



developer media

Entwickler- Almanach 2022

Fachliches. Unterhaltsames. Hilfreiches.

Entwickler-Almanach 2022

Fachliches, Unterhaltsames, Hilfreiches.

ChatGPT, Dr. Fabian Deitelhoff, Dr. Joachim Fuchs, Ben Hoyt,
Bernhard Lauer

Tilman Börner

Ebner Media Group GmbH & Co. KG

Entwickler-Almanach 2022

1. [Gut gestärkt ins neue Jahr: Der Entwickler-Almanach 2022](#)
2. [Impressum](#)
3. [E/A ist nicht länger der Engpass](#)
4. [WPF, C#: ItemsSource für Control implementieren](#)
5. [Spaß mit Awaitables: await 1000;](#)
6. [WPF/VB: Fenster ohne Fensterrahmen](#)
7. [Bewegte Bilder im GIF-Format](#)
8. [Treffen sich künstliche Intelligenz und Sitcom](#)
9. [Windows Explorer: VS Code aus der Adresszeile aufrufen](#)
10. [Die wichtigsten Trends im Jahr 2023 für ...](#)
11. [VS 2022: JSON als Klassen einfügen](#)

Gut gestärkt ins neue Jahr: Der Entwickler-Almanach 2022

Da bin ich wieder. Ich, der Almanach. Frisch gebacken, knusprig und knackig komme ich auf Ihren Tisch. Der Entwickler-Almanach ist ein kostenloses E-Book, das die Entwickler-Division developer media der Ebner Media Group um die Weihnachtszeit verschenkt.



developer media versorgt Sie mit Fachinformation zur professionellen Softwareentwicklung. Wir bieten Expertenwissen auf Konferenzen wie der Developer Week, der .NET Developer Conference, der Web Developer Conference oder der Dev-Cloud Conference. In Trainings schulen wir Sie intensiv beispielsweise auf Softwarearchitektur, Softwarequalität oder C#. In den Magazinen dotnetpro und web & mobile developer erhalten Sie die neuesten Information in die Tiefe der Softwareentwicklung

Unter dem Motto **Fachliches. Unterhaltsames. Hilfreiches.** wollen wir Ihnen für die stillen Tage zwischen den Jahren diese kleine Lektüre an die Hand geben, die zum Schmökern einlädt.

Gespannt? Na, dann los - stürzen Sie sich in den Entwickler-Almanach 2022. Wir wünschen Ihnen viel Spaß und kurzweilige Erkenntnisse.



developer media christmas

Das Team von developer media wünscht Ihnen ein gesundes und erfolgreiches Jahr 2022.

Fernando, Florian, Jonas, Max, Reinhold, Sabine, Susanne, Tilman

Impressum

Der Entwickler-Almanach 2022

Fachliches, Unterhaltsames, Hilfreiches.

published by developer media www.developer-media.de

E-Mail: info@developer-media.de

Ebner Media Group GmbH & Co. KG

Adresse: Karlstr. 3, 89073 Ulm

E-Mail: info@ebnermedia.de

Website: www.ebnermedia.de



developer media jobs

Der Stellenmarkt für Entwickler

**HIER FINDEN SIE
NICHT JEDEN.
SONDERN GENAU
WEN SIE SUCHEN.**

**JETZT
ANZEIGE
SCHALTEN!**



jobs.developer-media.de

Reinhold Fritsch, Tel.: +49 731 88005-8285 | Sabine Vockrodt, Tel.: +49 731 88005-8222
oder per E-Mail an: jobs@developer-media.de



He couldn't sleep for 2 days
because he missed her.

·
·
I couldn't sleep for 6 days
because I missed a ";" in my code.

<https://www.facebook.com/ProgrammersCreateLife/>



E/A ist nicht länger der Engpass

November 2022

Bei Vorstellungsgesprächen mit Programmierern bitte ich sie oft, ein einfaches Programm zum Zählen von Worthäufigkeiten in einer Textdatei zu programmieren. Das ist ein gutes Problem, das eine Reihe von Fähigkeiten testet und mit einigen Folgefragen eine überraschende Vertiefung ermöglicht.



Eine der Folgefragen, die ich stelle, lautet: “Was ist der Leistungsengpass in Ihrem Programm?” Die meisten Leute sagen etwas wie “Lesen aus der Eingabedatei”.

Tatsächlich wurde ich zu diesem Artikel inspiriert, nachdem ich jemandem auf Gopher Slack geantwortet hatte, der sagte: “Ich habe auch bemerkt, dass hier eine Menge zusätzlicher Arbeit beim Aufteilen der gesamten Zeile usw. anfällt, aber normalerweise ist das alles so viel schneller als I/O, dass es uns egal ist.”

Ich will nicht auf ihm herumhacken ... bevor ich die Leistung des Zählwortproblems analysierte, dachte ich dasselbe. Das hat man uns allen beigebracht, nicht wahr? "E/A ist langsam."

Die Zeiten haben sich verändert

Nicht mehr! Festplatten-E/A mag vor 10 oder 20 Jahren langsam gewesen sein, aber im Jahr 2022 ist das sequentielle Lesen einer Datei von der Festplatte sehr schnell.

Aber wie schnell? Ich habe die Lese- und Schreibgeschwindigkeit meines Entwicklungslaptops mit dieser Methode getestet, allerdings mit `count=4096`, so dass wir 4 GB lesen und schreiben.

Hier sind die Ergebnisse auf meinem 2022 Dell XPS 13 Plus mit einem Samsung PM9A1 NVMe-Laufwerk, auf dem Ubuntu 22.04 läuft:

I/O-Typ	Geschwindigkeit (GB/s)
Lesen (nicht zwischengespeichert)	1,7
Lesen (gecached)	10,8
Schreiben (inkl. Sync-Zeit)	1,2
Schreiben (nicht inkl. Sync)	1,6

Systemaufrufe sind natürlich relativ langsam, aber beim sequentiellen Lesen oder Schreiben müssen Sie nur alle 4 KB oder 64 KB oder wie groß auch immer Ihr Puffer ist, einen Systemaufruf ausführen.

Und E/A über ein Netzwerk ist immer noch langsam, besonders in einem nicht-lokalen Netzwerk.

Der Knackpunkt

Was ist also der Engpass in einem Programm, das Worthäufigkeiten wie das obige zählt? Es ist die Verarbeitung oder das Parsen der Eingabe und die damit verbundenen Speicherzuweisungen: Aufteilung der Eingabe in Wörter, Konvertierung in Kleinbuchstaben und Zählung der Häufigkeiten mit einer Hashtabelle.

Ich habe meine Python- und Go-Programme zum Zählen von Wörtern so modifiziert, dass die Zeiten der verschiedenen Phasen des Prozesses aufgezeichnet werden: - Lesen der Eingabe, - Verarbeitung (der langsame Teil), - Sortieren nach Häufigkeit und - Ausgabe.

Ich lasse das Programm mit einer 413 MB großen Textdatei laufen, also eine anständige Menge an Input (100 verkettete Kopien des Textes der King James Bibel).

Nachfolgend sind die Ergebnisse in Sekunden des besten von 3 Durchläufen aufgeführt:

Stufe	Python	Go (einfach)	Go (optimiert)
Lesen	0,384	0,499	0,154
Verarbeitung	7,980	3,492	2,249
Sortieren	0,005	0,002	0,002
Ausgeben	0,010	0,009	0,010
Insgesamt	8,386	4,000	2,414

Das Sortieren und Ausgeben ist hier vernachlässigbar: Da die Eingabe aus 100 Kopien besteht, ist die Zahl der eindeutigen

Wörter vergleichsweise gering.

Nebenbemerkung: Dies ist eine weitere interessante Folgefrage in Vorstellungsgesprächen. Einige Kandidaten sagen, dass das Sortieren der Engpass sein wird, weil es der Ordnung $O(N \log N)$ ist und nicht die der Eingabeverarbeitung, die $O(N)$ ist. Dabei wird jedoch leicht vergessen, dass wir es mit zwei verschiedenen N zu tun haben: der Gesamtzahl der Wörter in der Datei und der Anzahl der eindeutigen Wörter.

Der Kern der Python-Version

```
# Run with the following command to clear the read cac
#
# sudo sysctl vm.drop_caches=3; python3 simple.py <kjv

import collections
import sys
import time

t0 = time.time()
content = sys.stdin.read()
t1 = time.time()
read_time = t1 - t0

counts = collections.Counter(content.lower().split())
t2 = time.time()
process_time = t2 - t1

most_common = counts.most_common()
t3 = time.time()
sort_time = t3 - t2

for word, count in most_common:
    print(word, count)
```

```
t4 = time.time()
output_time = t4 - t3

total_time = t4 - t0

print('Reading    :', read_time, file=sys.stderr)
print('Processing:', process_time, file=sys.stderr)
print('Sorting    :', sort_time, file=sys.stderr)
print('Outputting:', output_time, file=sys.stderr)
print('TOTAL      :', total_time, file=sys.stderr)
```

besteht aus ein paar Zeilen Code:

```
content = sys.stdin.read()
counts = collections.Counter(content.lower().split())
most_common = counts.most_common()
for word, count in most_common:
    print(word, count)
```

In Python kann man leicht Zeile für Zeile lesen, aber das ist etwas langsamer. Deshalb lese ich hier einfach die ganze Datei in den Speicher und verarbeite sie auf einmal.

Die [einfache Go-Version](#) verfolgt denselben Ansatz, obwohl die Standardbibliothek von Go nicht über [collections.Counter](#) verfügt, so dass wir die “häufigste” Sortierung selbst vornehmen müssen.

Die [optimierte Go-Version](#) ist deutlich schneller, aber auch um einiges komplizierter. Wir vermeiden die meisten Speicherzuweisungen, indem wir in Kleinbuchstaben konvertieren und an den Wortgrenzen splitten. Das ist eine gute Faustregel für die Optimierung von CPU-gebundenem Code:

Speicherzuweisungen reduzieren. In [meinem Artikel über die](#)

[Optimierung der Anzahl der Wörter](#) finden Sie Informationen dazu, wie Sie das Profil erstellen können.

Ich habe keine optimierte Python-Version gezeigt, weil es schwierig ist, Python weiter zu optimieren! (Ich habe die Zeit von 8,4 auf 7,5 Sekunden gesenkt). Es ist so schnell wie es ist, weil die Kernoperationen in C-Code ablaufen - deshalb spielt es oft keine Rolle, dass "Python langsam ist".

Wie Sie sehen können, nimmt die Festplatten-E/A in der einfachen Go-Version nur 14 Prozent der Laufzeit in Anspruch. In der optimierten Version haben wir sowohl das Lesen als auch die Verarbeitung beschleunigt, so dass die Festplatten-E/A nur 7 Prozent der Gesamtzeit ausmacht.

Meine Schlussfolgerung? Wenn Sie "große Daten" verarbeiten, ist die Festplatten-E/A wahrscheinlich nicht der Engpass. Ein paar Messungen werden wahrscheinlich auf das Parsing und die Speicherzuweisung hindeuten.

Ben Hoyt (<https://benhoyt.com/>) ist mit der einzigen *Franci Hoyt* auf der Welt verheiratet und lebt mit seinen drei reizenden Töchtern in Christchurch, Neuseeland. Tagsüber ist er Software-Ingenieur bei *Canonical* - weitere Informationen finden Sie in [meinem Lebenslauf](#). Er hat ein paar [Open-Source-Projekte](#) entwickelt und schreibt gerne [technische](#) und [nicht-technische Artikel](#).

SAVE THE DATE

DEVELOPER WEEK '23

VOM 26. BIS 30. JUNI 2023

NCC OST | NÜRNBERG

DWX

DAS EVENT FÜR WEB, MOBILE, JAVA UND .NET

Die Top Themen: Softwarearchitektur | .NET | Cloud-Entwicklung
AI / Machine Learning | Angular | Lowcode | Core Java /
Java SE Container | Big Data | Softwarequalität
Agile Methoden | Enterprise Java / Jakarta EE | Programmiersprachen
Security | Accessibility | Softskills | Refactoring | DevOps | Testen

developer-week.de | #DWX23 | Find us on    

Veranstalter:  **developer
media**

Präsentiert von:  **dotnetpro**  **web & mobile
DEVELOPER**

WPF, C#: ItemsSource für Control implementieren

Über die Eigenschaft `ItemsSource` bindet man Datensammlungen an Controls wie `DataGrid` oder `ListView`. Wie implementiert man aber diese Eigenschaft in einem eigenen Control?



`ItemsSource` sorgt dafür, dass sich bei einer Veränderung in der angebundenen `Collection` das Control neu zeichnet. Was aber, wenn das Control, das man selbst bauen will, auch über einen solchen Mechanismus verfügen soll?

Es gibt Empfehlungen, bei einem solchen Ansinnen das `ItemsControl` zu verwenden. Das kennt die Eigenschaft `ItemsSource` bereits. Wer aber einen anderen Weg gehen und beispielsweise ein Grid erweitern will, kann auf den Quellcode von [Gaurav Verma](#) zurückgreifen.

```
public IEnumerable ItemsSource
{
    get { return (IEnumerable)GetValue(
```

```

        ItemsSourceProperty); }
    set { SetValue(ItemsSourceProperty, value); }
}

public static readonly DependencyProperty
    ItemsSourceProperty =
    DependencyProperty.Register(
        "ItemsSource", typeof(IEnumerable),
        typeof(UserControl1), new PropertyMetadata(
            new PropertyChangedCallback(
                OnItemsSourcePropertyChanged)));

private static void OnItemsSourcePropertyChanged(
    DependencyObject sender, <>
    DependencyPropertyChangedEventArgs e)
{
    var control = sender as UserControl1;
    if (control != null)
        control.OnItemsSourceChanged(
            (IEnumerable)e.OldValue,
            (IEnumerable)e.NewValue);
}

private void OnItemsSourceChanged(
    IEnumerable oldValue, IEnumerable newValue)
{
    // Remove handler for oldValue.CollectionChanged
    var oldValueINotifyCollectionChanged =
        newValue as INotifyCollectionChanged;

    if (null != oldValueINotifyCollectionChanged)
    {
        oldValueINotifyCollectionChanged
            .CollectionChanged -=
                new NotifyCollectionChangedEventHandler(
                    newValueINotifyCollectionChanged_
                        CollectionChanged);
    }
}

```

```

// If possible add handler for
// newValue.CollectionChanged
var newValueINotifyCollectionChanged =
    newValue as INotifyCollectionChanged;
if (null != newValueINotifyCollectionChanged)
{
    newValueINotifyCollectionChanged
        .CollectionChanged +=
            new NotifyCollectionChangedEventHandler(
                newValueINotifyCollectionChanged_
                    CollectionChanged);
}
}

void newValueINotifyCollectionChanged_CollectionChange
    object sender, NotifyCollectionChangedEventArgs e)
{
    /***Do your stuff here**
}

```

Bei */**Do your stuff here*** kommt dann die Aktualisierung des Controls hinein. Dort können zum Beispiel alle Kinder des Grids gelöscht und neu angelegt werden.



"In a world where you
can be anything,
be the person
who didn't deploy
to production Friday"

@iamdeveloper

Jetzt kostenlos testen!



**2x
gratis!**



Das Fachmagazin für .NET-Entwickler



Testen Sie jetzt 2 kostenlose Ausgaben und erhalten Sie unseren exklusiven Newsletter gratis dazu.

www.dotnetpro.de/dnpabo

Spaß mit Awaitables: await 1000;

Was soll das denn sein? Ein Syntaxfehler schon im Titel? So einen Unsinn lässt der C#-Compiler zum Glück nicht zu. Oder vielleicht doch?



Probieren wir es doch einfach einmal aus. Konsolenanwendung anlegen, das Statement einfügen und – siehe da - der erste Versuch scheint schon die Annahme zu bestätigen, dass das nicht gehen kann. Die Fehlermeldung in Bild 1

```
await 1000;
```

readonly struct System.Int32
Represents a 32-bit signed integer.

CS1061: 'int' does not contain a definition for 'GetAwaiter' and no accessible extension method 'GetAwaiter' accepting a first argument of type 'int' could be found (are you missing a using directive or an assembly reference?)

Wie erwartet – das kann doch nicht funktionieren (Bild 1)

kann nicht lügen. Der Hinweis, warum der Compiler das nicht mag, wirft allerdings Fragen auf. Was genau meint der Compiler mit

does not contain a definition for 'GetAwaiter'?

Was muss denn eigentlich hinter dem `await`-Schlüsselwort stehen? Na, da haben wir mal gelernt, dass da immer ein Ausdruck erwartet wird, der ein (bereits gestartetes) `Task`-Objekt liefert. In der Praxis wird das wohl auch meist stimmen, aber hinter den Kulissen passiert noch was anderes. Da gibt es noch eine verborgene, magische Namenskonvention.

Interessanterweise kann hinter dem `await`-Schlüsselwort jeder Ausdruckstyp stehen, der eine öffentliche Methode namens `GetAwaiter` besitzt. Diese muss eine Struktur vom Typ `TaskAwaiter` zurückgeben. Die Dokumentation zu [TaskAwaiter](#) sagt zwar:

This type is intended for compiler use only.

Aber das soll uns ja nicht davon abhalten, es doch mal zu versuchen. Also probieren wir mal folgendes: Wir definieren eine Klasse `MagicDelay`:

```
public class MagicDelay
{
    private readonly TimeSpan timeSpan;

    private MagicDelay(TimeSpan timeSpan)
    {
        this.timeSpan = timeSpan;
    }

    public static MagicDelay Milliseconds(int milliseconds)
    {
        return new MagicDelay(TimeSpan.FromMilliseconds(milli

// Magic function...
    public TaskAwaiter GetAwaiter()
    {
        return Task.Delay(timeSpan).GetAwaiter();
    }
}
```

```
}  
}
```

und sehen einen Konstruktor mit einem *TimeSpan* als Parameter vor, eine statische Factory-Methode zur einfacheren Instanzierung sowie die besagte *GetAwaiter*-Methode. Den *TaskAwaiter* bekommen wir von dem *Task*-Objekt, das von *Task.Delay* zurückgegeben wird. Und dann führt ein Aufruf wie

```
await MagicDelay.Milliseconds(1000);
```

plötzlich zu keinem Syntaxfehler mehr, sondern funktioniert tatsächlich. Das macht zwar nichts anderes als

```
await Task.Delay(1000);
```

Es ist aber trotzdem interessant, dass das so zulässig ist und ohne zu Mucken läuft.

Na gut, aber *1000* ist ein Integer und damit können wir keine zusätzliche Instanzmethode definieren. Also kommen wir so nicht weiter, oder? Es sei denn ...

Was wäre denn, wenn man statt der Instanzmethode namens *GetAwaiter* auch eine Erweiterungsmethode verwenden könnte? In der Fehlermeldung in **Bild 1** war ja so was angedeutet worden. Probieren wir doch vielleicht folgendes mal aus: Wir entfernen die *GetAwaiter*-Methode aus *MagicDelay*, ergänzen eine *ReadOnly*-Property für *TimeSpan* und verlagern *GetAwaiter* als Erweiterungsmethode in die Klasse *MagicExtensions*:

```
public class MagicDelay  
{  
    private readonly TimeSpan timeSpan;  
    public TimeSpan TimeSpan => timeSpan;
```

```

private MagicDelay(TimeSpan timeSpan)
{
    this.timeSpan = timeSpan;
}

public static MagicDelay Milliseconds(int milliseconds)
{
    return new MagicDelay(TimeSpan.FromMilliseconds(milli
}

public static class MagicExtensions
{
    public static TaskAwaiter GetAwaiter(this MagicDelay ma
    {
        return Task.Delay(magicDelay.TimeSpan).GetAwaiter();
    }
}

```

Und siehe da,

```
await MagicDelay.Milliseconds(1000);
```

funktioniert immer noch.

Ok – das ist doch ein Ansatz, da muss noch mehr gehen. Wozu der Umweg über die Klasse *MagicDelay*? Können wir das nicht direkt mit *TimeSpan* machen? Eine Erweiterungsmethode

```

public static class MagicExtensions
{
    public static TaskAwaiter GetAwaiter(this TimeSpan time
    {
        return Task.Delay(timeSpan).GetAwaiter();
    }
}

```

und dann ein Aufruf in dieser Form:

```
await TimeSpan.FromMilliseconds(2000);
```

Oh wow - das geht ja auch.

Also, dann müsste es doch auch möglich sein, eine solche Extension-Methode für den Typ Integer vorzusehen. Kann das mit dem Vierzeiler

```
public static class MagicExtensions
{
    public static TaskAwaiter GetAwaiter(this int milliseconds)
    {
        return Task.Delay(TimeSpan.FromMilliseconds(milliseconds));
    }
}
```

vielleicht schon funktionieren? Ein Integer-Wert als Zeitangabe in Millisekunden? Implementieren, ausprobieren:

```
await 1000;
```

Übersetzen - der Compiler meckert nicht - Syntax korrekt – Starten – Läuft!!! Wer hätte das gedacht? Es geht also doch.

Braucht man so was?

Nicht wirklich. Aber witzig ist es schon, dass man mit einfachen Tricks eine solche Syntax zulässig machen kann. „Bei `await 1000` weiß man doch nicht, welche Einheit sich dahinter verbergen soll“, werden Kritiker jetzt sagen. Sind das Sekunden, Millisekunden, oder was? Andererseits muss man zugeben, dass das bei `Task.Delay(1000)` auch nicht besser gelöst wurde.

Ob es wirklich praktische Anwendungen für diesen Ansatz gibt oder ob es nur eine Lösung für ein nicht existierendes Problem ist – wir wissen

es nicht. Vielleicht haben Sie als Leser kreative Ideen, was man mit *GetAwaiter*-Erweiterungsmethoden Sinnvolles umsetzen könnte? Lassen Sie es uns wissen.

[Hier finden die den Quellcode zu den Beispielen](#)

Inspiziert wurde dieser Artikel übrigens von dem Youtube-Video mit dem lustigen Titel [How to await ANYTHING in C#](#).

Wenn das nicht Anregung genug für Sie ist, es selbst mal auszuprobieren. Dann haben wir hier eine Übungsaufgabe für Sie: Bringen Sie doch mal folgendes zum Laufen:

```
await "nächste Ausgabe der dotnetpro";
```

Viel Erfolg!

***Dr. Joachim Fuchs** ist begeisterter Anhänger von Microsofts .NET-Philosophie. Er arbeitet als Softwarearchitekt, Berater und Dozent im Expertennetzwerk www.it-visions.de und ist Autor der dotnetpro. Seine Schwerpunkte liegen derzeit bei XAML- und Web-UI-Technologien.*

Jetzt kostenlos testen!



2x gratis!



Praxiswissen für Entwickler!

Testen Sie jetzt 2 kostenlose Ausgaben und erhalten Sie exklusiven Zugang zu unserem Archiv.

webundmobile.de/probelesen

WPF/VB: Fenster ohne Fensterrahmen

Soll das Anwendungsfenster anders als gewöhnlich aussehen, stört der für Windows typische Fensterrahmen. Den kann man aber ganz einfach weglassen.



Startet man die Arbeit an einer neuen WPF-Desktop-Anwendung, so gibt Visual Studio folgenden XAML-Code vor, der ein ganz normales Fenster mit dem üblichen Rahmen erzeugt:

```
<Window x:Class="MainWindow"
  xmlns="http://schemas.microsoft.com/
    winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/
    winfx/2006/xaml"
  xmlns:d="http://schemas.microsoft.com/
    expression/blend/2008"
  xmlns:mc="http://schemas.openxmlformats.
    org/markup-compatibility/2006"
  xmlns:local="clr-namespace:Test"
  mc:Ignorable="d"
  Title="MainWindow" Height="450"
  Width="800">
```

```
<Grid> </Grid>
</Window>
}
```

Um den Rahmen zu entfernen, muss die folgende Zeile hinzugefügt werden:

```
AllowsTransparency="True" WindowStyle="None"
```

Das resultierende Fenster hat keinen Rahmen mehr, allerdings kann man es auch nicht mehr mit der Maus verschieben, und auch das Vergrößern und Verkleinern klappt nicht mehr. Dafür braucht es jedoch ebenfalls nur eine kleine Erweiterung, so dass der Code dann beispielsweise so aussieht:

```
<Window x:Class="MainWindow"
    ...
    Title="MainWindow" Height="850" Width="1220"
    AllowsTransparency="True" WindowStyle="None"
    >
    <!-- Die nachfolgenden Zeilen sorgen dafür, dass
         man das Fenster mit der Maus vergrößern,
         verkleinern und verschieben kann -->
    <WindowChrome.WindowChrome>
        <WindowChrome />
    </WindowChrome.WindowChrome>
    ...
</Window>
```

Jetzt fehlen nur noch ein paar zusätzliche Steuerelemente, um den Nutzern zu erlauben, das Fenster zu minimieren (`Me.WindowState = WindowState.Minimized`) oder zu schließen (`Me.Close`).

Bewegte Bilder im GIF-Format

Animationen lassen sich hervorragend dazu nutzen, Abläufe oder Sachverhalte klar darzustellen. Davon profitieren textuelle Beschreibungen aller Art, wie zum Beispiel Dokumentationen oder FAQ-Bereiche. Mit ScreenToGif können im Handumdrehen kleine Screencasts aufgenommen und im GIF-Format abgespeichert werden.



Text ist im allgemeinen ein gutes Medium, um etwas zu erklären. Es ist sicherlich noch immer die vorherrschende Methode, um Informationen zu übertragen. Selbst in Zeiten von immer besserer Audio- und Video-Übertragungen ist das sicherlich keine Übertreibung.

Wird Text allerdings nicht gedruckt, sondern liegt in digitaler Form vor, bieten sich noch weitere Möglichkeiten, Informationen in einen Beitrag einzubetten. Natürlich stehen die bereits angesprochenen Audio- und Video-Daten wieder zur Verfügung.

Allerdings ist das nicht immer der beste Weg. Zum Beispiel bei Online-Artikeln und Blogposts, bei denen Videos häufig als störend empfunden werden.

Nicht zu vergessen der Einsatz als [Internet Meme](#), um einen Sachverhalt durch eine kurze Animation zusammenzufassen oder zu erklären, häufig auf eine pointierte Art und Weise.

Vom Problem zur Lösung

Ein Problem bei eingebetteten Videos ist nämlich, dass sie den Lesefluss und den Informationsfluss unterbrechen. Stellen Sie sich vor, der Artikel würde an dieser Stelle enden, die nächsten ein bis zwei Seiten wären als Video vorhanden und dann ginge der Text weiter. Eine sehr gruselige Vorstellung. Mit Audiodaten besteht das nahezu identische Problem.

Es gibt allerdings ein Zwischenformat, das zwar wie ein Video daherkommt, allerdings nur ein animiertes Bild ist. So lassen sich Teilinformationen, zum Beispiel in einem Tutorial, auf animierte Weise darstellen, ohne dass sich Leser und Leserinnen erst ein minutenlanges Video anschauen müssen. Gerade bei kurzen Beispielen hilft diese Art der Kombination von Informationen durch eine kurze Animation und den zugehörigen Text enorm beim Verständnis.

Programme, mit denen der Bildschirm aufgenommen und in einer GIF-Datei gespeichert wird, erfreuen sich in letzter Zeit immer größerer Beliebtheit. Insbesondere in Blogposts sind sie äußerst beliebt.

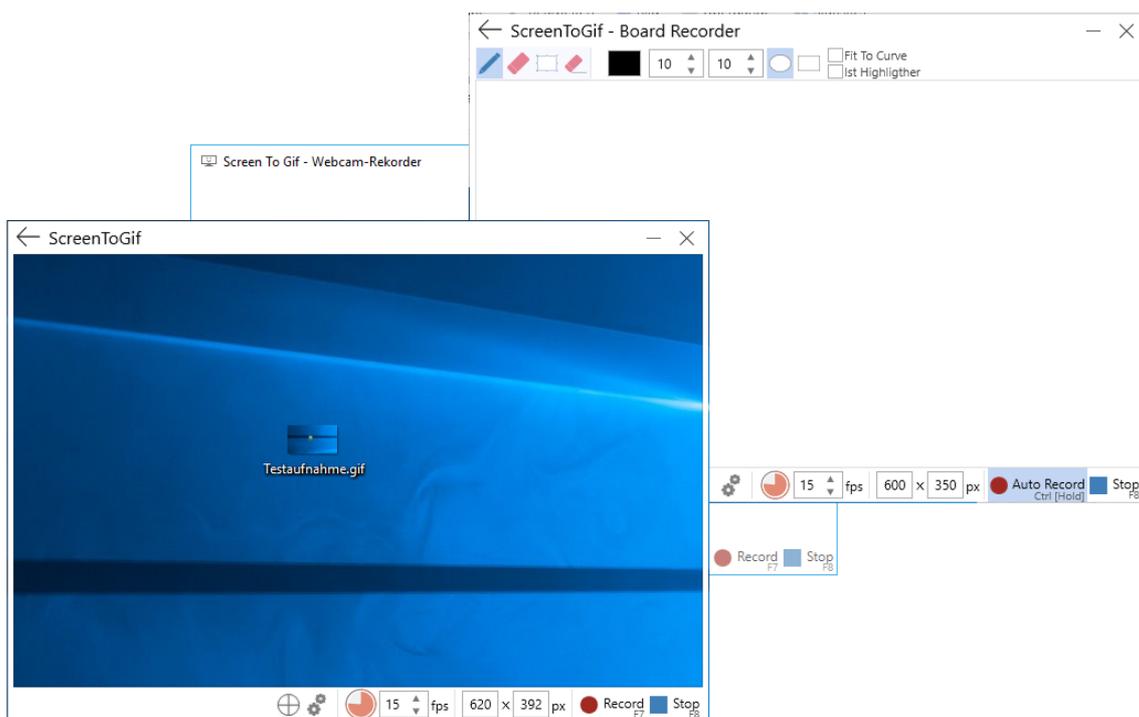
Der Vorteil ist, dass für moderne Browser diese animierten GIF-Dateien keinerlei Probleme mehr darstellen. Ebenfalls von Vorteil ist der geringe Speicherbedarf und damit die geringere Belastung für die Bandbreite, verglichen mit einem vollwertigen Video.

Der Artikel stellt im Folgenden das Tool ScreenToGif vor. Am Ende befindet sich zudem eine Übersicht über weitere Programme, die einen

ähnlichen Funktionsumfang haben. Die Fülle der Anwendungen macht es leider unmöglich, diese alle vollständig in einem Artikel vorzustellen.

Was ist ScreenToGif?

Die Anwendung [ScreenToGif](#) trägt einen sehr markanten Namen, der den Sinn und den Funktionsumfang des Tools fast vollständig beschreibt. Primär ist es damit möglich, den Bildschirm, das Bild einer Webcam oder einen im Programm eingebetteten Zeichenbereich aufzuzeichnen und zum Beispiel im GIF-Format abzuspeichern (siehe Bild 1). Als Bonus ist ein Editor integriert, mit dem sich das aufgezeichnete Material direkt in der Anwendung bearbeiten lässt.



Die verschiedenen Hauptfunktionen von ScreenToGif (Bild 1)

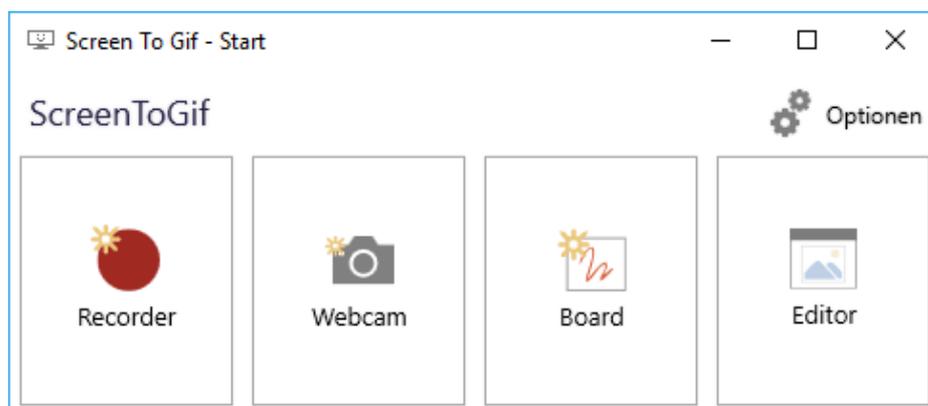
ScreenToGif ist auf der einen Seite aufgrund seiner zahlreichen Features so beliebt. Auf der anderen Seite aber auch deshalb, weil die Anwendung komplett kostenfrei ist und als Open-Source-Projekt im Quelltext zur Verfügung steht. Das Projekt wird auf GitHub in einem [Git](#)

[Repository](#) gehostet und steht unter der MS-PL Lizenz. Entwickelt wird ScreenToGif hauptsächlich von [Nicke Manarin](#).

Installation

Die Installation von ScreenToGif ist nicht der Rede wert, da es keine dedizierte Installation gibt. Heruntergeladen wird das Programm als zip-Archiv, in dem sich eine ausführbare Datei befindet.

Die so heruntergeladene Datei kann einfach gestartet werden. Voraussetzung ist das .NET Framework 4.6.1. Weitere Abhängigkeiten gibt es nicht. Nach einigen Sekunden erscheint das Startfenster, das in Bild 2 zu sehen ist. Darüber starten Sie die Aufzeichnung des Bildschirms, der Webcam, des Zeichenbereichs oder den Editor oder nehmen Einstellungen vor.

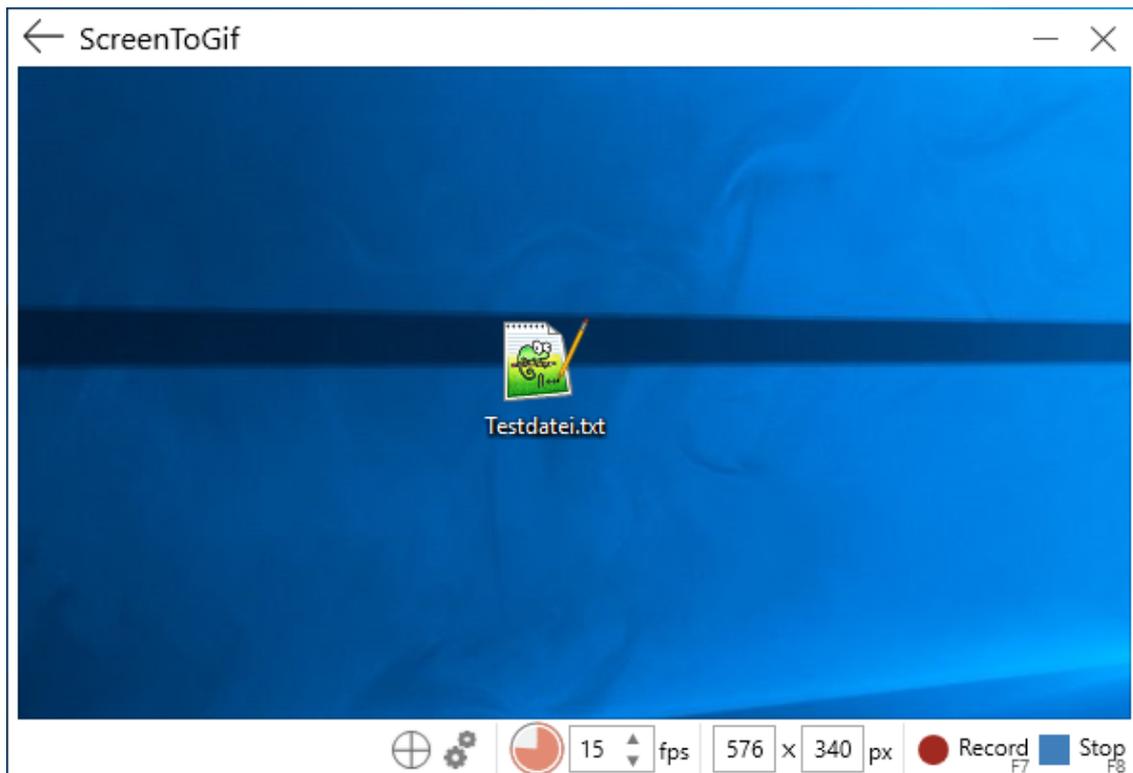


Das Startfenster von ScreenToGif (Bild 2)

Aufnehmen, Bearbeiten und Speichern

Die Hauptfeatures der Anwendung konzentrieren sich auf die bereits genannten Komponenten des Rekorders, der Webcam-Aufzeichnung, des Zeichenbereichs und des Editors. Dieser Artikel konzentriert sich auf den Rekorder als Hauptfunktion. Einmal im Startfenster ausgewählt, ergibt sich Bild 3. Die Bild zeigt keine optische Täuschung oder etwas in der Art. Der Hauptbereich des Fensters ist durchsichtig,

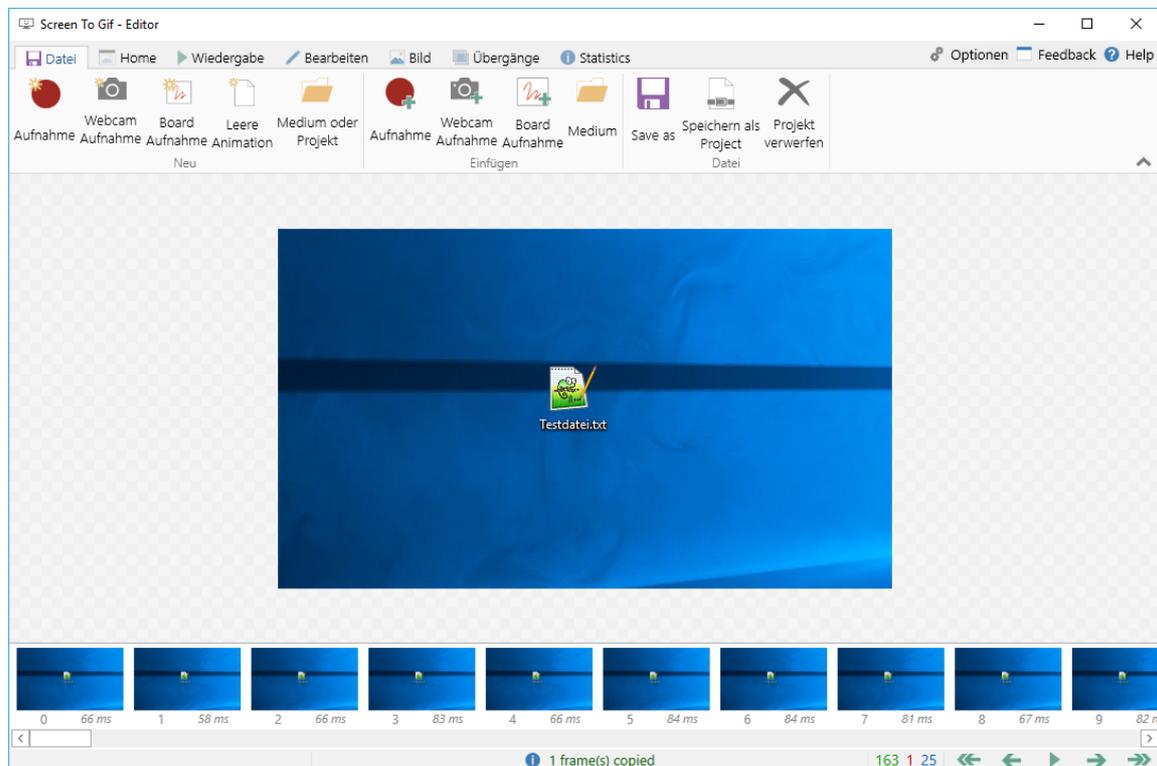
so dass die darunterliegende Anwendung zu sehen ist. Ganz genau betrachtet handelt es sich beim Rekorder-Fenster um den Rahmen, da der mittlere Bereich nicht nur durchsichtig ist, sondern alle Aktionen zum darunterliegenden Fenster durchlässt. Eine sehr interessante und gut durchdachte Funktion, da es so kinderleicht ist, den Bereich auszuwählen, der von ScreenToGif aufgezeichnet werden soll.



Der Rekorder mit dem relevanten Bildausschnitt in der Mitte (Bild 3)

Das weitere Handling zum Aufnehmen ist denkbar einfach. Am unteren Bildschirmrand befindet sich häufig notwendige Optionen wie die Frames pro Sekunde, die Größe des aufgenommenen Bereichs und Schaltfläche beziehungsweise deren Tastaturkürzel zum Aufnehmen (F7) und Stoppen (F8) einer Aufnahme. Läuft eine Aufnahme, ist das ganz deutlich an den durchlaufenden Frames in der Titelleiste von ScreenToGif zu erkennen. Wird die Aufnahme gestoppt, schließt sich das Rekorder-Fenster und der Editor wird mit der gerade

durchgeführten Aufnahme geöffnet (siehe Bild 4). Das Editorfenster bietet zahlreiche Optionen, um die Aufnahme zu bearbeiten. Ganz konkret können die einzelnen Frames, die am unteren Bildschirmrand zu sehen sind, durchlaufen und nach Belieben verändert werden.

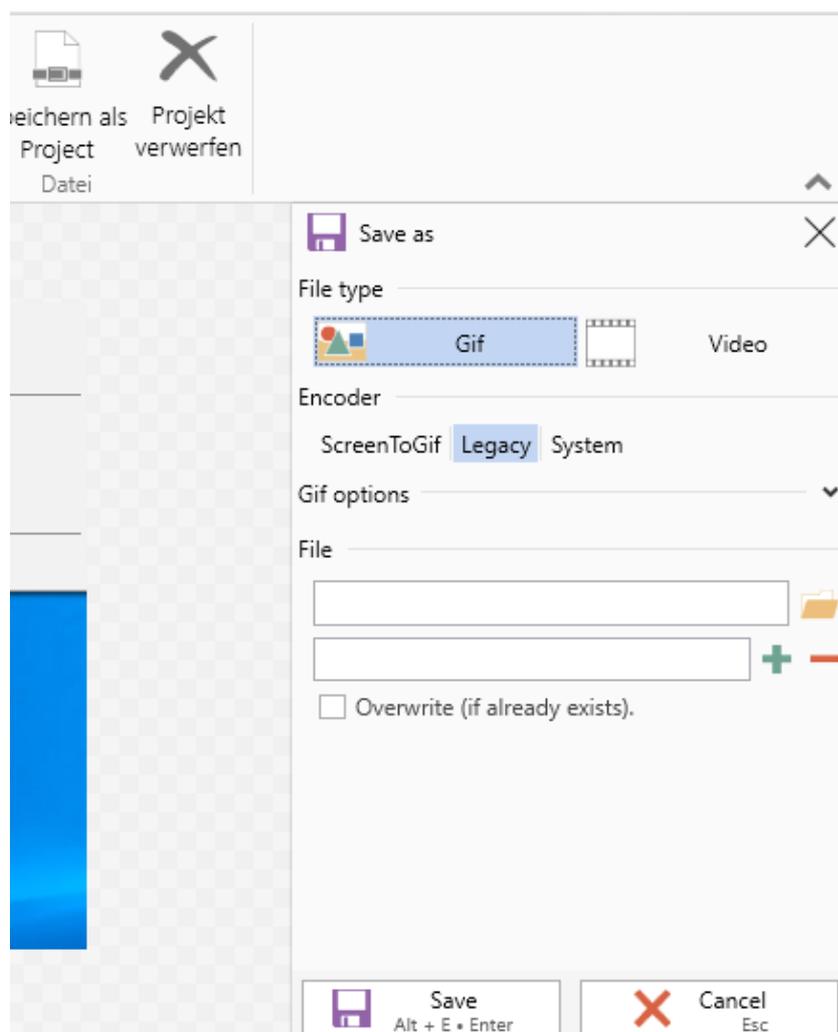


Die Testaufzeichnung im Editor (Bild 4)

Die im Rahmen dieses Artikels durchgeführte Beispielaufnahme ist zum Beispiel deutlich zu lang. Aufgenommen wurde, wie eine leere Textdatei mit Namen Testdatei.txt über das Kontextmenü vom Desktop eines Windows 10 Systems gelöscht wird. Von den aufgenommenen 162 Frames sind aber etliche redundant, weil sie keine Veränderung im Bild darstellen. Für animierte Bilder, die in Beiträge wie zum Beispiel Tutorials eingebunden werden sollen, ist es allerdings sehr relevant, dass die Animation genau den Kern des Pudels zeigt.

Steht die Animation etwas länger still als sie muss, wundert sich der Betrachter, ob etwas nicht richtig ist. Zudem stehen Animationen selten

für sich allein. In der Regel sollen sie eine Textpassage inhaltlich unterstützen. Hierfür ist ebenfalls eine zeitlich kurze Animation von Vorteil, die nur das Wesentliche zeigt. Frames lassen sich über die Auswahl am unteren Bildschirmrand löschen. Frame oder Framebereich auswählen, entfernen drücken und weg sind sie. Auf diese Weise konnte die Beispielaufnahme auf 52 Frameworks gekürzt werden. Sind die Bearbeitungen abgeschlossen, kann die Aufnahme als ScreenToGif-Projekt oder als GIF- beziehungsweise Video-Datei gespeichert werden (siehe Bild 5). Dabei stehen unterschiedliche Kodierungsverfahren zur Verfügung. Mit dem Legacy-Encoder ist das Resultat 176 Kilobyte groß und lässt sich mit den üblichen Bildbetrachtern anzeigen.



Die verschiedenen Möglichkeiten, das aufgezeichnete Material abzuspeichern (Bild 5)

Weitere Features des Editors sind unter anderem die Möglichkeit, Frames in der Größe zu verändern, zu stutzen oder zu spiegeln. Eigener Text ist ebenso möglich wie freies Zeichnen, das Einfügen von Wasserzeichen oder Fortschrittsbalken. Gerade letztes kann sich auszahlen, da Betrachter der Animationen sofort wissen, wie lange ebendiese Animation noch dauert.

Der Blick ins Repository

ScreenToGif ist nahezu vollständig in C# geschrieben. Als Oberflächen-Technologie kommt die Windows Presentation Foundation (WPF) zum Einsatz. Wenn es allerdings darum geht, mit Screenshots zu arbeiten - eine Aufzeichnung findet als Reihe von sequenziellen Screenshots statt - wird direkt auf Windows-API-Funktionen zurückgegriffen.

Wer sich für diese Implementierung interessiert, wird in der [Klasse ImageCapture in der Datei ImageCapture.cs](#).

Einen weiteren Blick wert sind die Implementierungen zu den Decodern und Encodern, die sich im [Verzeichnis ImageUtil](#) befinden.

Weitere Tools im Überblick

Am Anfang des Artikels wurde es bereits angekündigt, dass ein einzelner Beitrag nicht ausreicht, um alle Programme vorzustellen. ScreenToGif ist nämlich beileibe nicht die einzige Anwendung, die den Bildschirm aufzeichnen und im GIF-Format speichern kann.

Eine weitere Anwendung ist [Recordit](#), die ziemlich anders an das Thema herangeht. Der Download ist schnell erledigt und die Installation geht ebenfalls flott über die Bühne. Nach dem Start der

Anwendung ist nur ein Symbol in der Info-Leiste von Windows zu sehen. Mit einem Klick wird eine Selektion gestartet, um den Bildschirmbereich auszuwählen, der aufgenommen werden soll. Nach der Aufnahme öffnet sich nicht etwa ein Editor oder eine sonstige Möglichkeit zur Bearbeitung, sondern eine [Webseite mit einer eindeutigen URL](#). Das aufgezeichnete Material wird nämlich direkt hochgeladen und steht dort denjenigen zur Verfügung, die von der URL Kenntnis haben. Auf der einen Seite ist das schön puristisch, auf der anderen Seite muss jeder für sich überlegen, ob der automatische Upload für die Aufnahme überhaupt in Frage kommt.

[GifCam](#) ist ein weiteres Tool, das sich bei der Handhabung stark an ScreenToGif orientiert. Es kommt als einzelne ausführbare Datei und bringt einen rudimentären Editor mit. Für kleinere Aufgaben ist es gut einsetzbar. Wer ein paar mehr Features benötigt, gerade was den Editor betrifft, sollte direkt zu ScreenToGif greifen.

Und zu guter Letzt noch eine Auflistung all derer Anwendungen, die aus Platzgründen bisher komplett außen vor bleiben mussten:

- Capture Gif
- GIF Brewery 3
- Gif Screen Recorder
- GYAZO
- LICEcap
- Camstudio
- Icecream Screen Recorder
- Gif Recorder

Die Auflistung ließe sich beliebig fortsetzen. Interessant ist, dass es zahlreiche Anwendungen nur für Mac OS X gibt. Ob das daran liegt, dass Mac User häufiger Bedarf an GIF-Animationen haben oder ob es

durch den App Store einfacher ist, Anwendungen an den Mann und die Frau zu bringen, ist schwer zu sagen. Glücklicherweise hat sich das in den letzten Jahren geändert, so dass es mittlerweile eine ebenfalls stattliche Zahl von Screen Rekordern für die Windows Plattform gibt.

Fazit

Der Autor des Artikels ist ein großer Fan von ScreenToGif, was einer der Gründe ist, warum das Tool in diesem Artikel genauer unter die Lupe genommen wurde. Allerdings ist es auch nur ein Beispiel unter vielen, da zahlreiche andere Anwendungen einen ebenso großen Funktionsumfang bieten und ohne Probleme als Ersatz dienen können.

Allgemein hat sich der Einsatz von kurzen Animationen im GIF-Format mindestens gefühlt drastisch erhöht. Nicht nur als Meme, um einen Sachverhalt auf den Punkt zu bringen. Auch in Blogposts beziehungsweise Online-Texten aller Art kommen die Animationen immer häufiger vor. Wenn sie gut gemacht sind, können sie die Informationen im Text unterstützen und anreichern. Quellen

Fabian Deitelhoff arbeitet nach seiner Promotion zu "Source Code Comprehension" als Tech-Lead Domestic an Cloud-Themen bei Miele. Darüber hinaus ist er mit brickobotik in der MINT-Bildung und mit Loosely.Ruhr in der Cross-Plattform-Softwareentwicklung tätig. Seine Schwerpunkte sind Low- und No-Code sowie digitale Geschäftsmodelle. Daneben ist er als freier Autor unter anderem für die dotnetpro, Dozent und Softwareentwickler im .NET- und Web-Umfeld tätig. Sie erreichen ihn über deitelhoff.me, unter mail@deitelhoff.me oder auf Twitter als [@FDeitelhoff](https://twitter.com/FDeitelhoff).



A typical website visit in 2022:

1. Figure out how to decline all but essential cookies
2. Close the support widget asking if I need help
3. Stop the auto-playing video
4. Close the “subscribe to our newsletter” pop-up
5. Try and remember why I came here in the first place

@andybudd

.NET: Datenbank für Code-Snippets

Schon mal erlebt? Sie arbeiten an einem Projekt, stoßen auf ein Problem, wissen sofort, dass Sie die Lösung schon einmal wussten und beginnen fieberhaft mit der Suche.



In Büchern, in alten Projekten, im Web und so weiter. Eine Datenbank für Code-Snippets schafft hier Abhilfe.

Wer mit Code-Snippets arbeiten möchte, kann entweder die in Visual Studio eingebauten Möglichkeiten nutzen oder spezielle Tools dafür einsetzen. Mit [CodezBank](#) gibt es ein C++-Projekt, das Entwickler sbryndin schon 2008 gestartet hat und seither regelmäßig pflegt. Zuletzt wurde CodezBank Ende August 2022 an die neueste Version von Visual Studio angepasst.

Es handelt sich dabei um eine kleine Anwendung, die hierarchisch organisierte Quellcode-Schnipsel mit einfachen Suchfunktionen speichert. Man kopiert den Codeausschnitt und fügt ihn über die Zwischenablage in die Datenbank ein oder holt sie auf diesem Weg wieder heraus. [Details zum Projekt.](#)

DEV ACADEMY

HANDS-ON-WORKSHOPS UND
WEITERBILDUNG FÜR SOFTWARE-
ENTWICKLER UND -ARCHITEKTEN



Cloud und DevOps Hands-on Workshops

Kubernetes: Ab in die Cloud (2 Tage)

Erste Schritte
mit Docker für .NET (2 Tage)

Angewandte DevOps-Prozesse für
.NET Core Services mit Docker und Kubernetes (3 Tage)

Azure DevOps - CI/CD mit Azure Pipelines und Git (3 Tage)

Azure Pipeline - Multistage YAML Pipelines
für Kubernetes (3 Tage)

Verteilte Anwendungen bei Azure mit Docker
und Kubernetes (2 Tage)

DevOps mit GitHub **NEU** (3 Tage)

Git-Masterclass - Versionsverwaltung für die Praxis (3 Tage)

DevSecOps -
Security in den Prozess integrieren **NEU** (2 Tage)

REMOTE
ODER
INHOUSE



Weitere Informationen auf developer-academy.de
Preise auf Anfrage!

Ihre Ansprechpartnerin: Susanne Herl • +49 731 88005-8835 • susanne.herl@ebnermedia.de

Treffen sich künstliche Intelligenz und Sitcom

Na, **ChatGPT** schon ausprobiert? Die künstliche Intelligenz, die nicht nur mit einem chatten kann. Wir wollten wissen, ob sie auch in der Lage ist, Szenen einer Sitcom zu schreiben. Und in der Tat. Nach dieser Texteingabe:



Bitte schreiben Sie eine Szene in einer Sitcom, in der ein Softwareentwickler gegen Fehler in seinem Code kämpft.

gab die KI folgende Szene heraus. Immerhin, obwohl - lustig ist anders. Aber lesen sie selbst:

Szene: Die Hauptfigur, ein Softwareentwickler namens Dave, sitzt an seinem Schreibtisch vor seinem Computer. Er tippt hektisch auf seiner Tastatur und murmelt vor sich hin.

Dave: Komm schon, wo ist der Bug... Ich weiß, er ist hier irgendwo...

(Daves Arbeitskollegin Sarah betritt den Raum)

Sarah: Hey Dave, was gibt's? Du siehst aus, als hättest du einen harten Tag.

Dave: (frustriert) Habe ich auch. Ich arbeite schon seit Stunden an diesem Projekt und kann diesen blöden Fehler nicht finden.

Sarah: Oh nein, was ist denn das Problem?

Dave: (erklärt) Nun, ich habe diesen Code geschrieben, um eine Zufallszahl zu generieren, aber jedes Mal, wenn ich ihn ausführe, erhalte ich immer dieselbe Zahl. Das macht mich wahnsinnig.

Sarah: (mitfühlend) Ich weiß, wie du dich fühlst. Das habe ich auch schon erlebt. Hast du versucht, einen anderen Zufallszahlengenerator zu verwenden?

Dave: (seufzt) Ja, das habe ich versucht. Es hat nicht funktioniert.

Sarah: (denkt nach) Hmm... vielleicht solltest du eine Pause machen und deinen Kopf frei bekommen. Manchmal kann das helfen.

Dave: (entschlossen) Nein, ich kann jetzt nicht aufhören. Ich muss das herausfinden. (fängt wieder an zu tippen)

(Ein paar Minuten vergehen)

Dave: (aufgeregt) Ich habe ihn gefunden! Ich habe den Fehler gefunden!

Sarah: (beeindruckt) Wow, das ging aber schnell. Was war es denn?

Dave: (erklärt) Ich habe den Zufallszahlengenerator jedes Mal, wenn ich den Code ausgeführt habe, mit demselben Startwert initialisiert. Deshalb habe ich immer die gleiche Zahl erhalten.

Sarah: (amüsiert) Ah, der alte Trick mit dem gleichen Startwert. Auf den bin ich auch schon mal reingefallen.

Dave: (erleichtert) Nun, wenigstens habe ich es endlich herausgefunden. Jetzt kann ich mich dem nächsten Fehler widmen.

(Dave und Sarah klatschen sich ab)

Sarah: (aufmunternd) Du schaffst das, Dave. Du bist ein Softwareentwickler, ein Meister des Codes.

Dave: (zuversichtlich) Verdammt richtig, das bin ich. Her mit den Bugs. Ich werde sie alle zerquetschen.

(Sarah lacht und verlässt den Raum, während die Szene endet)

Windows Explorer: VS Code aus der Adresszeile aufrufen

In der Adresszeile zeigt der Windows Explorer den Pfad des dargestellten Ordnerinhalts an. Hier lässt sich auch ein Pfad aus der Zwischenablage einfügen, um zu diesem Ordner zu springen.



Über die Adresszeile lässt sich aber auch die Kommandozeile aufrufen, indem Sie *cmd* eintippen und mit [Return] bestätigen. Es öffnet sich die Kommandozeile – und zwar in dem Verzeichnis, das vom Explorer angezeigt wird. Schneller geht das nicht.

Nun liegt die Idee nahe, diesen Schnellstart mit folgender Eingabe auch für den Aufruf von Visual Studio Code zu nutzen:

```
code .
```

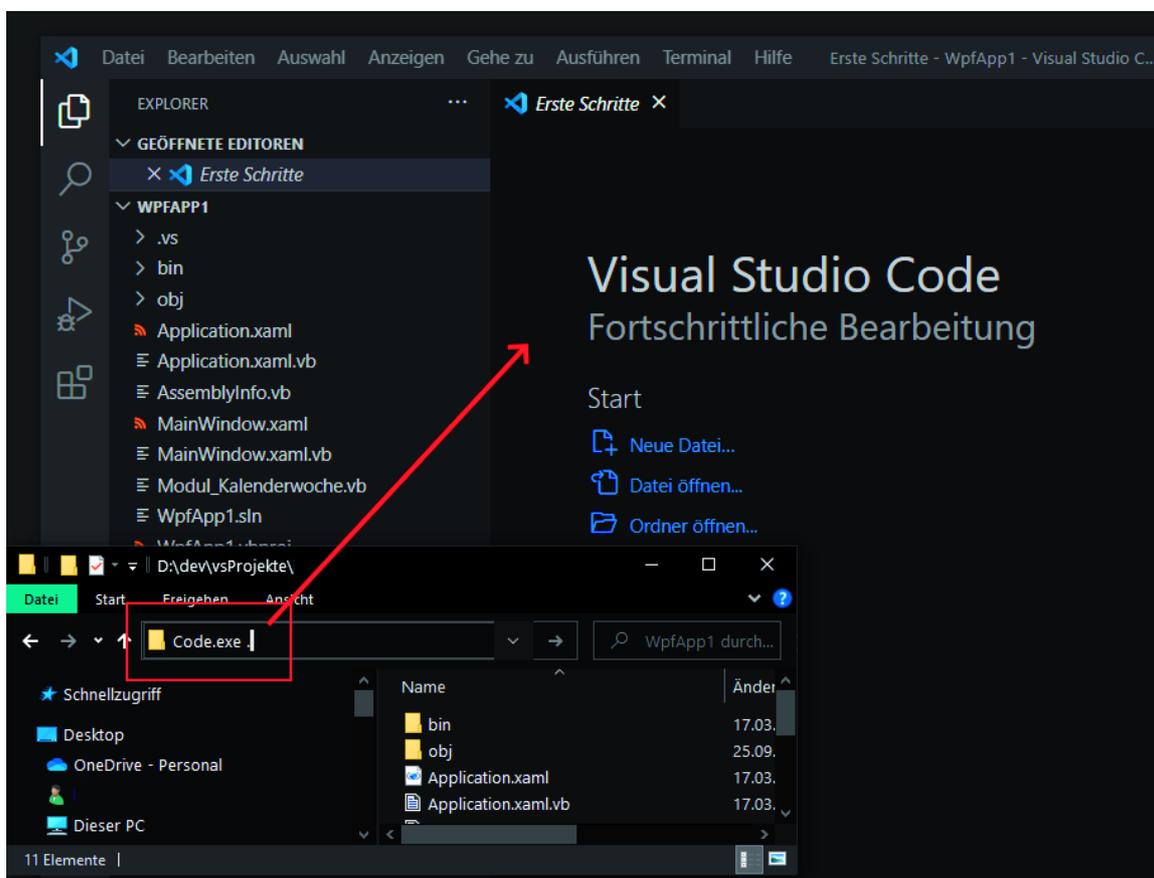
Diese Eingabe funktioniert jedoch zunächst nicht. Allerdings genügt es, den Speicherort des Visual-Studio-Code-Executables in den Systempfad aufzunehmen.

Drücken Sie dazu die Windows-Taste und geben Sie *Pfad* ein. Nach einem [Return] erscheint die Dialogbox für die Systemeigenschaften.

Wählen Sie dort den Knopf *Umgebungsvariablen...*. In der oberen Listbox der erscheinenden Dialogbox finden Sie den Eintrag *Path*. Fügen Sie nun über einen Klick auf *Bearbeiten...* den Pfad zum Programm *Code.exe* hinzu und speichern Sie die Änderungen. Geben Sie anschließend in die Adresszeile

```
code.exe
```

ein, so öffnet sich der Editor (Bild 1).



VS Code aus der Adresszeile des Windows-Explorers starten (Bild 1)

Auch die Funktion mit dem Punkt funktioniert.

```
code.exe .
```

öffnet den aktuellen Ordner in Visual Studio Code als Projekt-ordner. Leider ist die Angabe der Namenserverweiterung *exe* notwendig.

Die wichtigsten Trends im Jahr 2023 für ...

Jahresendzeit ist die Zeit der Hellseher: Welche Trends werden uns im kommenden Jahr heimsuchen? Womit müssen wir rechnen? Wir haben mal zusammengestellt, was so in den letzten Tagen hereingekommen ist. <ironie>Teilweise sind die erwarteten Entwicklungen tatsächlich total überraschend</ironie>. Ein Ausschnitt.



... Datenpflege

Die Kommunikation mit den Kund:innen/Nutzer:innen

wird sich intensivieren: Unternehmen werden in Zukunft nicht nur transparenter vermitteln müssen, welche Daten konkret erhoben und gespeichert werden. Sie müssen auch zunehmend in den Dialog mit den Kund:innen gehen, um diese zu ermutigen, Daten zu teilen, sodass deren User Experience mit dem Produkt stetig optimiert werden kann.

Bewusstsein für Nachhaltigkeit

wird sich schärfen: Es werden sich neue Wege und Ansätze für die Messung und Kontrolle von Nachhaltigkeit in Unternehmen etablieren.

Außerdem werden sich innovative Möglichkeiten auftun, wie Daten zur Verbesserung der Nachhaltigkeitsbemühungen beitragen können und wie Geschäftsprozesse im Hinblick auf ein nachhaltiges Ziel anstelle eines traditionellen Umsatzziels optimiert werden können.

Datenaustausch zwischen Unternehmen

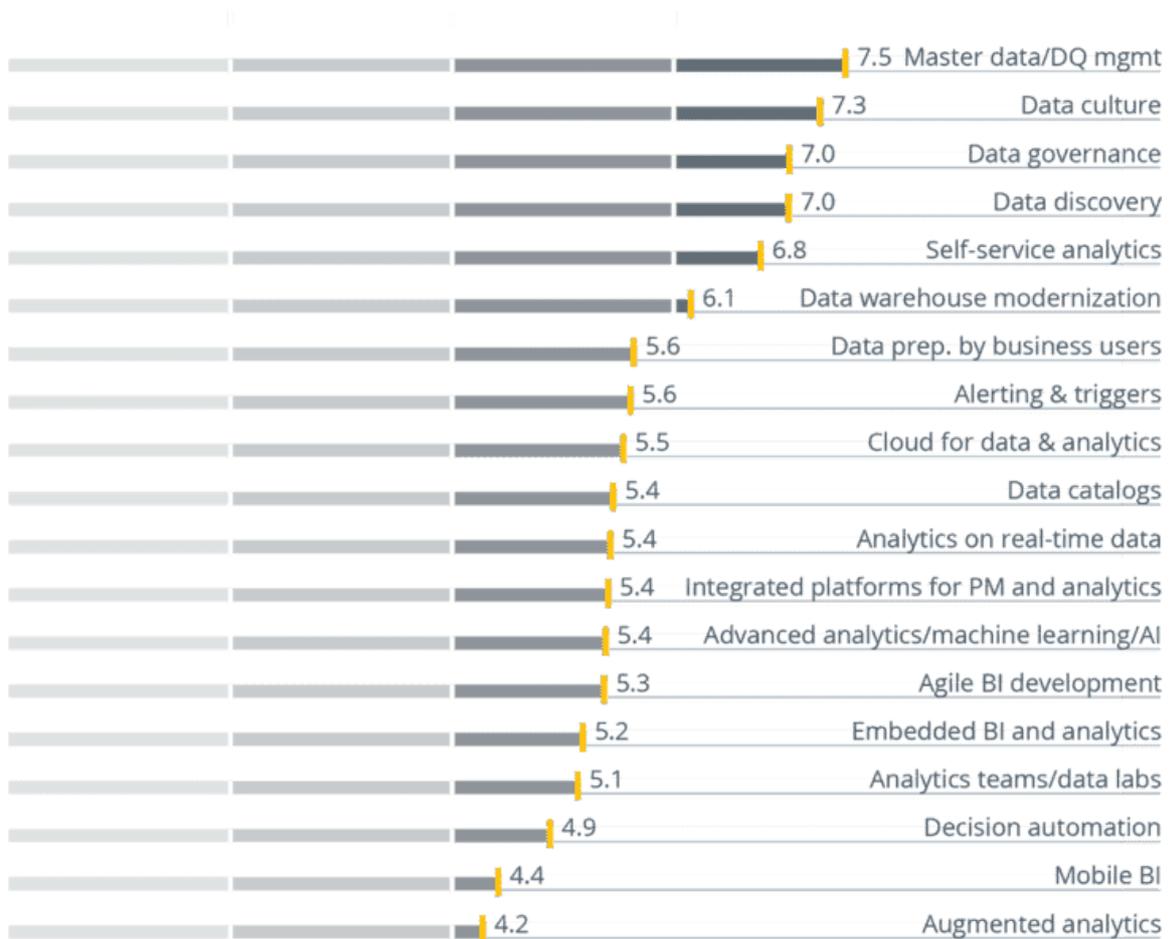
Traditionell halten Unternehmen ihre Daten unter Verschluss und kaum eines würde in Erwägung ziehen, Daten mit Konkurrenten zu teilen. Die Zusammenarbeit in diesem Bereich wird jedoch enorm zunehmen. Mit Hilfe von Datenaustauschdiensten können Unternehmen ihre eigenen Datensätze für andere Unternehmen zugänglich machen. Zwei Gründe sprechen dafür: Der offensichtlichste Grund ist, dass alle gesammelten Daten – auf natürlich Datenschutz-konformer Art und Weise – noch besser monetarisiert werden können. Der zweite Grund ist, dass die Entwicklung von KI-basierten Modellen stark von größeren Datensätzen profitieren wird, insbesondere mit dem Aufkommen generativer KI-Anwendungen.

Quelle: [Vinted](#)

... Datenqualität

Datenqualität und Stammdatenmanagement

werden im sechsten Jahr in Folge als wichtigster Trend der [Studie](#) “[Data, BI & Analytics Trend Monitor 2023](#)” eingestuft. Dies deckt sich mit den Ergebnissen anderer BARC-Studien, die immer wieder zeigen, dass Unternehmen mit unzureichender Datenqualität zu kämpfen haben. Damit zusammenhängend ist die stabile Bedeutung von Data Governance, die auch in diesem Jahr wieder an dritter Stelle rangiert.



Bedeutung von Trends im Bereich Data, BI und Analytics von „überhaupt nicht wichtig“ (0) bis „sehr wichtig“ (10) [n= 1,823, Copyright BARC]

Quelle: [Barc](#)

... Cybersecurity

Die Cyberkriminalität wird 2023 ein Ausmaß wie nie zuvor annehmen.

Die Angriffszahlen steigen aktuell schnell und das wird 2023 voraussichtlich so weiter gehen. Mit strengen Sicherheitskontrollen und dem Einsatz von Cybersicherheitslösungen entlang der kompletten Cyber-Kill-Chain müssen dafür jetzt Vorkehrungen getroffen werden.

Online-Betrug wird zunehmen.

Cyberkriminelle werden sich für ihre Betrugsmaschen Künstliche

Intelligenz (KI) zunutze machen und sie mit zielgerichteten Bots und Automatisierung kombinieren. Außerdem werden sie Deep-Fake-Technologien einsetzen, um mit künstlichen Identitäten zu betrügen sowie ihre Social-Engineering- und Phishing-Angriffe zu optimieren.

Sicherheitslösungen müssen Hybrid- und Multi-Clouds schützen.

Unternehmen werden bei der Cloud-Migration immer pragmatischer vorgehen und zunehmend auf Hybrid- und Multi-Clouds setzen. Diese werden zu den vorherrschenden Cloud-Betriebsmodellen avancieren. Entsprechend werden Sicherheitstools 2023 genauso flexibel wie diese Betriebsmodelle sein müssen.

Cyberkriminelle setzen verstärkt auf Machine Learning.

Cybersicherheitslösungen, die regelbasiert vorgehen, benötigen 2023 zusätzlich Funktionen für maschinelles Lernen (ML). Schon jetzt zeigt sich, dass Hacker vermehrt ML einsetzen, um Cybersicherheitstools automatisch zu erkennen und ihre Angriffe so anzupassen, dass sie die Verteidigungsmechanismen überwinden können. Ohne Sicherheitslösungen mit ML-Funktionen, die in Echtzeit neue Angriffsmethoden erkennen und darauf reagieren können, sind Unternehmen gefährlich ungeschützt.

Distributed-Denial-of-Service-Angriffe (DDoS-Angriffe) werden mehr, länger und umfangreicher.

Auch Staaten nutzen vermehrt DDoS-Angriffe als Ablenkungsmanöver und Mittel der Cyberkriegsführung, um unerkannt Daten zu stehlen – womit sie den eigentlichen Schaden anrichten. Daher ist es notwendig, dass Unternehmen aber insbesondere auch öffentliche Einrichtungen, sowohl DDoS-Angriffe abwehren als auch ihre Daten schützen können.

Viele Unternehmen werden versuchen, mit weniger Mitteln mehr zu erreichen, um der steigenden Inflation entgegenzuwirken.

Cybersicherheitsverantwortliche müssen ihre Budgets klug einsetzen und auf Automatisierung, KI und ML setzen, um die Effizienz ihrer Teams zu steigern. Sie werden außerdem ihren Cybersecurity-Stack konsolidieren. Fokussieren sie sich dabei auf Cloud-native Plattformen und -Technologien anstelle von Einzellösungen, hilft das, die Gesamtbetriebskosten (TCO) sowie die Ausgaben für Wartung und Schulungen zu senken. Gleichzeitig können sie so den Betrieb zukunftssicher machen und ihn vereinfachen.

Cyberkriminelle werden zunehmend API-Schwachstellen ausnutzen.

Die Nutzung von Application Programming Interfaces (APIs) wird weiter zunehmen. Der Trend zur Cloud-nativen App-Entwicklung wird dies vorantreiben, ebenso wie Initiativen von Regierungen und der Wirtschaft für mehr Datenaustausch und -portabilität. Cyberkriminelle werden diesem Trend folgen und mehr und mehr versuchen, API-Schwachstellen auszunutzen. Entwicklungs- und Sicherheitsteams müssen ihre Kräfte bündeln und die Sicherheit in den agilen Entwicklungsprozess einbinden.

Quelle: [Imperva](#)

... Recruiting

Viele Unternehmen können nicht länger klassisch rekrutieren.

Gut ausgebildete Arbeitnehmende sind mächtig wie nie zuvor. Sie können Bedingungen stellen und zwischen verschiedenen Angeboten wählen. Die Nachfrage nach ihnen steigt durch den demographischen Wandel und den Fachkräftemangel kontinuierlich. Dies wird für alle Unternehmen zum Problem, die nicht in der Lage oder nicht Willens sind, die Vorstellungen der neuen Generation an Young Professionals zu erfüllen. Ein negatives Arbeitnehmerimage wird zu einem immer

größeren Wettbewerbsnachteil. Weil sie ihre Traumkandidat:Inne nur mehr selten für sich gewinnen können, werden Unternehmen häufiger auf Quereinsteigende zurückgreifen oder auf Bewerbende, die noch nicht alle gewünschten Fähigkeiten mitbringen.

Re-Skilling ersetzt Umbrüche im Team.

Wenn sich Jobprofile und benötigte Kompetenzen verändert haben, war bislang der häufigste Ansatz, darauf mit Neueinstellungen zu reagieren. Stattdessen werden Unternehmen künftig verstärkt darüber nachdenken, wie sie vorhandene Mitarbeitende anders einsetzen, deren Fähigkeiten weniger relevant geworden sind oder deren Stärken in anderen Positionen noch wirksamer werden. Angesichts der wirtschaftlichen Lage bestimmt vielerorts Vorsicht das Vorgehen in der Personalplanung. Zudem wächst der finanzielle und zeitliche Aufwand von Recruiting-Prozessen rapide und übertrifft die Kosten von Lernen um ein Vielfaches. In Folge werden Unternehmen ihren Angestellten immer öfter Re-Skillings anbieten.

Der Skill-Gap zwischen Unternehmen wächst und wer Lernen nicht auf der Agenda hat, fällt zurück.

Um im Wettbewerb nicht zurückzufallen, wird der konstante Erwerb neuer Fertigkeiten zu einer Grundvoraussetzung für Unternehmen. Durch Digitalisierung, Automatisierung und das Bestreben zu „grünen Unternehmen“ sind viele Anforderungen von Morgen aktuell weder bekannt noch absehbar sind. Effektive Lernstrukturen im eigenen Unternehmen aufzubauen ist daher unabdingbar. Diese sind im Idealfall keine „out of the box“-Lösungen, sondern sind individuell auf das jeweilige Unternehmen angepasst. Wer den Wandel zur „Lernenden Organisation“ schafft, macht sich agil genug, um die Herausforderungen der Zukunft zu meistern.

Quelle: [Brights](#)

... Rechenzentren

Energiekrise überstehen

Das größte Problem, mit dem wir derzeit konfrontiert werden, sind die enormen Energiepreise. Die Kosten sind so in die Höhe geschneit, dass sie für große Energieverbraucher, wie Betreiber von Rechenzentren, zu einem echten Problem werden. Können sie diese Kosten an ihre Kunden weitergeben? Werden die Preise weiter steigen? Verfügen sie über den Cashflow, um dies in ihrem Geschäftsmodell zu bewältigen? Während das Argument für eine Strategie zur Erzeugung erneuerbarer Energien immer Nachhaltigkeit war, brauchen wir heute erneuerbare Energien in der Region, um die Versorgung der europäischen Länder vor allem aus Gründen der Energiesicherheit und der Kosten zu sichern. Microsoft geht beispielsweise bereits einen Schritt in diese Richtung. Sein Rechenzentrum in Dublin verfügt über Lithium-Ionen-Batterien, die für den Anschluss an das Stromnetz zugelassen sind, um die Netzbetreiber zu unterstützen, falls erneuerbare Energiequellen den aktuellen Bedarf nicht decken können.

Gestörte Lieferketten in den Griff bekommen

Die Corona-Pandemie hatte enorme Auswirkungen auf die globalen Lieferketten in vielen Sektoren. Nachdem die Pandemie abgeklungen war, wiegten sich die Unternehmen in falscher Sicherheit und glaubten, das Schlimmste hinter sich zu haben. Niemand hatte mit einem zweiten Schlag gerechnet, einer geopolitischen Krise, die sich für einige Lieferketten – insbesondere für die für den Bau von Rechenzentren wichtigen Halbleiter und Basismetalle – als noch fataler erweist. Als Wachstumsmarkt reagiert die Rechenzentrumsbranche sehr empfindlich auf Störungen in der Lieferkette, vor allem in einer Zeit, in der ein hohes Wachstum bevorsteht.

Komplexität reduzieren

Ein Rechenzentrum beherbergt eine Fülle unterschiedlicher Technologien – von HLK-Systemen über Maschinen- und Gebäudetechnik bis hin zu IT und Computern. Die Herausforderung besteht darin, solche hochkomplexen, voneinander abhängigen Umgebungen effizient zu verwalten. Zu diesem Zweck entwickeln Designer, Betreiber und Anbieter von Rechenzentren Systeme, die diese Komplexität reduzieren und gleichzeitig den geschäftskritischen Charakter von Anwendungen berücksichtigen. Die Modularisierung von Rechenzentren, bei der vorgefertigte, vorgeplante und vorintegrierte Einheiten an den Standort geliefert werden, ist eine Möglichkeit, die Planung und den Bau eines Rechenzentrums weniger komplex zu gestalten und gleichzeitig eine kürzere Time-to-Market zu realisieren.

Rechenzentren in die Fläche bringen

Bislang waren London, Dublin, Frankfurt, Amsterdam und Paris die traditionellen Standorte für Rechenzentren, entweder weil Unternehmen in diesen Städten ihren Hauptsitz haben oder weil es sich um Wirtschaftscluster mit einer Fülle von Telekommunikationsverbindungen handelt. Um eine hohe Servicequalität zu bieten und näher an den Bevölkerungs- und Wirtschaftszentren zu sein, wird es immer günstiger, Rechenzentren in weiteren Städten der wichtigsten Wirtschaftsnationen und in den Hauptstädten kleinerer Nationen zu bauen. Der Wettbewerb unter den Anbietern von Rechenzentren ist stark, so dass viele dieser Tier-II-Städte und -Länder bestehenden Betreibern Wachstum oder neuen Akteuren einen einfachen Marktzugang bieten. Aus diesem Grund tut sich gerade in Städten wie Warschau, Wien, Istanbul, Nairobi, Lagos und Dubai einiges. Diese Expansion ist jedoch nicht ohne Herausforderungen. Überlegungen zur Verfügbarkeit geeigneter Standorte, zur Stromversorgung und zu technischem Personal

erhöhen die Komplexität des Gesamtbetriebs eines Unternehmens. Ungeachtet solcher Herausforderungen werden jedoch weiterhin neue Märkte erschlossen, wobei viele Betreiber versuchen, einen First-Mover-Vorteil in sich entwickelnden Sekundärmärkten zu erzielen.

Quelle: [Eaton](#)

... Observability

Let's make 2023 less noisy!“

Um Lärm, Arbeit und Ermüdung zu reduzieren, brauchen Ingenieur:innen Scoring und Priorisierung, um feststellen zu können, welcher Alert gerade am wichtigsten ist. Observability-Tools helfen dabei, indem sie alle Vorfälle sortieren und diejenigen mit den potenziell gefährlichsten Folgen identifizieren. Wir gehen davon aus, dass AIOps im nächsten Jahr immer wichtiger werden werden, um Ingenieur:innen in die Lage zu versetzen, mögliche Probleme vorherzusagen, die Ursachen zu ermitteln und die Automatisierung zu deren Behebung intelligent voranzutreiben. Noch wichtiger ist, dass dies dazu beitragen wird, permanente Benachrichtigungen und die daraus resultierende Alarm-Müdigkeit sowie das Stresslevel der Teams zu reduzieren.

Software-Pipelines werden schneller: durch Automatisierung, Observability as Code und Everything as Code.

In einer Post-COVID-Welt, in der mehr als 50 Prozent der weltweiten Wirtschaftstätigkeit über digitale Kanäle abgewickelt wird, müssen Unternehmen immer reaktionsschneller werden. Vor allem Entwickler:innen müssen in der Lage sein, noch schneller von der Idee zur Implementierung und weiter zur Codebereitstellung zu gelangen. Durch die Kombination aus DevOps-Ansätzen, Microservices-Architekturen und automatisierten Tools können Entwickler:innen neue

Softwareversionen heute bei Bedarf viel häufiger und schneller bereitstellen. Ein Faktor, der 2023 einen großen Einfluss auf IT- und Developer-Teams haben wird.

Observability as Code/Everything as Code - Integration von Anfang an

Für 2023 erwarten wir den verstärkten Einsatz von Observability as Code, was bedeutet, dass Observability von Anfang an als Teil des Codes in die Systeme eines Unternehmens integriert wird. Mittel- und langfristig wird „Everything as Code“ eine der wichtigsten Lösungen für Systemintegrationen und Automatisierung in Unternehmen werden, da nur so die Wahrscheinlichkeit menschlicher Fehler maximal reduziert werden kann und den IT-Teams gleichzeitig die Möglichkeit gegeben wird, ihre Fähigkeiten für tiefergehende, anspruchsvollere Aufgaben einzusetzen.

Quelle: [New Relic](#)

... IT

Metaverse: neue Chancen fürs Business

Dem Metaverse stehen noch immer viele skeptisch gegenüber. Dabei bietet es Chancen für Unternehmen: Avatare und Chatbots, um die Kundenbetreuung zu verbessern, Gamification für Schulungen oder Augmented Reality (AR) für mehr Shopping Experience. Gartner prognostiziert, dass bis 2027 über 40 Prozent der großen Unternehmen weltweit in Metaverse-basierten Projekten eine Kombination aus Web3, AR-Cloud und digitalen Zwillingen einsetzen werden. »Das Metaverse bietet für Unternehmen Möglichkeiten, sich mit Kunden auf ungewöhnliche Weise zu vernetzen und Umsätze zu steigern«, sagt Ralf Klotzbücher, Dätwylers VP Sales Europa. »Es

lohnt sich auszuloten, welche Optionen für das eigene Unternehmen am vielversprechendsten sind.«

Nachhaltigkeit braucht einen Rahmen.

Das Thema Nachhaltigkeit steht im Mittelpunkt fast aller Business-Entscheidungen. Doch wie gelingt die effiziente Umsetzung in der Praxis? Laut Gartner benötigen Unternehmen ein »Sustainable Technology Framework«, also ein Rahmenwerk, das aus verschiedenen digitalen Lösungen besteht, die sie beim Erreichen der Nachhaltigkeitsziele unterstützen. Dazu gehören Analytics und KI, außerdem das Tracking von Gütern und spezifizierte Softwarelösungen für die Unternehmensführung. »Ein umfassendes Energiekonzept ist wichtig, um die Nachhaltigkeitsziele zu erreichen. Dazu zählt auch, die Energieeffizienz in Rechenzentren zu erhöhen«, ergänzt Ralf Klotzbücher. »Mit kompakten Edge-Rechenzentren sind Unternehmen gut aufgestellt, um in diesem Bereich Energie und Kosten zu sparen.«

Auf dem Vormarsch: Künstliche Intelligenz

Künstliche Intelligenz (KI) erobert immer mehr Unternehmensbereiche, von der Produktion bis zum Kundenkontakt. Viele Unternehmen beherrschen ihre KI-Risiken jedoch nicht ausreichend. Laut Gartner waren bereits 41 Prozent von einer Datenschutzverletzung oder einem Sicherheitsvorfall durch KI betroffen. Daher ist es ratsam, Funktionen zu implementieren, mit denen sich die Zuverlässigkeit, Vertrauenswürdigkeit, Sicherheit und der Datenschutz beim Einsatz von KI gewährleisten lässt. Eine Kooperation der verschiedenen Geschäftsbereiche ist notwendig, ebenso der Einsatz adaptiver KI-Systeme. Diese nutzen datenbasiertes Realtime-Feedback, um das eigene Lernen dynamisch zu verändern und die Ziele anzupassen.

Stabil und sicher dank digitaler Immunsysteme

Die Entwicklung widerstandsfähiger Anwendungen gewinnt durch die

Digitalisierung immer mehr an Bedeutung. Klassische Ansätze der Software-Entwicklung beeinträchtigen jedoch das Erstellen von skalierbaren, sicheren und stabilen Systemen. Es braucht Methoden wie Observability, Site Reliability Engineering und Chaos Engineering, um den Schutz von Infrastrukturen und Daten zu optimieren. Das Zusammenspiel von Analytics und KI erleichtert das Testen der Tools und verbessert gleichzeitig die End-to-End-Sicherheit der gesamten Lieferkette. Gartner schätzt, dass Unternehmen, die in digitale Immunität investieren, ihre Systemausfälle bis 2025 um bis zu 80 Prozent reduzieren werden.

Wachsende As-a-Service-Prozesse und smarte Produkte

Laut der International Data Cooperation (IDC) wird zukünftig ein wichtiger Fokus auf As-a-Service-Produkten und -Prozessen liegen. Physische Ressourcen wie Hard- und Software müssen somit nicht länger angeschafft werden, und auch die Ausgaben für Wartung und Erneuerung der Systeme können entfallen. Dies reduziert betriebliche Belastungen und ermöglicht Unternehmen einen schnelleren Zugang zu Innovationen. »Steigende Anforderungen an die Agilität, Automatisierung und Orchestrierung der IT führen dazu, dass Unternehmen immer häufiger zu As-a-Service-Angeboten greifen«, erläutert Ralf Klotzbücher. IDC schätzt, dass Unternehmen bis 2024 mehr als drei Viertel ihrer Anwendungen und mehr als 50 Prozent ihrer Rechenzentrumsinfrastruktur »As-a-Service« nutzen werden. »Viele bestehende IT- und OT-Infrastrukturen sind dafür nicht ausgelegt und müssen daher überholt werden«, so Klotzbücher.

Quelle: [Dätwyler IT Infra](#)

VS 2022: JSON als Klassen einfügen

Eines der besonders nützlichen Helferlein in Visual Studio 2022 legt die Klassenstruktur zu JSON-Daten automatisch an.



Es kommt immer wieder vor: Eine WebApp stellt ein API zur Datenübernahme zur Verfügung. Die JSON-Daten einzulesen ist noch einfach, aber dann beginnt die langweilige und fehlerträchtige Tipparbeit, denn es muss eine Klassenstruktur für die JSON-Daten angelegt werden.

Eigentlich ist das eine überflüssige Arbeit, denn in den Daten steckt die Struktur ja bereits drin. In der aktuellen Version kann Visual Studio die Klassenstruktur direkt aus den Daten ermitteln und in das Programm einfügen.

Das klappt schlicht durch den Aufruf des Menüpunkts *Bearbeiten | Inhalte einfügen | JSON als Klassen einfügen*. Findet Visual Studio im Clipbord eine gültige JSON-Struktur, wird der Klassen-Code daraus erzeugt. Hier ein simples Beispiel, zunächst der JSON-Text:

```
{  
  "colors": [  

```

```

    {
      "color": "black",
      "category": "hue",
      "type": "primary",
      "code": { ... }
    },
    {
      "color": "white",
      "category": "value",
      "code": { ... }
    }, ...
  ]
}

```

Daraus macht Visual Studio nach Aufruf des Menüpunkts *Als JSON einfügen* die folgende Klassenstruktur:

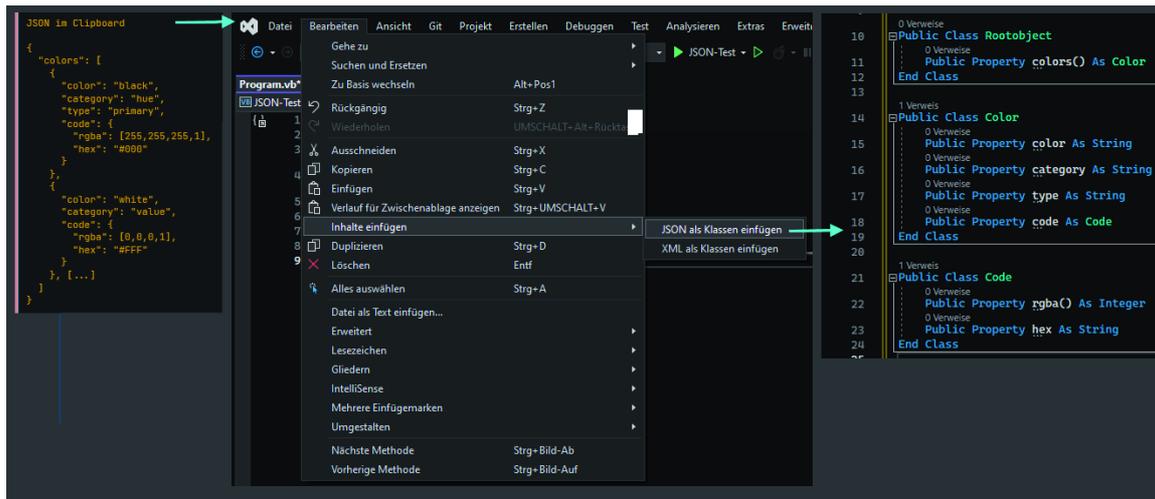
```

public class Rootobject
{
    public Color[] colors { get; set; }
}

public class Color
{
    public string color { get; set; }
    public string category { get; set; }
    public string type { get; set; }
    public Code code { get; set; }
}

public class Code
{
    public int[] rgba { get; set; }
    public string hex { get; set; }
}

```



VS Code aus der Adresszeile des Windows-Explorers starten (Bild 1)

Wie in Bild 1 zu sehen ist, funktioniert das Ganze auch mit Visual Basic sowie mit im Clipboard abgelegtem XML-Code.



"Does anyone use C in production?"

@d_feldman

"Is this a joke?"

@shanselman