

wiki2book

Aus Wikipedia eigene eBooks bauen

Hauke Stieler

27. Dezember 2022

Motivation
●○○○○○○

Wikipedia
○○○○○○○○○○○○○○○○

Technischer Aufbau
○○

Funktionsweise
○○○○○○○○○○○○

Take aways
○○○

Wikipedia hole

Kennt ihr das?

Wikipedia hole

[☰ WIKIPEDIA](#)

Echter Clownfisch

Art der Gattung Anemonenfische (Amphiprion)

[Artikel](#) [Diskussion](#)

[Sprache](#) [Beobachten](#) [Versionsgeschichte](#) [Bearbeiten](#) [Mehr](#)

Der **Echte Clownfisch** (*Amphiprion percula*), auch **Trauerband-Anemonenfisch** genannt, lebt an der Küste Nord-Queenslands (im nördlichen Great Barrier Reef), an der Nordküste Neuguineas und in den Korallenriffen Melanesiens (Neubritannien, Neuirland, Salomonen und Vanuatu).

☰ Inhaltsverzeichnis ▼

[▲ Merkmale](#)



Farbmorph mit hohem Schwarzanteil

Der Echte Clownfisch wird sechs bis elf Zentimeter lang. Die Länge beträgt das 2,1 bis 2,4 fache der Körperhöhe. Die Tiere sind leuchtend orange gefärbt, mit drei weißen Querstreifen, der mittlere mit einer nach vorne gerichteten Ausbuchtung. Die Streifen sind im Unterschied zu denen von *Amphiprion ocellaris* oft deutlich schwarz begrenzt. Die schwarzen Begrenzungen variieren in der Breite und können auch ineinander übergehen. Die Intensität der Streifung allein lässt eine sichere Artidentifizierung jedoch nicht immer zu. So gibt es

Trauerbandanemonenfische mit kaum begrenzten Streifen und umgekehrt Orangeringefische, deren Schwarzfärbung deutlicher hervortritt, bis hin zum Extremfall, melanistischen Farbmorphen. Daniel Knop empfiehlt zur Artdifferenzierung, die Anzahl der Hartstrahlen der oberen Rückenflosse

Brutpflegendes Paar



Systematik
Bibliographie

Barschverwandte
(Perciformes)
Ovalentaria

< □ ▶ < □ ▶ < □ ▶ < □ ▶ < □ ▶

Wikipedia hole

☰ WIKIPEDIA Q Wikipedia durchsuchen

☰ WIKIPEDIA Q Wikipedia durchsuchen

E

A Symbioseanemone

A Seeanemone

—

☰ Artikel Diskussion

D Sprache

Beobachten

Versionsgeschichte

Bearbeiten

Mehr

R Symbioseanemonen sind Seeanemonen (Anthozoa), die mit andersartigen Lebewesen in Symbiose leben. Am bekanntesten ist die Symbiose mit den Anemonenfischen (Amphiprion).

Die Symbioseanemonen bieten den Anemonenfischen, die alle schlechte Schwimmer sind, Schutz vor Raubfischen. Auch die Anemonenfische schützen ihre Symbiospartner vor Fressfeinden, z. B. Falterfischen. Annahmen, die Fische würden ihre Partner füttern, konnten nicht bestätigt werden, dagegen werden Symbioseanemonen, deren Fischpartner weggefangen wurden, bald von Falter- oder Feilenfischen gefressen. Die Anemonen werden sehr groß: 30 Zentimeter bis 1,5 Meter im Durchmesser. Alle Symbioseanemonen beherbergen auch Zoxanthellen, symbiotische Algen, die zur Ernährung des Wirtes beitragen.

Symbioseanemonen leben im tropischen Bereich des Indopazifik im Flachwasser der Korallenriffe.

☰ Arten

Arten, die Wirte für Anemonenfische und/oder andere Arten sind. Zu dieser Gruppe gehören zehn Arten, die nicht alle näher miteinander verwandt sind.

- Familie: *Actiniidae*
 - Blasenanemone, Kupferanemone (*Entacmaea quadricolor*)
 - Korkenzieheranemone (*Macrodactyla doreensis*)
- Familie: *Stichodactylidae*
 - Glasperlen-Anemone (*Heteractis aurora*)

Blasenanemone (*Entacmaea quadricolor*) mit Rotmeer-Anemonenfisch (*Amphiprion bicinctus*) und Dreipunkt-Preuerfische (*Dascyllus trimaculatus*)



CC BY-SA 3.0 Wikipedia contributors; <https://de.wikipedia.org/wiki/Symbioseanemone>

Wikipedia hole

Screenshot of a Wikipedia article on Sea Anemones (*Actiniaria*). The page shows the main content area with sections like 'Merkmale' and 'Geographisches Vorkommen, Verbreitung und Lebensweise'. To the right, there is a sidebar with a large image of a sea anemone, its common name 'Seeanemonen', and a section titled 'Systematik' with the rank 'Vielzellige Tiere (Metazoa)'.

WIKIPEDIA Q Wikipedia durchsuchen

WIKIPEDIA Q Wikipedia durchsuchen

WIKIPEDIA Q Wikipedia durchsuchen

Seeanemonen

Ordnung der Klasse Blumentiere (Anthozoa)

Artikel Diskussion

Sprache Beobachten Versionsgeschichte Bearbeiten Mehr

A Seeanemonen (Actiniaria), auch **Seerosen**, **Seenelken** oder **Aktinien** genannt^[1], sind eine arten- und gattungsreiche **Ordnung** der **Hexacorallia** innerhalb der **Blumentiere** (Anthozoa). Es handelt sich ausschließlich um im Meer vorkommende, stets **solitär** lebende, meist relativ große Tiere, die vom Flachwasser bis in **abyssale** Tiefen vorkommen. Derzeit sind etwa 1200 Arten bekannt.

A **Inhaltsverzeichnis**

A Merkmale

Seeanemonen besitzen kein **Skelett** und leben **solitär**, d. h., sie bilden keine Kolonien im Gegensatz zu den meisten anderen Vertretern der Blumentiere. Sie sind **halbsessil**; sie können sich durch langsames Kriechen auf ihrer Fußscheibe fortbewegen, mit der sie sich normalerweise auf hartem Untergrund festkrallen oder in Sand und Geröll eindringen. Ihr Körper ist muskulös. Die Größe kann, je nach Art, von einem bis 150 Zentimeter reichen. Ihre **Tentakel** sind einfach und in der Regel nicht verzweigt und oft durchscheinend. Manche **Arten** haben **Nesselfäden**, hier Acontien genannt, die durch den Mund oder durch Poren im Scapus, sog. Cnididien, ausgeschießt werden. Vielfältige Formen an **Fortpflanzungsmodi** sind bekannt. So existieren getrenntgeschlechtliche, aber auch zwittrige Arten. Sogar Querteilung oder Abschnüren von Fußpartien kommt vor.

CC BY

A Geographisches Vorkommen, Verbreitung und Lebensweise

Fischfressende Seeanemone (*Urticina piscivora*)

ohne Rang: Vielzellige Tiere (Metazoa)

Wikipedia hole

Screenshot of a Wikipedia article page for "Blumentiere". The page title is "Blumentiere". The sidebar on the left shows a navigation tree with categories like "A", "S", "D", "R", "B", "C", "E", "F", "T", "S", "K", "CC-BY", "CC-BY-SA", and "CC-BY-NC". The main content area has a search bar at the top. Below it, there are sections for "Artikel" and "Diskussion". A "Bearbeiten" (Edit) button is available. The main text discusses the class Anthozoa, mentioning the lack of medusa stage and various species. There is a "Inhaltsverzeichnis" (Table of Contents) section below the main text. To the right, there is a summary box titled "Blumentiere" showing a coral image and text about Steinkorallen. Below that is a "Systematik" section with taxonomic ranks.

Blumentiere

Klasse im Stamm Nesseltiere (Cnidaria)

Artikel Diskussion

Sprache Beobachten Versionsgeschichte Bearbeiten Mehr

Die **Blumentiere** (Anthozoa) sind mit etwa 7500 Arten die größte Klasse der **Nesseltiere** (Cnidaria). Innerhalb dieser Klasse fehlt die **Medusenform**, das heißt, die Tiere kommen nur als **Polypen** vor. Dies wurde früher als Reduktion interpretiert; heute wird angenommen, dass das Medusenstadium primär fehlt. Sie werden daher meist den **Medusozoa**, d. h. den anderen, Medusen bildenden Klassen der Nesseltiere gegenübergestellt. Die Tiere leben einzeln oder kolonial, als Klonen mit oder ohne Skelett, das organisch oder mineralisiert sein kann. Sie leben ausschließlich im Meer und kommen dort in allen Tiefenstufen bis in abyssale Tiefen vor. Die meisten Arten sind jedoch auf die obersten 100 m beschränkt.

Inhaltsverzeichnis

Merkmale

Die Blumentiere sind durch das primäre Fehlen des **Medusenstadium** charakterisiert. Das Merkmal ist ein ursprüngliches oder **plesiomorphes** Merkmal und kann nicht zur Begründung der **Monophylie** benutzt werden. Dafür zeigen die **Polypen** drei Merkmale, die nur bei Polypen der Blumentiere vorkommen: **Actinopharynx**, **Siphonoglyph** und **Mesenterien**.

Der **Actinopharynx** (auch **Stomodeum**) ist eine **ektodermale** ausgekleidete Röhre, die in den **Gastrovaskularraum** (Coelenteron, Körperhohlraum) hineinreicht. Sie ist bei allen Blumentieren, die man bisher eingehend untersuchen konnte, vorhanden, mit einer einzigen Ausnahme, der **schwarzen Koralle Sibopathes**. Der **Siphonoglyph** (auch **Sulcus**) ist eine dicht bewimperte und meist drüsige Region des **Actinopharynx**, die einzeln oder paarig vorkommt. Sie fehlt nur wenigen Gruppen (z. B. Vorhandensein eines

Steinkorallen der Gattungen *Montipora* und *Scyphomia*

ohne Rang: Opisthokonta

ohne Rang: Holozoa

Wikipedia hole

Screenshot of a Wikipedia article page for 'Nesseltiere' (Cnidaria). The page shows a sidebar with navigation links, a main content area with a summary and detailed information about Cnidaria, and a right sidebar with a summary and systematics section.

Main Content:

- Summary:** Die **Nesseltiere** (Cnidaria; alfrg. Kviðn̄ knidē Nessel) sind einfache gebaute, vielzellige Tiere, die durch den Besitz von **Nesselkapseln** gekennzeichnet sind und die Küsten, den Grund und das offene Wasser der Weltmeere und einige Süßgewässer bewohnen.
- Content:** Bekannte Untergruppen sind **Schirm- und Würfelquallen**, die **sessilen Blumentiere** mit den Seeanemonen, Stein- und Weichkorallen sowie die vielgestaltigen **Hydrozoen**, zu denen auch die **Staatsquallen** und der in Bächen und Flüssen in Mitteleuropa heimische **Soßwasserpoly** gehören. Sie umfassen derzeit über 11 000 rezente Arten.^[1] Einige Nesseltiere (z. B. *Polypodium hydriforme* und die *Myxozoa*) sind **Parasiten**.

Right Sidebar:

Nesseltiere	
	
Seeanemone (Actiniaria) und Lederkoralle (Alcyonacea)	
Systematik	
Domaine: Eukaryoten (Eucaryota)	
ohne Rang: Opisthokonta	

CC BY-SA 3.0 Wikipedia contributors. https://de.m.wikipedia.org/wiki/Nesseltiere (Version vom 21. Mai 2022, 15:26 Uhr, mso) Gezeigt ist der „Magen“ der Nesseltiere, den

Wikipedia hole

☰ WIKIPEDIA Q Wikipedia durchsuchen

☰ WIKIPEDIA Q Wikipedia durchsuchen

E ☰ WIKIPEDIA Q Wikipedia durchsuchen

A S ☰ WIKIPEDIA Q Wikipedia durchsuchen

A Sh S ☰ WIKIPEDIA Q Wikipedia durchsuchen

S Ai E ☰ WIKIPEDIA Q Wikipedia durchsuchen

D R S ☰ WIKIPEDIA Q Wikipedia durchsuchen

A S ☰ WIKIPEDIA Q Wikipedia durchsuchen

D S B D di D ☰ Sprache ⚡ Beobachten ⚡ Versionsgeschichte ⚡ Bearbeiten ⚡ Mehr

Nesselzelle
Giftschießende Zelle in der Epidermis von Nesseltieren

Artikel Diskussion

Die Nesselzellen, auch als Nematocyten oder Cnidocyten bezeichnet, sind ein spezieller Zelltyp, der ausschließlich bei den Nesseltieren (Cnidaria) zu finden ist. Es handelt sich dabei um Zellen, die in der äußeren Schicht (Epidermis) der Tiere eingebettet sind und zum Beutefang oder zur Abwehr von Feinden oder Konkurrenten eingesetzt werden können. Bei Reizung wird ein Nesselsslucht ausgeschleudert, der häufig ein hochwirksames Gift in das Opfer injiziert. Obwohl sie bei Menschen in der Regel nur Reizungen der Haut und leichtere Verbrennungen auslösen, sind die Nesselgifte einiger Arten so wirksam, dass sie zum Zusammenbruch des Herz-Kreislauf-Systems und somit zum Tode führen können.

☰ Inhaltsverzeichnis

☰ Lage

Die Anordnung der Zellen auf dem Körper der Tiere ist nicht gleichmäßig. In der Außenhaut, der Epidermis, finden sich Nesselzellen auf allen Strukturen, die zum Beutefang oder zur Verteidigung eingesetzt werden, in besonders hoher Dichte – man nennt diese Ansammlungen dann auch Batterien. Sie sitzen in erster Linie auf den Tentakeln, aber auch auf speziellen Strukturen: Dazu zählen etwa die Acontia der Seeanemonen (Actinaria), die normalerweise als feine Fäden im Körperhorraum, dem Coelenteron, liegen, aber bei Gefahr durch kleine Öffnungen desselben, die Cinciliden, nach außen geschleudert werden. Seeanemonen besitzen darüber hinaus zur Verteidigung gegen genetisch fremde Argenossen Acrorhagi, fadenförmige Strukturen, die unterhalb der echten Tentakel liegen und als Waffe eingesetzt werden – nach einem Acrorhagi-Duell zieht sich die unterlegene Anemone meist mit erheblichen Gewebebeschädigungen zurück. Steinkorallen (Scleractinia) haben zwar weder Acontia noch Acrorhagi, können aber Nesselzellen tragende



Lichtmikroskopische Aufnahme von Nematocyten, die von Tentakeln von Chironex fleckeri isoliert wurden (400fache Vergrößerung)

Quellen: die auf den Seitenrändern der Oberseite der Nesselzelle nach außen ausstülpen und zum Beutefang und zur externen Verdauung einsetzen. Bei den CC BY-SA 3.0 Wikipedia contributors, <https://de.wikipedia.org/wiki/Nesselzelle>

Wikipedia hole

*„Going on to Wikipedia to look something up, then unexpectedly being sucked into a seemingly **endless series of link clicking** to end up in a completely different part of wikipedia than you ever meant to go to.“*

— Urban Dictionary

Wikipedia hole



Motivation
○○○○●○○○

Wikipedia
○○○○○○○○○○○○○○○○

Technischer Aufbau
○○

Funktionsweise
○○○○○○○○○○○○

Take aways
○○○

Existierende Tools

- `pandoc`
- `mediawiki2latex / wb2pdf`
- `epub-press`
- `w2eb`
- `percollate`

Existierende Tools

Warum gehen die nicht?

Inhaltliche & visuelle Gründe:

- Formatierung, Schriftgrößen, etc. stimmt nicht
- Templates werden nicht/uneingeschränkt evaluiert
- $\text{\LaTeX}/\text{Math}$ wird nicht in Bild gerendert
- Tabellen funktionieren nicht

Existierende Tools

Warum gehen die nicht?

Technische Gründe:

- Kann nicht mehrere Artikel gleichzeitig
- Bilder werden nicht heruntergeladen
- Wird nicht mehr maintained
- Ist in JavaScript
- Ist in einer Programmiersprache, die ich nicht kann / mag
- Ergebnis ist kein EPUB
- Ergebnis lief nicht auf meinem Tolino

Motivation
oooooooo●

Wikipedia
oooooooooooooooooooo

Technischer Aufbau
○○

Funktionsweise
oooooooooooo

Take aways
○○○

Was will ich haben?

Generierte und gekaufte eBooks sollen sich qualitativ nicht unterscheiden.

Was will ich haben?

Generierte und gekaufte eBooks sollen sich qualitativ nicht unterscheiden.

Allgemeine Anforderungen:

- Formatierung stimmig
- Korrekte Übersetzung/Einbindung von Tabellen, Bilder, Listen, Quellenangaben, etc.
- Wikipedia-spezifische Templates & Kategorien ignorieren

Was will ich haben?

Generierte und gekaufte eBooks sollen sich qualitativ nicht unterscheiden.

Allgemeine Anforderungen:

- Formatierung stimmig
- Korrekte Übersetzung/Einbindung von Tabellen, Bilder, Listen, Quellenangaben, etc.
- Wikipedia-spezifische Templates & Kategorien ignorieren

Persönliche Anforderungen:

- Soll auf meinem Tolino eBook-Reader laufen
- Go als Programmiersprache
- Caching aller heruntergeladenen Daten (zum Coden im Zug)

Wikitext – Die Sprache der Wikipedia

Formatierung

Wikitext „kann“ auch „„Formatierung““.

Und „„sogar „alles““ durcheinander“ geht.

Wikitext *kann* auch **Formatierung**.

Und **sogar alles** *durcheinander* geht.

Wikitext – Die Sprache der Wikipedia

Links

Interne [[Hyperlink|Links]] gehen.

Auch ins Internetz [<https://externe-links>] , sogar mit [<https://foo.bar> Namen].

Interne [Links](#) gehen.

Auch ins Internetz [\[1\]](#), sogar mit [Namen](#).

Wikitext – Die Sprache der Wikipedia

Referenzen & Templates

Hi<ref name="foo">{{Internetquelle|url=http://bar.de
|abruf=2022-10-12|titel=Ref mit Template}}</ref>!

Die selbe Ref. nochmal!<ref name="foo" />

Hi^[1]!

Die selbe Ref. nochmal!^[1]

1. ↑^{a b} *Ref mit Template.* ↗ Abgerufen am 12. Oktober 2022.

Wikitext – Die Sprache der Wikipedia

Überschriften

= Level 1 =

Wird nicht aktiv benutzt, da Titel der Seite h1 ist.

===== Level 4 =====

Die hier wird benutzt.

Level 1

Wird nicht aktiv benutzt, da Titel der Seite h1 ist.

Level 4

Die hier wird benutzt.

Wikitext – Die Sprache der Wikipedia

Listen

- * Listen
- ** gibt
- es
- # auch
- ## noch

- Listen
 - gibt
- es
- 1. auch
 - 1. noch

Wikitext – Die Sprache der Wikipedia

Tabellen

```
{| class="wikitable"
|-  
! Spalte 1 !! Spalte 2
|-  
| Hier
| könnte
|-  
| ihre || Werbung stehen
|}
```

Spalte 1	Spalte 2
Hier	könnte
ihre	Werbung stehen

Wikitext – Die Sprache der Wikipedia

Bilder

Hier ein Bild:

[[Datei:Full moon partially obscured by atmosphere.jpg
|mini|Mit Unterschrift.]]

Hier ein Bild:



Mit Unterschrift.

Wikitext – Die Sprache der Wikipedia

Und vieles mehr

- Description list
- Zitate
- Einrückungen
- Code
- \LaTeX -Mathe-Zeug
- Musiknoten
- Gallerien
- Inline Bilder
- Diverse Parameter an allen möglichen Dingen

Instanzen & APIs

Instanzen – Artikel

- Instanz pro Land/Sprache → z.B. [en|de|nds].wikipedia.org
- Verlinkungen ggf. zu anderen Instanzen möglich

Instanzen & APIs

Instanzen – Bilder

- Wikimedia commons (commons.wikimedia.org)
- Normal:
upload.wikimedia.org/wikipedia/commons/0/06/Foo.jpg
- Aber auch:
upload.wikimedia.org/wikipedia/de/2/26/Son-3.jpg

Instanzen & APIs

Instanzen – Bilder

- Wikimedia commons (commons.wikimedia.org)
- Normal:
upload.wikimedia.org/wikipedia/commons/0/06/Foo.jpg
- Aber auch:
upload.wikimedia.org/wikipedia/de/2/26/Son-3.jpg
- Redirects möglich
 - ▶ Beispiel: [File:MET00506.jpg](https://commons.wikimedia.org/wiki/File:MET00506.jpg)
 - ▶ Ggf. ist Dateiname im Artikel \neq Dateiname bei Wikimedia commons
 - ▶ Nach Bild-Artikel abfragen
 - ▶ `redirects=true` Parameter hilft

Instanzen & APIs

Instanzen – Bilder

- Wikimedia commons (commons.wikimedia.org)
- Normal:
upload.wikimedia.org/wikipedia/commons/0/06/Foo.jpg
- Aber auch:
upload.wikimedia.org/wikipedia/de/2/26/Son-3.jpg
- Redirects möglich
 - ▶ Beispiel: [File:MET00506.jpg](https://commons.wikimedia.org/wiki/File:MET00506.jpg)
 - ▶ Ggf. ist Dateiname im Artikel \neq Dateiname bei Wikimedia commons
 - ▶ Nach Bild-Artikel abfragen
 - ▶ `redirects=true` Parameter hilft
- In Deutschen Artikeln wird natürlich `Datei:Sol-3.jpg` benutzt

Instanzen & APIs

APIs – Bilder

Aufbau:

`upload.wikimedia.org/wikipedia/{instance}/
{MD5[0]}/{MD5[0]MD5[1]}/{filename}`

MD5:

$\text{MD5}[i]$ = Das i-te Zeichen des MD5-Hashes von filename

Instanzen & APIs

APIs – Artikel abfragen

Anfrage:

Puren Wikitext in JSON Antwort verpackt:

```
GET de.wikipedia.org/w/api.php
    ?action=parse
    &format=json
    &prop=wikitext
    &page={article name}
```

Antwort:

```
{
  "parse": {
    "title": "Erde",
    "wikitext": {
      "*": "..."
    }
  }
}
```

Instanzen & APIs

APIs – Templates evaluieren

Anfrage:

Wie bei Artikeln nur andere Parameter.

```
GET de.wikipedia.org/w/api.php
    ?action=expandtemplates
    &format=json
    &prop=wikitext
    &text={{mein tolles template}}
```

Antwort:

```
{
  "expandtemplates": {
    "wikitext": "..."
  }
}
```

Instanzen & APIs

APIs – \LaTeX -Mathe in Bild umwandeln

1. Math-check API für Resource location anfragen
2. Eigentliches Bild abfragen

Instanzen & APIs

\LaTeX zu Bild: 1. Resource location bekommen

Anfrage:

URL: POST https://wikimedia.org/api/rest_v1/media/math/check/tex

Body: URL encoded form Element q mit dem \LaTeX -Code:

```
q:\sqrt{x}
```

Antwort:

Header x-resource-location auslesen:

```
x-resource-location: 73b85c4ec364802ad746381712d10a43f073d50a
```

Instanzen & APIs

\LaTeX zu Bild: 2. Bild abfragen

Anfrage:

Einfaches GET mit Hash an

`wikimedia.org/api/rest_v1/media/math/render/{svg|png}/73b85c4...`

Technischer Aufbau

Idee & Annahmen

Annahmen:

- Heruntergeladener wikitext ist korrekt → Keine Syntaxprüfung nötig.
- Formatierung vom HTML (Einrückung, Leerzeilen, etc.) ist egal

Technischer Aufbau

Idee & Annahmen

Annahmen:

- Heruntergeladener wikitext ist korrekt → Keine Syntaxprüfung nötig.
- Formatierung vom HTML (Einrückung, Leerzeilen, etc.) ist egal

Idee:

- Elemente im wikitext rekursiv durch Token ersetzen
- Token in Map speichern: [token] → [token-content]
- Token-Content kann weitere Token enthalten
- Token sollen einfach zu HTML ersetzt/relaxiert werden können

Technischer Aufbau

Meine „Compiler“ „Pipeline“

wikitext

Die „„Erde““ ist
ein [[Planet]] ...

Technischer Aufbau

Meine „Compiler“ „Pipeline“

wikitext —————→ Token/AST

Die '''Erde''' ist
ein [[Planet]] ...

Die \$\$TOK_BOLD_1\$\$\$ ist
ein \$\$TOK_LINK_2\$\$...

Token Map:
\$\$TOK_BOLD_1\$\$\$ → Erde
\$\$TOK_LINK_2\$\$\$ → Planet

Technischer Aufbau

Meine „Compiler“ „Pipeline“

wikitext —————→ Token/AST —————→ HTML

Die '''Erde''' ist
ein [[Planet]] ...

Die \$\$TOK_BOLD_1\$\$ ist
ein \$\$TOK_LINK_2\$\$...

Token Map:

\$\$TOK_BOLD_1\$\$ → Erde
\$\$TOK_LINK_2\$\$ → Planet

buch.html

Die Erde
ist ein <a href=
"...">Planet
...

Technischer Aufbau

Meine „Compiler“ „Pipeline“

wikitext → Token/AST → HTML → EPUB

Die '''Erde''' ist
ein [[Planet]] ...

Die \$\$TOK_BOLD_1\$\$ ist
ein \$\$TOK_LINK_2\$\$...

Token Map:
\$\$TOK_BOLD_1\$\$ → Erde
\$\$TOK_LINK_2\$\$ → Planet

buch.html

Die Erde
ist ein <a href=
"...">Planet
...

erde.epub

Technischer Aufbau

Meine „Compiler“ „Pipeline“



Die '''Erde''' ist
ein [[Planet]] ...

Die \$\$TOK_BOLD_1\$\$ ist
ein \$\$TOK_LINK_2\$\$...

Token Map:
\$\$TOK_BOLD_1\$\$ → Erde
\$\$TOK_LINK_2\$\$ → Planet

buch.html

Die Erde
ist ein <a href=
"...">Planet
...

erde.epub

Technischer Aufbau

Meine „Compiler“ „Pipeline“



Die '''Erde''' ist
ein [[Planet]] ...

Die \$\$TOK_BOLD_1\$\$ ist
ein \$\$TOK_LINK_2\$\$...

Token Map:
\$\$TOK_BOLD_1\$\$ → Erde
\$\$TOK_LINK_2\$\$ → Planet

buch.html

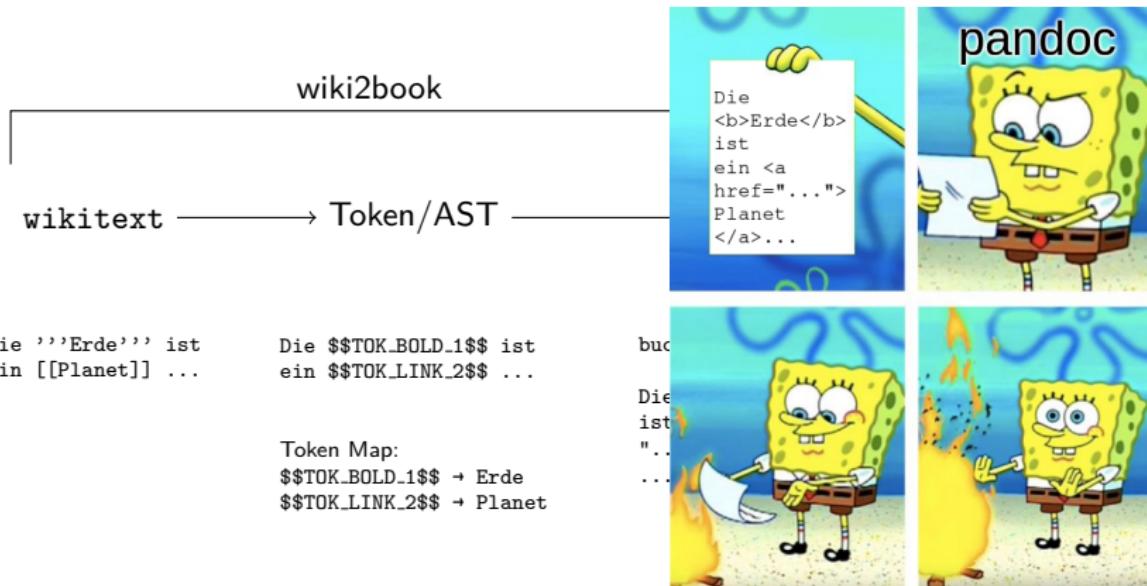
Die Erde
ist ein <a href=
"...">Planet
...

what
the

buch.epub

Technischer Aufbau

Meine „Compiler“ „Pipeline“



Motivation
ooooooooo

Wikipedia
oooooooooooooooooooo

Technischer Aufbau
oo

Funktionsweise
●oooooooooooo

Take aways
ooo

Ablauf

1. Load wikitext

Ablauf

1. Load wikitext



2. Clean data

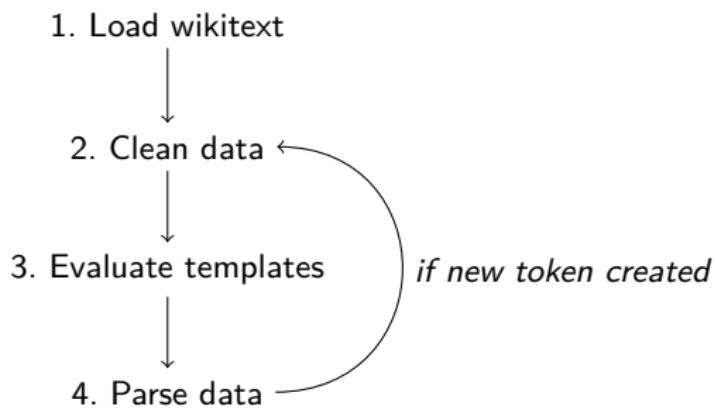
Ablauf

1. Load wikitext
2. Clean data
3. Evaluate templates

Ablauf

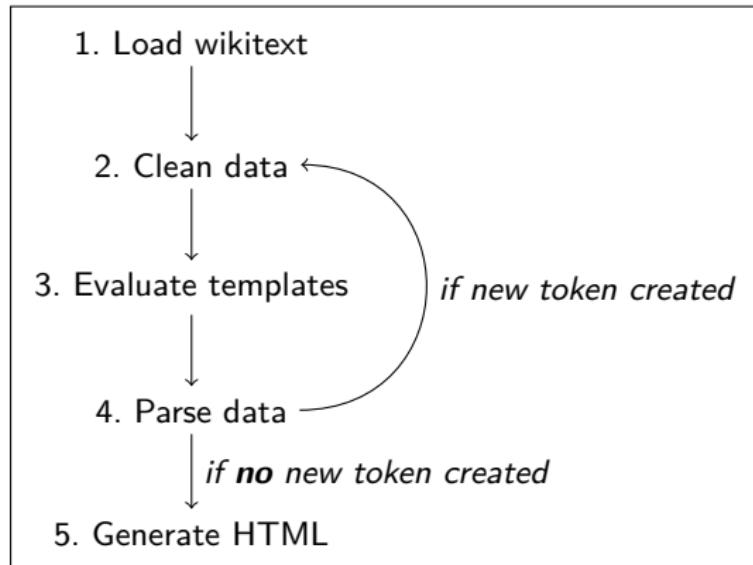
1. Load wikitext
2. Clean data
3. Evaluate templates
4. Parse data

Ablauf



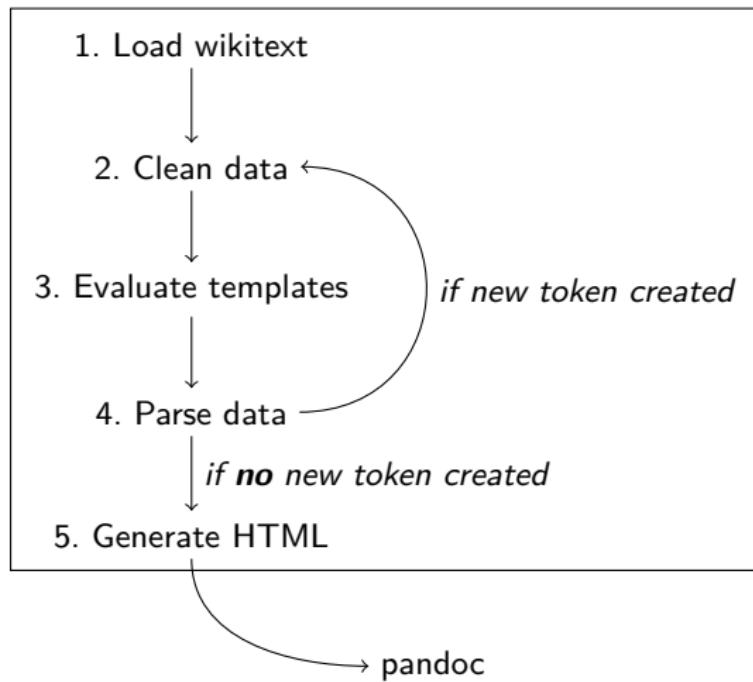
Ablauf

wiki2book



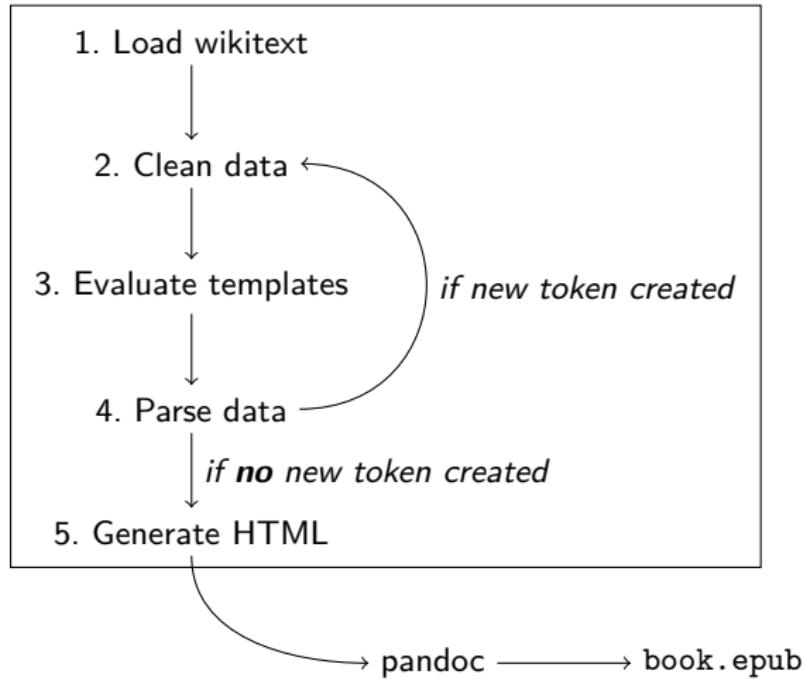
Ablauf

wiki2book



Ablauf

wiki2book



Pipeline

1. Artikel laden

Via CLI über drei mögliche Wege:

- Aus Datei (`standalone <file>`)
- Einzelnen Artikel (`article <name>`)
- Projekt-Datei (`project <project-file>`)

Pipeline

2. Cleanup

- Entfernt Kategorien
- Entfernt unerwünschte Templates
- Entfernt HTML (div und span Elemente)
- Entfernt leere Abschnitte

Pipeline

2. Cleanup – Irgendwas mit Performance oder so

```
func removeUnwantedTemplates(content string) string {
    // All lower case. Makes things easier below.
    ignoreTemplates := []string{
        "alpha centauri",
        "benriffel5grundscheinweis",
        "wiktionary",
        "toter link",
    }

    // Find all templates that actually appear in the text
    lowerCaseContent := strings.ToLower(content)
    var ignoreRegexes []*regexp.Regexp
    for _, template := range ignoreTemplates { 1
        if strings.Contains(lowerCaseContent, template) {
            ignoreRegexes = append(ignoreRegexes, regexp.MustCompile(`str: ` + template + `[^`]*?`))
        }
    }

    var matches []string
    for _, regex := range ignoreRegexes { 2
        matches = append(matches, regex.FindAllString(content, -1)...)
    }

    for _, match := range matches { 3
        content = strings.ReplaceAll(content, match, new: "")
    }

    return content
}
```

Pipeline

3. Templates evaluieren

- Finde templates
- Bitte Wikipedia-API diese zu wikitext zu konvertieren
- String-Replace

Pipeline

3. Templates evaluieren

- Finde templates
- Bitte Wikipedia-API diese zu wikitext zu konvertieren
- String-Replace



Pipeline

4. Parsen

Kann man nicht einfach mit Regexes lösen?

Pipeline

4. Parsen

Kann man nicht einfach mit Regexes lösen?

Haha.

Motivation
oooooooo

Wikipedia
oooooooooooooooooooo

Technischer Aufbau
oo

Funktionsweise
oooo●oooooooo

Take aways
ooo

Pipeline

4. Parsen

Kann man nicht einfach mit Regexes lösen?

Haha. Nein.

Motivation
oooooooo

Wikipedia
oooooooooooooooooooo

Technischer Aufbau
oo

Funktionsweise
oooo●oooooooo

Take aways
ooo

Pipeline

4. Parsen

Kann man nicht einfach mit Regexes lösen?
Haha. Nein... Manchmal.

Pipeline

4. Parsen – Beispiel: Referenzen

Wikitext:

```
Foo.<ref name="foo">...</ref> Bar.<ref name="foo"/>
```

Pipeline

4. Parsen – Beispiel: Referenzen

Wikitext:

```
Foo.<ref name="foo">...</ref> Bar.<ref name="foo"/>
```

Regex für ref-Definition:

```
<ref[^>]*?name="?(^>)*"?([>]*?=[>]*?)*(.|\n)*?></ref>
```

Regex für ref-Nutzung:

```
<ref name="(.*?)"\s?/>
```

Pipeline

4. Parsen – Beispiel: **bold** und *italic*

Wikitext:

’’foo’’ ’’’bar’’’ → ***foo bar***

¹... glaube ich

Pipeline

4. Parsen – Beispiel: **bold** und *italic*

Wikitext:

’’foo’’, ’’’bar’’’ → ***foo bar***
’’’’foo

¹... glaube ich

Pipeline

4. Parsen – Beispiel: **bold** und *italic*

Wikitext:

’’foo’’ ’’’bar’’’ → ***foo bar***
’’’’foo’’’

¹... glaube ich

Pipeline

4. Parsen – Beispiel: **bold** und *italic*

Wikitext:

''foo'' ''bar''' → **foo** *bar*

''''foo''' → '**foo**

¹... glaube ich

Pipeline

4. Parsen – Beispiel: **bold** und *italic*

Wikitext:

''foo'', ''bar''' → **foo** *bar*
''''foo''' → '**foo**'

Kein Regex möglich, da *kontextsensitive Grammatik*¹ nötig wäre.

Entscheidbarkeit bei kontextsensitiven Grammatiken in $\mathcal{O}(2^n)$.

¹... glaube ich

Pipeline

4. Parsen – Beispiel: **bold** und *italic*

Wikitext:

''foo'', ''bar''' → **foo** *bar*
''''foo''' → '**foo**'

Kein Regex möglich, da *kontextsensitive Grammatik*¹ nötig wäre.

Entscheidbarkeit bei kontextsensitiven Grammatiken in $\mathcal{O}(2^n)$. Laufzeit meines Algorithmus: $\mathcal{O}(2^n)$.

¹... glaube ich

Pipeline

4. Parsen – Beispiel: **bold** und *italic*

Wikitext:

''foo'', ''bar''' → **foo** *bar*
''''foo''' → '**foo**'

Kein Regex möglich, da *kontextsensitive Grammatik*¹ nötig wäre.

Entscheidbarkeit bei kontextsensitiven Grammatiken in $\mathcal{O}(2^n)$. Laufzeit meines Algorithmus: $\mathcal{O}(2^n)$... Yeii...

¹... glaube ich

Pipeline

4. Parsen – Beispiel: Tabellen

Wikitext (vor Preprocessing):

```
{| class="wikitable"
|+ Text der Überschrift
|-
! Überschrift 1 !! Überschrift 2
|-
|style="text-align:center"|Foo || Bar
|}
```

→ Preprocessing kann Dinge vereinfachen

Pipeline

4. Parsen – Beispiel: Tabellen

Wikitext (nach Preprocessing):

```
{| class="wikitable"
|+ Text der Überschrift
|-
! Überschrift 1
! Überschrift 2
|-
|style="text-align:center" |Foo
| Bar
|}
```

Pipeline

4. Parsen – Beispiel: Tabellen

Wikitext (nach Preprocessing):

```
{| class="wikitable"
|+ Text der Überschrift
|-
! Überschrift 1
! Überschrift 2
|-
|style="text-align:center" |Foo
| Bar
|}
```

- Zeilenpräfix anschauen
- Styles beachten
- Tabellen in Tabellen dann auch easy

Token-Map

Beispiel

Wikitext: (vorher)

Die Sonne ist ein:

* [https://de.wikipedia.org/wiki/Stern Stern]

Token-Map

Beispiel

Wikitext: (vorher)

Die Sonne ist ein:

* [https://de.wikipedia.org/wiki/Stern Stern]

Wikitext: (nachher)

Die Sonne ist ein:

\$\$TOKEN_UNORDERED_LIST_4\$\$

Token-Map

Beispiel

Wikitext: (vorher)

Die Sonne ist ein:

* [https://de.wikipedia.org/wiki/Stern Stern]

Wikitext: (nachher)

Die Sonne ist ein:

\$\$TOKEN_UNORDERED_LIST_4\$\$\$

Token-Map:

\$\$TOKEN_UNORDERED_LIST_4\$\$ → \$\$TOKEN_LIST_ITEM_3\$\$

\$\$TOKEN_LIST_ITEM_3\$\$ → \$\$TOKEN_EXTERNAL_LINK_2\$\$

\$\$TOKEN_EXTERNAL_LINK_2\$\$ → \$\$TOKEN_EXTERNAL_LINK_URL_0\$\$

\$\$TOKEN_EXTERNAL_LINK_TEXT_1\$\$

\$\$TOKEN_EXTERNAL_LINK_URL_0\$\$ → https://de.wikipedia.org/wiki/Stern

\$\$TOKEN_EXTERNAL_LINK_TEXT_1\$\$ → Stern

HTML Generierung

Token-Map:

\$\$TOKEN_EXTERNAL_LINK_2\$\$ → \$\$TOKEN_EXTERNAL_LINK_URL_0\$\$
 \$\$TOKEN_EXTERNAL_LINK_TEXT_1\$\$

\$\$TOKEN_EXTERNAL_LINK_URL_0\$\$ → <https://de.wikipedia.org/wiki/Stern>
\$\$TOKEN_EXTERNAL_LINK_TEXT_1\$\$ → Stern

HTML Generierung

Token-Map:

\$\$TOKEN_EXTERNAL_LINK_2\$\$ → \$\$TOKEN_EXTERNAL_LINK_URL_0\$\$
\$\$TOKEN_EXTERNAL_LINK_TEXT_1\$\$

\$\$TOKEN_EXTERNAL_LINK_URL_0\$\$ → <https://de.wikipedia.org/wiki/Stern>
\$\$TOKEN_EXTERNAL_LINK_TEXT_1\$\$ → Stern

Template ausfüllen:

```
<a href="%s" %s > %s </a>  
<a href="https://de.wikipedia.org/wiki/Stern">Stern</a>
```

HTML Generierung

Token-Map:

\$\$TOKEN_EXTERNAL_LINK_2\$\$ → \$\$TOKEN_EXTERNAL_LINK_URL_0\$\$
 \$\$TOKEN_EXTERNAL_LINK_TEXT_1\$\$

\$\$TOKEN_EXTERNAL_LINK_URL_0\$\$ → <https://de.wikipedia.org/wiki/Stern>
\$\$TOKEN_EXTERNAL_LINK_TEXT_1\$\$ → Stern

Template ausfüllen:

```
const HREF_TEMPLATE = "<a href=\"%s\">%s</a>"  
  
func (g *HtmlGenerator) expandExternalLink(tokenString string, tokenMap map[string]string) (string, error) {  
    splitToken := strings.Split(tokenMap[tokenString], sep: " ")  
    url := tokenMap[splitToken[0]]  
    text, err := g.expand(tokenMap[splitToken[1]]), tokenMap)  
    if err != nil {  
        return "", err  
    }  
    return fmt.Sprintf(HREF_TEMPLATE, url, text), nil  
}
```

Sonstige Problemchen

- Tolino ist weird
 - ▶ Absturz bei Nutzung von CSS gap Eigenschaft
 - ▶ Absturz bei anderen Kleinigkeiten

Sonstige Problemchen

- Tolino ist weird
 - ▶ Absturz bei Nutzung von CSS gap Eigenschaft
 - ▶ Absturz bei anderen Kleinigkeiten
- Pandoc ist weird
 - ▶ Baut HTML um & ergänzt eigenmächtig <p> Elemente
 - ▶ <th> in <tr> werden umgewandelt

Sonstige Problemchen

- Tolino ist weird
 - ▶ Absturz bei Nutzung von CSS gap Eigenschaft
 - ▶ Absturz bei anderen Kleinigkeiten
- Pandoc ist weird
 - ▶ Baut HTML um & ergänzt eigenmächtig <p> Elemente
 - ▶ <th> in <tr> werden umgewandelt
- Wikipedia-Artikel sind weird/ungünstig
 - ▶ Random Leerzeilen
 - ▶ Uneinheitliche Template-Nutzungen
 - ▶ Verweise auf andere Artikel (ala „Siehe hierzu Artikel [[foo]].“)

Motivation
ooooooooo

Wikipedia
oooooooooooooooooooo

Technischer Aufbau
oo

Funktionsweise
oooooooooooo

Take aways
oo

github.com/hauke96/wiki2book

Motivation
oooooooo

Wikipedia
oooooooooooooooooooo

Technischer Aufbau
oo

Funktionsweise
oooooooooooo

Take aways
oo●

Hands-on