

neuland - Bfi GmbH, Konsul-Smidt-Straße 8g, 28217 Bremen

RAL Deutsches Institut für Gütesicherung
und Kennzeichnung e.V.
Fränkische Straße 7
53229 Bonn

Holmes - Blauer Engel für Software
Anlage 7

neuland - Büro für Informatik GmbH
Konsul-Smidt-Str. 8g (Speicher 1)
28217 Bremen
info@neuland-bfi.de
www.neuland-bfi.de
Fon +49 (0) 421/38 01 07 - 0
Fax +49 (0) 421/38 01 07 - 99

Geschäftsführung

Maike Conrads
Jens Himmelreich
Handelsregister HRB 23395 HB
USt. ID DE 246585501

Bankverbindung

Sparkasse Bremen
IBAN DE88 2905 0101 0001 0273 33
SWIFT-BIC SBREDE22XXX

Bremen, 17.06.2025

Diese Anlage enthält:

- a) Minimale Systemvoraussetzungen (siehe Abschnitt 3.1.1.1)
- b) Dokumentation des Messsystems (siehe Abschnitt 3.1.1.2)
- c) Messergebnisse im Leerlaufzustand (siehe Abschnitt 3.1.1.3)
- d) Messergebnisse bei der Nutzung (siehe Abschnitt 3.1.1.4)
- e) Kalenderjahr und Daten zur Abwärtskompatibilität (siehe Abschnitt 3.1.2.1)
- f) Dokumentation der Datenformate (siehe Abschnitt 3.1.3.1)
- g) Schnittstellendokumentation, ggf. Softwarequelle, Softwarelizenzen (siehe Abschnitt 3.1.3.2)
- h) Hinweise auf Softwareupdates (siehe Abschnitt 3.1.3.3)
- i) Beschreibung der Deinstallation (siehe Abschnitt 3.1.3.4)
- j) Angaben zur Modularität und über Informationen zur Reduzierung von Ressourceninanspruchnahme (siehe Abschnitt 3.1.3.5)
- k) Benennung der zulässigen Tracking-Funktionen und ggf. Hinweise zu deren Deaktivierung (siehe Abschnitt 3.1.3.6)

Minimale Systemvoraussetzungen

(Alle Werte in SI-Einheiten)

- CPU: Es wird mindestens ein SSE2-kompatibler Prozessor vorausgesetzt; für Mac OS X wird ein 64bit-kompatibler Intelprozessor (Core2Duo oder neuer) vorausgesetzt oder ein M1 ARM oder neuer.

- Arbeitsspeicher: 4583 MB

- Festplattenspeicher: 5461 MB

- Betriebssystem und Softwarepakete: Linux Betriebssystem (z.B. Ubuntu 18.04 oder neuer) mit Kernel 4.0 oder größer o. Windows 10 oder neuer o. macOS 10.15 oder neuer. Weiterhin wird ein Web-Browser benötigt Firefox ab Version 75 oder neuer sowie vergleichbare.

- Erforderliche externe Dienste und Referenz-Systeme: Google für Schriften (nicht zwingend erforderlich, Fallback auf System-Schriften funktioniert auch); Amazon für die Bilder des verwendeten Produkt-Datensatzes (im Livebetrieb von Holmes wäre das ein Bilder-Dienst oder, als Fallback, keine Bilder)

- Zusätzliche Hardware: Keine

- Online Dokumentation:

<https://www.neuland-bfi.de/assets/documentation/UZ-215-de-anlage-7-neuland-holmes.pdf>

Dokumentation des Messaufbaus

(Alle Werte in SI-Einheiten)

Die Messung wurde durchgeführt auf einem dedizierten Bare-Metal System (Fujitsu PRIMERGY TX1330 M3) an welches das Leistungs-Messgerät (MCP39F511N- <https://www.microchip.com/en-us/development-tool/ADM00706>) via USB angeschlossen wurde.

Eine Synchronisation der Zeitstempel zwischen mehreren Systemen ist nicht erforderlich, da lediglich auf einem System gemessen wird.

Das Leistungs-Messgerät wurde auf eine kontinuierliche Übertragung der Leistungs-Messdaten eingestellt in einem 99ms Intervall.

Entsprechend der Anforderung der Vergabekriterien wurden die Zeitstempel der Leistungsdaten direkt beim Auslesen aus dem Messgerät erzeugt und in einer Log-Datei abgelegt.

Das Messgerät führt keine Mittelung der Werte durch sondern liefert instantan-Werte.

Durch das gewählte Abtastintervall ist dies jedoch identisch mit den in den Vergabekriterien vorgeschlagenen Geräten (z.B. Janitza UMG-604 - <https://www.janitza.de/files/download/manuals/current/UMG604-PRO/janitza-bhb-umg604pro-en.pdf>). Das Janitza UMG-604 z.B nimmt laut Datenblatt intern eine 200ms Mittelung der Effektivwerte vor.

Das System wurde weiterhin auf eine feste CPU-Frequenz eingestellt und das Intel Turbo-Boost Feature sowie Intel SpeedStep deaktiviert.

Vor jeder Messung wurde weiterhin folgendes ausgeführt:

- sync

- Offene Schreiboperationen auf der Festplatte werden ausgeführt, damit diese nicht während der Messung passieren

- `sudo /usr/sbin/sysctl -w vm.drop_caches=3`

- Weist das Betriebssystem an, alle bestehenden Caches freizugeben, damit das Szenario nicht "falsch-niedrige" Werte ausliefert durch bestehende Speicher- oder Datei-Caches

Mess-Software / Version

Als Mess-Software wurde das Green Metrics Tool genutzt in der Version v2.1 mit Commit-Version badf5ee061c8474d098434bf4e2830078c5f986b

Automatisierungs-Software

Als Automatisierungs-Software wurde das Green Metrics Tool genutzt in der Version v2.1 mit Commit-Version badf5ee061c8474d098434bf4e2830078c5f986b.
Sowie mcr.microsoft.com/playwright:v1.51.1-noble as Browser-Automatisierung und alpine/jmeter:5.6.3

Erhobene Kennwerte

Für die Kennwerte innerhalb des Systems wurden aus dem OS die Werte über Sammel-Skripte ausgelesen.

Hierbei respektive:

- Permanentspeicherbelegung über stavfs syscall als C Programm
- Arbeitsspeicherbelegung über /proc/meminfo
- Netzwerkverkehr auf IP Ebene über tcpdump
- Netzwerkauslastung über /proc/net/dev
- CPU-Auslastung über /proc/stat

Messmethode: Einsatz von Virtualisierung

Wie in den Vergabekriterien unter 3.1.1.2 als Option genannt, wird hier als Messmethode der Einsatz von Virtualisierung genutzt.

Hierbei werden die Komponenten Client/Server auf einem System gestartet und durch Virtualisierung voneinander getrennt.

Dabei haben Client / Server jeweils folgende Unterkomponenten:

- Client
 - dargestellt durch einen Webbrowser (Firefox) (Backoffice Szenario) / o. JMeter Lastgenerator (Suche Szenario)
- Server
 - holmes-postgres
 - holmes-zoo1
 - holmes-solr1
 - backoffice

Bei Einsatz einer Virtualisierung werden zunächst die Werte für die Grundauslastung normal erhoben. Zu diesem Zeitpunkt findet noch keine Virtualisierung statt.

Die Werte für die Leerlaufauslastung werden ebenfalls kumuliert erhoben und auch als solche ausgewiesen. Zuzüglich müssen hier jedoch die unter "Erhobene Kennwerte" genannten Metriken für Client / Server getrennt erhoben werden.

Gleiches gilt analog für das Nutzungsszenario.

Im Sonderfall der Messung dieser Client/Server Anwendung, welches mit einem Browser o. einem CLI Lastgenerator instrumentiert wird, ist eine Aufsplittung der Messdaten im Leerlauf nicht notwendig, da diese identisch mit den Werten des Gesamtsystems sind (Siehe Anmerkung zu Qualitäts-Anforderung 1.4).

Anmerkung zu Qualitäts-Anforderung 1.4

In den Vergabekriterien wird für den Leerlauf gefordert, dass die Anwendung aktiv ist und gestartet ist.

Im Fall dieser Client/Server Anwendung findet die Steuerung nicht durch eine eigenständige App, sondern durch einen Browser als Web-App bzw. einen CLI Lastgenerator statt. Es ist keine exakte Vorgabe vorhanden, ob der Browser in einem solchen Fall bereits geöffnet und ob zusätzlich die Startseite der Server- Applikation geöffnet sein sollen. Der CLI Lastgenerator hat keinen "geöffneten" Zustand. Er kann nur aktiv sein oder geschlossen.

Für die Leerlauf-Messung haben wir den Browser noch nicht geöffnet. Der Hintergrund für diese Entscheidung ist, dass der Browser beim Öffnen der Startseite der Anwendung bereits mit der Anwendung interagiert und auch beim reinen Öffnen bereits als Automatisierungstool Abfragen ausführt wie z.B. Zertifikatsabfragen und DNS-Abfragen.

Die Idee des Leerlaufs gem. den Vergabekriterien ist nach unserem Verständnis, zu ermitteln, ob die Server-/Client Anwendung ohne Interaktion Last erzeugt. Dies ist am besten mit einem geschlossenen Browser zu ermitteln.

Durchführung der Messung zu Anforderung 3.1.1.4

In den Vergabekriterien wird gefordert, dass:

- Arbeitsspeicherarbeit (MByte*s)
- Permanentpeicherarbeit (Lesen und Schreiben) (MByte/s*s)
- Übertragene Datenmenge außerhalb des lokalen Netzwerks (Mbit/s*s)
- Energiebedarf (Wh) (netto beim Szenario-Test, brutto beim Langzeit-Test)
- Liste aller aufgerufenen Internetadressen (IP-Adresse oder Domain Name), örtliche Zuordnung (Ländercode nach ISO 3166-1), Angabe zum Eigentümer (eigener oder externer Dienst), Häufigkeit des Aufrufs (Frequenz oder Anzahl pro Zeiteinheit)

erhoben werden. Entsprechend müssen diese Werte beim Einsatz von Virtualisierung ebenfalls getrennt ermittelt werden.

Eine Besonderheit ist hierbei, wie der Overhead aus dem Automatisierungs-Tool und dem Mess-Tool zu berücksichtigen ist.

In einer Messung ohne Einsatz von Virtualisierung findet der Overhead durch die Mess-Tools Einzug in die Grundauslastung und der Overhead durch die Automatisierung Einzug in die Messung während der Nutzung.

Somit ist das Mess-Tool, welches ebenfalls in der Grundauslastung schon läuft, aus der später auszuweisenden Arbeit bereits herausgerechnet.

Die Automatisierungs-Software und der damit entstehende Overhead werden jedoch den Kosten der Software selbst in der Messung während der Nutzung zugeschlagen.

Entsprechend ist dies auch bei Einsatz von Virtualisierung zu handhaben. Hierzu muss der Overhead der Automatisierungs-Software rechnerisch aus der Differenz zwischen den getrennt für Client / Server erhobenen Kennwerten berechnet werden.

Dies geschieht durch:

$$CPU_{BruttoauslastungGesamtsystem} - CPU_{NettoauslastungClient} - CPU_{NettoauslastungServer} - CPU_{BaselineauslastungGesamtsystem} = Overhead_{Automatisierung}$$

Der Overhead der Automatisierungs-Software muss dann dort zugeschlagen werden, wo in einem nicht-virtualisierten System die Automatisierungs-Software laufen würde. In diesem Fall wäre dies der Client.

Für die Energie sind diese Werte nicht getrennt zu erheben, daher wird hier ein Virtualisierungs-Allokations-Faktor (VAF) erstellt, welcher sich aus der jeweiligen CPU-Auslastung von Server / Client berechnet und mit der Netto-Auslastung des Systems in Bezug gesetzt wird.

Konkret:

$$VAF_{Server} = \frac{CPU_{NettoauslastungServer}}{CPU_{NettoauslastungGesamtsystem}}$$

$$VAF_{Client} = \frac{CPU_{NettoauslastungClient} + CPU_{NettoauslastungAutomatisierungsOverhead}}{CPU_{NettoauslastungGesamtsystem}}$$

Und es muss gelten:

$$VAF_{Server} + VAF_{Client} = 1$$

Die Verwendung der CPU-Auslastung ist hierbei ein plausibel nutzbarer Wert, da diese repräsentativ für die Arbeit aller Komponenten gilt. Schlussendlich ist die CPU dafür verantwortlich, sowohl die elektrische Arbeit der CPU selbst, des Speichers als auch des PermanentSpeichers und des Netzwerkadapters zu verursachen.

Die Methodik wird in folgenden akademischen und technischen Quellen beschrieben:

- Scaphandre Dokumentation - <https://hubblo-org.github.io/scaphandre-documentation/explanations/how-scaph-computes-per-process-power-consumption.html>
- Kepler: A Framework to Calculate the Energy Consumption of Containerized Applications - <https://ieeexplore.ieee.org/abstract/document/10254956>
- Development and evaluation of a reference measurement model for assessing the resource and energy efficiency of software products and components— Green Software Measurement Model (GSMM) - <https://www.sciencedirect.com/science/article/pii/S0167739X24000384>

Wichtig: Zu beachten hier ist, dass diese Methodik nur dann erlaubt ist, wenn das Hardware-System mit einer festen CPU Frequenz und ohne DVFS (Dynamic Voltage and Frequency Scaling) eingestellt ist.

Der Hintergrund ist, dass sonst die relativen Prozentangaben keine gleiche Grundgesamtheit haben. Im Falle von DVFS kann die CPU während einer Messung den Takt ändern und somit in einer Zeiteinheit unterschiedlich viele CPU-Zyklen durchlaufen.

Plausibilitätsbetrachtung des Zuschlags zum Client

Um die virtualisierte Testmethode anzuwenden, ist eine Plausibilitätsbetrachtung notwendig, welche darlegt, welcher Komponente der Overhead zuzuschlagen ist.

Die Begründung, dass dieser dort zugeschlagen wird, wo auch das Automatisierungs-Tool läuft, ist hinreichend, jedoch nicht abschließend.

Es könnte ebenfalls der Fall sein, dass der Kernel im Betriebssystem Arbeiten für die *Server Komponente* erledigt.

Um dies besser darzulegen, fügen wir eine Plausibilitätsbetrachtung an:

<https://metrics.green-coding.io/stats.html?id=42f956c7-303b-435a-89c7-418eee66007c>

In dieser Messung wurde die Automatisierung des Rendering der grafischen Oberfläche (GUI) des Browsers durchgeführt und der sog. "headless mode" verwendet.

Ersichtlich ist, dass die Werte für die Komponenten Server (nc, ncs, db) und für den Client (gcb_playwright) sich summiert nur um 0,09% von dem CPU% Wert des Gesamtsystems unterscheiden.

Die notwendige Annahme, für die Nutzung der Messmethode *Einsatz von Virtualisierung* jeglichen Overhead hier dem Client zuzuschlagen, ist somit gerechtfertigt.

Die gleiche Betrachtung muss auch für den Arbeitsspeicher und auch das Schreiben auf die Festplatte gemacht werden. Kleine Abweichungen sind hier wieder völlig normal, da die Werte sequentiell erfasst werden und somit auch zwischen dem Auslesen Speicher neu allokiert werden kann und ebenfalls Daten auf der Festplatte in einen temporären Cache geschrieben und erst später final auf die Festplatte gespeichert werden.

Anmerkung zur übertragenen Datenmenge für Netzzugang

Für den Netzwerkverkehr wird kein Virtualisierungsfaktor bestimmt, da die Server / Client Komponenten bereits über virtuelle Interfaces den Netzwerkverkehr schicken. Dies bedeutet, dass jeglicher Netzwerktraffic der Komponenten voll in diesen virtualisierten Interfaces erfasst wird.

Würde man diese in Bezug zu dem Netzwerkverkehr des Gesamtsystems setzen, würde hier ein "Double-Counting" erfolgen, da der Traffic im unterliegenden Betriebssystem ebenfalls separat gezählt wird.

Es ist in Folge ausreichend, nur den Netzwerktraffic der virtualisierten Komponenten zu betrachten und als Arbeit auszuweisen.

Anmerkung zu den Markern in den Zeitreihen (Start / Ende des Szenarios)

Das Green Metrics Tool führt immer einen vollen Durchlauf von Baseline, Install, Boot, Idle, Runtime und Remove durch.

Für die Messprotokolle des Blauen Engel relevant ist immer der Teil zwischen "Starting Phase [RUNTIME]" und "Ending Phase [RUNTIME]".

Ausnahme: Die Grundaustauslastung. Hier wird die Phase "Baseline" des Green Metrics Tool genutzt, welche analog beginnt bei "Starting Phase [BASELINE]" und endet bei "Ending Phase [BASELINE]".

Anmerkung zum Hilfsblatt in der Tabellenkalkulations-Datei

Die Permanentspeicherauslastung von Container benötigt zur Ermittlung drei Komponenten:

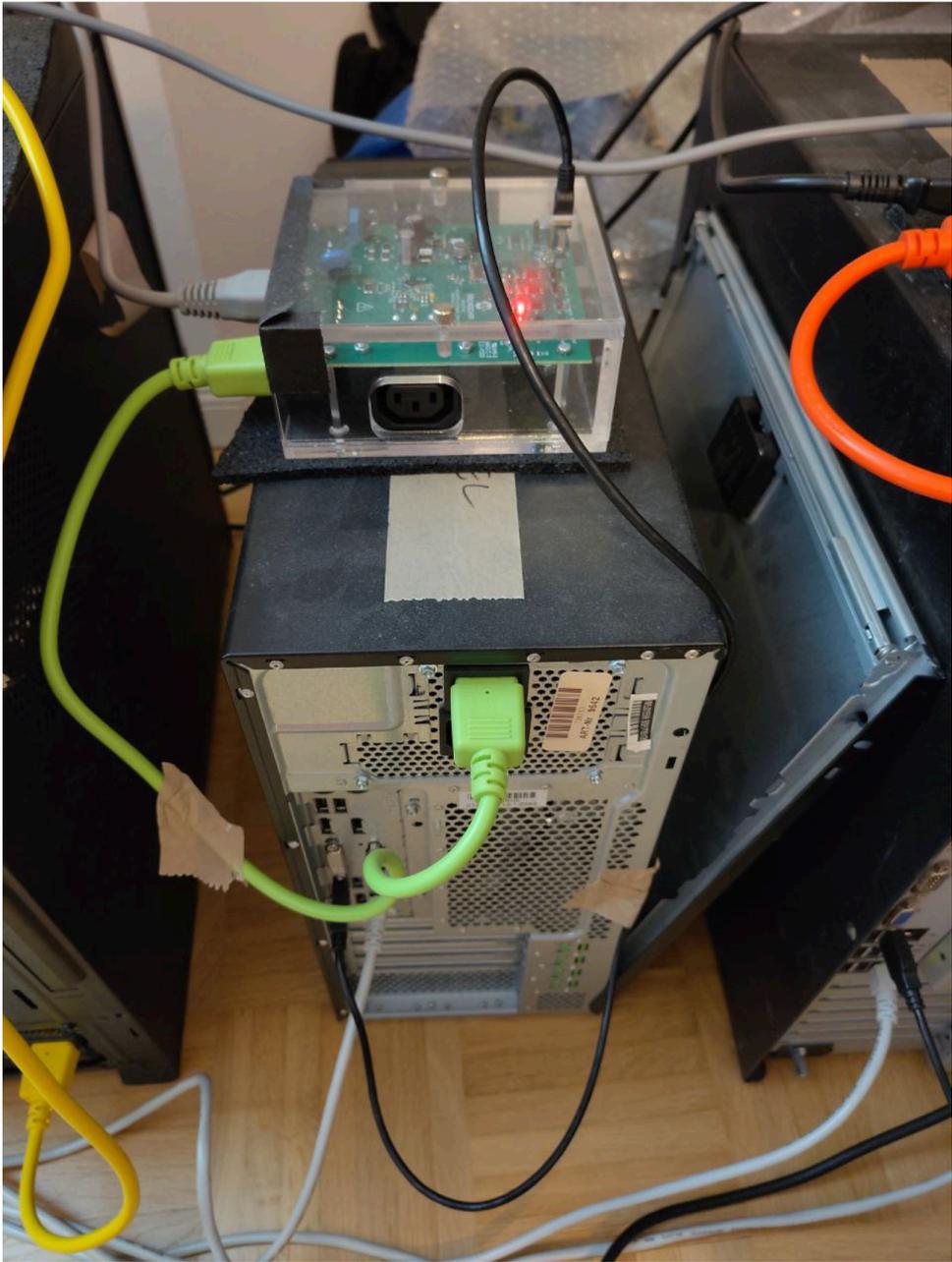
- A: Die Größe des Container Images
- B: Alle Daten, die während der Ausführung des Standardnutzungs-Szenario im Container geschrieben werden

A ist im Hilfsblatt eingetragen und wird in dem Berechnungsbogen über eine Formel zu B hinzuaddiert.

Die Ermittlung der Werte von A sind in der Anlage 3 im Ordner "Image Size Logs" hinterlegt. Dies erfolgt über die Container-Orchestrierungs-Software Docker.

Die Ermittlung der Werte von B erfolgt über die Messprotokolle.

Foto des Messaufbaus



Messergebnisse im Leerlaufzustand

(Alle Werte in SI-Einheiten)

Messung der Grundauslastung und der Last des Softwareprodukts im Leerlaufzustand				
ID	Bezeichnung	Hinweis	Ergebnis	Einheit
Messung der Grundauslastung				
3.1.1.3 a)	Mittlere Prozessorauslastung		0,25	%
3.1.1.3 b)	Mittlere Arbeitsspeicherbelegung		1536,29	MByte
3.1.1.3 c)	Mittlere Permanentspeicherbelegung		29509,23	MByte
3.1.1.3 d)	Mittlere beanspruchte Bandbreite für Datenübertragung	Die mittlere Bandbreite des Messsystems wird ohne das installierte Softwareprodukt gemessen	0	Mbit/s
3.1.1.3 e)	Mittlere elektrische Leistungsaufnahme (brutto)		32,72	W
Messung der Last des Softwareprodukts im Leerlaufzustand				
3.1.1.3 a1)	Mittlere Prozessorauslastung		0,35	%
3.1.1.3 b1)	Mittlere Arbeitsspeicherbelegung		3.765	MByte
3.1.1.3 c1)	Mittlere Permanentspeicherbelegung		6.306	MByte
3.1.1.3 d1)	Mittlere beanspruchte Bandbreite für Datenübertragung	Die zusätzliche mittlere Bandbreite, die durch den Leerlaufbetrieb der Software entsteht wird gemessen	0	Mbit/s
3.1.1.3 e1)	Mittlere elektrische Leistungsaufnahme (netto)		0,28	W

Messergebnisse während der Nutzung

(Alle Werte in SI-Einheiten)

Es gibt zwei gemessene Szenarien:

1. Szenario „Backoffice“:

Jemand ruft das Backoffice auf, stellt die Sprache ein, geht auf die Suchvorschau und sucht nach etwas. Es gibt 5 Ergebnisse. Dann wechselt die Person zum Reiter „Suchregeln“, legt eine neue Regel mit 3 Synonymen und Booster an, speichert und wechselt zur Vorschau. Die Suche nach dem o.g. Suchbegriff liefert jetzt 44 Ergebnisse. Dann veröffentlicht die Person die Konfiguration und prüft nochmal die Live-Ergebnisse.

Messung des Softwareprodukts während der Nutzung				
ID	Bezeichnung	Hinweis	Ergebnis	Einheit
3.1.1.4 a)	Prozessorarbeit		581	%*s
3.1.1.4 b)	Arbeitsspeicherarbeit		100.660	MByte*s
3.1.1.4 c)	Permanentspeicherarbeit (Lesen und Schreiben)		126.548	MByte/s*s
3.1.1.4 d)	Übertragene Datenmenge für Netzzugang		8	Mbit/s*s
3.1.1.4 e)	Energiebedarf	netto beim Szenario-Test; brutto beim Langzeit-Test	0,05	Wh
Messung des Softwareprodukts während der Nutzung - Für virtualisierte Systeme - Server				
ID	Bezeichnung	Hinweis	Ergebnis	Einheit
3.1.1.4 a)	Prozessorarbeit		182	%*s
3.1.1.4 b)	Arbeitsspeicherarbeit		75.815	MByte*s
3.1.1.4 c)	Permanentspeicherarbeit (Lesen und Schreiben)		41.302	MByte/s*s
3.1.1.4 d)	Übertragene Datenmenge für Netzzugang		48	Mbit/s*s
3.1.1.4 e)	Energiebedarf	netto beim Szenario-Test; brutto beim Langzeit-Test	0,01	Wh
Messung des Softwareprodukts während der Nutzung - Für virtualisierte Systeme - Client				
ID	Bezeichnung	Hinweis	Ergebnis	Einheit
3.1.1.4 a)	Prozessorarbeit		399	%*s
3.1.1.4 b)	Arbeitsspeicherarbeit		24.845	MByte*s
3.1.1.4 c)	Permanentspeicherarbeit (Lesen und Schreiben)		78.997	MByte/s*s
3.1.1.4 d)	Übertragene Datenmenge für Netzzugang		38	Mbit/s*s
3.1.1.4 e)	Energiebedarf	netto beim Szenario-Test; brutto beim Langzeit-Test	0,03	Wh

2. Szenario „Suche“:

Ein Prozess (z.B. eine Nutzer:in in einem Online-Shop) sucht nach 3 verschiedenen Begriffen, filtert das Ergebnis weiter und blättert dann mehrmals auf die nächste Seite.

Messung des Softwareprodukts während der Nutzung				
ID	Bezeichnung	Hinweis	Ergebnis	Einheit
3.1.1.4 a)	Prozessorarbeit		242	%*s
3.1.1.4 b)	Arbeitsspeicherarbeit		24.355	MByte*s
3.1.1.4 c)	Permanentspeicherarbeit (Lesen und Schreiben)		32.844	MByte/s*s
3.1.1.4 d)	Übertragene Datenmenge für Netzzugang		0	Mbit/s*s
3.1.1.4 e)	Energiebedarf	netto beim Szenario-Test; brutto beim Langzeit-Test	0,02	Wh
Messung des Softwareprodukts während der Nutzung - Für virtualisierte Systeme - Server				
ID	Bezeichnung	Hinweis	Ergebnis	Einheit
3.1.1.4 a)	Prozessorarbeit		26	%*s
3.1.1.4 b)	Arbeitsspeicherarbeit		20.791	MByte*s
3.1.1.4 c)	Permanentspeicherarbeit (Lesen und Schreiben)		11.478	MByte/s*s
3.1.1.4 d)	Übertragene Datenmenge für Netzzugang		1	Mbit/s*s
3.1.1.4 e)	Energiebedarf	netto beim Szenario-Test; brutto beim Langzeit-Test	0,00	Wh
Messung des Softwareprodukts während der Nutzung - Für virtualisierte Systeme - Client				
ID	Bezeichnung	Hinweis	Ergebnis	Einheit
3.1.1.4 a)	Prozessorarbeit		216	%*s
3.1.1.4 b)	Arbeitsspeicherarbeit		3.564	MByte*s
3.1.1.4 c)	Permanentspeicherarbeit (Lesen und Schreiben)		4.345	MByte/s*s
3.1.1.4 d)	Übertragene Datenmenge für Netzzugang		0	Mbit/s*s
3.1.1.4 e)	Energiebedarf	netto beim Szenario-Test; brutto beim Langzeit-Test	0,02	Wh

Kalenderjahr und Daten zur Abwärtskompatibilität

(Alle Werte in SI-Einheiten)

Die Software ist mind. auf einem 5 Jahre alten System lauffähig.
Aufgeführt ist das genutzte Test-System für die Energiemessungen:

System Jahr	2018
Modell	Fujitsu PRIMERGY TX1330 M3
Prozessor:	Intel(R) Xeon(R) CPU E3-1220 v6 @ 3.00GHz
Cores	4
Taktfrequenz	3 GHz
RAM:	16 GB
Festplatte (SSD / HDD)	INTENSO SSD, V0601B0, 220 GB
Grafikkarte:	Matrox Electronics Systems Ltd. MGA G200e
Netzwerk:	Intel Corporation I210 Gigabit Network Co
Netzteil:	Fujitsu 450W hot-plug, 94% (Platinum efficiency), 100-240V, 50 / 60Hz
Mainboard:	Fujitsu D3373 mit Intel® C236 Chipset
Betriebssystem:	Ubuntu 24.04.1 LTS

Das System ist mindestens mit einem 5 Jahre alten Betriebssystem kompatibel, respektive:

- Linux Betriebssystem (z.B. Ubuntu 18.04 oder neuer) mit Kernel 4.0 oder größer
- Windows 10 (Mid-2019) oder neuer
- macOS 10.15 oder neuer

Dokumentation der Datenformate

Holmes nutzt bzw. erzeugt SQL (.sql) und JSON (.json). Über die API oder das Backoffice kann die Konfiguration als JSON importiert und exportiert werden.

Die Datenbank ist quasi Schema-los und hat JSON-Felder in der Tabelle. Das JSON sollte gut lesbar sein, die Feldnamen sind Domänenbegriffe (sind also im Sprech der Such-Domäne benannt).

Schnittstellendokumentation, ggf. Softwarequelle, Softwarelizenzen

API-Dokumentation für Holmes Suchfunktionen

Diese Dokumentation beschreibt die API-Endpunkte für die Suchfunktionen von Holmes, die für externe Anbindungen an einen Shop verwendet werden können.

Basis-URL

Alle Endpunkte verwenden die folgende Basis-URL:

```
/suche/{indexLabel}
```

Wobei `{indexLabel}` ein Parameter ist, der den zu verwendenden Suchindex identifiziert.

1. Shop-Suche

Dieser Endpunkt führt eine Suche im Shop durch und gibt Suchergebnisse zurück.

Anfrage

```
GET /suche/{indexLabel}/shop
```

Parameter

Parameter	Typ	erforderlich	Beschreibung
q	String	ja	Der Suchbegriff
page	Integer	nein	Seitennummer (0-basiert, Standard: 0)
size	Integer	nein	Anzahl der Ergebnisse pro Seite (Standard: 20)
sort	String	nein	Sortierkriterien im Format <code>`eigenschaft1,eigenschaft2(,asc desc)`</code>
f.*	String	nein	Filter-Eingaben im Format <code>`f.Feldname=Wert`</code>
step	String	nein	ID des Suchschritts
sessionId	String	nein	ID der Session
sortierung	String	nein	Label für die Sortierung
ranking	String	nein	Label für das Ranking

Beispiel

```
GET /suche/shop1/shop?  
q=lampe&size=10&f.Lichtverteilung=Diffus+Allgemein&sortierung=relevanz
```

2. Kategorie-Suche

Dieser Endpunkt führt eine Suche innerhalb einer bestimmten Kategorie durch.

Anfrage

```
GET /suche/{indexLabel}/kategorie
```

Parameter

Parameter	Typ	erforderlich	Beschreibung
kategorieid	String	ja	ID der Kategorie
page	Integer	nein	Seitennummer (0-basiert, Standard: 0)
size	Integer	nein	Anzahl der Ergebnisse pro Seite (Standard: 20)
sort	String	nein	Sortierkriterien
sessionId	String	nein	ID der Session
sortierung	String	nein	Label für die Sortierung
ranking	String	nein	Label für das Ranking

Beispiel

```
GET /suche/shop1/kategorie?  
kategorieId=beleuchtung&size=20&sortierung=preis_asc
```

3. Suchvorschläge (Suggest)

Dieser Endpunkt liefert Suchvorschläge basierend auf einem eingegebenen Begriff.

Anfrage

```
GET /suche/{indexLabel}/suggest
```

Parameter

Parameter	Typ	erforderlich	Beschreibung
q	String	ja	Der Suchbegriff für Vorschläge

Beispiel

```
GET /suche/shop1/suggest?q=lam
```

Antwortformate

Alle Endpunkte liefern Antworten im JSON-Format. Die genaue Struktur der Antworten kann je nach Endpunkt variieren, enthält aber typischerweise:

- Suchergebnisse mit Produktinformationen
- Metadaten zur Suche (Gesamtanzahl, Seitenzahl, etc.)
- Facetten für die Filterung

Authentifizierung

Die API verwendet aktuell Basic Authentication für die Zugangskontrolle. Bei Anfragen müssen gültige Zugangsdaten im HTTP-Header mitgesendet werden.

Fehlerbehandlung

Bei Fehlern werden entsprechende HTTP-Statuscodes zurückgegeben:

- 400 Bad Request: Bei ungültigen Parametern
- 404 Not Found: Wenn der angeforderte Index nicht gefunden wird
- 500 Internal Server Error: Bei internen Serverfehlern

Lizenz:

Als Kunde von Holmes erwirbt man über eine einmalige Softwarenutzungsgebühr ein nicht-exklusives, nicht übertragbares und unbefristetes Nutzungsrecht an den im Projekt entwickelten Funktionalitäten zur Nutzung für eigene Geschäftszwecke.

Funktionalitäten, die im Rahmen von Holmes für einen Kunden entwickelt werden, können auch von anderen Kunden von neuland erneut beauftragt werden.

Kunden bekommen nach Zahlung der Softwarenutzungsgebühr den gesamten Quellcode zur Verfügung gestellt und können diesen ggf. auch selbst oder mithilfe Dritter weiterentwickeln.

Hinweise auf Softwareupdates

Software-Updates werden entweder von neuland durchgeführt oder können alternativ vom Kunden selbst oder einer vom Kunden beauftragten dritten Instanz durchgeführt werden. Kunden erhalten den Quellcode von Holmes (siehe „Lizenz“).

Zudem verpflichtet sich neuland, für 5 Jahre kostenlose Sicherheitsupdates durchzuführen.

Beschreibung der Deinstallation

Dienste stoppen:

```
docker compose down
```

Persistente Daten löschen:

```
docker volume rm projekt_pgdata projekt_solrdata
```

Volumen-Namen ggf. per `docker volume ls` prüfen. Angaben zur Modularität und über Informationen zur Reduzierung von Ressourceninanspruchnahme

Sollten Teile von Holmes bei einem Kunden nicht benötigt werden, können sie im Quellcode gelöscht werden.

Benennung der zulässigen Tracking-Funktionen und ggf. Hinweise zu deren Deaktivierung

Holmes trackt **keinerlei kundenbezogene Daten** und enthält auch **keinerlei Werbung**.

Es gibt ein auf die Funktionalität von Holmes bezogenes Tracking:

- welche Suchbegriffe wurden wie oft gesucht?
- wieviele Ergebnisse wurden für diesen Suchbegriff gefunden?
- wie schnell war die Suche?
- in welchem Suchschritt wurden Ergebnisse gefunden?
- wie oft führte eine Suche nicht zu Treffern?

Diese Informationen werden genutzt, um im Backoffice eine Statistik anzuzeigen.