs
.θ--.
|.→.|
||0||
|'~'|
|'ε--'

        ldisplay 0/cc2            A APL+1
.θ--.
|.---|
||0||
|'~'|
|'ε--'

        ldisplay (1 2)(3 4)~3 4   A both APLs
.→-----
|.→---...→---.|
||1 2||3 4||
|'~---'~---'|
|'ε-----'

```

```

      ldisplay (1 2)(3 4)~c3 4      A both APLs
.→----.
|.→--.|
||1 2||
|'~--'|
|'ε----'|
      ldisplay (c3 4)~c3 4      A both APLs
.θ----.
|.→--.|
||0 0||
|'~--'|
|'ε----'|
      ldisplay (c,3)~3          A both APLs
.→--.
|.→. |
||3||
|'~'|
|'ε--'|
      ldisplay (c,3)~,3        A both APLs
.→--.
|.→. |
||3||
|'~'|
|'ε--'|
      ldisplay (,3)~3          A both APLs
.θ.
|0|
|~'|
      ldisplay (c3)~3          A APL+1
.θ.
|0|
|~'|
      ldisplay 2∈2              A should be 0
1
      ldisplay (c2 3)∈c2 3      A should be 1
0
      ldisplay 2∈2              A both APLs
1
      ldisplay 2∈,2            A both APLs
.→.
|1|
|~'|
      ldisplay (,2)∈2          A both APLs
0
      ldisplay (,2)∈c2        A APL+1
0
      ldisplay a←c2,2          A both APLs, depth 3
.-----
|.----.|
|.→. |
||2||
|'~'|
|'ε--'|
|'ε----'|

```

```

      |display (⍳0)(⍳0)⊃a      ⌘ get innermost vector; both APLs
.→.
|2|
|~|

      ((⍳0)(⍳0)⊃a)←2        ⌘ replace inner vector with simple scalar; both APLs
      |display a              ⌘ TAPL, collapsing tower
2
      |display a              ⌘ APL+1, depth-2 tower
.---.
|.→.|
||2||
|'~'|
|ε--|

      |display (⍳0)⍳2        ⌘ both APLs
2
      |display (⍳0)⍳c,2     ⌘ both APLs
.---.
|.→.|
||2||
|'~'|
|ε--|

      |display 1⍳2 3        ⌘ TAPL, depth-0 scalar, same as v[1]
2
      |display 1⍳2 3        ⌘ APL+1, depth-1 scalar, same as v[1]
.→.
|2|
|~|

      |display (c1)⍳2 3     ⌘ APL+1, depth-1 scalar, same as v[1]
.→.
|2|
|~|

      |display (,1)⍳2 3     ⌘ TAPL, depth-0 scalar, same as v[1]
2
      |display (,1)⍳2 3     ⌘ APL+1, depth-1 scalar, same as v[1]
.→.
|2|
|~|

      |display (c,1)⍳2 3    ⌘ both APLs, same as v[,1]
.→.
|2|
|~|

      |display (c1 2)⍳2 3   ⌘ both APLs, same as v[1 2]
.→---.
|2 3|
|~---|

      |display (⍳0)⍳2 3     ⌘ TAPL, depth-0 scalar
2
      |display (⍳0)⍳2 3     ⌘ APL+1, depth-1 scalar, depth preserving
.→.
|2|
|~|

```

```

        ]display (⊔0)ρ(2 3)(4 5)  ⌘ both APLs, depth preserving
.-----
|.→--.|
||2 3||
|'~--'|
|'ε-----'

        1⊕2 3 4                    ⌘ both APLs
3 4 2
        (,1)⊕2 3 4                 ⌘ both APLs
3 4 2
        (⊂1)⊕2 3 4                 ⌘ APL+1
3 4 2
        1⊗2 3 4                    ⌘ both APLs
2 3 4
        (,1)⊗2 3 4                 ⌘ both APLs
2 3 4
        (⊂1)⊗2 3 4                 ⌘ APL+1
2 3 4
        ]display ,¨2 3 4           ⌘ both APLs
.→-----
|.→..→..→.|
||2||3||4||
|'~'|~'|~'|
|'ε-----'

        ]display ⊃,¨2 3 4         ⌘ both APLs
.→.
↓2|
|3|
|4|
|~|

        ]display ⋈,¨2 3 4        ⌘ both APLs
.→-----
|.---.---.---.|
||.→.→.→.→.→.| | | | | | | | | |
|||2|||3|||4|||
||'|~'|~'|~'|~'|~'|
|'ε--'ε--'ε--'|
|'ε-----'

        ]display ⊃⋈,¨2 3 4      ⌘ both APLs
.→-----
|.→..→..→.|
||2||3||4||
|'~'|~'|~'|
|'ε-----'

        ]display ⋈⋈,¨2 3 4      ⌘ both APLs
.→-----
|.---.---.---.|
||.→.→.→.→.→.| | | | | | | | | |
|||2|||3|||4|||
||'|~'|~'|~'|~'|~'|
|'ε--'ε--'ε--'|
|'ε-----'ε-----'ε-----'
|'ε-----'

```

```

]display ⍳c''c'',2 3 4      ⌘ both APLs
┌───────────┐
│.-----┐
│|.-----┐
│||.→.|||.→.|||.→.||
│||2|||3|||4||
│|'|~'|'|~'|'|~'|
│|'ε--'|'ε--'|'ε--'|
│'ε-----┘
└───────────┘

]display c''2 3 4          ⌘ APL+1
┌──────────┐
│.-----┐
│|.-----┐
│||2||3||4||
│|'|~'|'|~'|'|~'|
│|'ε-----┘
└──────────┘

]display ⍳c''2 3 4        ⌘ both APLs
┌──────────┐
│.-----┐
│|2 3 4|
│|~-----┘
└──────────┘

L f.g R ↔ f/'' (c[ρρL]L)∘.g c[1]R      ⌘ APL2 Lang. Ref. Manual
Z[I;J] ↔ c f/L[I;] g R[;J]              ⌘ APL2 Lang. Ref. Manual

]display 2+.×,20 20      ⌘ both APLs (APL2, APL+WIN)
┌──────────┐
│.-----┐
│|.-----┐
│||.→-||
│||60||
│|'|~-'|
│|'ε-----┘
└──────────┘

]display 2+.×,20 20      ⌘ both APLs (Dyalog)
┌──────────┐
│.-----┐
│|.-----┐
│||60||
│|'|~-'|
│|'ε-----┘
└──────────┘

]display 2+.×c'',20 20   ⌘ both APLs (APL2, APL+WIN)
┌──────────┐
│.-----┐
│|.-----┐
│||.→-||
│||60||
│|'|~-'|
│|'ε-----┘
└──────────┘

]display 2+.×c'',20 20   ⌘ both APLs (Dyalog)
┌──────────┐
│.-----┐
│|.-----┐
│||60||
│|'|~-'|
│|'ε-----┘
└──────────┘

```

```

      ldisplay 2+.×c''10 20      ⍝ APL+1 (APL2, APL+WIN)
┌-----┐
│.----.│
│|.---.│
│||60||
│|'~-'|
│|'ε---'|
│|'ε-----'|
└-----┘
      ldisplay 2+.×c''10 20      ⍝ APL+1 (Dyalog)
┌-----┐
│.----.│
│||60||
│|'~-'|
│|'ε---'|
└-----┘
      ldisplay 2+.×c''10 20      ⍝ TAPL
60
      ldisplay 1 2+.×10 20      ⍝ TAPL
50
      ldisplay 1 2+.×10 20      ⍝ APL+1 (APL2, APL+WIN), depth-2 scalar tower
┌-----┐
│.----.│
│|.---.│
│||50||
│|'~-'|
│|'ε---'|
└-----┘
      ldisplay 1 2+.×10 20      ⍝ APL+1 (Dyalog), depth-1 scalar
┌-----┐
│.----.│
│||50||
│|'~-'|
└-----┘
      L ⊥ R ↔ ((ρL)↑Φ1,×\Φ1↓[ρρL]L) +.× R      ⍝ APL2 Lang. Ref. Manual
      ldisplay 10⊥2 3 4          ⍝ TAPL
234
      ldisplay 10⊥2 3 4          ⍝ APL+1 (APL2, APL+WIN), depth-2 scalar tower
┌-----┐
│.----.│
│|.----.│
│||234||
│|'~---'|
│|'ε-----'|
└-----┘
      ldisplay 10⊥2 3 4          ⍝ APL+1 (Dyalog), depth-1 scalar
┌-----┐
│.----.│
│||234||
│|'~---'|
└-----┘
      ldisplay (,10)⊥2 3 4      ⍝ TAPL
234
      ldisplay (,10)⊥2 3 4      ⍝ APL+1 (APL2, APL+WIN), depth-2 scalar tower
┌-----┐
│.----.│
│|.----.│
│||234||
│|'~---'|
│|'ε-----'|
└-----┘
      ldisplay (,10)⊥2 3 4      ⍝ APL+1 (Dyalog), depth-1 scalar
┌-----┐
│.----.│
│||234||
│|'~---'|
└-----┘

```

<pre>]display (⊢10)⊥2 3 4 ┌-----┐ │.----.│ │ .----.│ │ 234 │ '~--' │ ε----- └-----┘ </pre>	<p>⊞ APL+1 (APL2, APL+WIN), depth-2 scalar tower</p>
<pre>]display (⊢10)⊥2 3 4 ┌-----┐ │234│ │~--│ └-----┘ </pre>	<p>⊞ APL+1 (Dyalog), depth-1 scalar</p>

Innsbrucker Quantencomputer in der Cloud

Innsbruck 16.09.2019. Die Quantencomputer von Alpine Quantum Technologies (AQT) und Universität Innsbruck unterstützen jetzt das Google-Framework Cirq für die Entwicklung von Quantenalgorithmen. Damit können Forscher und Industriepartner ihre Quantensoftware auf den Innsbrucker Quantencomputern ausführen.

Quantencomputer versprechen, Probleme zu lösen, die für heutige Supercomputer unerreichbar sind. Das Programmieren von Quantencomputern unterscheidet sich grundlegend von den heute verwendeten Methoden und erfordert daher neue Programmiersprachen. Eine Zusammenarbeit von Alpine Quantum Technologies (AQT) und der Universität Innsbruck ermöglicht den direkten Zugriff auf den Ionenfallen-Quantencomputer in Innsbruck über Cirq. Cirq ist ein von Google entwickeltes Framework, das sich auf die Entwicklung und Implementierung von Quantenalgorithmen konzentriert. Mit Cirq können Quantenalgorithmen für die verschiedenen Hardwarearchitekturen, supraleitende Elektronik und gespeicherte Ionen untersucht werden.

Mehrere Forschungseinrichtungen und Unternehmen arbeiten an der Realisierung von Quantencomputern. Es gibt mehrere physikalische Plattformen, auf denen sich ein zukünftiger Quantencomputer aufbauen lässt. AQT verfolgt einen Ansatz basierend auf gefangenen Ionen und Google favorisiert supraleitende Elektronik. Jeder Ansatz weist unterschiedliche Funktionen und Einschränkungen auf, die sich in der Regel in verschiedenen, vom Gerät abhängigen Programmiersprachen widerspiegeln. Diese Mischung von Programmiersprachen erschwert es Softwareentwicklern und Programmierern, diese Quantencomputer-Prototypen zu verwenden und die Fähigkeiten verschiedener Architekturen zu erkunden.

Cirq, ein von Google entwickeltes Python-Framework, dient zum Erstellen, Bearbeiten und Aufrufen von NISQ-Schaltkreisen (Noisy Intermediate Scale Quantum). Dr. Markus Hoffmann von Google München erklärt: „Es ist toll zu sehen, wie Cirq im Geiste der Open Source-Lizenz Apache 2.0 eingeführt und für weitere Hardwareplattformen zugänglich gemacht wurde. Die Bibliothek unterstützt mehrere Hardware-Architekturen, die auf supraleitenden elektronischen und atomaren Systemen basieren. Jetzt können Forscher und Industriepartner ihre Quantensoftware problemlos auf den Innsbrucker Quantencomputern ausführen. Schüler und Studierende sind in der Lage, ihr Wissen zum Ausführen von Quantenalgorithmen auf tatsächlicher Hardware aufzubauen.“ Diese Bemühungen kommen neben der Forschung auch dem Quantencomputer-Startup AQT in Österreich zugute. CEO Dr. Thomas Monz freut sich, „eine so einfache und effektive Schnittstelle zwischen internationalen Quanten-Software-Entwicklern und unserer in Innsbruck ansässigen Quanten-Computer-Infrastruktur bereitzustellen, um die Realisierung einer ganzen Sammlung von Quantenanwendungen für Forschungs- und Industriepartner zu ermöglichen.“ Dr. Philipp Schindler an der Universität Innsbruck ist überzeugt, dass die Schnittstelle neue Kooperationen mit Forschungspartnern auf der ganzen Welt ermöglichen wird.

Wissenschaftliche Ansprechpartner:

Dr. Thomas Monz, Alpine Quantum Technologies (AQT),
 Telefon: +43 512 507 52452, E-Mail: thomas.monz@aqt.eu,
 Web: <https://www.aqt.eu/>

Dr. Philipp Schindler, Institut für Experimentalphysik, Uni-
 versität Innsbruck, Telefon: +43 512 507-52466, E-Mail:
philipp.schindler@uibk.ac.at, Web: <https://www.uibk.ac.at>

Quaternionen und Bildererkennung

Dieter Kilsch
Technische Hochschule Bingen

Zusammenfassung

In diesem Artikel werden Quaternionen als Schiefkörper in der komplexen Ebene \mathbb{C}^2 und im reellen vierdimensionalen Raum \mathbb{R}^4 eingeführt. Spezielle Quaternionen operieren auf dem von den drei nicht-reellen Achsen aufgespannten Unterraum als Drehungen. Da die Hintereinanderausführung von dreidimensionalen Drehmatrizen, also die Matrizenmultiplikation, numerisch aufwändiger ist als die Multiplikation von Quaternionen, haben diese in der Berechnung von Bilddarstellung und Bildererkennung große Vorteile. Hinzu kommt die Möglichkeit in der Bildererkennung die Berechnung der Drehung auf ein lineares Gleichungssystem zurückzuführen.

Die von mir gewählte Einführung der Quaternionen als komplexe 2×2 -Matrizen kann in Analogie zur Einführung der komplexen Zahlen als reelle 2×2 -Matrizen gesehen werden. Diese wiederhole ich daher und verweise zum weiteren Lesen auf Bücher zur Linearen Algebra und Körpertheorie.

1 Definition der komplexen Zahlen als reelle 2-2 Matrizen

Die Gleichung $x^2 + 1 = 0$ hat in der Menge der reellen Zahlen keine Lösung. Eine Lösung kann man nur in einem Zahlbereich jenseits der reellen Zahlen erhalten, erfolgreich ist man in der Menge der Punkte der reellen Ebene versehen mit der Vektoraddition und einer geeigneten Multiplikation.

Eine einfache Möglichkeit für die Definition komplexer Zahlen \mathbb{C} mit den benötigten Verknüpfungen ist die Einbettung in die Menge der reellen 2×2 -Matrizen, die \mathbb{C} als Untermenge von $GL(2, \mathbb{R})$ ¹ betrachtet:

Definition 1.1 (Komplexe Zahlen) *Mit den Matrizen*

$$\text{Id} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, i = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \tag{1}$$

setzen wir

$$\mathbb{C} = \{a \text{Id} + bi \mid a, b \in \mathbb{R}\} = \left\{ \begin{pmatrix} a & -b \\ b & a \end{pmatrix} \mid a, b \in \mathbb{R} \right\}. \tag{2}$$

$a \text{Id}$ wirkt als Streckung und bi als Streckung und Drehung um 90° . \mathbb{C} besteht genau aus den Drehstreckungen der Ebene \mathbb{R}^2 , die Wurzel der Determinante $\sqrt{a^2 + b^2}$ ist der Streckfaktor.

\mathbb{C} ist abgeschlossen unter der Addition, enthält die Nullmatrix als Nullelement und die additiv Inversen ihrer Matrizen. Wegen

$$\begin{aligned} \begin{pmatrix} a & -b \\ b & a \end{pmatrix} \cdot \begin{pmatrix} c & -d \\ d & c \end{pmatrix} &= \begin{pmatrix} ac - bd & ad + bc \\ -ad - bc & ac - bd \end{pmatrix}, \\ \begin{pmatrix} a & -b \\ b & a \end{pmatrix}^{-1} &= \frac{1}{\sqrt{a^2 + b^2}} \begin{pmatrix} a & b \\ -b & a \end{pmatrix} \quad \text{und} \\ \begin{pmatrix} a & -d \\ d & a \end{pmatrix} \cdot \begin{pmatrix} a & -b \\ b & a \end{pmatrix} &= \begin{pmatrix} ca - db & cb + da \\ -cb - da & ca - db \end{pmatrix}, \end{aligned}$$

ist \mathbb{C} auch abgeschlossen unter der Multiplikation, enthält die Inversen der Matrizen (außer der Nullmatrix). Aus einem Vergleich der ersten und dritten Zeile folgt die Kommutativität der Multiplikation. Damit ist $(\mathbb{C}, +, \cdot)$ ein Körper, der Körper der komplexen Zahlen mit der Dimension 2 über \mathbb{R} .

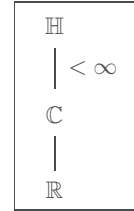
¹allgemeine lineare Gruppe über \mathbb{R} , Menge der invertierbaren reellen 2×2 -Matrizen

Durch die Identifikation einer reellen Zahl $r \in \mathbb{R}$ mit der Skalarmatrix $r \cdot \text{Id}$ werden die reellen Zahlen in die komplexen Zahlen eingebettet und man erhält eine einfachere Darstellung

$$\mathbb{C} = \{a + bi \mid a, b \in \mathbb{R}\} .$$

Bis auf Isomorphie ist \mathbb{C} der einzige Körper endlicher Dimension über \mathbb{R} .

Da \mathbb{C} algebraisch abgeschlossen² ist, kann es keine Körper endlicher Dimension über \mathbb{C} geben und über \mathbb{R} gibt es nur \mathbb{C} . Nach einem Satz von Gelfand-Mazur (s. Satz 2.11) sind endlich-dimensionale Schiefkörper³ immer isomorph zu \mathbb{R} , \mathbb{C} oder dem hier zu definierenden Schiefkörper der Quaternionen \mathbb{H} , die im englischen Sprachraum zu Ehren von William Hamilton⁴ Hamiltonians genannt werden.



2 Definition der Quaternionen

2.1 Quaternionen als komplexe 2-2 Matrizen

Die Menge der Quaternionen \mathbb{H} wird als Untermenge von $GL(2, \mathbb{C})$ festgelegt:

Definition 2.1 (Quaternionen) *Mit den Matrizen*

$$h_0 = \text{Id} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, h_1 = \begin{pmatrix} i & 0 \\ 0 & -i \end{pmatrix}, h_2 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, h_3 = \begin{pmatrix} 0 & i \\ i & 0 \end{pmatrix} \tag{3}$$

setzen wir

$$\begin{aligned} \mathbb{H} &= \{a h_0 + b h_1 + c h_2 + d h_3 \mid a, b, c, d \in \mathbb{R}\} \\ &= \left\{ \begin{pmatrix} a + bi & c + di \\ -c + di & a - bi \end{pmatrix} \mid a, b, c, d \in \mathbb{R} \right\} = \left\{ \begin{pmatrix} v & w \\ -\bar{w} & \bar{v} \end{pmatrix} \mid v, w \in \mathbb{C} \right\} . \end{aligned} \tag{4}$$

Bemerkung 2.2

- (a) \mathbb{H} ist abgeschlossen unter der Matrizenaddition und der -multiplikation und enthält die identische Matrix. \mathbb{H} ist also ein Ring mit Eins.
- (b) $h_1^2 = h_2^2 = h_3^2 = -h_0$.
Damit enthält \mathbb{H} mehrere Kopien der komplexen Zahlen. Ferner hat das Polynom $x^2 + h_0$ mindestens drei Lösungen, so dass \mathbb{H} keine Körper sein kann.
- (c) $h_1 h_2 = h_3, h_2 h_3 = h_1, h_3 h_1 = h_2$ und $h_2 h_1 = -h_3, h_3 h_2 = -h_1, h_1 h_3 = -h_2$.
Die Multiplikation auf \mathbb{H} ist damit nicht kommutativ. Diese Regeln erinnern an die Rechenregeln des Kreuzproduktes in \mathbb{R}^3 .
- (d) Die Abbildung

$$\Phi : \left\{ \begin{array}{l} (\mathbb{R}^4, +) \rightarrow (\mathbb{H}, +) \\ (a, b, c, d) \mapsto \begin{pmatrix} a + bi & c + di \\ -c + di & a - bi \end{pmatrix} \end{array} \right\}$$

ist verträglich mit der Vektoraddition / Matrizenaddition und der Skalarmultiplikation. Folglich ist Φ ein Vektorraumhomomorphismus, der bijektiv und somit ein Vektorraumisomorphismus ist.

Satz 2.3 \mathbb{H} ist ein Schiefkörper mit Zentrum⁵ $\mathbb{R} h_0$.

BEWEIS: Durch Nachrechnen zeigt man:⁶

$$(a) \quad \begin{pmatrix} a + bi & c + di \\ -c + di & a - bi \end{pmatrix}^{-1} = \frac{1}{a^2 + b^2 + c^2 + d^2} \begin{pmatrix} a - bi & -c - di \\ c - di & a + bi \end{pmatrix}$$

⁴Jedes Polynom mit komplexen Koeffizienten hat eine Lösung in \mathbb{C} .

³In einem Schiefkörper ist die Multiplikation nicht kommutativ.

⁴William Rowan Hamilton, ir. Mathematiker und Physiker, 1805 (Dublin) - 1865 (Dunsink bei Dublin)

⁵Das Zentrum enthält alle Elemente, die mit jedem multiplikativ vertauschen

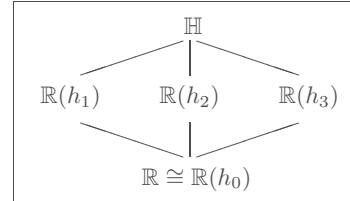
⁶Die einfache Regel für das Invertieren einer zweidimensionalen Matrix lautet: „Vertauschen der Diagonalelemente, Vorzeichen ändern bei den Nichtdiagonalelementen und Division durch die Determinante.“

$$\begin{aligned}
 (b) \quad & \begin{pmatrix} a_1 + b_1 i & c_1 + d_1 i \\ -c_1 + d_1 i & a_1 - b_1 i \end{pmatrix} \cdot \begin{pmatrix} a_2 + b_2 i & c_2 + d_2 i \\ -c_2 + d_2 i & a_2 - b_2 i \end{pmatrix} \\
 &= \begin{pmatrix} a_1 a_2 - b_1 b_2 - c_1 c_2 - d_1 d_2 + (a_1 b_2 + b_1 a_2 + c_1 d_2 - d_1 c_2) i \\ -c_1 a_2 - d_1 b_2 - a_1 c_2 + b_1 d_2 + (-c_1 b_2 + d_1 a_2 + a_1 d_2 + b_1 c_2) i \\ a_1 c_2 - b_1 d_2 + c_1 a_2 + d_1 b_2 + (a_1 d_2 + b_1 c_2 - c_1 b_2 + d_1 a_2) i \\ -c_1 c_2 - d_1 d_2 + a_1 a_2 - b_1 b_2 + (-c_1 d_2 + d_1 c_2 - a_1 b_2 - b_1 a_2) i \end{pmatrix}
 \end{aligned}$$

(c) Das Zentrum kann durch Nachrechnen verifiziert werden. ◇

Bemerkung 2.4

- (a) $\{h_0, h_1, h_2, h_3\}$ bildet eine reelle Basis des Vektorraums \mathbb{H} .
- (b) Mit der Einbettung von \mathbb{R} als $\mathbb{R}(h_0)$ enthält \mathbb{H} mit $\mathbb{R}(h_i)$, ($i, \dots, 3$) drei Kopien der komplexen Zahlen, deren Schnitt $\mathbb{R} = \mathbb{R}(h_0)$ ist.



2.2 Quaternionenmultiplikation auf \mathbb{R}^4

Mit Hilfe des Vektorraumisomorphismus Φ aus 2.2 (d) wird die Multiplikation auf \mathbb{R}^4 übertragen:

Bemerkung 2.5 $(\mathbb{R}^4, +, \cdot)$ ist mit der Vektoraddition und der Multiplikation nach 2.3 (b)

$$\begin{aligned}
 \begin{pmatrix} a_1 \\ b_1 \\ c_1 \\ d_1 \end{pmatrix} \cdot \begin{pmatrix} a_2 \\ b_2 \\ c_2 \\ d_2 \end{pmatrix} &\cong \begin{pmatrix} a_1 + b_1 i & c_1 + d_1 i \\ -c_1 + d_1 i & a_1 - b_1 i \end{pmatrix} \cdot \begin{pmatrix} a_2 + b_2 i & c_2 + d_2 i \\ -c_2 + d_2 i & a_2 - b_2 i \end{pmatrix} \\
 &= \begin{pmatrix} a_1 a_2 - b_1 b_2 - c_1 c_2 - d_1 d_2 + (a_1 b_2 + b_1 a_2 + c_1 d_2 - d_1 c_2) i \\ -c_1 a_2 - d_1 b_2 - a_1 c_2 + b_1 d_2 + (-c_1 b_2 + d_1 a_2 + a_1 d_2 + b_1 c_2) i \\ a_1 c_2 - b_1 d_2 + c_1 a_2 + d_1 b_2 + (a_1 d_2 + b_1 c_2 - c_1 b_2 + d_1 a_2) i \\ -c_1 c_2 - d_1 d_2 + a_1 a_2 - b_1 b_2 + (-c_1 d_2 + d_1 c_2 - a_1 b_2 - b_1 a_2) i \end{pmatrix} \\
 &\cong \begin{pmatrix} a_1 a_2 - b_1 b_2 - c_1 c_2 - d_1 d_2 \\ a_1 b_2 + b_1 a_2 + c_1 d_2 - d_1 c_2 \\ a_1 c_2 - b_1 d_2 + c_1 a_2 + d_1 b_2 \\ a_1 d_2 + b_1 c_2 - c_1 b_2 + d_1 a_2 \end{pmatrix}
 \end{aligned}$$

ein zu $(\mathbb{H}, +, \cdot)$ isomorpher Schiefkörper, der ebenfalls mit $(\mathbb{H}, +, \cdot)$ bezeichnet wird. Das inverse Element wird durch

$$\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}^{-1} = \frac{1}{a^2 + b^2 + c^2 + d^2} \begin{pmatrix} a \\ -b \\ -c \\ -d \end{pmatrix}$$

berechnet.

Definition 2.6 (Konjugation, Norm)

(a) Die Konjugation $*$: $\mathbb{H} \rightarrow \mathbb{H}$ wird definiert durch

$$\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}^* = \begin{pmatrix} a \\ -b \\ -c \\ -d \end{pmatrix} \text{ oder } \begin{pmatrix} a + b i & c + d i \\ -c + d i & a - b i \end{pmatrix}^* = \begin{pmatrix} a - b i & -c - d i \\ c - d i & a + b i \end{pmatrix}.$$

Sie ist ein additiver Automorphismus und ein multiplikativer Antiautomorphismus⁷ auf \mathbb{H} .

(b) Die Norm $N : \mathbb{H} \rightarrow \mathbb{R}_{\geq 0}$ einer Quaternione ist

$$N \left(\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \right) = a^2 + b^2 + c^2 + d^2 = \left| \begin{pmatrix} a + b i & c + d i \\ -c + d i & a - b i \end{pmatrix} \right|.$$

⁷ $(q_1 q_2)^* = q_2^* q_1^*$

Bemerkung 2.7 Für $q, q_1, q_2 \in \mathbb{H}$ gelten

- (a) $N(q) = \det(q)$ und damit
- (b) $N(q_1 \cdot q_2) = N(q_1)N(q_2)$. N ist also ein Homomorphismus von (\mathbb{H}, \cdot) auf $(\mathbb{R}_{\geq 0}, \cdot)$.
- (c) Mit dem üblichen Skalarprodukt auf \mathbb{R}^4 folgt

$$2 \langle q_1, q_2 \rangle h_0 = q_1 q_2^* + q_2 q_1^*$$

durch Auswerten der beiden Seiten mit Hilfe von (4) oder 3.4 (a), siehe auch [1, Satz 9.3.2].

Aufgrund der Definition von \mathbb{H} über komplexen Matrizen gilt offensichtlich der

Satz 2.8 Für $S := N^{-1}\{1\} = \{q \in \mathbb{H} \mid N(q) = 1\}$ gilt $S \cong \text{SU}(2, \mathbb{C})$, der speziellen unitären Matrizen des Grades zwei, also der komplexen 2×2 -Matrizen mit Determinante 1. S ist die Menge aller Einheitsquaternionen.

BEWEIS: Da die Norm und die Determinante denselben Wert liefern, entsprechen die Einheitsquaternionen den Elementen der speziellen linearen Gruppe. Nach (4) sind die Spalten der Matrizen Einheitsvektoren und stehen orthogonal (nach dem Hermiteschen Skalarprodukt) zueinander. Damit sind sie unitär. \diamond

2.3 Besondere Bedeutung der Quaternionen

Die Bedeutung der Quaternionen wird durch ihre Einzigartigkeit als 4-dimensionaler Schiefkörper über \mathbb{R} und die Nähe zu Paulis Spin-Matrizen hervorgehoben. Ich beginne mit den charakterisierenden Sätzen aus der Algebra:

Satz 2.9 (G. Frobenius⁸, 1877) [1, S. 526] Eine endlichdimensionale, assoziative, reelle Algebra mit Einselement ist isomorph zu \mathbb{R} , \mathbb{C} oder \mathbb{H} .

Satz 2.10 (Hurwitz⁹, 1898) [1, S. 526] Eine endlichdimensionale, reelle Algebra mit Einselement ist isomorph zu \mathbb{R} , \mathbb{C} , \mathbb{H} oder \mathbb{O} , der acht-dimensionalen nicht-assoziativen reellen Algebra der Cayley-schen¹⁰ Oktaven.

Satz 2.11 (Gelfand¹¹-Mazur¹²) [1, S. 529] Eine reelle Algebra mit einer Algebrennorm ($\|xy\| \leq \|x\| \cdot \|y\|$), die vollständig und ein Schiefkörper ist, ist isomorph zu \mathbb{R} , \mathbb{C} oder \mathbb{H} .

Bemerkung 2.12 Die Spin-Matrizen nach Pauli¹³ werden in der Beschreibung der Spin-Eigenschaften der Teilchen bei der Bewegung in einem elektromagnetischen Feld benutzt, s. [7]. Sie sind definiert als

$$\sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = -i h_3, \quad \sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} = -i h_2, \quad \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = -i h_1$$

Alle Quadrate der Spin-Matrizen ergeben die Einheitsmatrix.

2.4 Implementierung der Verknüpfungen in APL

Addition und Subtraktion sind zur Addition und Subtraktion von Vektoren im \mathbb{R}^4 identisch und damit in APL vorhanden. Nars2000¹⁴ hat die Multiplikation und Division implementiert. Für andere Dialekte schreibt man die entsprechende Funktionen schnell selbst, Abb. 1 enthält sie für Dyalog APL. Diese Abbildung enthält auch eine Implementierung für die Konjugation und die symmetrische Differenz. Abb. 2 stellt einfache Rechenbeispiele vor.

3 Quaternionen und Drehungen

In diesem Paragrafen wird der Zusammenhang von Drehungen im \mathbb{R}^3 und Einheitsquaternionen hergestellt. Hierzu wird der \mathbb{R}^3 als die Teilmenge der Imaginärteile der Quaternionen betrachtet. Das wichtige Ergebnis wird in Satz 3.5 formuliert.

⁸Ferdinand Georg Frobenius, dt. Mathematiker, 1849 - 1917 (Charlottenburg / Berlin)

⁹Adolf Hurwitz, dt. Mathematiker, 1859 (Hildesheim) - 1919 (Zürich)

¹⁰Arthur Cayley, engl. Mathematiker, 1821 (Rochmond upon Thames, Surrey, UK) - 1895 (Cambridge, UK)

¹¹Israil Moiseevic Gelfand, russ. Mathematiker, 2.9.1913 (Ozny Ukraine) - 5.10.2009 (New Brunswick USA)

¹²Stanislaw Mazur, poln. Mathematiker, 1.1.1905 (Lemberg) - 5.11.1981 (Warschau)

¹³Wolfgang Ernst Pauli, österr. Physiker, 25.4.1900 (Wien) - 15.12.1958 (Zürich)

¹⁴This APL-dialect has been developed and is maintained by Sudley Place Software, see <http://www.nars2000.org>

```

r←a Hmul b
r←a[1]×b
r←r+a[2]×-1 1 -1 1×b[2 1 4 3]
r←r+a[3]×-1 1 1 -1×b[3 4 1 2]
r←r+a[4]×-1 -1 1 1×b

Hinv←((1↑ω), -1↓ω)÷+/ω×ω}
Hdiv←{α Hmul Hinv ω}
Hcon←{(1↑ω), -1↓ω}
HSDi←{(α Hmul ω)-ω Hmul α}
    
```

Abb. 1: APL-Routinen in Dyalog APL

```

      Hinv 0 0 1 1
0 0 -0.5 -0.5
      0 1 0 0 Hmul 0 0 1 0
0 0 0 1
      0 1 0 0 HSDi 0 0 1 0
0 0 0 2
    
```

Abb. 2: Einfache Beispiele

3.1 Real- und Imaginärteil, Drehungen

Definition 3.1 Der Realteil einer Quaternion $a h_0 + b h_1 + c h_2 + d h_3$ ist a , ihr Imaginärteil $\begin{pmatrix} b \\ c \\ d \end{pmatrix}$. Bei der Zerlegung $\mathbb{H} = h_0 \mathbb{R} \oplus V$ bezeichnet $V := h_1 \mathbb{R} \oplus h_2 \mathbb{R} \oplus h_3 \mathbb{R} \cong \mathbb{R}^3$ die Menge der Imaginärteile.

Definition und Satz 3.2 (Polardarstellung der Einheitsquaternionen) Für die Menge der Einheitsquaternionen gilt

$$S = \left\{ \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \hat{\omega} \end{pmatrix} \mid \alpha \in [0, 2\pi) \wedge \hat{\omega} \in \{\vec{v} \in \mathbb{R}^3 \mid \|\vec{v}\| = 1\} \right\}.$$

Diese Form heißt Polardarstellung einer Einheitsquaternione.

Da die Norm und damit ihr Betrag einer Einheitsquaternione 1 ist, gilt dies offensichtlich. ◇

Für die weiteren Berechnungen stelle ich wichtige Formeln zusammen und zitiere zu Beginn den für die Beweise benötigten

Hilfssatz 3.3 (Graßmannscher¹⁵ Entwicklungssatz, [4, 11.9, S. 60]) Für $\vec{a}, \vec{b}, \vec{c} \in \mathbb{E}^3$ gilt

$$\vec{a} \times (\vec{b} \times \vec{c}) = (\vec{a} \cdot \vec{c})\vec{b} - (\vec{a} \cdot \vec{b})\vec{c}.$$

Bemerkung 3.4 (Multiplikation, Inverse, Konjugieren) Für $a, a_i \in \mathbb{R}$ und $\vec{v}, \vec{v}_i \in V (i = 1, 2)$ gelten (vgl. [7])

(a) Für $\vec{v} \in V$ und das reelle Skalarprodukt gilt $\langle \vec{v}, \vec{v} \rangle = N(\vec{v})$.

(b)
$$\begin{pmatrix} a_1 \\ \vec{v}_1 \end{pmatrix} \cdot \begin{pmatrix} a_2 \\ \vec{v}_2 \end{pmatrix} = \begin{pmatrix} a_1 a_2 - \langle \vec{v}_1, \vec{v}_2 \rangle \\ a_1 \vec{v}_2 + a_2 \vec{v}_1 + \vec{v}_1 \times \vec{v}_2 \end{pmatrix}.$$

Die auf V eingeschränkte Multiplikation entspricht dem Kreuzprodukt.

(c)
$$\begin{pmatrix} a \\ \vec{v} \end{pmatrix}^{-1} = \frac{1}{a^2 + \|\vec{v}\|^2} \begin{pmatrix} a \\ -\vec{v} \end{pmatrix}$$

(d) Konjugieren mit der Einheitsquaternion $\cos(\alpha)h_0 + \sin(\alpha)\hat{\omega}$ liefert

$$\begin{aligned} & \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \hat{\omega} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ \vec{v} \end{pmatrix} \cdot \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \hat{\omega} \end{pmatrix}^{-1} \\ &= \begin{pmatrix} 0 \\ (\cos^2(\alpha) - \sin^2(\alpha))\vec{v} + 2 \langle \hat{\omega}, \vec{v} \rangle \hat{\omega} + 2 \cos(\alpha) \hat{\omega} \times \vec{v} \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ \cos(2\alpha)\vec{v} + 2 \sin^2(\alpha) \langle \hat{\omega}, \vec{v} \rangle \hat{\omega} + \sin(2\alpha) \hat{\omega} \times \vec{v} \end{pmatrix} \end{aligned}$$

(e) Zum Konjugieren mit der Einheitsquaternion $\begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \hat{\omega} \end{pmatrix} = \begin{pmatrix} \omega_0 \\ \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}$ gehört die Drehmatrix

$$D_{\omega, \alpha} = \begin{pmatrix} \omega_0^2 + \omega_x^2 - \omega_y^2 - \omega_z^2 & 2(\omega_x \omega_y - 2\omega_0 \omega_z) & 2(\omega_0 \omega_y + \omega_x \omega_z) \\ 2(\omega_0 \omega_z + \omega_x \omega_y) & \omega_0^2 - \omega_x^2 + \omega_y^2 - \omega_z^2 & 2(\omega_y \omega_z - \omega_0 \omega_x) \\ 2(\omega_x \omega_z - \omega_0 \omega_y) & 2(\omega_0 \omega_x + \omega_y \omega_z) & \omega_0^2 - \omega_x^2 - \omega_y^2 + \omega_z^2 \end{pmatrix} \quad \text{auf } V.$$

¹⁵Hermann Günther Graßmann, dt. Mathematiker und Sprachwissenschaftler, 1809 (Stettin) - 1877 (Stettin)

BEWEIS:

(a) dies folgt direkt aus der Definition der Norm.

(b) Folgt durch Nachrechnen aus Bemerkung 2.5.

$$(c) \quad \begin{aligned} \begin{pmatrix} a \\ \vec{v} \end{pmatrix} \cdot \frac{1}{a^2 + \|\vec{v}\|^2} \begin{pmatrix} a \\ -\vec{v} \end{pmatrix} &= \frac{1}{a^2 + \|\vec{v}\|^2} \begin{pmatrix} a \\ \vec{v} \end{pmatrix} \cdot \begin{pmatrix} a \\ -\vec{v} \end{pmatrix} \\ &\stackrel{(b)}{=} \frac{1}{a^2 + \|\vec{v}\|^2} \begin{pmatrix} a^2 + \|\vec{v}\|^2 \\ a\vec{v} - a\vec{v} + \vec{v} \times \vec{v} \end{pmatrix} = \begin{pmatrix} 1 \\ \vec{0} \end{pmatrix} \end{aligned}$$

(d) Mit $\vec{\omega} = \sin(\alpha)\hat{\omega} = \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}$ folgt

$$\begin{aligned} &\begin{pmatrix} \cos(\alpha) \\ \vec{\omega} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ \vec{v} \end{pmatrix} \cdot \begin{pmatrix} \cos(\alpha) \\ \vec{\omega} \end{pmatrix}^{-1} = \left(\begin{pmatrix} \cos(\alpha) \\ \vec{\omega} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ \vec{v} \end{pmatrix} \right) \cdot \begin{pmatrix} \cos(\alpha) \\ -\vec{\omega} \end{pmatrix} \\ &= \begin{pmatrix} -\langle \vec{\omega}, \vec{v} \rangle \\ \cos(\alpha)\vec{v} + \vec{\omega} \times \vec{v} \end{pmatrix} \cdot \begin{pmatrix} \cos(\alpha) \\ -\vec{\omega} \end{pmatrix} \\ &= \begin{pmatrix} -\cos(\alpha)\langle \vec{\omega}, \vec{v} \rangle + \cos(\alpha)\langle \vec{v}, \vec{\omega} \rangle + \langle \vec{\omega} \times \vec{v}, \vec{\omega} \rangle \\ \langle \vec{\omega}, \vec{v} \rangle \vec{\omega} + \cos^2(\alpha)\vec{v} + \cos(\alpha)\vec{\omega} \times \vec{v} - \cos(\alpha)\vec{v} \times \vec{\omega} - (\vec{\omega} \times \vec{v}) \times \vec{\omega} \end{pmatrix} \\ &\stackrel{3.3}{=} \begin{pmatrix} 0 \\ \langle \vec{\omega}, \vec{v} \rangle \vec{\omega} + \cos^2(\alpha)\vec{v} + 2\cos(\alpha)\vec{\omega} \times \vec{v} + (\langle \vec{\omega}, \vec{v} \rangle \vec{\omega} - \vec{\omega}^2 \vec{v}) \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ (\cos^2(\alpha) - \sin^2(\alpha))\vec{v} + 2\cos(\alpha)\vec{\omega} \times \vec{v} + 2\langle \vec{\omega}, \vec{v} \rangle \vec{\omega} \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ \cos(2\alpha)\vec{v} + \sin(2\alpha)\hat{\omega} \times \vec{v} + 2\sin^2(\alpha)\langle \hat{\omega}, \vec{v} \rangle \hat{\omega} \end{pmatrix} \end{aligned}$$

(e) Aus dem vorletzten Ausdruck folgt mit $\cos(\alpha) = \omega_0$ und $\sin^2(\alpha) = \omega_x^2 + \omega_y^2 + \omega_z^2$ die Berechnung

$$\begin{aligned} \begin{pmatrix} x \\ y \\ z \end{pmatrix} &\mapsto (\omega_0^2 - \omega_x^2 - \omega_y^2 - \omega_z^2) \begin{pmatrix} x \\ y \\ z \end{pmatrix} + 2\omega_0 \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} \times \begin{pmatrix} x \\ y \\ z \end{pmatrix} + 2 \left\langle \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}, \begin{pmatrix} x \\ y \\ z \end{pmatrix} \right\rangle \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} \\ &= (\omega_0^2 - \omega_x^2 - \omega_y^2 - \omega_z^2) \begin{pmatrix} x \\ y \\ z \end{pmatrix} + 2\omega_0 \begin{pmatrix} \omega_y z - \omega_z y \\ \omega_z x - \omega_x z \\ \omega_x y - \omega_y x \end{pmatrix} + 2(\omega_x x + \omega_y y + \omega_z z) \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} \\ &= \begin{pmatrix} \omega_0^2 + \omega_x^2 - \omega_y^2 - \omega_z^2 & -2\omega_0\omega_z + 2\omega_x\omega_y & 2\omega_0\omega_y + 2\omega_x\omega_z \\ 2\omega_0\omega_z + 2\omega_x\omega_y & \omega_0^2 - \omega_x^2 + \omega_y^2 - \omega_z^2 & -2\omega_0\omega_x + 2\omega_y\omega_z \\ -2\omega_0\omega_y + 2\omega_x\omega_z & 2\omega_0\omega_x + 2\omega_y\omega_z & \omega_0^2 - \omega_x^2 - \omega_y^2 + \omega_z^2 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad \diamond \end{aligned}$$

Satz 3.5 Konjugieren mit einer Einheitsquaternion $\begin{pmatrix} \cos(\alpha) \\ \sin(\alpha)\hat{\omega} \end{pmatrix}$ bewirkt eine Drehung um die Achse $\hat{\omega}$ mit den Winkel 2α .

BEWEIS: Mit einem beliebigen Vektor $\hat{e} \in V$, der orthogonal zu $\hat{\omega}$ liegt, z.B. $\hat{e} := \widehat{\vec{f} \times \hat{\omega}}$ mit $\vec{f} \perp \hat{\omega}$, bilden $\{\hat{\omega}, \hat{e}, \hat{\omega} \times \hat{e}\}$ eine Orthonormalbasis von V , deren Bilder unter Konjugieren berechnet werden:

$$\begin{aligned} \hat{\omega} &\mapsto \cos(2\alpha)\hat{\omega} + 2\sin^2(\alpha)\underbrace{\langle \hat{\omega}, \hat{\omega} \rangle}_{=1}\hat{\omega} + \sin(2\alpha)\underbrace{\hat{\omega} \times \hat{\omega}}_{=\vec{0}} \\ &= (\cos^2(\alpha) - \sin^2(\alpha) + 2\sin^2(\alpha))\hat{\omega} = \hat{\omega} \\ \hat{e} &\mapsto \cos(2\alpha)\hat{e} + 2\sin^2(\alpha)\underbrace{\langle \hat{\omega}, \hat{e} \rangle}_{=0}\hat{\omega} + \sin(2\alpha)\hat{\omega} \times \hat{e} = \cos(2\alpha)\hat{e} + \sin(2\alpha)\hat{\omega} \times \hat{e} \\ \hat{\omega} \times \hat{e} &\mapsto \cos(2\alpha)\hat{\omega} \times \hat{e} + 2\sin^2(\alpha)\underbrace{\langle \hat{\omega}, \hat{\omega} \times \hat{e} \rangle}_{=0}\hat{\omega} + \sin(2\alpha)\hat{\omega} \times (\hat{\omega} \times \hat{e}) \\ &\stackrel{3.3}{=} \cos(2\alpha)\hat{\omega} \times \hat{e} + \sin(2\alpha)\left(\underbrace{\langle \hat{\omega}, \hat{e} \rangle}_{=0}\hat{\omega} - \underbrace{\langle \hat{\omega}, \hat{\omega} \rangle}_{=1}\hat{e}\right) = \cos(2\alpha)\hat{\omega} \times \hat{e} - \sin(2\alpha)\hat{e}. \end{aligned}$$

$\hat{\omega}$ ist nach der ersten Gleichung die Drehachse. In der dazu orthogonalen Ebene $E_{\hat{\omega}, \hat{\omega} \times \hat{e}}$ wird um den Winkel 2α gedreht. Bezogen auf die Basis $\{\hat{\omega}, \hat{e}, \hat{\omega} \times \hat{e}\}$ gehört zu dieser Drehung die Drehmatrix

$$D_{\hat{\omega}, 2\alpha} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(2\alpha) & -\sin(2\alpha) \\ 0 & \sin(2\alpha) & \cos(2\alpha) \end{pmatrix}.$$

◇

Abb. 3 enthält ein Beispiel zur Berechnung der Drehung des Vektors $\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$ um 15° um die z-Achse. Die Einheits-

quaternion s wird hierzu mit $\begin{pmatrix} \cos(15^\circ) \\ \sin(15^\circ)\hat{e}_z \end{pmatrix}$ belegt und „s Hdreh 0,v“ berechnet die Drehung des Vektors mit Hilfe der Konjugation, s. Abb. 4. Zur Kontrolle wird der Vektor v mit der Drehmatrix „Hdmat s“ gedreht, Hdmat ist in Abb. 5 dargestellt.

```

,s*←(2 10015+180)×**1 (0 0 1)
0.9659258263 0 0 0.2588190451

Hdmat s
0.8660254038 -0.5 0
0.5 0.8660254038 0
0 0 1

s Hdreh 0,v*1 2 3
0 -0.1339745962 2.232050808 3
(Hdmat s)+.*v
-0.1339745962 2.232050808 3
    
```

Abb. 3: Beispiel einer Drehung um 15°

```

r←s Hdreh v;rr
A V1.1 11.04.2018 D.Kilsch
A Drehung durch Konjugieren
A s V[4] Einheitsquaternione
A v V drei- oder vierdimensionaler Vektor
A r V 3 oder 4-dimensioaler Vektor, gedreht durch Konjugieren
A=====
r←(-ρv)†s Hmul(⌊4†0,v)Hmul Hin v s
    
```

Abb. 4: APL-Routine Hdreh

```

r←Hdmat s;rr
A V 1.1 23.10.2018 D.Kilsch
A Festlegen der 3-dimensionalen Drehmatrix zu einer Einheitsquaternione.
A s V[4] Einheitsquaternione
A r M[3;3] Drehmatrix
A=====
A Nicht-Diagonalelemente
r←3 3ρs[1]×0,0,s[3 4],0,0,0,s[2],0
rr←3 3ρ0,(s[2]×s[3]),(s[2]×s[4]),0,0,(s[3]×s[4]),3ρ0
r←2×(r-ρr)+rr+ρrr
A Diagonalelemente
(1 1ρr)←(1 1 -1 -1+.×rr)(1 -1 1 -1+.×rr)(1 -1 -1 1+.×rr←s×s)
    
```

Abb. 5: APL-Routine Hdmat

3.2 Quaternionen und Drehgruppen

Wie in 3.1 bezeichnet $V = \langle h_1, h_2, h_3 \rangle = \mathbb{R}h_1 \oplus \mathbb{R}h_2 \oplus \mathbb{R}h_3$ das dreidimensionale orthogonale Komplement des Zentrums $\mathbb{R}h_0$ in \mathbb{H} . Die Quaternionenmultiplikation eingeschränkt auf V ist nach 3.4 (a) das dreidimensionale Vektorprodukt. Nach 2.8 ist $S = \{s \in \mathbb{H} \mid N(s) = 1\}$ die Menge der Einheitsquaternionen (mit Norm 1).

Satz 3.6 Für die Abbildung von S in die spezielle orthogonale Gruppe der Drehmatrizen des Grades 3 mit Determinante 1

$$\tau : \left\{ \begin{array}{l} S \rightarrow \text{SO}(\mathbb{R}, 3) \\ s \mapsto \tau(s) : \left\{ \begin{array}{l} V \rightarrow V \\ v \mapsto sv s^{-1} \end{array} \right\} \end{array} \right\}$$

gelten:

- (a) $\tau(s)$ ist eine spezielle orthogonale Abbildung des Vektorraums V in sich, $\tau(s) \in \text{SO}(\mathbb{R}, 3)$.
- (b) τ ist ein Epimorphismus mit $\ker \tau = \langle -h_0 \rangle = \{h_0, -h_0\} = S \cap Z(\mathbb{H})$.

BEWEIS:

- (a) • $\tau(s) \in O(V)$: Mit 3.4 (a) folgt:

$$\langle \tau(s)v, \tau(s)v \rangle = N(svs^{-1}) = N(s)N(v)N(s)^{-1} = N(v) = \langle v, v \rangle .$$
 Damit ist $\tau(s)$ eine orthogonale Abbildung auf $\{\hat{\omega}, \hat{e}, \hat{\omega} \times \hat{e}\}$. Sie lässt das Zentrum $\mathbb{R}h_0$ invariant, also auch dessen orthogonales Komplement V .
 - τ ist ein Homomorphismus:

$$\tau(s_1s_2)q = (s_1s_2)q(s_1s_2)^{-1} = s_1(s_2qs_2^{-1})s_1^{-1} = \tau(s_1)\tau(s_2)q .$$
- (b) • τ ist surjektiv: die spezielle orthogonale Abbildung $D \in SO(\mathbb{R}^3)$ besitzt einen Eigenvektor $\hat{\omega}$ zum Eigenwert 1 (Drehachse) und einen hierzu orthogonalen invarianten Unterraum $W = \langle \widehat{h_1 \times \hat{\omega}}, \widehat{\hat{\omega} \times h_1} \rangle$ der Dimension 2, auf dem D als Drehung um einen Winkel α wirkt. (Im Falle $\hat{\omega} = h_1$ muss man h_1 durch h_2 ersetzen.) Nach 3.5 bildet τ die Einheitsquaternionen $\begin{pmatrix} \cos(\frac{\alpha}{2}) \\ \sin(\frac{\alpha}{2})\hat{\omega} \end{pmatrix}$ und $\begin{pmatrix} \cos(\pi + \frac{\alpha}{2}) \\ \sin(\pi + \frac{\alpha}{2})\hat{\omega} \end{pmatrix}$ auf D ab.
 - $\ker \tau = S \cap Z(\mathbb{H})$: Dies ist mit dem eben gezeigten klar, Einheitsquaternionen mit $\alpha = 0^\circ$ und $\alpha = 180^\circ$ werden auf die Drehung um 0° abgebildet. Alternativ kann argumentiert werden: Für $s \in \ker \tau$ gilt $h_i = sh_i s^{-1}$, $i = 1, 2, 3$. Damit liegt s im Zentrum von \mathbb{H} , also in $\langle h_0 \rangle = \mathbb{R}h_0$. Wegen $N(s) = 1$ folgt $s = \pm h_0$. \diamond

Zusammenfassung 3.7 $S/\{\pm 1\} \cong \text{SU}(2, \mathbb{C})/\{\pm \text{Id}\} \cong \text{SO}(3, \mathbb{R})$

4 Bedeutung für die Computergrafik

4.1 Anzahl Multiplikationen

Der Aufwand (Anzahl der Multiplikationen) bei der Multiplikation einer Matrix mit einem Vektor wird mit der Konjugation eines rein imaginären Vektors verglichen, ebenso die Multiplikation zweier Matrizen und zweier Einheitsquaternionen, vgl. hierzu [7].

- (a) Anwendung einer Matrix auf einen Vektor: 3 mal 3, also 9 Multiplikation.
- (b) Einheitsquaternion angewandt auf rein imaginären Vektor durch Konjugieren nach 3.4 (d): 18.
- (c) Multiplikation zweier Matrizen: 27 Multiplikationen.
- (d) Multiplikation zweier Einheitsquaternionen nach 2.5: 16.
- (e) Berechnung der Drehmatrix aus einer Einheitsquaternion nach 3.4(e): 10.

Es ist der unterschiedliche Aufwand in (c) und (d) der beim Arbeiten mit Quaternionen den Aufwand reduziert. Die Hintereinanderausführung von Drehungen kann viel einfacher zusammengefasst werden.

4.2 Modell in der Szene identifizieren

Eine Standardaufgabe in der Bilderkennung besteht darin, ein Modellobjekt $\{\vec{m}_i | i = 1, \dots, n\}$ auf das Szenenobjekt $\{\vec{s}_i | i = 1, \dots, n\}$ abzubilden, s. Abb. 6. Durch Translation des Szenenobjekts kann erreicht werden, dass ein Punkt des Modells mit einem Punkt der Szene übereinstimmt (Abb. 7). Dieser Punkt wird als Ursprung des Koordinatensystems gewählt. Gesucht ist jetzt die Drehung D , die den Fehler

$$E(D) = \sum_{i=1}^n \|\vec{s}_i - D\vec{m}_i\|^2$$

minimiert. Diese Optimierungsaufgabe wird mit Einheitsquaternionen $q = \begin{pmatrix} \cos(\frac{\alpha}{2}) \\ \sin(\frac{\alpha}{2})\hat{\omega} \end{pmatrix}$ berechnet:

$$E(q) = \sum_{i=1}^n \|\vec{s}_i - q\vec{m}_i q^{-1}\|^2 \cdot 1 = \sum_{i=1}^n \|\vec{s}_i - q\vec{m}_i q^{-1}\|^2 \cdot \|q\|^2 = \sum_{i=1}^n \|\vec{s}_i q - q\vec{m}_i\|^2 \tag{5}$$

$$= \sum_{i=1}^n \|A_i \vec{q}\|^2 = \sum_{i=1}^n \vec{q}^t A_i^t A_i \vec{q} = \vec{q}^t \left(\sum_{i=1}^n A_i^t A_i \right) \vec{q} \tag{6}$$

$$= \vec{q}^t \cdot B \cdot \vec{q} \tag{7}$$

Dabei erhält man die erste Gleichheit in (6) durch die Beobachtung, dass

$$q \mapsto \vec{s}_i q - q\vec{m}_i$$

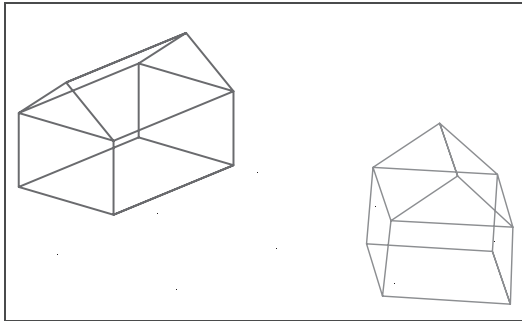


Abb. 6: Model (blau) und Szene (grün)

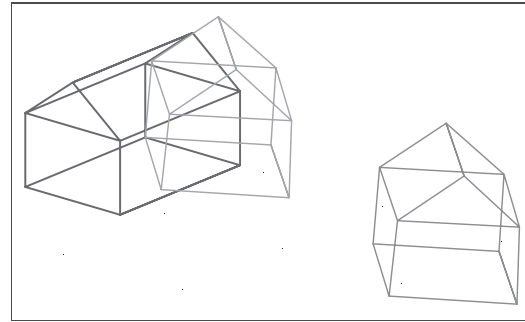


Abb. 7: Model (blau) und verschobene Szene (rot)

eine in q \mathbb{R} -lineare Abbildung $\mathbb{H} \rightarrow \mathbb{H}$ ist und diese durch eine Matrix $A_i \in \text{GL}(\mathbb{R}^4)$ dargestellt wird.

Die Matrix B ist als Summe symmetrischer Matrizen symmetrisch. Wegen

$$\langle B\vec{q}, \vec{q} \rangle = \left\langle \left(\sum_{i=1}^n A_i^t A_i \right) \vec{q}, \vec{q} \right\rangle = \sum_{i=1}^n \langle A_i^t A_i \vec{q}, \vec{q} \rangle = \sum_{i=1}^n \langle A_i \vec{q}, A_i \vec{q} \rangle = \sum_{i=1}^n \|A_i \vec{q}\|^2$$

ist sie semi-definit. Im Falle $\det B = 0$ gibt es eine Quaternion $\vec{q} \in \mathbb{H}$ mit $B \cdot \vec{q} = \vec{0}$. Die normierte Einheitsquaternion löst dann die Aufgabe fehlerfrei. Im Weiteren sei B positiv definit. Der Eigenvektor zum kleinsten nicht-negativen Eigenwert minimiert den Fehler. Dies wird zusammengefasst:

Verfahren 4.1 *Seien*

- $A_i : \begin{cases} \mathbb{H} & \rightarrow & \mathbb{H} \\ q & \mapsto & \bar{s}_i q - q \bar{m}_i \end{cases} \in \text{GL}_{\mathbb{R}}(\mathbb{H})$
- $B = \sum_{i=1}^n A_i^t A_i$

Dann wird der Fehler $E(D)$ durch die Matrix D zum Einheitsvektor \vec{q} zum kleinsten Eigenwert der Matrix B minimiert. Der kleinste Eigenwert und der dazugehörige Eigenvektor können durch das Verfahren von Mises¹⁶ und die Wielandt¹⁷-Iteration, s. [3, §13] berechnet werden.

Das Verfahren nach Mises (Abb. 8) startet mit einem Vektor außerhalb des Kerns der Matrix B . Es multipliziert ihn mit B und normiert ihn wiederholend, bis die Änderung kleiner als ein Abbruchkriterium, z.B. 10^{-8} , wird. Es konvergiert gegen einen Eigenvektor zum größten Eigenwert.

```
r←Mises1 mat
a V1.1 23.10.2018
a Verfahren von Mises zur Berechnung des Eigenvektors zum größten
a Eigenwert einer semi-definiten Matrix.
a mat M      symmetrische, semi-definite Matrix
a r      AV[1] S  größter Eigenwert
a      [2] V  Eigenvektor
a=====
x←(1⊖mat)⊕1

DO:
x←mat+.×xalt←x      a Vektor mit Matrix multiplizieren
x←x÷(+/x×x)*0.5    a Vektor normieren
→((|x-xalt)|>1E-8)/DO  a Iteration beenden, wenn Änderung ≤ 1E-8

r←((mat+.×x)⊖x)x
```

Abb. 8: APL-Routine Mises1

Zur Berechnung eines Eigenwerts zum kleinsten Eigenwert einer positiv-definiten Matrix A setzt Wielandt das Verfahren von Mises für die inverse Matrix A^{-1} ein, s. Abb. 9. Statt die Inverse wirklich zu bestimmen und $\vec{x}_{n+1} = A^{-1} \cdot \vec{x}_n$ direkt zu berechnen, ermittelt er x_{n+1} durch Lösen des linearen Gleichungssystems $A \cdot \vec{x}_{n+1} = x_n$. Die Funktion Hdrfnd in Abb. 10 fasst die Vorbereitung und das Anwenden dieses Algorithmus zusammen. Sie löst die Aufgabenstellung, die gefundene Matrix dreht das Modell auf das grüne Bild in Abb. 11 mit geringer Abweichung vom verschobenen Szenenbild!

Der Aufbau des Modells und der verschobenen Szene, der Aufruf von Hdrfnd und eine Fehlerberechnung ist in Abb. 12 enthalten.

¹⁶Richard von Mises, österr. Mathematiker, 1883 (Lemberg, heute Lviv/Ukraine) - 1953 (Boston/Mass.)

¹⁷Helmut Wielandt, dt. Mathematiker, 1910 (Niedereggenen) - 2001 (Schliersee)


```

r←Wielal mat
A V1.1 23.10.2018
A Verfahren von Wieland: Iteratives Bestimmen des Eigenvektors zum
A kleinsten Eigenwert: Verfahren von Mises für inverse Matrix.
A mat M symmetrische, positiv-definite Matrix
A r AV[1] S kleinster Eigenwert
A [2] V Eigenvektor
A=====
x←(1⊖pmat)†1

DO:
x←(xalt+x)⊖mat a Statt mit Inverser mult.: LGS lösen
x←x†(+/x×x)*0.5 a Vektor normieren
→((†/|x-xalt)>1E-8)/DO a Iteration beenden, wenn Änderung ≤ 1E"8

r←((mat+.×x)⊖x)x a Ergebnis
    
```

Abb. 9: APL-Routine Wielal

```

r←sc Hdrfnd mo
A V1.1 23.10.2018
A Bestimmen der Einheitsquaternione, die das Modell auf die Szene abbildet.
A mo M Modell: Zeilenweise Punkte
A sc M Modell: Zeilenweise Punkte
A r V[4] Einheitsquaternione
A=
A lokale Variable
A A AV M[4,4] lineare Abbildung je korrespondierendem Punktepaar
A B M[4,4] Summe der Produkte lin. Abb. in A und ihrer Transpon.
A q M[4,4] Einheitsmatrix
A e V[4] Eigenvektor: Einheitquaternione
A w S Eigenwert
A=====
q←c[2]4 4p5†1
A←q††c[2]((c[2]0,sc)◦.Hmul q)-q◦.Hmulc[2]0,mo
B←††/(q†A)+.×"A a pos. definite Matrix

(w e)←Wielal B a Eigenwert und -vektor
r←e
    
```

Abb. 10: APL-Routine Hdrfnd: Berechnen der Einheitsquaternione

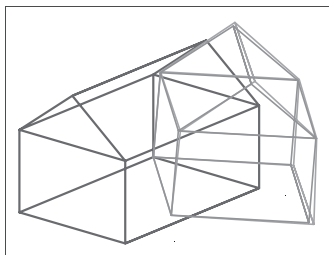


Abb. 11: gedrehtes Modell (grün) und verschobene Szene (rot)

```

Modell
mo←4 3p0 0 0 12 0 0 12 8 0 0 8 0
mo←mo,[1]0 0 5+[2]mo
q←mo←mo,[1]2 3p0 4 8 12 4 8 a Modell

sc←mo+.×1 Dm3 -45 4 5 a Drehen des Modells
sc←(0.99+(psc)ρ0.02×ε((ρ,sc)ρ1)?"2)×sc a Verrauschen
q←sc+14 31 4+[2]sc a Verschobene Szene

e←sc Hdrfnd mo
(sc-†(cHdrmat,e)+.×"c[2]mo)÷sc a Fehler
    
```

Abb. 12: Aufbau Modell, Szene und Berechnung

Literatur

- [1] B. Huppert und W. Willems. Lineare Algebra. Teubner, 2006. ISBN: 3-8351-0089-0.
- [2] X. Jung und H. Bunke. Dreidimensionales Computertsehen. Gewinnung und Analyse von Tiefenbildern. Berlin: Springer, 1997. ISBN: 3-540-60797-8.
- [3] D. Kilsch. Informatik und Numerik. FH Bingen, WS 2015/16. <https://seafilerlp.net/d/9a29c9397c/files/?p=/lehrvera/Math/Skr4.pdf&dl=1>.
- [4] D. Kilsch. Mathematik A. FH Bingen, WS 2007/08. <https://seafilerlp.net/d/9a29c9397c/files/?p=/lehrvera/Math/Skr1.pdf&dl=1>.
- [5] Mathepedia. Quaternionen. <http://www.mathepedia.de/Quaternionen.html> (besucht am 30.01.2018).
- [6] D. J. Robinson. Abstract Algebra: An Introduction with Applications. 2nd ed. De Gruyter Textbook. Berlin: Walter de Gruyter, 2015. ISBN: 978-3-11-034086-0.
- [7] Wikipedia. Quaternionen. <https://de.wikipedia.org/wiki/Quaternion> (besucht am 30.01.2018).

Weil das Gute nur kurze Zeit das Neue ist¹

Barghoorn@zedat.fu-berlin.de

1. Welche Computer sind gemeint

Desktop-PCs, Notebooks und Thin Clients in privater Nutzung oder im Büro.

Obwohl Tablets und Mobiltelefone thematisch dazugehören, werden sie hier nur am Rande erwähnt. Dasselbe gilt für Großrechner und Workstations, weil dort die Möglichkeiten von individueller Modernisierung erfahrungsgemäß beschränkt sind.

2. Wann sind Computer alt?

Computer sind Gebrauchsgüter, die wie andere technische Produkte veralten, etwa durch technischen Fortschritt und Abnutzung oder weil sie durch ihr Äußeres als unmodern empfunden werden. Für dieses Phänomen gibt es den Begriff Obsoleszenz. Wenn Obsoleszenz künstlich erzeugt wird, z.B. durch Marketing und Profitstreben, spricht man von geplanter Obsoleszenz. Das heißt, die Lebensdauer der Produkte wird schon in der Planung absichtlich verkürzt. Es ist möglich und auch aufgrund der Wirtschaftsweise wahrscheinlich, dass Produzenten ein Interesse daran haben, die Gebrauchsdauer ihrer Produkte künstlich zu verkürzen, um ihren Absatz und Gewinn zu steigern.

Da Computer hauptsächlich durch ihre Funktionalität und weniger durch ihr Äußeres zu charakterisieren sind, ist bei dieser Art der Produkte die Ausmusterung wegen eines geänderten Designs (Mode) weniger von Bedeutung. Bei Mobiltelefonen dagegen spielt das Design eine größere Rolle, deshalb gibt es dort eher die Möglichkeit, durch Designänderung, neue Modelle oder Modeströmungen eine Obsoleszenz künstlich zu erzeugen.

Auch Veränderungen der Computer-Software durch Updates oder neue Versionen lässt Computer veralten. Das betrifft sowohl das Betriebssystem als auch die Anwendungsprogramme. Zum Beispiel haben in der Vergangenheit Banken oder Finanzämter keine Kommunikation mehr ermöglicht, wenn noch das vermeintlich *veraltete* Betriebssystem Windows XP ermittelt wurde. Diese Restriktion wurde später wieder fallengelassen, so dass aktuell Onlinebanking mit Windows XP und vielen Banken wieder möglich ist.

Ab wann Computer als veraltet angesehen werden können, hängt von vielen Einflussfaktoren ab, auf die im Folgenden eingegangen wird. Erfreulicherweise ist die Gebrauchsdauer einiger wichtiger technischer Produkte in den vergangenen Jahren deutlich gestiegen, so z.B. bei PKW und auch Computern. Überwiegend als schnelllebige Produkte können dagegen leider die Computer-Drucker und Scanner weiterhin bezeichnet werden.

Nach meinen aktuellen praktischen Erfahrungen lassen sich Computer – je nach Zustand - noch sinnvoll modernisieren, wenn sie jünger als 10-15 Jahre sind. Danach sind sich nur noch einzelne Komponenten nutzbar bzw. verwertbar. Es sollte aber möglich sein, Computer 15 Jahre und länger zu nutzen.

¹ *Das Neue ist selten das Gute, weil das Gute nur kurze Zeit das Neue ist*, Arthur Schopenhauer, Parerga und Paralipomena, 1851

Gemäß dem aktuellen Avast „PC Trend Report 2019“² haben PC in Deutschland und weltweit ein Durchschnittsalter von 6 Jahren erreicht. Mehr als 97 % der Computer waren älter als 2 Jahre. Relativ die meisten der aktiven Computer wurden 2009 verkauft (14,3 %). Immerhin noch 10,2 % der Computer waren aus dem Jahr 2003. Etwa die Hälfte der Computer war schon älter als 9 Jahre.

3. Anforderungen an Computer, Langlebigkeit und Entsorgung

Folgende Anforderungen können allgemein an Computer bei der Produktion und beim Produkterwerb gestellt werden:

- Schnelligkeit, die Bauteile passen zueinander, Komponenten bremsen sich nicht gegenseitig, das System läuft flüssig
- das System startet schnell, Standfestigkeit, Langlebigkeit aller Einzelteile
- eine hohe Qualität aller Bauteile sollte gewährleistet sein, keine Billigteile verbaut
- Wartungsfreundlichkeit, die Bauteile lassen sich ohne großen Aufwand wechseln
- der Computer ist leistungsstark und hat gleichzeitig wenig Stromverbrauch
- Betriebssystem und Anwendungen sind schnell, bedienungsfreundlich und sicher
- Computer sollen nicht überteuert sein, die Entsorgung gesichert und nicht umweltschädlich im In- und Ausland.

In Bezug auf die genannten Forderungen ist festzustellen, dass diese infolge der rasanten Entwicklung der Computertechnik heute deutlich besser erfüllt werden als noch vor 25 Jahren. Die Wartungsfreundlichkeit von Desktop-PC und Notebooks kann im Vergleich mit anderen technischen elektrischen Produkten als gut bezeichnet werden. Mit relativ geringen handwerklichen bzw. mechanischen Kenntnissen und Fähigkeiten lassen sich fast alle Modernisierungsarbeiten leicht bewältigen. Auch das Internet und diverse YouTube-Videos sind fast immer geeignet, die nötige Hilfestellung zu geben.

Die Forderungen nach mehr Leistung und höherer Energieeffizienz stehen naturgemäß im Widerspruch. Trotz schneller Weiterentwicklung der Betriebssysteme und der peripheren Geräte, wie z.B. Drucker und Scanner, sollte eine möglichst lange Nutzungsdauer garantiert sein. Serviceeinschränkungen für ältere Betriebssysteme, wie sie bei Windows XP vorkamen und derzeit bei Windows 7 stattfinden, sollten unbedingt vermieden werden. Alternativ lassen sich auch Open Source Systeme einsetzen, bei denen der Service zeitlich nicht beschränkt ist.

4. Alterung und Ausmusterung von Geräten

Die Entscheidung, wann bestimmte elektrische Geräte ausgemustert und verwertet werden, wird nicht ausschließlich durch technischen Parameter und Gesetze (z.B. Ausfallwahrscheinlichkeit von Bauteilen oder Verschleiß) bestimmt. Hinzu kommen weitere wirtschaftliche und gesellschaftliche Einflussvariable.

Einflussvariable für die Modernisierung oder Ausmusterung von technischen Geräten allgemein:

- Verfügbarkeit von Ersatzteilen, Service, Umweltfreundlichkeit

² Avast® PC Trends Report 2019

- Preis der Ersatzteile und Reparaturen und Reparaturfreundlichkeit
- veränderte Interoperabilität/Kompatibilität, technischer Fortschritt, Energiesparen, Wasserverbrauch etc.
- Persönliche Umstände (z.B. Umzug, Geschenk, Erbschaft)
- Mode, Werbung, Image, veraltete Modelle, Gesetze (z.B. Diesel NOx Emission).

Um gezielt zu prüfen, welche Bauteile bzw. Baugruppen im Einzelnen ersetzbar sind und mit welchem Aufwand, wird zunächst der typische Aufbau eines Desktop-PCs untersucht werden. Dieser Aufbau hat sich in den zurückliegenden 25 Jahren kaum geändert.

5. Die wichtigsten Komponenten eines Desktop-Computers

Gehäuse, Desktop, Midi und Big Tower

Mainboard oder Motherboard

Prozessor

Arbeitsspeicher (RAM)

Grafikkarte teilweise integriert

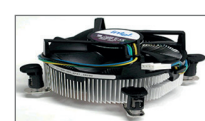
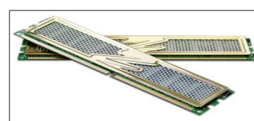
CPU-Kühler

Netzteil

Festplatte(n)

Laufwerke CD, DVD, USB etc.

Diverse Anschlüsse, Schnittstellen



6. Verschleiß, technische und spezifische Alterung

Die Alterung von Computern und anderen technischen Produkten macht sich bemerkbar, indem ihre Funktionalität ganz oder teilweise ausfällt³ oder subjektiv oder objektiv beeinträchtigt wird, z.B., dass die Geräte als zu langsam empfunden werden oder tatsächlich absolut oder relativ langsamer werden. Weiterhin können einzelne Programme oder der Computer insgesamt zu oft *abstürzen*. Dies kann durch Soft- oder Hardware verursacht werden. Reine Computerlaien können meist nicht unterscheiden, was die Ursachen sind. Sie können nur feststellen, dass der Bildschirm schwarz bleibt.

Die technische Weiterentwicklung der einzelnen Bauteile des Computers verläuft erfahrungsgemäß mit unterschiedlicher Geschwindigkeit und lässt sich kaum vorhersagen. Wenn im Zeitablauf einzelne Komponenten deutlich leistungsfähiger (schneller, mehr Kapazität) werden, kann umgekehrt der Computer als Ganzes beschleunigt altern. Dadurch können sich aber

³ Man hört oft den Spruch, dass Computer oder andere Produkte ihren *Geist* aufgeben.

auch Chancen ergeben, durch Modernisierung einzelner Bauteile den Computer wieder schneller, leistungsfähiger und somit langlebiger zu machen.

7. Ralf Herminghaus' Benchmarks zur Ermittlung der Obsoleszenz von Computern
 Durch Anwendung einiger APL-Programme von Ralf Herminghaus wird beispielhaft die Leistungsfähigkeit von Computern aus unterschiedlichen Baujahren ermittelt werden. Die Performance von Computern wird durch Simulationen quantifiziert und verglichen werden. Diverse numerische Programme (Median, Mittelwert und Zufallszahlen) sowie Funktionen, mit denen große Mengen von Strings zufällig erzeugt und verarbeitet wurden, kamen zum Einsatz.

Performance-Messungen vor und danach sind außerdem gute geeignet, Verbesserungen der Computermmodernisierung in einzelnen Phasen und insgesamt zu beurteilen.

Abbildung: diverse Hermingshaus Simulationen

```

A-----
A Test-Objects
A-----
X←(?Np1000)×0.001  A---NUM-Vector-(Floating Point)---
Y←(?Np1000)×0.001  A---NUM-Vector-(Floating Point)---
XI←[100×X          A---NUM-Vector-(Integer)-----
YI←[100×Y          A---NUM-Vector-(Integer)-----
B←X>Y              A---Boolean Vector-----
IND_B←B/ιtpB      A---Index Representation of B-----

sN←10              A---Size of short Strings-----
L←NpAV[65+(?N)p60] A---Long String---
S←sNpAV[65+(?sN)p60] A---Short String---
K←(N sN)pAV[65+(?N×sN)p60] A---Character-Key-Table----
KX←c[2]K          A---Character-Key-Vector---

AV_FIRST←tAV
AV_LAST←t-1tAV

sN2←20             A---Size of short Strings-----
S2←sN2pAV[65+(?sN2)p60] A---Short String---
K2←(N sN2)pAV[65+(?N×sN2)p60] A---Character-Key-Table----
KX2←c[2]K2        A---Character-Key-Vector---

PI←01              A---3.14159265...-----

M←Xo_+Y
MX←c[2]M
MC←(N N)pAV[65+(?N×N)p60] A---Character-Matrix---
H1←(+/+M)÷(N×N)
MB←(N N)p(M>H1)    A---Boolean-Matrix-----
    
```

Beispielhaft wird hier meine APL-Funktion *CHAR* wiedergegeben werden. Die verbrauchte CPU-Zeit wurde mit den Funktionen *TIME 0* und *TIME 1* aus dem TIME-Workspace on APL2 verglichen.

```

r←CHAR N;t
i←1∅r←''
m:
t←(N N)pAV[65+(?N×N)p60]
t←+/,tε'H'
r←r,t
i←i+1
→(i≤500)/m
    
```

Insgesamt sechs Computer aus den Baujahren 2003 bis 2012 wurden von mir untersucht und es ergaben sich folgende Messwerte:

Modell Baujahr	CPU / Frequenz	RAM	APL-Benchmark
Tower 2003	Pentium 4 2 GHz	2 Gbyte	45 sek
R52 2005	Pentium M 2 GHz	1,5 Gbyte	64 sek
T60 2007	Pentium 2600 2,1 GHz	1 Gbyte	24 sek
X61 s 2009	Pentium L7500 1,6 GHz	3 Gbyte	27 sek
T 410 2011	Pentium M580 2,67 GHz	4 Gbyte	11 sek
T420 2012	Pentium i5 2530 2,5 GHz	4 Gbyte	11 sek

8. Computer-Performance-Messungen mit dem R-System

Hierfür gibt es bei R die Funktion `system.time`, welche die verbrauchte CPU-Zeit außerdem auftrennt nach der Anwendung (R) und dem Betriebssystem.

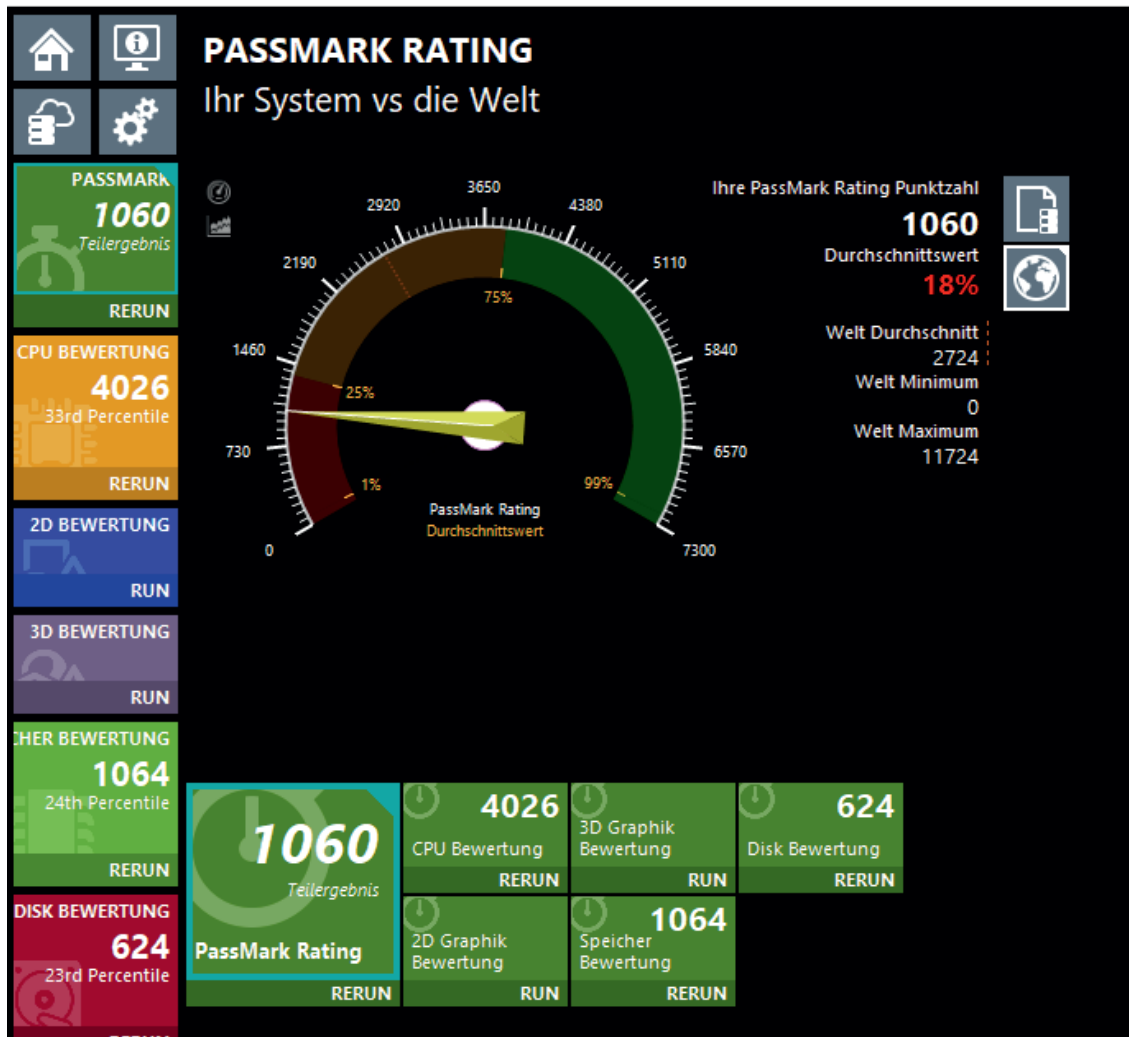
```
> test
function(anz) {
  i<-1 ; r<- NULL
  for(i in 1:anz) {
    r <- c(r, median.apl(sample(5e6,1000)))
  }
  r
}
> system.time(h<-test(1000))
User      System verstrichen
3.53      4.20      7.73
```

Funktion `median.apl`:

```
> median.apl
function(dat)
{
# Median von Vektoren
  h <- sort(dat)
  l <- 0.5 * length(h)
  i <- h[c(ceiling(l), floor(l + 1))]
  res <- 0.5 * sum(i)
  res
}
```

9. Professionelle Software für Performance Messungen mit *Passmark*

Mit *Passmark* lässt sich jeweils ein ausführliches Protokoll des gesamten Systems und der Messwerte erzeugen. Gemessen und verglichen werden jeweils CPU, Grafikkarte und Arbeitsspeicher.



10 Konkrete Arbeiten und Maßnahmen für Computer-Modernisierung

10.1 Hardware

- Computer reinigen und entstauben, Computer sollte freistehen, für gute Belüftung sorgen, saubere Luft, kein Rauch(er)
- Rechtzeitiger Austausch defekter Komponenten, mit preiswerten, aber nicht mit den billigsten Bauteilen ersetzen, unbedingt auf Qualität achten.
- SSD einbauen anstelle von HDD
SSDs kommen auf Lese-/Schreibgeschwindigkeiten von bis zu 550 Megabyte pro Sekunde. HDDs schaffen nur 120 MB/s. SSDs sind also viel schneller, mobiler, robuster und tendenziell länger haltbar. Außerdem sind sie absolut geräuschlos, Durch Ersatz der HDD wird der Computer in der Regel deutlich schneller. Dies gilt auch für Notebooks.
- Arbeitsspeicher erweitern (RAM)
Allgemein gilt, je mehr Arbeitsspeicher der Computer besitzt, desto schneller werden die Programme ausgeführt. Falls das System aufgrund unzureichendem RAM langsam arbeitet, kann man auch den virtuellen Speicher erhöhen. Das Hinzufügen von *echtem* RAM ist jedoch die bessere Lösung, weil der Prozessor Daten aus dem RAM viel schneller lesen kann, als von einer Festplatte.

12.2 Software

Unter Windows sollten die Anwendungen und Prozesse mit dem Task-Manager kontrolliert werden. Generell ist darauf zu achten, dass die Anwendungen nicht schon beim Hochfahren des Computers automatisch gestartet werden, sondern erst, wenn sie benötigt werden. Hierdurch wird der Arbeitsspeicher entlastet. Kontrolle und Einstellung mit dem Programm msconfig bzw. dem Task-Manager.

13 Zusammenfassung

Erneuern und Beschleunigen Sie Ihren Computer. Das kostet sehr viel weniger als ein neuer PC, ist umweltfreundlich und ist auch nicht kompliziert. Sie müssen dazu weder Techniker noch Handwerker sein. Ein Schraubenzieher (Kreuzschlitz) und evtl. eine Lupe sind die einzigen Werkzeuge, die Sie benötigen. Sie tun es für sich und die Umwelt, viel Spaß!



DPC-Computer in Solingen 2013

Wie man alte Computer wieder flott macht

Stefan.Ebelt@Ebelt-Beratung.de

Einen ‚alten‘ Computer wieder flott machen, ist wirtschaftlich und wenn man es selbst macht, kann es sogar Spaß machen. Aber wir wollen das Wort ‚alt‘ lieber nicht mehr verwenden, sondern von ‚gebrauchten‘ IT-Geräten sprechen. Wer ist schon gerne ‚alt‘?

Es erreichen mich häufig Anfragen, ob wir alte bzw. gebrauchte Rechner annehmen. Bei der Nachfrage, wie alt die Geräte denn seien, wird meistens gesagt, dass man das gar nicht mehr wisse. Nach Erhalt dieser Geräte stellt sich heraus, dass diese Rechner Tatsache sechs bis über zehn Jahre alt sind. Bei diesen Geräten ist dann wirklich nichts mehr ‚flott‘ zu machen - die PCs werden zerlegt und nach Komponenten bzw. Materialien sortiert; die Materialien und einige Komponenten werden einem Schrotthändler verkauft (Stahl, Eisen, Kupfer, Alu, Platinen, CPU und Speicher, etc.). Brauchbare Komponenten können in ältere PCs wieder eingebaut werden, nachdem sie geprüft wurden.

Bei Notebooks ist die Verfahrensweise ähnlich: Hier fallen jedoch mit dem TFT kostenpflichtige Abfälle an, wenn diese nicht mehr verwendet werden können. Prinzipiell ist im Notebook dieselbe Technik verbaut, jedoch wurde jede Komponente für den Miniatureinsatz umkonstruiert, sodass die Geräte möglichst klein werden können und auch das Gewicht nach und nach wesentlich reduziert wird. Erstaunlicherweise sind manche Tablets genauso leistungsfähig, wie kleine einfache Notebooks. Noch erstaunlicher ist es, dass größere bzw. teure Smartphones fast so leistungsfähig sind, wie durchschnittliche Notebooks.

Tablets

Während PCs heute immer weniger verkauft werden, und die Tendenz zum Notebook bzw. Zweit-Notebook geht und zusätzlich mindestens noch ein Smartphone in Benutzung ist, eventuell auch ein Tablett, ist die Reparierfreudigkeit an den hier aufgezählten Geräten ablesbar: PCs können sehr gut repariert, auf- oder umgerüstet werden, bei Notebooks gibt es schon Einschränkungen, noch weniger kann bei Smartphones repariert werden und bei Tablets ist eine Reparatur meistens vollständig unmöglich. Dass Tablets nicht repariert werden, liegt an der Miniaturisierung und an der Bauweise, die meisten Komponenten sind verklebt. Zudem sind Tablets in einem Preissegment angeordnet, wo eine Reparatur in jedem Falle den Preis eines neuen Gerätes erreicht - in der Einstufung des Umweltschutzes und Ressourcenverbrauchs schneiden diese Geräte am schlechtesten ab.

Smartphones

Bei Smartphones ist in Reparaturfällen sehr häufig die Touch-Oberfläche beschädigt. In den allermeisten Fällen kann das Glas ausgetauscht werden. Da die Smartphone-Hersteller ebenfalls dazu übergehen, die Glasflächen einzuleben, ist ein Austausch in Reparaturfällen wesentlich aufwendiger, als in früheren Fällen, womit Dichtungen und kleinen Schrauben bzw. Klippsen gearbeitet wurde. Für den Konsumenten, der selbst seine Glasoberfläche austauschen möchte, gibt es mannigfaltige Hilfsmittel und Werkzeuge. Das



reicht allerdings nicht, um eine erfolgreiche Reparatur durchzuführen. Der Eigentümer muss mindestens ein wenig technischen Verstand und Gefühl für Materialien haben. Nicht selten kann der Eigentümer die Glasoberfläche austauschen, nur mit dem wieder zusammen montieren sieht es weniger gut aus. Der alte Kleber muss entfernt, und mit neuem Kleber die Glasoberfläche eingesetzt werden. Die Schwierigkeit besteht darin, erstens einen Kleber zu finden und zweitens den schnell trocknen Kleber dann so zu verwenden, dass die Glasoberfläche gut passend eingesetzt wird. Wenn das ohne Schaden und Nebeneffekte durchgeführt wurde – Gratulation!

Einen Schwierigkeitsgrad kleiner kann der Akku selber ausgetauscht werden, auch der ist selbstverständlich verklebt. Hier besteht die Kunst darin, den meistens sehr festen Kleber zu lösen, um den alten Akku herausnehmen zu können. Hierfür gibt es Spezialwerkzeuge und Lösungsmittel, um nicht nur den Akku zu entfernen, sondern auch den Kleber, der in jedem Falle entfernt werden muss. Wenn ein schlauer Mensch meint, den Akku einfach einlegen zu können, ohne ihn fest zu kleben, besteht Gefahr, wenn nicht sogar Lebensgefahr: Wenn das Smartphone längere Zeit benutzt wird, erhitzt sich der Akku. Diese Wärme muss abgeführt werden. Bleibt die Wärme im Gerät, kann die darüber liegende Elektronik Schaden nehmen oder der Akku überhitzt sich selber und gerät bzw. setzt das Gerät in Brand! Der Kleber funktioniert nicht nur als Festhalter für den Akku, sondern ist gleichzeitig dazu da, die Wärme an die Außenschale abzuführen (von einer Wärmeleitpaste, wie sie im PC- bzw. Notebookbau verwendet wird, ist abzuraten). Wurde der Austausch des Akkus richtig durchgeführt, kann ein Smartphone die nächsten 2-4 Jahre überstehen. Im Allgemeinen haben Smartphones eine Lebensdauer von mindestens 4-10 Jahren; Das hängt meistens noch vom Preis ab, mindestens aber von der herstellenden Firma und den verkauften Modellen. Sollte bei der Produktion beim Hersteller kein Flüchtigkeitsfehler passiert sein, ist die Lebensdauer umso länger, je teurer das Smartphone ist. Allerdings gibt es verschiedene Modellreihen, die für verschiedene Preissegmente gedacht sind. So gibt es Modelle, die im unteren Preissegment angeordnet sind, und bei mäßiger Beanspruchung nur ein bis drei Jahre halten sollen.

Den höchsten Schwierigkeitsgrad hat das Austauschen zum Beispiel einer Kamera oder eines Mikrofons. Hier muss ein verklebtes Smartphone geöffnet werden (erste Schwierigkeit: der Kleber), dann muss die Hauptplatine gelöst werden (zweiter Schwierigkeitsgrad) und nun muss die Verbindung (ein kleines empfindliches Kabel) gefunden und vollständig entfernt werden, als auch das Modul mit der Kamera (dritter Schwierigkeitsgrad). Der vierte Schwierigkeitsgrad ergibt sich, eine neue Kamera mit Verkabelung und dem richtigen Modell zu kaufen. Der fünfte Schwierigkeitsgrad besteht nun darin, den Montageweg im Rückwärtsgang vorzunehmen und dabei keine Sekundärschäden zu verursachen. Wer das hin bekommt: Gratulation! (Melden sie sich bei mir als Techniker).

Daneben abgebildet ein paar brauchbare Werkzeuge, die für die Öffnung, das Abschrauben und für weitere Manipulationen am Gerät zu empfehlen sind.

Wenn man diese ganzen Reparaturen nicht selbst machen möchte, kann man selbstverständlich auch gute Reparaturläden besuchen. Diese führen dann die oben genannten Tätigkeiten für Sie aus - aber



bitte: Sie können meistens nicht auf die Reparatur warten. Da die Tätigkeiten, wie z.B. Kleber entfernen, mehrere Stunden dauern kann, auch das Trocknen von Kleber und anderen Materialien benötigt seine Zeit. Im Internet findet man viele Reparaturbetriebe, die Ihnen diese Leistung anbieten. Bitte achten Sie hier auf Bewertungen oder Kommentare von Nutzern, die schon einmal dort etwas haben reparieren lassen - diese Hinweise geben einen ersten Eindruck, ob der Reparaturladen die Leistungen auch wirklich erbringen kann. Es gibt gute Reparaturläden, die lehnen eine Reparatur ab, wo sich andere Reparateure schon versucht haben, um aufgestaunten Ärger der Kunden oder unberechtigte Kosten zu vermeiden.

Notebooks

Ein wenig besser stellt sich die Situation bei der Reparatur oder Umbau von Notebooks dar. Da hier zum größten Teil noch Schrauben verwendet werden, um das Gerät zu schließen bzw. zu öffnen, ist der Ärger mit dem Kleber beim Reparieren dieser Geräte noch nicht vorgedrungen. Allerdings, je kleiner Notebooks werden und zu Tablets mutieren, ist die Problematik sofort wieder vorhanden.

Bei Notebooks können Sie meistens Speicher aufrüsten bzw. austauschen, Festplatten gegen größere oder bessere austauschen (SSD) und wenn wieder zu viel Kaffee in der Tasse war, auch die meisten Tastaturen austauschen. Sollte ein Display einen Schaden haben, so kann oft auch dieses ausgetauscht werden. Die guten Reparateure haben entweder gleiche Ersatzteile am Lager, oder wissen, wo sie diese bestellen können. Noch schwieriger wird es, wenn festgestellt wird, dass Komponenten im Notebook defekt sind. Ein CPU-Lüfter gehört zu den einfachen Aufgaben, wenn dieser ausgetauscht werden soll. Bei jedem Eingriff in ein Notebook, bei dem viele Schrauben entfernt werden müssen, um an die entlegensten Stellen zu kommen, benötigt man zumindest Hinweise aus dem Internet, wie die Demontage in einzelnen Schritten vor sich geht. Kennt man die Reihenfolge der Demontage, ist in den meisten Fällen der rückwärtige Gang der Montage möglich. Hierbei hat man auf die Werkzeuge und die Zeit, die man beim Öffnen als auch beim Schließen des Gerätes verbraucht, zu achten. Soll zum Beispiel eine Tastatur oder ein CPU-Lüfter ausgetauscht werden, kann das durchaus bedeuten, dass sämtliche Komponenten des Notebooks ausgebaut bzw. gelockert werden müssen. Dabei können ein bis zwei Stunden auch von Fachkräften benötigt werden.

Wir unterscheiden bei Notebooks und PCs zwischen Businessgeräten und Konsumergeräten. Grob gesagt liegt der Unterschied darin, dass Konsumergeräte auf Leistung getrimmt sind und Businessgeräte auf Qualität (und Leistung). Konsumergeräten sollen eine Lebensdauer von ca. 2-5 Jahren haben, Businessgeräte sollen ca. vier bis über zehn Jahre halten. Mittlerweile hat jeder Hersteller, sei er auch noch so namhaft, Modelle für jedes Marktsegment im Angebot. So kann zum Beispiel ein Notebook von Lenovo schon für 200 bis 300 € gekauft werden, oder man bestellt ein Businessgerät für 3.500 € oder mehr bei einem Fachhändler.

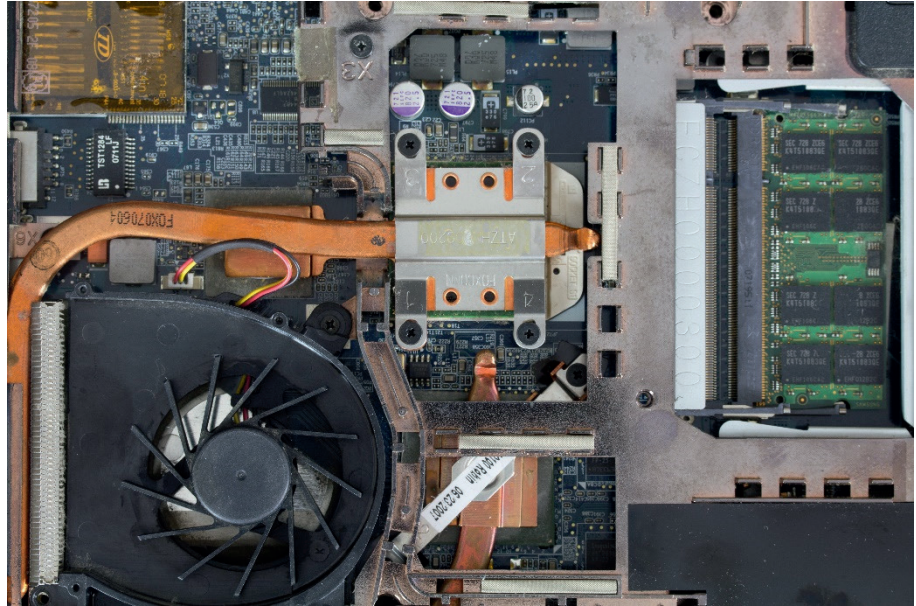
In entsprechender Weise verhält sich dann eine Reparatur - bei Businessgeräten lohnt sich eine Reparatur fast in allen Fällen, bei Konsumergeräten ist bei einem Reparaturpreis von 100 bis 300 € meistens der wirtschaftliche Totalschaden eingetreten. Dann hängt es nur noch von der Einstellung des Eigentümers ab, ob das Gerät doch noch repariert werden soll.

Betrachten wir ein paar Reparaturszenarien. Bevor sie jedoch irgendwelche Schraubarbeiten oder andere Putzarbeiten am Notebook vornehmen, trennen sie das Stromkabel vom Notebook ab und entfernen zur Sicherheit ebenfalls den Akku. Sie brauchen keine Angst haben, dass der Rechner irgendwelche Einstellungen vergisst, da die Ausstattung des Notebooks in einem Festwertspeicher

gespeichert ist, der von einer dafür vorgesehenen Batterie gespeist wird. (Haben sie jedoch das Pech, dass diese Batterie am Ende ihrer Lebensdauer ist, hat er trotzdem alles vergessen, aber dann muss diese Batterie sowieso ersetzt werden!) Als letztes sollten Sie sich erden (fassen sie an einen Metallleiter an, der zum Beispiel mit dem Haus verbunden ist [Heizung, Wasserleitung, etc.]), um eine statische Aufladung abzuleiten.

1. Speichererweiterung:

Entweder gibt es im Handbuch des Gerätes einen Hinweis, wo der Speicher verbaut ist und eventuell auch, welche Schrauben zu lösen sind, um an diese Stelle zu gelangen. Bessere Notebooks sind so konstruiert, dass man lediglich auf der Rückseite eine Klappe mit zwei oder drei Schrauben lösen muss, sodass ein weiterer Speicherriegel eingesetzt werden kann. In komplizierteren Fällen muss das Gehäuse bzw. der



Rückdeckel aufgeschraubt und eine andere Komponente entfernt werden, um an den versteckten Steckplatz eines Speichers zu gelangen. Vergewissern Sie sich in jedem Falle vorher, welche Art von Speicherriegel in ihr Notebook passt. Auch heute gibt es noch Inkompatibilitäten zwischen verschiedenen Firmen bzw. Takten, die verwendet werden dürfen. Achten Sie darauf, dass sie möglichst zwei gleiche Speicherriegel in das Notebook einbauen – zumindest die Taktung des Speichers muss bei allen Speicherriegeln kompatibel sein. Die meisten Notebooks, die eine größere Bauform haben, also 13 Zoll und mehr, haben auch mehr Platz, um weitere Komponenten aufzunehmen. Hier sind meistens zwei Steckplätze für Speicher vorhanden. Die Erfahrung zeigt, dass es in jedem Falle besser ist, beide Steckplätze zu bestücken, als ein Steckplatz mit einem doppelt so großen Speicher zu betreiben. Sind beide Steckplätze mit dem gleichen Speicher (Größe und Taktung) besetzt, kann das System parallel auf beide Speicher zugreifen - ist nur ein Steckplatz besetzt, geht das natürlich nicht. Wenn sie neuen Speicher kaufen, sollten Sie sicher sein, dass er auch wie vorgesehen funktioniert. In den allermeisten Fällen nehmen Online-Anbieter die aufgerissene Verpackung mit dem Speicher nicht mehr zurück. Tipp: Gehen Sie in ein Fachgeschäft und lassen Sie sich den richtigen Speicher geben. Meistens prüft der Verkäufer, welcher Speicherriegel zu verwenden ist. Sollte der neue Speicherriegel dann nicht funktionieren, nimmt der Fachhändler diesen Speicherriegel zurück.

2. Festplatten: Ebenfalls eine etwas leichtere Aufgabe ist der Austausch der Festplatte. Meistens finden Sie am Boden des Gerätes Symbole, die darauf schließen lassen, wo die Festplatte verbaut ist und welche Schrauben sie zu lösen haben. Auch hier ist es sinnvoll, einmal ins Internet zu schauen, wie der Aus- und Einbau vor sich geht. In den einfachen Fällen lösen sie eine Klappe, die mit 2 bis 3 Schrauben festgeschraubt ist. Wenn Sie die Klappe ausgehackt haben, sehen Sie schon die Festplatte mit ihren Anschlüssen. Zu über 90 % ist eine SATA-Platte verbaut. Da dieser Anschlusstyp für Festplatten schon sehr lange besteht und sicherlich auch noch lange Bedeutung haben wird,

haben Sie hier ein unübersehbares Angebot an neuen Festplatten bzw. SSDs, die sie einbauen können. Wenn Sie eine Vorstellung von der Größe der neuen Platte haben, filtern sie bei einem Anbieter nach dieser Größe. Wenn Sie zusätzlich wissen, ob sie eine HDD (Hard Disk Drive) oder eine SSD (Solid State Drive) haben möchten, filtern Sie weiter - in jedem Falle handelt es sich um eine interne Platte. Kurzer Exkurs: eine SSD besteht lediglich aus Speicher, ein drehendes Medium oder Mechanik ist nicht mehr vorhanden und ähnelt dem Arbeitsspeicher des Rechners.

Haben Sie die neue Platte in den Händen, müssen Sie meistens noch eine Befestigungsschraube des alten Laufwerks im Notebook lösen. Als nächstes finden Sie eine Lasche an der alten Platte, an der sie in Gegenrichtung des Anschlusses ziehen müssen, um sie vom Anschluss zu trennen. Allgemein sitzt die alte Platte in einem Rahmen, an dem diese Laschen befestigt sind und für einen besseren Aus- und Einbau verhelfen. Ist keine Lasche zu sehen, müssen Sie vorsichtig, am besten mit einem Werkzeug aus Plastik, zwischen die Anschlussstellen und vorsichtig den (SATA-)Anschluss auseinanderdrücken. An dieser Stelle müssen sie sanftmütig arbeiten, bringen Sie zu viel Kraft auf, könnten Sie die Anschlüsse zerstören. Wenn Sie die alte Platte ausgebaut haben, gehen Sie in umgekehrter Reihenfolge an den Einbau der neuen Platte - verwenden Sie exakt dieselben Handgriffe wie beim Deinstallieren. Nach ihrer Aktion sollte keine Schraube oder der Plattenrahmen übrigbleiben.

Selbstverständlich haben Sie vor der Handlung eine Datensicherung gemacht, die ihre Daten auf ein externes Medium geschrieben hat. Wenn Sie den Rechner jetzt starten, meckert er ein fehlendes Betriebssystem an. Hatten Sie die Absicht nur ein größeres oder schnelleres Medium im Notebook zu haben, müssen Sie vor dem Aus- und Einbau eine exakte Kopie von der alten Festplatte auf die neue Platte kopieren. Dazu benötigen Sie spezielle Programme, die Festplatten bzw. Partitionen kopieren können. Diese Arbeiten sind jedoch mit Überlegung durchzuführen, da Missverständnisse sehr leicht zum Verlust ihrer Benutzerdaten führen können. Bitte beachten Sie: Sie hantieren beim Austausch von Platten am offenen Herzen ihres Notebooks! Und noch einmal der Hinweis: Vergessen Sie niemals ein Back-up oder ein Image von ihrer alten Festplatte auf ein externes Medium zu machen (eine DVD reicht für diesen Zweck nicht aus, da sie eine zu kleine Kapazität hat).

Wenn Sie von ihrer alten Festplatte mit geeigneten Tools ein Image auf ihre neue Platte kopieren konnten, also eine exakte Kopie auf die neue Platte aufgebracht haben, sollten Sie in den Programmen gefragt werden, was mit dem übrigen Speicher passieren soll. Haben Sie von 256 GB auf 512 GB vergrößert, schlagen die meisten Programme vor, den vorhandenen Platz auf der neuen Platte proportional anzupassen an den verwendeten Platz der alten Festplatte. Haben Sie auf der alten Festplatte eine Partition für das Betriebssystem und eine andere Partition für Ihre Benutzerdaten, wäre die erste Alternative, diese beiden Partitionen von 256 auf 512 GB proportional anzupassen. Sie können aber auch auswählen und bestimmen, dass der restliche Platz erst einmal ungenutzt bleiben soll (in einem späteren Schritt, wenn die neue Platte eingebaut ist, können Sie unter Windows den noch freien Platz der angrenzenden Partition hinzufügen - dies sollte dann die Partition mit den Benutzerdaten sein). Wenn der Kopiervorgang beendet ist, können Sie mit dem Umbau beginnen.

Wenn der Umbau fertig ist und sie ihren Rechner wieder einschalten, sollten Sie beim Starten keinen Unterschied merken, lediglich die Geschwindigkeit (wenn sie schnellere Platten eingebaut haben) sollte deutlich zu spüren sein. Wenn sie sich dann wie immer anmelden können, haben Sie die gesamte Aufgabe erfolgreich und gut gelöst.

3. Akku: Eine der leichtesten Übungen ist der Austausch des Akkus. Hierzu müssen Sie sich lediglich ihr Notebookmodell merken und herausfinden welches Modell des Akkus in ihrem Notebook steckt. Wenn sie selbst im Internet nach einem Ersatz suchen, werden Sie schnell feststellen, dass auch hier das Angebot überwältigend ist. Und es gibt für Ihr Notebookmodell nicht den einzigen richtigen Akku, sondern viele und sogar mit verschiedenen Kapazitätswerten. Sehen Sie sich daher die Angaben auf ihrem Akku genau an, und suchen im Internet den zu ihrem Notebookmodell korrekt passenden Akku mit den gleichen Verbrauchswerten vom alten Akku. Achten Sie dabei auch auf die Kompatibilität des Akkus bei der Bauform. Da es keine einheitliche Form für den Stromanschluss innerhalb des Notebooks für eine Modellreihe oder für alle Modelle oder Herstellerübergreifend gibt, kann hier die Stromübergabe (Anschlüsse / Übergabepins aus dem Akku in das Notebook) minimale Änderungen haben und von der Bauform einige Millimeter abweichen und der Akku passt nicht. Verwenden Sie hier keine Kraft, um den Akku einzudrücken und zu verschließen - das endet immer in einem Defekt des Akkus, schlimmstenfalls des Notebooks.

Tipp für eine lange Lebensdauer des Akkus: Akkus lieben es, wenn sie benutzt werden. Sie mögen es gar nicht, wenn sie vollständig entleert werden. Sie sind zufrieden, wenn sie vollständig geladen werden. Am besten fühlen sich Akkus, wenn sie bis zu 20 oder 10 % entleert / benutzt werden und sich dann mit neuem Strom erholen und mindestens 80 bis 90 % des aktuellen Ladezustands erreichen können. Sollten Sie trotzdem einmal feststellen, dass ihr Akku ungewöhnliche Hitze entwickelt, fühlt sich ihr Akku nicht wohl. Dann kann es sein, dass eine von den 4, 6 oder neun Zellen defekt ist und die anderen nun die Arbeit mitverrichten müssen. Die defekte Zelle wird dann heiß, weil sie nicht mehr korrekt arbeitet und die anderen Zellen werden heiß, weil sie die Aufgabe einer oder zweier defekter Zellen übernehmen müssen. Lassen Sie diesen Akku von einem Fachmann prüfen, teilen Sie ihm jedoch mit, wie alt der Akku ist. Akkus verwalten ihre Lebensdauer selbst, aber die ausgelesenen Angaben sind nicht immer korrekt. Es gibt Firmen, die den Akku öffnen und defekte Zellen austauschen. Dieses Angebot reicht nicht ganz an den Preis von neun Akkus heran. Allerdings ist es wahrscheinlich besser, defekte Akkus einem professionellen Entsorger zu übergeben, der die Zellen (eigentlich normale Akkus wie wir sie kennen, Form AA oder AAA, etc.) öffnet und die Materialien für neue Akkuzellen wiederverwendet.

4. Tastatur und Lüfter: Wenn Eigentümer eines Notebooks technisch begabt sind, ist es durchaus möglich, dass auch diese Personen eine Tastatur bzw. den CPU-Lüfter austauschen können. In jedem Falle benötigen Sie mindestens das Handbuch oder technische Dokumente die zeigen, wie Sie an diese Komponenten kommen, sprich: in welcher Reihenfolge sind welche Schrauben zu lösen und welche Komponenten auszubauen. Bei Tastaturen ist es meistens möglich, diese mit Schnappnasen normal von oben zu öffnen. Dazu muss man jedoch wissen, an welcher Stelle diese Schnappnasen sitzen und wie viel Druck man wo in welcher Richtung aufbringen darf, damit die 2., 3. und weitere Schnappsnase ebenfalls ohne abbrechen sich öffnen lässt. Meistens sind diese ‚Tricks‘ im Internet mit kleinen Filmbeiträgen zu sehen. Bei einigen Modellen, die keine große Bauhöhe mehr haben, ist eine kleine hinterhältige Schraube zu lösen, die die Tastatur von hinten am Boden festhält, oder man muss erst einen Deckel öffnen, um die versteckte Schraube für die Tastatur lösen zu können. In wenigen Fällen muss von hinten das ganze Innenleben des Notebooks ausgebaut werden, um an die Tastatur zu gelangen. In diesen Fällen muss sich der Eigentümer sehr gut mit der Technik auskennen oder aufgeben und zu einem Fachgeschäft gehen.

Auch hier ein Tipp: Bevor sie anfangen auszubauen, sollten Sie die korrekten Ersatzteile schon vor sich haben. Eine Ersatztastatur ist meistens langfristig vom Hersteller bzw. deren Partnern zu kau-

fen; Gleichfalls gibt es im Internet viele Anbieter - auch bei eBay, wo viele gewerbliche Wiederverwendungsbetriebe ihre Angebote platzieren - bei denen man die passenden Tastaturen kaufen kann. Selbstverständlich gibt es auch bei Tastaturen diverse Varianten und Farben, auch die Nationalität spielt hier eine entscheidende Rolle, da oft preisgünstige anderssprachige Tastaturen (z.B. englische, russisch, etc.) angeboten werden und die Überraschung nach der Lieferung groß ist, dass die Umlaute Ä, Ü, Ö fehlen.

Genauso kompliziert kann es sein, den CPU-Lüfter auszutauschen. Bitte informieren Sie sich auch hier in den technischen Dokumenten des Gerätes oder im Internet - die Tipps sind in jedem Falle hilfreich und geben Ihnen Ideen, wie die einzelnen Komponenten zusammengesraubt oder aufgesteckt sind. Bei den bisherigen Arbeiten - Akku tauschen, Tastatur tauschen, Platte tauschen oder Speicher aufrüsten, haben sie teilweise tief ins System eingegriffen. Wenn Sie jetzt den CPU-Lüfter austauschen möchten, arbeiten Sie hier genau wie mit der Festplatte bei der CPU am technischen Herzen ihres Gerätes. Sie sollten hier alle Vorsichtsmaßnahmen vorher treffen, die oben schon angesprochen worden: Back-up, Image, richtiges Werkzeug, Vermeidung statischer Ladungen und Kenntnisse über die Technik. Fast alle Notebooks haben technische Schaltpläne, wo gezeigt wird, an welcher Stelle die CPU verbaut ist. Demontieren Sie den Bodendeckel und alle Komponenten, bis Sie an den Lüfter und die CPU gelangt sind - natürlich merken Sie sich hier wieder die Reihenfolge. Prüfen Sie, wie der CPU-Lüfter befestigt ist! Meistens ist er auf die CPU geklemmt bzw. mit Federn am Mainboard befestigt und wird auf die CPU gedrückt. Sie müssen diese Federbefestigung vorsichtig lösen. In den meisten Fällen ist der CPU-Lüfter entweder mit Wärmeleitpaste verbunden oder wurde mit einer Wärmeleitklebefolie auf die CPU geklebt. Bei der Wärmeleitpaste kann es vorkommen, dass diese so getrocknet ist, dass sie beim Lockern zerbröseln. Da die Wärmeleitpaste normalerweise nicht aushärtet, können Sie den CPU-Lüfter abnehmen, indem sie leichte Drehbewegungen durchführen, um die Adhäsionskraft der Wärmeleitpaste / -folie zu überwinden und den CPU-Lüfter abnehmen (ähnlich wie bei zwei Magneten, die sie trennen wollen). Sie müssen dabei besonders vorsichtig Kraft aufwenden, damit sie die CPU nicht aus ihrer Verankerung ziehen oder ihr ein oder mehrere Beinchen abreißen! Versuchen Sie nicht, mit Lösungsmitteln die Wärmeleitpaste lockern zu wollen - Flüssigkeiten haben im Innern eines Notebooks nichts zu suchen und richten mehr Schaden an, als dass sie überhaupt etwas nutzen! Wenn Sie sich nicht sicher sind, ob sie den CPU-Lüfter gelockert haben, stoppen Sie Ihre Aktion und bringen alle Teile inklusive der letzten Schraube einem Fachmann, der die Arbeit für sie übernimmt. Haben Sie jedoch den CPU-Lüfter entfernen können, und sie wollen ihn wieder benutzen, säubern Sie den Lüfter restlos von der noch anhaftenden Wärmeleitpaste oder der Folie. Gleiches müssen sie vorsichtig mit der CPU durchführen, wischen Sie die Reste der Wärmeleitpaste vorsichtig ab.

Ist Ihnen vertrocknete Wärmeleitpaste ins Gerät gefallen, könnten sie mit einem geeigneten Staubsauger das Gerät aussaugen, ABER: dabei müssen sie extrem vorsichtig sein, dass sie sich nicht statisch aufladen oder der Saugrüssel sich an einzelnen Komponenten festhackt und diese dann durch eine kräftige Saugbewegung abreißen - diese Methode ist nur etwas für Experten und Fachleute, die einen geeigneten Sauger haben, der lediglich Staub und lose Teile aus versteckten Ecken der Platine und dem Gehäuse saugt - meistens ist ein Bürstenaufsatz standardmäßig aufgesetzt. Das Gegenteil von Saugen ist Pusten. Die zweitbeste Alternative ist, mit technischer Druckluft das Gerät vorsichtig auszublasen. Achten Sie hierbei darauf, den Sie aufgewirbelten Staub nicht einatmen und der Luftstrahl nicht in andere Lüfter bläst, sodass sie sich anfangen zu drehen. Wenn sie bemerken, dass durch ihren Luftstrom sich ein Lüfter zu drehen beginnt, halten Sie ihn mit dem Finger fest, damit er sich nicht mehr dreht. Warum? Der Lüfter produziert durch das Drehen einen Strom, der

dann allerdings im Gerät in die andere Richtung fließt und somit Schaden verursachen könnte. Haben Sie einen neuen Lüfter oder Sie haben den alten Lüfter gesäubert, müssen Sie am besten eine Wärmeleitpaste auf beide Seiten auftragen: einmal auf den Rücken der CPU und andererseits auf die Fläche des Lüfters. Achten Sie darauf, dass die gesamte Fläche der CPU mit der Wärmeleitpaste versorgt ist, verwenden Sie aber die Wärmeleitpaste sparsam. Die Wärmeleitpaste ist nur dafür gedacht, die aufeinander liegenden Flächen der CPU und des Lüfters besser zu verbinden und nicht dafür vorgesehen, dass die Wärme der CPU über die Wärmeleitpaste auf den Lüfter übertragen wird. Die Wärmeleitpaste ist dafür gedacht, dass an jeder Stelle der Flächen der Wärmeaustausch optimal vorgenommen werden kann - dafür reichen auch geringe Mengen, allerdings auf der gesamten Fläche! Nach unserer Erfahrung sollten Sie eine Verwendung einer Wärmeleitfolie vermeiden.

Nun kommt die Aufgabe, den Lüfter wieder vorsichtig in seine federnde Verankerung zu bringen. Dabei dürfen sie auf keine Ecke der CPU größeren Druck ausüben, da sie ansonsten die CPU brechen könnten oder andere Kontakte abreißen. Wenn sie auch hierbei Bedenken haben, stoppen sie ihre Aktionen und gehen mit dem gesamten Konglomerat zu einem Reparaturbetrieb. Sollten Sie diese Hürde aber überwunden haben, müssen Sie nur noch die restlichen Komponenten in rückwärtige Reihenfolge montieren...!

5. TFT: Sollten Sie die verwegene Idee haben, ein TFT selbst auszutauschen, beherzigen sie auch hier alle vorher genannten Ratschläge. Wer das selbst machen möchte, dem brauche ich hier keine weiteren Ratschläge oder Vorgaben mehr zu machen - sollte Ihr Vorhaben gelingen, melden Sie sich gern bei mir als Techniker...! Ansonsten kann ich nur raten, diese hochkomplizierten arbeiten einem Reparatuer oder der Fachwerkstatt zu überlassen. Hier ist nicht nur im Inneren des Notebooks zu arbeiten, sondern Sie müssen hier auch die Klappmechanik öffnen und schließen, was nicht nur weiteres Werkzeug beansprucht, sondern in einer Techniker Ausbildung die höchste Stufe ist.

PCs

Schauen wir auf die ‚alten‘ PCs. Von allen genannten Geräten haben die PCs diejenigen Eigentümer, die meisten an ihnen herumbauen. Fast alle PCs haben Gehäuse der Art, dass eingeplant ist, Geräte und Komponenten zusätzlich zu installieren, auszutauschen oder den PCs neue Fähigkeiten (Steckplätze) hinzufügen zu können. So kommt es nicht selten vor, dass wir PCs sehen, wo das Gehäuse und das Netzteil ein ansehnliches Alter haben, alle anderen Komponenten jedoch wesentlich neuer sind (Mainboard, Platte, etc.). Hin und wieder trifft es aber auch ein Netzteil, das nicht mehr genug Strom liefert, um die stärkste Grafikkarte für bestimmte Spiele zu versorgen. Grundsätzlich kann am PC jede Komponente ausgetauscht werden. Der Eigentümer muss dafür nicht unbedingt große Erfahrung und Fingerspitzengefühl haben, wie bei einem Notebook. Teilweise reicht es aus, dass man weiß, wie das Gehäuse zu öffnen ist, eine Festplatte eingeschoben und festgeschraubt und wie sie am Mainboard angeschlossen wird.

Im Folgenden besprechen wir einzelne Komponenten separat.

1. (Arbeits)Speicher: Als Abkürzung auch RAM genannt (Random Access Memory). Schauen Sie in ihre technischen Unterlagen oder ins Handbuch, welchen Speicher ihr Rechner trägt. Meistens ist es hilfreicher, das Gehäuse zu öffnen, einen Riegel des vorhandenen Speichers herauszunehmen und sich die dort aufgedruckten Daten abzuschreiben. Wie bei den Notebooks schon erwähnt, sollten Sie Speicherriegel nur von gleicher Bauart, möglichst sogar vom gleichen Hersteller, verwenden.

Ein Mischen der Hersteller gelingt meistens, aber nicht immer und ein Zurücksenden der Speicherriegel ist meistens ausgeschlossen. Sind die Preise für Speicher günstig, schauen Sie in den technischen Dokumenten nach, ob schnellere Speicherriegel verwendet werden können. Dann prüfen Sie, wie viel Steckplätze Sie für Speicherriegel haben - meistens sind das zwei oder vier Steckplätze (entspricht 1 oder 2 Speicherbänke). Wenn Sie Ihr Gerät mit 16 GB ausrüsten möchten, haben sie verschiedene Alternativen: a) 1x 16 GB, b) 2x 8 GB, c) 4x 4 GB. Unter der Bedingung, dass alle Speicherriegel die gleiche technische Ausstattung haben (GHz), und dass das Mainboard parallelen Zugriff auf Speicher erlaubt, ist die Alternative b) und c) am günstigsten. Meistens müssen die Speicherbänke, die immer paarweise vorhanden sind, auch paarweise bestückt werden. Sie müssen in ihre technischen Dokumente schauen, welche Speichergrößen das Mainboard unterstützt - dort ist in Tabellen angegeben, wie Speicherriegel auf die Speicherbänke verteilt werden dürfen. In aller Regel ist die Bestückung einer Hälfte einer Speicherbank nicht gestattet oder führt zu Fehlfunktionen bzw. es wird nur die Hälfte der Kapazität des Speicherriegels erkannt. Sollte nur eine Speicherbank (zwei Slots) vorhanden sein, sollte diese Randbegrenzung allerdings fehlen, sodass es möglich ist, die zwei Slots einer Speicherbank auch nur mit einem Speicherriegel bestücken zu können. Möchten Sie jedoch in der nächsten Zukunft einen weiteren Ausbau des Arbeitsspeichers vornehmen, dann sollten Sie die zukünftig zu verwendenden Spezifikationen der Speicherriegel kaufen (GHz & Größe).

2. Laufwerke: Unter Laufwerken sind Festplatten (HDD, SSD), CDs oder DVDs, evtl. Bandgeräte gemeint. Bandgeräte kommen nur noch in den seltensten Fällen vor, waren aber in der Vergangenheit extrem zuverlässige Speichergeräte, die meistens für Backups verwendet wurden. Heutzutage kommen sie, wenn überhaupt, nur noch in Rechenzentren zum Einsatz. Auch die Zeit der CDs ist vorüber. Diese digitalen Aufzeichnungssysteme hatten eine wesentlich größere Kapazität, als die damals noch vorhandenen Disketten. Der nächste technische Schritt betraf die DVDs. Auch diese Geräte sind auf dem Rückzug. Allerdings werden noch viele Softwarepakete auf CDs bzw. DVDs angeboten, um sie installieren zu können - als Back-up-Medium taugen sie nichts mehr. Auch die letzten Softwareversionen von z.B. MS-Office 2019 kann man noch auf DVD-Installationsmedien erhalten. Möchte man ein DVD-Laufwerk in einen PC einbauen, gibt es hier kaum Probleme und sollte für jeden, der einen Schraubenzieher beherrschen kann, möglich sein. Das Gehäuse ist zu öffnen und ein freier Einschubplatz zu identifizieren. In den vorgesehenen Einschüben können Sie das DVD-Laufwerk hineinschieben und Stromkabel mit dem Netzgerät, sowie Datenkabel am Mainboard verbinden. Als ‚Schnittstelle‘ sprich ‚Datenkabel‘ wird das SATA-Format verwendet. Dieses Kabel hat Anschlüsse, wobei die Form des Anschlusses sehr lange Gültigkeit hat, jedoch die Geschwindigkeit von 3 GBit auf 6 GBit erhöht wurde. Die Anzahl der SATA-Anschlüsse auf dem Mainboard haben sich seit Jahren erhöht - zum Teil findet man 8 bis 10 SATA-Anschlüsse auf einem Mainboard. Das DVD-Laufwerk ist in der Lage, auch CDs lesen zu können. Ist das DVD-Laufwerk ein ‚RW‘, dann kann es Medien auch beschreiben. Wenn das Laufwerk einen separaten Audio-Anschluss hat, prüfen Sie auf dem Mainboard, ob auch dort eine Anschlussmöglichkeit für ein CD-/DVD-Laufwerk vorhanden ist. Dieses Verbindungskabel wird meistens mitgeliefert.

Klassische Festplatten gibt es in zwei Größen: 3,5“ oder 2,5“ und enthalten ein drehendes Medium (Plattenstapel). Eine 3,5“ Platte kann prinzipiell mehr Daten speichern als eine 2,5“ Platte. Ein 3,5“ Laufwerk passt sofort in einen Einschub eines PCs. Prüfen Sie, ob es einen Mechanismus gibt, der selbstständig das Laufwerk arretiert, ansonsten wird das Laufwerk mit vier Schrauben am Rahmen verschraubt. Vergessen Sie nicht die zwei Kabel anzuschließen. Ein 2,5“-Medium ist wesentlich kleiner und leichter. Hier gibt es Einschubvorrichtungen für die kleinen Laufwerke, um sie in einem normalen Schacht verankern zu können. Entweder werden 2,5“-Medien mechanisch in der Einschubvorrichtung arretiert oder ebenfalls festgeschraubt, dann muss die Einschubvorrichtung in einen

Einschub des PCs geschoben und verankert werden. Als letztes sind die beiden Kabel wieder anzuschließen.

Bei den Festplatten setzt sich eine neue Form der Speicherung durch: SSD. Wie bei den Notebooks schon erwähnt, bestehen diese Laufwerke nur noch aus nicht-bewegenden Teilen - spezielle Speicherchips. Da diese Medien wesentlich schneller arbeiten als die bisherigen HDDs, können wir eine Umrüstung von PCs jederzeit empfehlen. Zumindest das Betriebssystem sollte auf einer SSD installiert sein - ein Medium mit einer Größe von 256 GB reicht hier vollständig aus. Die Benutzerdaten können auf einer zweiten oder weiteren Festplatte als HDD oder ebenfalls SSD gespeichert werden. In einem normalen PC-Gehäuse haben mindestens 3 bis 5 Laufwerke Platz. Tipp: Sollte kein Einschubschacht mehr verfügbar sein, jedoch weitere SATA-Anschlüsse, so kann man eine SSD vorsichtig in den Rechner legen, ohne sie festzuschreiben - sie sollte dann behutsam festgeklemmt oder festgeklebt werden, damit sie sich beim PC-Transport nicht im Gehäuse bewegen kann. Wenn Sie eine Festplatte mit Betriebssystem gegen eine SSD austauschen, achten Sie darauf, dass sie zuerst eine Kopie Ihres Betriebssystems von der alten Platte auf die neue SSD übertragen. Sie sollten dann alle Festplatten vom Mainboard trennen und nur die neue SSD anschließen, die das Betriebssystem kopiert bekommen hat. Wenn Sie den Rechner starten, sollte er sich so verhalten, wie Sie es von vorher kennen. Nun können Sie die anderen Platten wieder anschließen (natürlich im Ausgeschalteten Zustand). Möglicherweise erkennt das Betriebssystem auf der neuen SSD, dass ein weiteres Betriebssystem (auf der alten Platte) vorhanden ist. Entweder bietet Ihnen das Betriebssystem ein Menü an, in dem Sie auswählen können, welches Betriebssystem sie starten möchten. Merken Sie sich, welchen Eintrag Sie aktiviert haben, damit Sie auch das neue Betriebssystem auf der neuen SSD beim Start erwischen. Selbstverständlich gibt es Tools, die zumindest die Bezeichnung eines Mediums ändern können, damit beim Start keine doppelten Namen und Mehrdeutigkeiten auftreten. Sollte Ihnen auf einem Laufwerk wieder der Platz ausgehen, oder Sie benötigen ein separates Back-up Laufwerk, können Sie eine weitere HDD oder SSD in einen freien Einschubschacht einbauen. In den Systemeinstellungen finden Sie ein Menü, mit dem Sie die neue Platte initialisieren, um sie verwendbar zu machen.

3. Steckkarten: Ein eindeutiger Unterschied zum Notebook ist beim PC sofort zu erkennen: die freien Steckplätze für Erweiterungskarten. Das Angebot von Steckkarten ist dermaßen überwältigend und kann einen PC unglaublich viele neue Funktionalität hinzufügen, dass hier nur einige genannt werden können: Grafikkarten, Schnittstellenkarten, Ethernetkarten, Vervielfältigung der USB Schnittstellen durch USB-Karten, usw. Wenn Sie vorhaben, eine neue schnellere Grafikkarte einzubauen, öffnen Sie zuerst Ihren PC und schauen sich die dafür vorgesehenen Steckplätze an. Möglicherweise erkennen sie zwei oder drei verschiedene Arten von Steckplätzen. Die momentan noch gültige interne Schnittstellentechnik ist PCI, AGP, PCIe. Nehmen Sie die technische Dokumentation zu Ihrem PC zur Hand, um die richtige Schnittstellentechnik zu erkennen. Danach müssen Sie Ihre neue Grafikkarte kaufen. Die drei Slotarten der Schnittstellen sind inkompatibel und passen mechanisch nicht. Gleichfalls stehen die verschiedenen Schnittstellen für verschiedene Geschwindigkeiten der Grafik - beachten Sie auch hier, welche Schnittstelle Sie am besten verwenden, da Sie eventuell mehrere Slotarten auf ihrem Mainboard haben. Gleichfalls prüfen Sie, ob Sie die externen Schnittstellen der Grafikkarte mit ihrem vorhandenen Monitor anschließen können. Hier gibt es meistens die Schnittstellentechnik HDMI, DP und die älteren DVI oder gar VGA. Alle Schnittstellen sind mechanisch anders konstruiert - keine passt in den jeweils anderen Anschluss. Hat man an seinem Monitor nicht die Schnittstelle, die die Grafikkarte liefert, kann man Adapter an die jeweiligen Kabel anschließen. Dabei sollten Sie auf die Qualität der Adapter achten, da diese einen Signalverlust bedeuten. Wenn ihr Monitor eine sehr hohe Auflösung liefern kann (UHD 4K bis WXGA), sollte auch

ihre neue Grafikkarte in der Lage sein, diese ohne Emulation liefern zu können. Grafikkarten haben ein äußerst unterschiedliches Leistungsvermögen: Sie können günstige Grafikkarten kaufen, die normalen Büroanwendungen genügen, wenn Sie aber High-Speed-Spiele spielen möchten, muss die Grafikkarte eine hohe Qualität haben und damit meistens einen hohen Preis ab ca. 250 € bis weit über 1.000 €. Allerdings muss auch die Technik des PCs dafür ausgerüstet sein, die da eine sehr leistungsfähige Grafikkarte sehr viel Strom verbraucht (evtl. sogar mehr, als im normalen Betrieb insgesamt anfällt), auf der anderen Seite muss der Monitor die schnellen Schaltzeiten bei der Darstellung anzeigen können (1 bis 3 ms). Spiele-Grafikkarten haben selbst aktive Kühlsysteme, um die enorme Wärme nach außen und nicht ins PC-Gehäuse ableiten zu können. Wenn Sie Ihre Grafikkarte gefunden haben, schrauben Sie das Gehäuse des PCs auf, stecken die Grafikkarte in den passenden Slot - möglicherweise müssen Sie ein zusätzliches Stromkabel an die Grafikkarte anschließen, was vom Netzgerät kommt; Das sollten Sie nicht vergessen, da die Grafikkarte sonst den Strom über das Mainboard zieht und auf längere Sicht damit das Mainboard schädigen kann, bzw. die Grafikkarte meldet einen Fehler, wenn der zusätzliche Stromanschluss nicht angeschlossen ist. Wenn Sie Ihren PC verschlossen und das Stromkabel wieder angeschlossen haben, sollte die neue Grafikkarte in einen Standardmodus starten. Möglicherweise werden Ihnen Treiber auf einer CD / DVD mitgeliefert - suchen Sie jedoch eher im Internet beim Hersteller nach den neuesten Treibern, da diese fast immer neuer sind, als auf der CD / DVD.

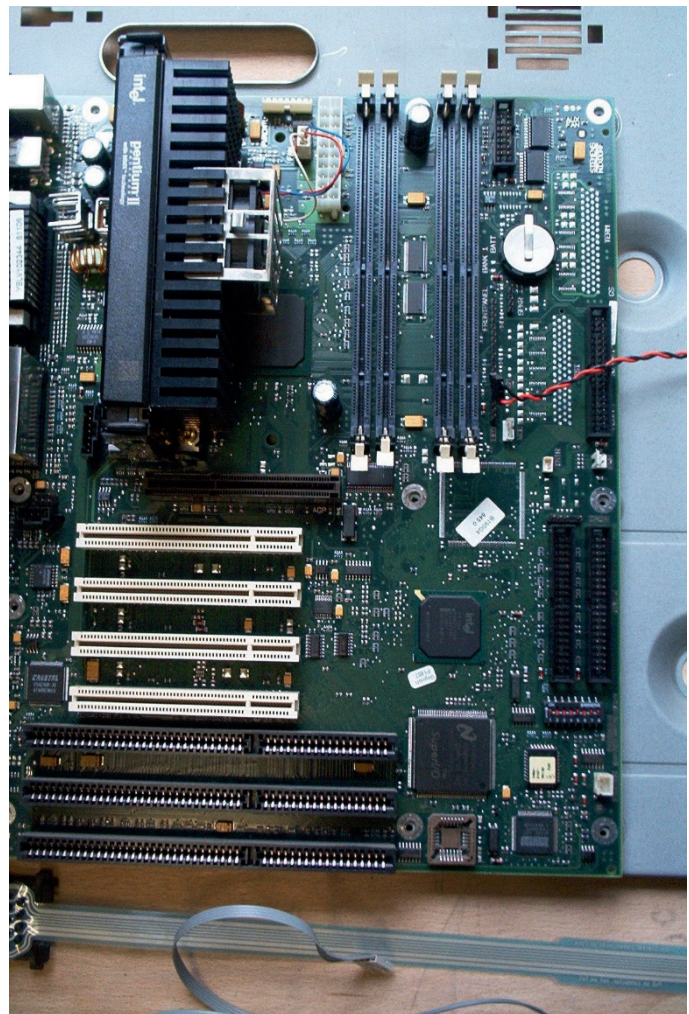
Vielfältige Varianten an Steckkarten gibt es auch für Schnittstellen. Gelegentlich benötigt ein Benutzer eine serielle Schnittstelle und muss dazu eine Steckkarte einstecken, da die PCs bzw. Mainboards diese Schnittstellenart schon lange nicht mehr anbieten. Auch eher exotische Schnittstellen (z.B. FireWire) gehört zu der Art der Schnittstellen, die ein normales Mainboard von sich aus nicht unterstützt. Je nachdem was ihnen fehlt, können Sie im Internet recherchieren und bei einem namhaften Anbieter ihre Schnittstellenkarte kaufen. Wenn Sie die Karte in den Händen halten, verfahren sie wie bei dem Einsetzen einer Grafikkarte: Suchen Sie den entsprechenden Slot, entfernen ein Slotblech und setzen die Karte ein. Natürlich haben Sie vor den Arbeiten den Rechner ausgeschaltet und vom Stromnetz getrennt. Verschließen Sie Ihren PC wieder, starten ihn und warten ab, was ihr Betriebssystem meldet. In vielen Fällen ist ein Windows- oder Linux-System in der Lage, die entsprechenden und richtigen Treiber nachzuladen. Sollte das nicht gelingen oder sie möchten andere Treiber laden, recherchieren sie wieder auf der Internetseite beim Hersteller und laden sich die richtigen Treiber herunter, um diese zu installieren - dann sollten alle Funktionalitäten der Steckkarte ausgenutzt werden können.

In ähnlicher Weise gehen Sie vor, wenn Sie eine Ethernet-Schnittstellenkarte benötigen. Die meisten Mainboards haben einen Ethernetanschluss auf dem Mainboard, der mit den anderen Anschlüssen herausgeführt ist. Benötigen Sie zwei oder weitere Ethernet-Anschlüsse, wird eine Ethernet-Steckkarte mit ein, zwei oder vier Anschlüssen in ein PC-Slot gesteckt. Mit den entsprechenden Treibern kann ihr Betriebssystem dann mehr als einen Ethernet-Anschluss steuern, praktisch, wenn man verschiedene Internetleitungen mit eigenen IP-Adressen hat, oder zwischen Eingangsleitung und Ausgangsleitung unterscheiden möchte.

Auf den meisten Mainboards sind Stecker angebracht, die mehrere USB Anschlüsse zur Verfügung stellen. Einige der Anschlüsse sind intern im PC verkabelt und nach außen geführt. Weitere auf dem Mainboard enthaltenen Anschlüsse, können zusätzlich angeschlossen werden, müssen jedoch noch nach außen geführt werden. Dafür gibt es z.B. Einschübe, die zusätzlich am PC montiert werden können. Sind auf dem Mainboard keine weiteren Anschlüsse für USB enthalten, verwenden Sie eine Steckkarte, die diese Möglichkeit zur Verfügung stellt. Vorteil dieser Steckkarte ist, dass die USB-

Anschlüsse gleichzeitig aus dem Rechner geführt werden. Meistens sind sofort zwei oder vier Anschlüsse zusätzlich vorhanden. Ein weiterer Vorteil dieser Steckkarten ist, dass die Stromversorgung für die USB-Unterstützung vom Mainboard geliefert wird. Eine Alternative wäre es, wenn man einen sogenannten USB-Hub an einen USB-Anschluss ansteckt, der die Anzahl der Anschlüsse eventuell verdoppelt oder vervierfacht. Dabei muss die Stromversorgung über einen USB-Anschluss geliefert werden, was bedeutet, dass nur geringe Stromverbraucher angeschlossen werden sollten. Alternativ könnte ein ‚aktiver‘ USB-Hub verwendet werden, der mit einem Transformator eine eigene Stromversorgung hat. Sie sollten darauf achten, welche Geräte Sie an welchen USB-Anschluss anschließen, damit Kommunikationsfehler zwischen den Geräten vermieden werden oder die Funktionalität einiger Geräte sichergestellt ist. Wenn Sie eine USB-Steckkarte kaufen, achten Sie darauf, mit welchem ‚Protokoll‘ sie betrieben werden kann: USB (2.0) / 3.0 / 3.1 / 3.2 - je nachdem richtet sich die Übertragungsgeschwindigkeit danach. Allerdings muss auch die Gegenstelle (Notebook, PC) diese Geschwindigkeit verdauen, damit man in den Genuss der Schnelligkeit kommt.

4. Mainboards: Wer den großen Schritt liebt, kann auch sein Mainboard austauschen. Diese Angelegenheit kann jedoch etwas teurer werden. So wird nicht nur das Mainboard neu gekauft, meistens muss auch eine neue CPU und ein neuer Arbeitsspeicher angeschafft werden. Sind ältere Schnittstellen auf dem neuen Mainboard nicht mehr vorhanden, wo Geräte aber noch verwendet werden, muss nach Adaptern Ausschau gehalten werden. Sie haben jedoch die Chance, das alte Mainboard mitsamt der CPU und dem Speicher jemanden anderen zu verkaufen, dadurch erhalten Sie das Startgeld für ihren Umbau. Bevor sie alles auseinander bauen und verkaufen, nutzen Sie Ihren alten Rechner, um alle neuen Komponenten, die Sie haben möchten, auszusuchen und zu kaufen. Erst wenn sie überzeugt sind, dass alle Komponenten miteinander laufen werden, sollten Sie Ihren alten Rechner umrüsten. Fangen Sie damit an, das Gehäuse zu öffnen und alle Kabel vom Mainboard zu lösen. Gehen Sie dabei schrittweise vor - kleben Sie an jedes Kabel eine Notiz, zu welchem Gerät und an welchen Stecker es gehört. Haben Sie alle



Kabel von den Steckern gelöst, müssen Sie prüfen ob, Sie an alle Schrauben kommen, womit das Mainboard befestigt wurde. Bei ATX-Boards müssen Sie meistens 9 bis 10 Schrauben lösen, bei größeren Mainboards können es auch mehr Schrauben sein. Geben Sie dabei acht, dass sie die Abstandhalter nicht mit abschrauben, die benötigen Sie noch für ihr neues Mainboard. Sollten Sie das Mainboard inklusive CPU und Speicher nicht aus ihrem Gehäuse heben können, müssen sie störende Käfige oder Kabel weiter demontieren. Merken Sie sich dabei immer, wo die Komponente war und

wieviele Schrauben wo verschraubt waren. Haben Sie es geschafft, das alte Mainboard zu demontieren, geht es in umgekehrter Reihenfolge an die Montage des neuen Mainboards. Um es wieder an die Rückseite des PC-Gehäuses bzw. -Käfigs zu schrauben, sehen Sie sich an, wieviel Montagematerial dem neuen Mainboard beigelegt ist. Prüfen Sie, an welcher Stelle des neuen Mainboards die Stege / Abstandhalter montiert werden müssen. Gehen Sie dabei möglichst gegenüberliegend vor und ziehen Sie die Schrauben nicht sofort vollkommen fest. Wenn Sie alle Stege korrekt montiert haben und das Mainboard mit den losen Schrauben richtig sitzt, können Sie die Schrauben möglichst gegenüberliegend festziehen. Installieren Sie nun die neue CPU auf dem Sockel. Meistens ist dazu ein Hebel nach vorne zu drücken, um die Beinchen der CPU fest zu klemmen. Sehen Sie sich nun den CPU-Lüfter an, legen ihn einmal an die CPU, um zu sehen, ob der Haltemechanismus korrekt funktioniert. Wenn das der Fall ist, können Sie die Wärmeleitpaste, wie oben beschrieben, auf die CPU und die Kühlfläche des Lüfters auftragen und den Kühlkörper über der CPU auf dem Mainboard befestigen. Nun stecken Sie ihren Arbeitsspeicher in die dafür vorgesehenen Slots, wobei Sie sich vorher informiert haben, wie die Verteilung der Speicherriegel am besten funktioniert. Als Letztes kommt die Sisyphusarbeit und Sie müssen alle Kabel von allen Geräten wieder auf dem Mainboard anschließen. Dabei ist natürlich die technische Information des Mainboards wesentlich hilfreich, da dort alle Schnittstellen beschrieben sind. Wenn Sie sicher sind, alles korrekt angeschlossen zu haben, schließen Sie den Rechner wieder ans Stromnetz an und starten ihn. Ein vertrautes ‚Piep‘ sollte zur Freude beitragen. Startet der Rechner wie erwartet, prüfen Sie, ob alle Komponenten korrekt arbeiten, Kontroll-LEDs leuchten und alle Laufwerke richtig funktionieren.

Wartungsarbeiten

In vielen Fällen merkt man erst spät, dass der PC langsam wird. Das kann mehrere Ursachen haben. Als erstes sind Sie von anderen Rechnern z.B. im Büro gewohnt, dass Programme schneller starten oder reagieren. Sitzen Sie wieder an ihrem Rechner (PC oder Notebook) zu Hause, kommt ihnen die Arbeit damit extrem langsam vor. Das kann daran liegen, dass ihr Rechner nach und nach eingestaubt ist und durch die nicht mehr abgeführte Wärme die CPU die Taktfrequenz von sich aus absenkt, damit sich die CPU nicht selbst überhitzt. Sie sollten regelmäßig ihren Rechner (PC) öffnen und die Staubmatten oder den an den Komponenten aufliegenden Staub entfernen (ACHTUNG: Staub leitet Strom – er kann zu Kurzschlüssen führen). Wie oben schon beschrieben, können Sie das mit einem Staubsauger machen, der einen Bürstenaufsatz hat, auf kleinster Stufe betrieben wird und Sie sich statisch entladen haben. Natürlich ist bei dieser Arbeit höchste Konzentration geboten, um keine Sekundärschäden zu verursachen. Am besten lassen Sie das einen Fachmann bzw. Reparatereure ausführen - diese Leute sind trainiert, diese Arbeiten ohne Nebenwirkungen durchzuführen. Nur wenige Reparatereure prüfen dann, ob die Wärmeleitpaste auf der CPU bzw. den Grafikchips noch ihre Funktion 100-prozentig erfüllt - bei einem Reinigungsauftrag wird die Wärmeleitpaste erneuert und damit die optimale Betriebstemperatur der CPU / Grafikchips sichergestellt.

Eine weitere Beschleunigung kann der Rechner erfahren, wenn alte Laufwerke (HDD) gegen neue Platten (SSD) ausgetauscht werden. Dieser Vorgang erfordert mehrere Handgriffe, benötigt aber nur wenig Erfahrung. Der Geschwindigkeitsvorteil von HDD zu SSD liegt bei 1:5 bis 1:10 oder mehr - wenn die SSDs über andere Schnittstellen betrieben werden (Notebooks mit M.2-Schnittstelle), kann eine Geschwindigkeitssteigerung von 1:20 bis 1:30 erreicht werden. Hier macht es den größten Sinn, die Betriebssystempartition bzw. -platte mit einer SSD zu ersetzen. Wer für seine Benutzerdaten eine andere Platte eingebaut hat, kann auch hier eine SSD verwenden, und man wird deutlich eine Geschwindigkeitserhöhung bemerken.

Wenn keine neue Platte angeschafft wird, und auf einer HDD das Betriebssystem läuft, können Optimierungsprogramme manchmal auch Wunder bewirken. Die Optimierungsprogramme sortieren die Dateifragmente eines Programmes oder anderer Dateien zusammen und speichern sie in einem Stück. Die HDD muss nun nur noch einmal ansetzen, um die gesamte Datei auf einen Schlag zu lesen, ohne neu positionieren zu müssen. Das beschleunigt Programmstarts erheblich und auch Datendateien können schneller gelesen und bearbeitet werden. Bei SSDs wird normalerweise diese Datenoptimierung nicht vorgenommen - alle Speicherzellen sind in gleicher Weise gleich schnell zu erreichen. Allerdings bieten die neuesten SSD selbst Optimierungsprogramme, um fragmentierte Dateien zusammen in einem Stück abzuspeichern. Den Vorgang des optimieren der Platten, zumindest der HDDs, sollte je nach Benutzung des Rechners gelegentlich durchgeführt werden. Meistens haben diese Programme einen vorgeschalteten Analysemechanismus, der Ihnen Auskunft gibt, wieviel Dateien fragmentiert und wieviel freie Fragmente auf der Festplatte vorhanden sind. Sollten sich die Fragmente auf einer Betriebssystemplatte über den Bereich von 2.000 bis 4.000 bewegen, sollte eine Optimierung in Gang gesetzt werden. Einige Programme nehmen eine solche Entscheidung ab, indem sie eigene Grenzen definieren, bei denen verschiedene Optimierungen vorgeschlagen werden.

Ein weiterer Flaschenhals kann in der Grafikkarte bestehen. Vorausgesetzt, das Mainboard hat entsprechende schnelle Slots zur Verfügung, kann eine neue schnellere Grafikkarte verwendet werden, die eine schnellere Bildanzeige ermöglicht - auch diesen Effekt kann man deutlich spüren - selbstverständlich auch messen.

Die Meinung, dass zu viele Programme installiert seien und daher der Rechner langsam wird, ist ein weitverbreiteter Trugschluss. Es spielt keine Rolle, wie viele Programme installiert sind oder wie viele Benutzerdateien Sie gespeichert haben. Es spielt lediglich eine Rolle, wie oft Sie Windows upgedatet haben. Sollten Sie auf Ihrem Rechner auf einer HDD einmal Windows 7 frisch installiert und alle weiteren Windows-Versionen aktualisiert (Updates) und Sie jetzt Windows 10 in der neuesten Version installiert haben, merkt sich das Betriebssystem in der letzten Version viele Einstellungen von den älteren Windows-Versionen. Wenn Sie in das Windowsverzeichnis schauen, stellen Sie fest, dass ihr Betriebssystem durchaus 80 bis 200 GB groß geworden ist. An dieser Stelle sollten Sie überlegen, auf einer neuen Platte (SSD) das aktuellste Betriebssystem neu zu installieren. Natürlich müssen Sie dann auch alle Programme, die Sie benötigen, erneut installieren und die Lizenzen wieder eingeben. Wenn Sie sich dazu entschließen, macht es Sinn, teuer lizenzierte Programme vorher zu deinstallieren, damit die Lizenz wieder frei wird. Sind alle derartigen Programme deinstalliert, können Sie die neue SSD einbauen, und Ihre gekaufte Software mit allen Updates neu installieren. Dabei werden Sie Ihre gekauften Lizenzen wieder eingeben müssen. Diese Lizenzen sollten zugelassen werden, da der Lizenzzähler mit dem Deinstallieren dieser Lizenz den Zähler heruntergezählt hat. Möglicherweise müssen Sie bei Ihren Programmen jedoch prüfen, ob es auch hier wesentliche Neuerungen gibt, und es sich lohnt, die neuesten Versionen zu installieren – meistens müssen dann auch neue Lizenzen gekauft werden. Somit verhindern Sie, dass Sie ältere / alte Programme, die nicht mehr für das aktuelle Betriebssystem geschrieben wurden, erneut installieren.

Ein letzter Tipp: Wie Sie beim Lesen des Textes schon bemerkt haben, unterscheiden wir zwischen Betriebssystempartition oder -platte und Partition / Platte für Benutzerdaten. Es ist zumindest sinnvoll, wenn nur eine Platte zur Verfügung steht, zwei Partitionen einzurichten: eine kleinere für das Betriebssystem (bis zu 100 GB) und der Rest in der anderen Partition für die Benutzerdaten. Wenn das Betriebssystem seinen Dienst versagt oder durch ungewöhnliche Umstände (auch Updates) nicht mehr funktioniert, gelangen sie immer noch an ihre Benutzerdaten. Natürlich sollten Sie auf

einem externen Laufwerk ihre Benutzerdaten kopiert und möglichst aktuell gehalten haben. Besser ist es, wenn in PCs (gelegentlich auch bei Notebooks) ein zweites oder weiteres Plattenlaufwerk eingebaut wird, auf denen die Benutzerdaten ein zweites Mal gespeichert werden. In dieser Situation darf die Systemplatte ausfallen, ohne dass Benutzerdaten gefährdet sind. So wie der Benutzer von seinen Benutzerdaten öfter ein Back-up anfertigt, sollte nach einigen Programminstallationen ebenfalls ein Image der Systempartition angefertigt und auf einer externen Festplatte gespeichert werden. Damit ist ein jederzeitiges Wiederherstellen der Betriebssystempartition und der Benutzerdaten möglich, ohne große Datenverluste zu erleiden. Allerdings ist das ein Tipp, den nur die wirklich beherzigen, die ein Benutzerdatenverlust schon einmal schmerzlich erlebt haben...

Stefan Ebelt, ReUse e.V. gemeinnützig, Am Forstacker 7a, 13587 Berlin

Der gemeinnützige ReUse e.V. auf dem Gebiet des Natur- und Umweltschutzes, der Ressourcenschonung und Nachhaltigkeit tätig, fördert die Bildung und Erziehung auf dem Gebiet der Wiederverwendung von elektrischen und elektronischen Produkten, unterstützt die Herstellung netzwerkbasierter Kooperationsstrukturen, informiert die Öffentlichkeit über die Nutzungsmöglichkeiten von gebrauchter IT-Technik und anderen elektr(on)ischen Geräten und beteiligt sich an wissenschaftlichen Forschungsprojekten.

Modernes Tool für alte Texte

Würzburg, 23.04.2019. Historische Druckschriften in computerlesbaren Text umwandeln: Dafür sorgt das Werkzeug OCR4all, das sehr zuverlässig arbeitet, leicht zu bedienen und frei verfügbar ist. Wissenschaftler der Uni Würzburg haben es entwickelt.

Historiker, Germanisten und andere Geisteswissenschaftler haben es oft mit jahrhundertealten Druckwerken zu tun, die sich nicht leicht entziffern lassen und die oft schlecht erhalten sind. Viele dieser Dokumente sind inzwischen digitalisiert – in der Regel abfotografiert oder eingescannt – und stehen weltweit online zur Verfügung. Es gibt aber immer noch eine Herausforderung zu meistern: die digitalisierten alten Schriften mit Texterkennungs-Software in eine moderne Form zu bringen, die auch für Nicht-Fachleute und für Computer lesbar ist. Auf diesem Gebiet haben Wissenschaftler vom Zentrum für Philologie und Digitalität der Julius-Maximilians-Universität Würzburg (JMU) für eine deutliche Weiterentwicklung gesorgt.

Mit OCR4all stellt das JMU-Forschungsteam der Fachwelt ein neues Werkzeug zur Verfügung. Es setzt digitalisierte historische Drucke mit einer Fehlerquote von weniger als einem Prozent in computerlesbaren Text um. Und es bietet eine grafische Benutzeroberfläche, für deren Bedienung kein Informatik-Fachwissen nötig ist. Bei bisherigen Tools dieser Art war die Nutzerfreundlichkeit nicht sonderlich ausgeprägt, meist musste mit Programmierbefehlen hantiert werden.

Das neue Werkzeug OCR4all wurde unter der Leitung von Christian Reul mit seinen Informatik-Fachkollegen Professor Frank Puppe (Lehrstuhl für Künstliche Intelligenz und

angewandte Informatik) und Christoph Wick sowie mit Uwe Springmann, Fachmann für Digital Humanities, und zahlreichen Studierenden und Hilfskräften entwickelt.

Seine Wurzeln hat OCR4all im Kallimachos-Verbundprojekt der JMU, das vom Bundesministerium für Bildung und Forschung gefördert wird. Diese Kooperation zwischen Geisteswissenschaften und Informatik wird im neu gegründeten Zentrum für Philologie und Digitalität (ZPD) weitergeführt und institutionalisiert.

OCR4all steht der Öffentlichkeit auf der Plattform GitHub (mit Anleitungen und Anschauungsbeispielen) frei zur Verfügung: <https://github.com/OCR4all>

Wissenschaftlicher Ansprechpartner:

Christian Reul, kommissarischer Leiter der Digitalisierungseinheit, Zentrum für Philologie und Digitalität der Universität Würzburg, christian.reul@uni-wuerzburg.de

Weitere Informationen:

OCR4all auf GitHub: <https://github.com/OCR4all>

Calamari auf GitHub: <https://github.com/Calamari-OCR>

Das Projekt „Narragonien digital“: <https://www.uni-wuerzburg.de/aktuelles/einblick/archiv/single/news/narrenschi/>

Link zur Publikation der Fallstudie mit sechs historischen Büchern: https://jcl.org/content/2-allissues/1-heft1-2018/jcl_2018-1_1.pdf

Publikation über methodische und technische Verbesserungen: https://jcl.org/content/2-allissues/1-heft1-2018/jcl_2018-1_4.pdf

Neue Arbeitsgruppe zu mathematischer Software

Exzellente Verstärkung für mathematische Forschung in Leipzig

Leipzig, 02.12.2019. Prof. Dr. Michael Joswig, Einstein-Professor für Diskrete Mathematik und Geometrie an der Technischen Universität Berlin, wurde von der Max-Planck-Gesellschaft zum Max-Planck-Fellow berufen. Parallel zu seiner Tätigkeit in Berlin leitet er ab Dezember 2019 eine Arbeitsgruppe zu mathematischer Software am Max-Planck-Institut für Mathematik in den Naturwissenschaften in Leipzig. Das Max-Planck-Fellow-Programm fördert die Zusammenarbeit von herausragenden Hochschullehrerinnen und -lehrern mit Wissenschaftlerinnen und Wissenschaftlern der Max-Planck-Gesellschaft.

Michael Joswig gilt als weltweit führender Forscher auf den Gebieten der polyedrischen und geometrischen Kombinatorik als auch der Entwicklung und Integration von mathematischer Software. Er interessiert sich für alle Aspekte der polyedrischen, tropischen und algorithmischen Geometrie, einschließlich Themen wie Optimierung und kombinatorische Topologie. Eine seiner größten wissenschaftlichen Errungenschaften ist das von ihm mitentwickelte Softwaresystem „Polymake“, das sich weltweit als Standardsoftware im Bereich der Diskreten Geometrie und in zahlreichen weiteren Anwendungen etabliert hat.

„Wir versprechen uns von dieser doppelten fachlichen Expertise Michael Joswigs eine große Bereicherung unseres

Forschungsspektrums und viele neue Impulse für unsere wissenschaftliche Arbeit“, so Prof. Bernd Sturmfels, Direktor des Instituts und Leiter der Arbeitsgruppe Nichtlineare Algebra, welche in enger wissenschaftlicher Zusammenarbeit mit der neugegründeten Forschungsgruppe stehen wird.



Foto von Prof. Dr. Michael Joswig
PR / TU Berlin / Ulrich Dahl

Michael Joswig unterhält weitreichende Forschungstätigkeiten auf höchstem Niveau. Er ist aktiv in mehreren DFG-geförderten Verbundprojekten und spielt eine zentrale und visionäre Rolle in Berlins neuem Exzellenzcluster MATH+. Aktuell arbeitet er an einer Reihe offener Probleme in der tropischen Geometrie, wie der Klassifizierung glatter tropischer kubischer Flächen und der f-Vektor-Vermutung für tropische lineare Räume. Einige dieser Projekte finden unmittelbare Anwendung in der Optimierung und in der Datenmathematik, was am Leipziger Max-Planck-Institut ein wachsendes Interesse hervorruft.

Der Hauptfokus der jüngst gegründeten Gruppe am Leipziger Max-Planck-Institut

liegt auf der Entwicklung von mathematischer Software. Computer Algebra und diskrete Algorithmen haben eine wachsende Relevanz in einer Vielzahl von Anwendungen. Mathematische Software kann hier, laut Joswig, als eine Brücke fungieren, nicht nur zwischen den mathematischen Disziplinen, sondern auch interdisziplinär zu diversen Forschungsbereichen. Das Ziel ist es, mathematisches Wissen mithilfe von Software weitreichend zugänglich zu machen. Unterstützend werden ihm zunächst ein Postdoc sowie ein aus der renommierten IMPRS-Graduiertenschule rekrutierter Doktorand zur Seite stehen. Am 3. Dezember wird die neue Arbeitsgruppe mit einem eintägigen Workshop, dem Mathematical Software Day, thematisch-passend initiiert.

Die Software Polymake hat sich zu einem leistungsstarken System mit einer Vielzahl von Anwendungen entwickelt. Hierzu gehören beispielsweise die tropische und algebraische Geometrie, kombinatorische Topologie sowie die lineare und diskrete Optimierung. Polymake wird von zahlreichen nationalen und internationalen Forschungsgruppen in Mathematik, Informatik, Biologie und Chemie konsequent eingesetzt. Aus dem Projekt gingen mehrere viel beachtete Spin-off-Projekte hervor, die von Tools für den Mathematikunterricht, beispielsweise mit dem Spiel „MatchTheNet“, bis hin zu Spezialsoftware für parallele Höchstleistungsrechner

reichen. Michael Joswig ist ein Verfechter der Nützlichkeit und Bedeutung von mathematischer Software in allen Bereichen der mathematischen Forschung und darüber hinaus. Er glaubt, dass der Computer als Werkzeug eine ähnlich bedeutende Rolle für die Mathematik spielt wie das Mikroskop für die Biologie. Michael Joswig fungiert als Generalvorsitzender der Internationalen Konferenz für Mathematische Software (ICMS), die 2020 in Braunschweig stattfinden wird, ist im Beirat der Oberwolfach-Referenzgruppe für Mathematische Software (ORMS) und als Principal Investigator des Sonderforschungsbereiches „Symbolische Werkzeuge in der Mathematik und ihre Anwendung“ aktiv. Mit seinen Kolleginnen und Kollegen will er das Computeralgebrasystem der nächsten Generation OSCAR (= Open Source Computer Algebra Research) entwickeln.

Weitere Informationen:

<http://page.math.tu-berlin.de/~joswig/>
TU Berlin - Homepage der Discrete Mathematics/Geometry Gruppe von Prof. Dr. Michael Joswig

<https://www.einsteinfoundation.de/personen-projekte/einstein-professoren/michael-joswig/>
Einstein Stiftung - Prof. Dr. Michael Joswig im Portrait

<https://www.polymake.org/doku.php>
Informationen zur Software Polymake

APL-Journal

38. Jg. 2019, ISSN 1438-4531

Herausgeber: Prof. Dr. Dieter Kilsch, APL-Germany e.V., Mannheim, Homepage: <https://apl-germany.de/>, E-Mail: d.kilsch@th-bingen.de

Redaktion: Dipl.-Volksw. Martin Barghoorn (verantw.), Lückhoffstr. 8, 14129 Berlin, E-Mail: Martin@Barghoorn.com

Verlag: RHOMBOS-VERLAG, Berlin, Kurfürstenstr. 15/16, D-10785 Berlin, Tel. (030) 261 9461, eMail: verlag@rhombos.de, Internet: <https://rhombos.de/>

Erscheinungsweise: halbjährlich

Erscheinungsort: Berlin

Druck: dbusiness.de GmbH, Berlin

Copyright: APL Germany e.V. (für alle Beiträge, die als Erstveröffentlichung erscheinen)

Fotonachweis: Martin Barghoorn (Umschlagseite 1 und 4, Seite 26, 28, 29, 30), Stefan Ebelt: 31, 32, 34, 42

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutzgesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürfen. Eine Haftung für die Richtigkeit der veröffentlichten Informationen kann trotz sorgfältiger Prüfung von Herausgeber und Verlag nicht übernommen werden. Mit Namen gekennzeichnete Artikel geben nicht unbedingt die Meinung des Herausgebers oder der Redaktion wieder. Für unverlangte Einsendungen wird keine Haftung übernommen. Nachdruck ist nur mit Zustimmung des Herausgebers sowie mit Quellenangabe und Einsendung eines Beleges gestattet. Überarbeitungen eingesandter Manuskripte liegen im Ermessen der Redaktion.



Allgemeine Informationen

(Stand 2019)

APL-Germany e.V. ist ein gemeinnütziger Verein mit Sitz in Düsseldorf. Sein Zweck ist es, die Programmiersprache APL, sowie die Verbreitung des Verständnisses der Mensch-Maschine Kommunikation zu fördern. Für Interessenten, die zum Gedankenaustausch den Kontakt zu anderen APL-Benutzern suchen, sowie für solche, die sich aktiv an der Weiterverbreitung der Sprache

APL beteiligen wollen, bietet APL-Germany den adäquaten organisatorischen Rahmen.

Auf Antrag, über den der Vorstand entscheidet, kann jede natürliche oder juristische Person Mitglied werden. Organe des Vereins sind die mindestens einmal jährlich stattfindende Mitgliederversammlung sowie der jeweils auf zwei Jahre gewählte Vorstand.

1. Vorstandsvorsitzender

Prof. Dr. Dieter Kilsch,
Dumontstraße 12, 55313 Mainz,
Tel. 06131 6982200, E-Mail:
d.kilsch@th-bingen.de.

2. Vorstandsvorsitzender:

Martin Barghoorn
Lückhoffstr. 8, 14129 Berlin,
E-Mail: Martin@Barghoorn.com

Schatzmeister

Jürgen Beckmann
Im Freudenheimer Grün 10
68259 Mannheim
Tel. 0621 7 98 08 40,
eMail: JBecki@onlinehome.de

Beitragsätze

Persönliche Mitglieder:

Natürliche Personen 32,- EUR*
Studenten / Schüler 11,- EUR*

Institutionelle Mitglieder:

Jurist./natürl. Pers. 500,- EUR*

* Jahresbeitrag

Bankverbindung

BVB Volksbank eG Bad Vilbel
BLZ 518 613 25
Konto-Nr. 523 2694

Hinweis:

Wir bitten alle Mitglieder, uns Adressänderungen und neue Bankverbindungen immer sofort mitzuteilen. Geben Sie bei Überweisungen den Namen und/oder die Mitgliedsnummer an.

<https://apl-germany.de/>

