

TEXTE

62/2019

# Leitfaden zur umweltfreundlichen öffentlichen Beschaffung von Software



TEXTE 62/2019

Ressortforschungsplan des Bundesministerium für  
Umwelt, Naturschutz und nukleare Sicherheit

Forschungskennzahl 3715 37 601 0

FB000101/ZW

## **Leitfaden zur umweltfreundlichen öffentlichen Beschaffung von Software**

Empfehlungen aus dem UFOPLAN-Projekt 3715 37 601 0

„Entwicklung und Anwendung von  
Bewertungsgrundlagen für ressourceneffiziente Software  
unter Berücksichtigung bestehender Methodik“

von

Jens Gröger

Im Auftrag des Umweltbundesamtes

## Impressum

### Herausgeber

Umweltbundesamt  
Wörlitzer Platz 1  
06844 Dessau-Roßlau  
Tel: +49 340-2103-0  
Fax: +49 340-2103-2285  
[info@umweltbundesamt.de](mailto:info@umweltbundesamt.de)  
Internet: [www.umweltbundesamt.de](http://www.umweltbundesamt.de)

[f/umweltbundesamt.de](https://www.facebook.com/umweltbundesamt.de)

[t/umweltbundesamt](https://twitter.com/umweltbundesamt)

### Erstellung des Leitfadens:

Jens Gröger, Öko-Institut e.V., Postfach 17 71, 79017 Freiburg

### Auf Grundlage des Forschungsberichtes:

Gröger, J.; Naumann, S.; Hilty, L.; Filler, A.; Guldner, A.; Kern, E.; Köhler, A. R.; Maksimov, Y.; Entwicklung und Anwendung von Bewertungsgrundlagen für ressourceneffiziente Software unter Berücksichtigung bestehender Methodik; Öko-Institut e.V. in Kooperation mit Hochschule Trier, Umwelt-Campus Birkenfeld und Universität Zürich, Institut für Informatik im Auftrag des Umweltbundesamtes, Dessau-Roßlau, September 2018

### Abschlussdatum:

Dezember 2018 - Korrekturversion Januar 2020

### Redaktion:

Beratungsstelle nachhaltige Informations- und Kommunikationstechnik (Green IT)  
Marina Köhn

Publikationen als pdf:

<http://www.umweltbundesamt.de/publikationen>

ISSN 1862-4804

Dessau-Roßlau, Mai 2019

Dieser Leitfaden basiert auf den Erkenntnissen und Ergebnissen des Forschungsvorhabens Entwicklung und Anwendung von Bewertungsgrundlagen für ressourceneffiziente Software unter Berücksichtigung bestehender Methodik (UBA-Texte 105/2018). Trotz sorgfältiger Prüfung sämtlicher Angaben des Leitfadens können Fehler nicht mit letzter Sicherheit ausgeschlossen werden. Die Richtigkeit, Vollständigkeit und Aktualität des Inhalts ist daher ohne Gewähr. Eine Haftung des Herausgebers auch für die mit dem Inhalt verbundenen potentiellen Folgen ist ausgeschlossen.

Wir erlauben das Kopieren sowie die sonstige Nutzung aller in diesem Leitfaden enthaltenen Inhalte, sofern sie nicht verfälscht oder auf sonstige missbräuchliche Art und Weise genutzt werden.

**Inhaltsverzeichnis**

Abbildungsverzeichnis..... 6

Tabellenverzeichnis..... 6

Begriffsbestimmungen ..... 7

1 Einleitung..... 10

2 Verwendung dieses Leitfadens ..... 11

    2.1 Auswahl der Beschaffungskriterien ..... 11

    2.2 Weitere Hinweise zur Anwendung der Beschaffungskriterien..... 13

3 Anforderungen an nachhaltige Software ..... 15

    3.1 Ressourceneffizienz ..... 15

        3.1.1 Erforderliche minimale Systemvoraussetzungen ..... 15

        3.1.2 Hardware-Auslastung im Leerlauf der Software ..... 16

        3.1.3 Hardware-Inanspruchnahme und Energiebedarf bei Ausführung eines  
Standardnutzungsszenarios ..... 16

        3.1.4 Unterstützung des Energiemanagements ..... 17

    3.2 Potentielle Hardware-Nutzungsdauer ..... 18

        3.2.1 Abwärtskompatibilität ..... 18

        3.2.2 Plattformunabhängigkeit und Portabilität..... 18

    3.3 Nutzungsautonomie..... 19

        3.3.1 Transparenz der Datenformate ..... 19

        3.3.2 Transparenz des Programmcodes ..... 20

        3.3.3 Kontinuität des Softwareproduktes..... 20

        3.3.4 Deinstallierbarkeit..... 21

        3.3.5 Offlinefähigkeit ..... 22

        3.3.6 Dokumentation des Softwareprodukts, der Lizenz- und Nutzungsbedingungen..... 22

4 Erläuterungen zur Durchführung von Messungen ..... 24

    4.1 Referenzsystem..... 25

    4.2 Standardnutzungsszenario..... 26

    4.3 Grundlegende Definitionen für die Messung und Berechnung von Softwareeigenschaften28

5 Literaturverzeichnis..... 31

## Abbildungsverzeichnis

Abbildung 1:	Exemplarischer Messaufbau Energie- und Ressourceneffizienzmessung von Software .....	24
Abbildung 2:	Beispielhafter Messzyklus zur Bestimmung der Hardware-Auslastung .....	30

## Tabellenverzeichnis

Tabelle 1:	Auswahl der Kriterien nach Art des zu beschaffenden Softwareprodukts .....	12
Tabelle 2:	Referenzsystems als Zeitreihe 2014 – 2017 .....	26
Tabelle 3:	Überblick der durchgeführten Aktionen für ausgewählte Softwareprodukte .....	27
Tabelle 4:	Grundlegende Definitionen .....	28

## Begriffsbestimmungen

<b>Ausschlusskriterium</b>	Anforderungen an ein Produkt oder eine Dienstleistung, deren Nichteinhaltung zu einem Ausschluss aus der Angebotsbewertung führt.
<b>Bewertungskriterium</b>	Anforderung an ein Produkt oder eine Dienstleistung, die qualitativ oder quantitativ bewertet werden kann. Beim Vergleich von Produkten werden Produkte mit einer besseren Bewertung (z. B. geringere Kosten, geringerer Ressourcenverbrauch, längere Lebensdauer, bessere Gebrauchstauglichkeit) positiver bewertet.
<b>Content Management System</b>	Ein Content Management System (CMS) ist ein Softwareprodukt zur Erstellung von Inhalten (Content) durch eine Gruppe von Autoren. Dies kann beispielsweise der Inhalt einer Internetseite sein oder ein unternehmensinternes Informationssystem.
<b>Hardware</b>	Gesamtheit der für die Ausführung von Programmen, die Speicherung oder den Transport von Daten benötigten Sachgüter.
<b>Hardwarekapazität</b>	Quantifizierbare Eigenschaft eines Hardwaresystems, die die Grenze seiner Leistungsfähigkeit auf einer gegebenen Leistungsdimension darstellt (z. B. Arbeitsspeicherkapazität, Rechenkapazität, Bandbreite).
<b>Hardwaresystem</b>	Abgrenzbare Einheit von Hardware, die definierte Funktionen erbringt.
<b>Indikator</b>	Eine empirisch bestimmbare Größe, die Aufschluss über einen nicht direkt messbaren Sachverhalt gibt. Das Skalenniveau der in diesem Dokument vorgeschlagenen Indikatoren ist unterschiedlich. In einigen Fällen wird eine qualitative Ordinalskala angenommen (z. B. „ungenügend“, „genügend“, „gut“, „sehr gut“ oder auch nur „erfüllt“, „nicht erfüllt“). Damit soll vermieden werden, dass durch die Verwendung einer Kardinalskala eine nicht vorhandene Präzision suggeriert wird.
<b>Lasttreiber</b>	Ein Lasttreiber simuliert eine Arbeitsleistung auf einem Computer, um Software zu testen. Ein Lasttreiber wird i.d.R. durch Automatisierungssoftware oder ein Benchmark-Programm realisiert.
<b>Log-Datei</b>	Von einem Programm automatisch erstellte Datei, die während des Ablaufs einer Messung (z. B. eines Standardnutzungsszenarios) die durchgeführten Aktionen inklusive Zeitstempel protokolliert.
<b>Nutzungsmuster</b>	Abstrahierte Form einer Sequenz von Interaktionen mit einem gegebenen Softwareprodukt.
<b>Nutzungsszenario</b>	Beschreibung eines Nutzungsmusters, in der Regel maschinell ausführbar.
<b>Plugin</b>	Ein Softwaremodul, das in ein Softwareprodukt eingebunden werden kann, um dessen Funktionalität zu erweitern.

<b>Rechenleistung</b>	Unter (maximaler) Rechenleistung einer Central Processing Unit (CPU) wird im Rahmen dieses Vorhabens vereinfacht das Produkt aus Taktfrequenz, Anzahl der CPU-Kerne und Datenbusbreite verstanden. Ein anteiliger Wert der Rechenleistung (in Prozent) wird erreicht, indem die Abarbeitung von Befehlen zeitlich gestaffelt, mit reduzierter Taktfrequenz, unter Nutzung von weniger als der verfügbaren Prozessorkerne und/oder geringerer Datenbusbreite erfolgt.
<b>Referenzsystem</b>	Ein Hardwaresystem, das hinsichtlich seiner wichtigsten Kapazitäten (z. B. Arbeitsspeicher, Rechenleistung) während einer festgelegten Zeitperiode (z. B. ein Jahr) als allgemein üblich definiert wird. Das Referenzsystem dient dazu, Indikatoren wie z. B. „minimaler lokaler Arbeitsspeicher“ relativ zu einer Referenzgröße (der „aktuell üblichen“ Speicherkapazität) ausdrücken zu können.
<b>Ressource</b>	Im Kontext dieses Dokuments eine natürliche Ressource, insbesondere ein Rohstoff, eine Energieform oder auch die Absorptionsfähigkeit eines Umweltmediums für Emissionen. Zur Abgrenzung gegen technische Ressourcen, insbesondere Hardwareressourcen, werden letztere hier präziser als „Hardwarekapazitäten“ bezeichnet. Da die Beanspruchung von Hardwarekapazitäten stets zur Beanspruchung natürlicher Ressourcen führt, ist diese Abgrenzung (die in letzter Konsequenz auf eine definitorisch schwierige Grenzziehung zwischen Ökosphäre und Technosphäre hinausläuft) hier nicht von entscheidender Bedeutung.
<b>Ressourceneffizienz</b>	Allgemein die Menge an „nützlicher Arbeit“ dividiert durch die dabei beanspruchten Ressourcen. Im Kontext dieses Dokuments wird „nützliche Arbeit“ als erfolgreiche Ausführung von Standardnutzungsszenarien operationalisiert.
<b>Software</b>	Ausführbare Programme und die dazugehörigen Daten in digitaler Form.
<b>Softwareprodukt</b>	Eine abgrenzbare Einheit von Programmen und Daten, die zur Ausführung und Verarbeitung definierter Aufgaben auf einem Hardwaresystem bestimmt sind.
<b>Standardkonfiguration</b>	Eine als Referenz definierte Menge von Bedingungen, unter denen ein gegebenes Softwareprodukt betrieben wird. Sie umfasst die am Softwareprodukt während der Installation oder des Betriebs vorgenommenen Parametereinstellungen, die bereitgestellte Systemsoftware, ggf. weitere zum Betrieb benötigte Softwareprodukte sowie auf Hardwareebene das Referenzsystem.
<b>Standardnutzungsszenario</b>	Ein Nutzungsszenario, das zum Testen eines Softwareprodukts verwendet wird und möglichst repräsentativ für den üblichen Anwendungsfall sein soll.
<b>System Under Test (SUT)</b>	Ein Hardwaresystem, dessen verbrauchte Energie und verwendete Hardwarekapazitäten gemessen werden (z. B. während des Ablaufs eines Standardnutzungsszenarios). Beinhaltet neben der Hardware auch sämtliche zum Betrieb des Softwareprodukts notwendige Software (z. B. Betriebssystem, Laufzeitumgebungen, etc.).



**Zeitstempel**

Datum und Uhrzeit in einem definierten digitalen Format, das den genauen Zeitpunkt der Aufnahme eines Messwerts oder anderen Ereignisses (z. B. einer Aktion eines Standardnutzungsszenarios) festhält.

## 1 Einleitung

Software hat einen messbaren Einfluss auf den Energiebedarf von Computer-Hardware und kann durch steigende Anforderungen dazu beitragen, dass Hardware vorzeitig ausgetauscht werden muss („Software bedingte Obsoleszenz“).

In dem Forschungsprojekt „Entwicklung und Anwendung von Bewertungsgrundlagen für ressourceneffiziente Software unter Berücksichtigung bestehender Methodik“ des Umweltbundesamtes (UBA 2018) wurde eine Bewertungsmethodik entwickelt, anhand der der Energiebedarf, die Inanspruchnahme von Hardware-Ressourcen sowie weitere umweltbezogene Eigenschaften von Softwareprodukten ermittelt werden können. Der Vergleich verschiedener Softwareprodukte mit gleicher Funktionalität macht deutlich, dass es teils erhebliche Unterschiede zwischen den Produkten gibt. Das Forschungsprojekt hat aufgezeigt, dass beispielsweise der Energiebedarf für die Ausführung eines standardisierten Nutzungsszenarios bei untersuchten Fallbeispielen um einen Faktor von bis zu vier variieren kann. D.h. ein ineffizient programmiertes Softwareprodukt verbraucht viermal so viel Energie wie ein effizienter programmiertes Softwareprodukt. Die Ergebnisse des Forschungsprojektes zeigen außerdem, dass es auch bei der Hardwareeffizienz (bezogen auf Prozessorauslastung, Arbeitsspeicher, Permanentspeicher, Datenübertragung) erkennbare Unterschiede zwischen den Softwareprodukten gibt. Dies ist vor allem vor dem Hintergrund relevant, dass die übermäßige Beanspruchung von Hardware dazu führt, dass die Programmausführung zu lange dauert und Unternehmen, öffentliche Verwaltungen aber auch Privatpersonen diese vermeintlich langsame Hardware ausmustern und durch neue, schnellere Hardware ersetzen. Weitere Bewertungskriterien betreffen die Nutzungsautonomie und Anwendungsfreundlichkeit einer Software, die letztlich auf deren mögliche Nutzungsdauer Einfluss haben.

Ausgehend von dem im oben genannten Forschungsprojekt entwickelten Kriterienkatalog mit insgesamt 25 Kriterien und 76 Indikatoren zur Überprüfung der Umweltauswirkung von Softwareprodukten, wurde für diesen Beschaffungsleitfaden ein vereinfachter und reduzierter Kriterienkatalog mit 13 Kriterien und 32 Unterkriterien entwickelt, der sich für die Beschaffung von Software eignet. Die Kriterien dienen dazu, nachhaltige Software zu erkennen und zu entwickeln. Der Kriterienkatalog kann sowohl bei der Beschaffung fertiger Standardsoftware angewendet werden, als auch bei der Beauftragung einer Software-Entwicklung. Bei der Entwicklung kann die Software durch die iterative Überprüfung der Kriterien kontinuierlich verbessert und effizienter werden.

Bei der Beschaffung von Software sollten die durch die Kriterien beschriebenen Software-Eigenschaften bei den Anbietern abgefragt und bei der Beauftragung von Programmierleistungen durch die Vertragsbedingungen eingefordert werden.

Der vorliegende Leitfaden zur öffentlichen Beschaffung von umweltfreundlicher Software richtet sich an Beschaffungsstellen und erläutert den Kriterienkatalog für nachhaltige Software sowie die Möglichkeiten, diesen bei Ausschreibungen einzusetzen. Der Leitfaden unterstützt damit die Erstellung einer umweltbezogenen Leistungsbeschreibung für energie- und ressourceneffiziente Softwareprodukte. Weitere Qualitätskriterien, die über die umweltbezogenen Anforderungen hinausgehen, wie beispielsweise die Orientierung an der Norm ISO/IEC 25000 („System und Software-Engineering - Qualitätskriterien und Bewertung von System- und Softwareprodukten (SQuARE)“), die Anwendung des Softwareentwicklungsmodells des Bundesinnenministeriums („V-Modell XT“), Beachtung der gesetzlichen Grundlagen für die Barrierefreiheit oder der Sicherheitsanforderungen des Bundesamtes für Sicherheit in der Informationstechnik (BSI), die für die vollständige Leistungsbeschreibung erforderlich sind, werden im vorliegenden Leitfaden nicht behandelt.

## 2 Verwendung dieses Leitfadens

### 2.1 Auswahl der Beschaffungskriterien

Softwareprodukte sind so unterschiedlich und vielfältig, wie es auch physische Produkte sind. Daher kann ein einzelner Beschaffungsleitfaden für alle denkbaren Softwareprodukte nur die allgemeingültigen Anforderungen beschreiben und die grundsätzliche Herangehensweise zur Beschaffung von nachhaltiger Software skizzieren. Der Leitfaden kann nicht auf alle Einzelheiten und Einzelanforderungen eingehen.

Die Beschaffungsstelle bzw. der Bedarfsträger müssen daher teilweise von den genannten Kriterien abstrahieren und diese so anpassen, dass sie für die jeweilige Softwarebeschaffung anwendbar sind. So sind an einer Software, die für Arbeitsplatzcomputer beschafft werden soll, beispielsweise andere Anforderungen an die maximale Auslastung der Computersysteme zu stellen, als bei einer Software für Mobiltelefone oder für Hochleistungsserver in Rechenzentren.

Viele der Anforderungen sind im vorliegenden Anforderungskatalog zunächst als Bewertungskriterien formuliert und nicht als Ausschlusskriterium. Das heißt, die Anbieter von Softwareprodukten müssen Auskunft darüber geben, in welchem Umfang das Softwareprodukt die Anforderungen erfüllt bzw. welchen Messwert es erreicht. Dieser Auskunft schließt sich seitens der Beschaffungsstelle eine Bewertung der Softwareprodukte an, wobei jene Produkte, die bessere Nachhaltigkeitseigenschaften aufweisen, besser bewertet werden. Die Bewertung kann beispielsweise über ein Punktesystem oder über eine Nutzwertanalyse<sup>1</sup> erfolgen.

Dabei muss bereits in der Ausschreibung bekannt gegeben werden, wie die Bewertung erfolgt (Reihenfolge oder Gewichtung), damit die Anbietenden die Möglichkeit haben, Produkte auszuwählen, die die Anforderungen am besten erfüllen.

Manche Kriterien sollten jedoch als Ausschlusskriterium formuliert sein. Softwareprodukte, die diese Kriterien nicht einhalten, werden von der weiteren Bewertung und damit von der Beschaffung ausgeschlossen. Dies betrifft insbesondere das erste Kriterium *Erforderliche minimale Systemvoraussetzungen*. Der Bedarfsträger sollte vor der Ausschreibung hausintern ermitteln, auf welcher Hardware und welchem Betriebssystem (mit welcher Version) die zu beschaffende Software lauffähig sein soll, um auf dieser Grundlage Mindestanforderungen (Ausschlusskriterien) festzulegen und sicherzustellen, dass das Produkt auf der vorhandenen Hardware eingesetzt und genutzt werden kann.

Bei der Festlegung von Ausschlusskriterien wird das Angebot in der Regel eingeschränkt. Wird beispielsweise grundsätzlich gefordert, dass der Quellcode beschaffter Software offengelegt werden muss (Kriterium *Transparenz des Programmcodes* Unterkriterium b), so werden proprietäre Softwareprodukte ausgeschlossen. Sollen dagegen offene und proprietäre Software gegeneinander abgewogen werden, so sollte dieses Kriterium als Bewertungskriterium gesetzt und offene Quellcodes entsprechend besser bewertet werden.

Für vier typische Softwareprodukte werden in der nachfolgenden Tabelle 1 die Kriterien in einer Übersicht dargestellt. Die Bandbreite der Softwareprodukte vom Betriebssystem über Standardsoftware, durch einen Entwicklungsauftrag entstandene Software (Softwareentwicklung) und Weiterentwicklung bestehender Software macht deutlich, dass die Kriterien in unterschied-

---

<sup>1</sup> Ein Beispiel für eine Nutzwertanalyse befindet sich im „Leitfaden zur umweltfreundlichen öffentlichen Beschaffung: Produkte und Dienstleistungen für Rechenzentren und Serverräume“ des Umweltbundesamtes, <https://www.umweltbundesamt.de/publikationen/leitfaden-zur-umweltfreundlichen-oeffentlichen-14>

lichen Phasen des Beschaffungsprozesses genutzt werden können. Bei der Beschaffung des Betriebssystems können sie dazu genutzt werden, den Beschaffungsgegenstand festzulegen, d. h. vorab eine Entscheidung zu treffen, welches Betriebssystem (bzw. welche Version) beschafft werden soll. Bei der Beschaffung von Standardsoftware (beispielsweise Textverarbeitung, Tabellenkalkulation, Buchhaltung, Adressverwaltung usw.) können die Kriterien direkt zum Vergleich verschiedener Angebote genutzt werden. Bei der Beauftragung einer Programmierdienstleistung werden die Kriterien Bestandteil der Vertragsausführungsbedingungen. Die Programmierer werden dazu verpflichtet, die jeweiligen Anforderungen bei der Ausführung ihrer Leistungen umzusetzen.

In Tabelle 1 wird eine Empfehlung gegeben, welche Kriterien für die umweltbezogene Leistungsbeschreibung genutzt werden sollten und ob es sich bei diesen Kriterien um ein Ausschluss- (A) oder Bewertungskriterium (B) handeln sollte. Auch Kombinationen von Ausschluss- und Bewertungskriterium sind möglich (Mindestanforderung mit Bewertung der Übererfüllung). Wird die Entwicklung oder Weiterentwicklung einer Software beauftragt, sind einige der abgefragten Produkteigenschaften erst nach der Auftragsdurchführung am fertigen Produkt bestimmbar. Solche Anforderungen sollten als Dokumentationspflichten (D) in den Vertragsausführungsbedingungen formuliert werden.

Die Zuordnung der Kriterien zu Ausschluss- (A), Bewertungskriterien (B) oder Dokumentationspflichten (D) sollte durch den Bedarfsträger für die Beschaffung jedes spezifischen Softwareproduktes kritisch geprüft und bei Bedarf angepasst werden. Ebenso sollte geprüft werden, ob für die jeweiligen Unterkriterien Mindestanforderungen als Ausschlusskriterium festgelegt und/oder ob diese Kriterien als Bewertungskriterien verwendet werden können.

**Tabelle 1: Auswahl der Kriterien nach Art des zu beschaffenden Softwareprodukts**

Kriterium	Betriebssystem	Standard-Software	Softwareentwicklung	Weiterentwicklung bestehender Software
<b>1 Ressourceneffizienz</b>				
Erforderliche minimale Systemvoraussetzungen	A u. B	A u. B	A u. B	A u. B
Hardware-Auslastung im Leerlauf der Software	B	B	D	D
Hardware-Inanspruchnahme und Energiebedarf bei Ausführung eines Standardnutzungsszenarios	-	B	D	D
Unterstützung des Energiemanagements	A	A	A	A
<b>2 Potentielle Hardware-Nutzungsdauer</b>				
Abwärtskompatibilität	B	B	D	D
Plattformunabhängigkeit und Portabilität	-	-	B	-
<b>3 Nutzungsautonomie</b>				
Transparenz der Datenformate	-	A	A	A
Transparenz des Programmcodes	B	B	A o. B	A o. B

Kriterium	Betriebssystem	Standard-Software	Softwareentwicklung	Weiterentwicklung bestehender Software
Kontinuität des Softwareproduktes	A u. B	A u. B	A	A
Deinstallierbarkeit	-	B	A	D
Offlinefähigkeit	B	B	D	D
Dokumentation des Softwareprodukts, der Lizenz- und Nutzungsbedingungen	B	B	A	A

**Legende:**

- Kriterium wird nicht angewendet
- A** Ausschlusskriterium,
- B** Bewertungskriterium,
- A u. B** Ausschluss wenn Mindestanforderung nicht erfüllt und Bewertung der Übererfüllung,
- A o. B** Festlegung vor der Ausschreibung, ob Ausschluss- oder Bewertungskriterium,
- D** Dokumentationspflicht nach Ausführung der Leistung

## 2.2 Weitere Hinweise zur Anwendung der Beschaffungskriterien

Die hier vorgestellte Methodik zur Bewertung von Softwareprodukten ist noch sehr jung und noch nicht weit verbreitet. Insbesondere solche Kriterien, die die Durchführung von Messungen an einem Referenzsystem erfordern und für die ein Standardnutzungsszenario festgelegt werden muss, um verschiedene Produkte vergleichbar zu machen, werden die Anbieter und Entwickler von Software derzeit vor Herausforderungen stellen. Trotzdem ist es wichtig, den Prozess einzuläutern und die Effizienz von Software einzufordern, um unnötiger Energie- und Ressourcenverschwendung Einhalt zu gebieten.

Solche besonders ambitionierten Kriterien sind die folgenden:

- ▶ Hardware-Auslastung im Leerlauf der Software und
- ▶ Hardware-Inanspruchnahme und Energiebedarf bei Ausführung eines Standardnutzungsszenarios

Um die Systematik dennoch anwendbar zu machen, sollten die Kriterien deshalb pragmatisch verwendet werden, ohne dabei jedoch die Vergleichbarkeit von Softwareprodukten zu gefährden.

Im Abschlussbericht zum Forschungsprojekt „Entwicklung und Anwendung von Bewertungsgrundlagen für ressourceneffiziente Software unter Berücksichtigung bestehender Methodik“ (UBA 2018) wird im Anhang 3 „Operationalisierung für die Aufnahme der Indikatoren“ beschrieben, wie die Kriterien im Rahmen des Forschungsprojektes angewendet wurden. Die dort beschriebene „Operationalisierung“ sollte auch im Rahmen der Beschaffung berücksichtigt werden.

Im Abschnitt 4 Erläuterungen zur Durchführung von Messungen dieses Leitfadens werden außerdem Hinweise gegeben, wie die Messungen durchzuführen sind. Es wird eine Zeitreihe für ein Referenzsystem für Desktop-Computer vorgegeben sowie beschrieben, wie das Standardnutzungsszenario festgelegt werden kann. Ergänzend dazu stellt das Umweltbundesamt aus dem oben genannten Forschungsprojekt ein Softwarewerkzeug bereit, mit dem Log-Dateien getesteter Software ausgewertet und die erforderlichen Angaben berechnet werden können (siehe Literaturverzeichnis OSCAR 2018).

Die Informationen aus Abschnitt 4 sowie dieser gesamte Leitfaden sollten dem Software-Anbieter im Rahmen der Ausschreibung zur Verfügung gestellt werden. Bei einer konkreten Ausschreibung kann es darüber hinaus sinnvoll sein, entweder als ausschreibende Stelle einen Computer als Referenzsystem zur Verfügung zu stellen, auf dem der Anbietende seine Software testen und messen kann oder die Messungen einheitlich durch Dritte (benannte Prüflabore) durchführen zu lassen. Zusätzlich kann durch einen Bieterdialog vor der Angebotsabgabe sichergestellt werden, dass alle Anbietenden das gleiche Verständnis von den abgefragten Kriterien haben, um qualifiziert und vergleichbar anbieten zu können.

### 3 Anforderungen an nachhaltige Software

Die Anforderungen in Kapitel 3 verweisen auf den "Kriterienkatalog nachhaltige Software", der als Anhang 1 im Forschungsbericht UBA 2018 enthalten ist (dort S. 92 ff).

#### 3.1 Ressourceneffizienz

Ein Softwareprodukt soll seine Funktionalität mit einem minimalen Ressourcenaufwand erbringen. Die Ressourceneffizienz eines Produktes soll nach Möglichkeit maximiert werden. Die Ressourceneffizienz wird hier definiert als Menge an „nützlicher Arbeit“ dividiert durch die dabei beanspruchten Ressourcen. Als „nützliche Arbeit“ wird die erfolgreiche Ausführung eines definierten Standardnutzungsszenarios festgelegt (siehe Abschnitt 4.2). Als Ressourcen können für die Ressourceneffizienz grundsätzlich alle natürlichen Ressourcen wie Rohstoffe, Energie und Aufnahmefähigkeit der Umwelt für Emissionen berücksichtigt werden. Zur Operationalisierung der Ressourceneffizienz im Rahmen dieses Beschaffungsleitfadens, werden als Bezugsgröße jedoch technische und energetische Ressourcen verwendet, insbesondere Hardwareressourcen und elektrischer Energiebedarf. Die Beanspruchung von Hardwareressourcen und die Nutzung von Energie führen immer auch zur Beanspruchung natürlicher Ressourcen, weshalb dies hier mit dem Ressourcenverbrauch gleichgesetzt wird.

##### 3.1.1 Erforderliche minimale Systemvoraussetzungen

Die erforderlichen minimalen Systemvoraussetzungen, die für den Betrieb des Softwareprodukts Voraussetzung sind, müssen angegeben werden.

Folgende Angaben sind zu machen:

- a) Minimal erforderliche Prozessor-Architektur (z.B. Intel Core i5, Intel Atom x7)
- b) Minimal erforderlicher lokaler Arbeitsspeicher (MByte)
- c) Minimal erforderlicher lokaler Permanentspeicher (MByte)
- d) Minimal erforderliche Displayauflösung (Pixel x Pixel)
- e) Minimale Bandbreite für Netzzugang (Mbit/s)
- f) Erforderliche Voraussetzungen an weitere Software (Betriebssystem, Middleware und Hilfsanwendungen) (z.B. Windows 7, .NET Framework und Browser Version XY)

Bei der Bewertung verschiedener Produkte sind solche zu bevorzugen, die mit geringeren Systemvoraussetzungen auskommen.

Als Ausschlusskriterium sollten für diese Anforderung zusätzlich konkrete Werte für Prozessor-Architektur, Arbeitsspeicher, Permanentspeicher usw. vorgegeben werden, die durch die Mindestanforderungen der angebotenen Software nicht überschritten werden dürfen. Dies ist dann sinnvoll, wenn sichergestellt werden soll, dass bestehende Hardware beim Bedarfsträger zum Betrieb dieser Software genutzt werden kann und nicht eigens dafür neue Hardware beschafft werden muss.

Anforderung:	Ausschlusskriterium und Bewertungskriterium
Nachweis:	Dokumentation des Anbieters
Verweis:	Kriterienkatalog nachhaltige Software, Kriterium 1.1.2

### 3.1.2 Hardware-Auslastung im Leerlauf der Software

Die Hardware-Auslastung, die durch den Leerlauf des Softwareprodukts entsteht, muss angegeben werden. Hierzu muss die Software testhalber auf einem Referenzsystem (siehe Abschnitt 4.1) installiert und die Hardware-Auslastung entsprechend der in Abschnitt 4 *Erläuterungen zur Durchführung von Messungen* beschriebenen Messvorschriften bestimmt werden. Die Hardware-Auslastung beinhaltet dabei gemäß den Festlegungen des Kriterienkataloges sowohl die zusätzliche Auslastung, die durch das aktive Softwareprodukt entsteht, als auch einen prozentualen Anteil an der Grundauslastung (siehe Abschnitt 4.3).

Folgende Angaben sind zu machen:

- a) Mittlere Prozessorauslastung im Leerlauf unter Standardkonfiguration (%)
- b) Mittlere Arbeitsspeicherbelegung im Leerlauf unter Standardkonfiguration (MByte)
- c) Mittlere Permanentspeicherbelegung im Leerlauf unter Standardkonfiguration (MByte)
- d) Mittlere beanspruchten Bandbreite für Netzzugang im Leerlauf unter Standardkonfiguration (Mbit/s)

Bei der Bewertung verschiedener Produkte sind solche zu bevorzugen, die eine geringere Hardware-Auslastung im Leerlauf verursachen.

Anforderung:	Bewertungskriterium
Nachweis:	Messprotokoll durch Messung an einem Referenzsystem
Verweis:	Kriterienkatalog nachhaltige Software, Kriterium 1.1.3

### 3.1.3 Hardware-Inanspruchnahme und Energiebedarf bei Ausführung eines Standardnutzungsszenarios

Die Hardware-Inanspruchnahme und der Energiebedarf bei Nutzung des Softwareprodukts zur Ausführung eines Standardnutzungsszenarios müssen angegeben werden. Hierzu muss die Software testhalber auf einem Referenzsystem (siehe Abschnitt 4.1) installiert und die Hardware-Inanspruchnahme entsprechend den in Abschnitt 4 *Erläuterungen zur Durchführung von Messungen* beschriebenen Messvorschriften bestimmt werden.

Die Hardware-Inanspruchnahme wird durch Integration (d.h. Summenbildung über Zeitintervalle) der Hardware-Auslastung über die Ausführungsdauer eines Standardnutzungsszenarios definiert. Grundlage für die Berechnung sind Protokolldateien, die das Referenzsystem bei der Ausführung der Software aufzeichnet. Die Hardware-Inanspruchnahme beinhaltet dabei gemäß den Festlegungen des Kriterienkataloges sowohl die zusätzliche Inanspruchnahme, die durch das Softwareprodukt entsteht, als auch einen prozentualen Anteil an der Grundauslastung (siehe Abschnitt 0).

Die Maßeinheiten für die Hardware-Inanspruchnahme sind Einheiten für Arbeitsleistung, wie %\*s (Prozessorarbeit), MByte\*s (Arbeitsspeicherarbeit), MByte/s\*s = MByte (Permanentspeicherarbeit, Lesen und Schreiben) und MBit/s\*s = MBit (im Netzwerk übertragene Datenmenge).



Zusätzlich zur Hardware-Inanspruchnahme muss der elektrische Energiebedarf des Referenzsystems bestimmt werden. Die benötigte elektrische Energie ergibt sich durch Integration der elektrischen Leistungsaufnahme über die Ausführungsdauer des Standardnutzungsszenarios. Abweichend von den Festlegungen für die Hardware-Inanspruchnahme werden für die Messungen der elektrischen Energie nur die Leistungswerte, die über den elektrischen Grundverbrauch des Referenzsystems hinausgehen (Netto-Leistung), berücksichtigt.

Folgende Angaben sind jeweils bei Ausführung des Standardnutzungsszenarios unter Standardkonfiguration zu machen:

- a) Prozessorarbeit (%\*s)
- b) Arbeitsspeicherarbeit (MByte\*s)
- c) Permanentpeicherarbeit (Lesen und Schreiben) (MByte/s\*s)
- d) Übertragene Datenmenge für Netzzugang (Mbit/s\*s)
- e) Energiebedarf (netto) (Wh)

Bei der Bewertung verschiedener Produkte sind solche zu bevorzugen, die geringere Hardware-Inanspruchnahme und geringeren Energiebedarf verursachen.

Anforderung:	Bewertungskriterium
Nachweis:	Messprotokoll durch Messung an einem Referenzsystem
Verweis:	Kriterienkatalog nachhaltige Software, Kriterien 1.1.4 und 1.2

### 3.1.4 Unterstützung des Energiemanagements

Ein Softwareprodukt soll ein Computersystem nicht daran hindern, im ruhenden Betrieb den Bedarf an Hardwarekapazitäten und damit auch den Energieverbrauch zu reduzieren (bspw. Standby-Modus / Ruhezustand). Das Energiemanagement des Computers oder verbundener Clientsysteme darf nicht beeinträchtigt werden. Außerdem darf die Funktionalität der Software durch das Energiemanagement nicht negativ beeinflusst werden (z.B. durch eintretenden Datenverlust, Beeinträchtigung der Bedienbarkeit). Die Notwendigkeit einer (datenverlustfreien) Wiederanmeldung an eine serverbasierte Software oder die Zeit zum Reaktivieren des Computersystems aus einem Energiesparmodus wird nicht als Beeinträchtigung der Bedienbarkeit und Funktionalität gewertet.

Folgende Anforderung ist zu bestätigen:

- a) Die Software ist auch bei aktiviertem Energiemanagement der darunterliegenden Systemschichten oder der verbundenen Clientsysteme uneingeschränkt funktional nutzbar. Bei serverbasierter Software, beinhaltet dies, dass das Aktivieren des Energiemanagements auf Client-Seite die Funktionalität nicht beeinträchtigt (z.B. darf es nicht zu einem Verlust von Session-Information führen, wenn der Client in einen Ruhezustand wechselt).

Die Nichterfüllung dieses Kriteriums führt zu einem Ausschluss des Angebots.

Anforderung:	Ausschlusskriterium
Nachweis:	Eigenerklärung des Anbieters
Verweis:	Kriterienkatalog nachhaltige Software, Kriterium 1.3.2

## 3.2 Potentielle Hardware-Nutzungsdauer

Eine beschaffte Software oder beauftragte Softwareentwicklung soll nicht dazu beitragen, dass bestehende Hardware ausgemustert werden muss, weil sie den Leistungsanforderungen der Software nicht gerecht wird. Software-Updates sollen darüber hinaus nicht dazu führen, dass auch Hardware-Updates erfolgen müssen. Stattdessen soll dem Nutzenden ein voneinander unabhängiges Management von Software- und Hardware-Erneuerung ermöglicht werden. Die Anforderungen innerhalb dieses Abschnitts adressieren deshalb die potenzielle Hardware-Nutzungsdauer.

### 3.2.1 Abwärtskompatibilität

Der Anbieter des Softwareprodukts muss verbindlich erklären, mit welchem Referenzsystem aus welchem Kalenderjahr die Software noch lauffähig ist. Die Referenzsysteme sind hierbei vorgegeben (siehe Abschnitt 4.1). Erforderliche Zusatzsoftware (Betriebssystem, Framework, Anwendersoftware) sind bei der Bestimmung dieser Jahreszahl zu berücksichtigen.

Folgende Angabe ist zu machen:

- a) Jahreszahl des Referenzsystems (Kalenderjahr)

Bei der Bewertung verschiedener Produkte sind solche zu bevorzugen, die auf älteren Referenzsystemen noch lauffähig sind.

Anforderung:	Bewertungskriterium
Nachweis:	Dokumentation des Anbieters
Verweis:	Kriterienkatalog nachhaltige Software, Kriterium 2.1

### 3.2.2 Plattformunabhängigkeit und Portabilität

Ein Softwareprodukt soll im Idealfall auf verschiedenen aktuell verbreiteten produktiven Systemumgebungen (Hardware und Software) betrieben werden können. Anwender der Software sollen ohne Nachteil zwischen verschiedenen Systemumgebungen wechseln können.

Folgende Angabe ist zu machen:

- a) Nennung der Betriebssysteme und deren Versionen (ggf. zusätzlich Laufzeitumgebungen), auf der das Softwareprodukt lauffähig ist.

Bei der Bewertung verschiedener Produkte sind solche zu bevorzugen, die auf einer Vielzahl von produktiven Systemumgebungen lauffähig sind. Hierbei sind insbesondere die aktuell im Unternehmen (bzw. in der Behörde oder Einrichtung) eingesetzten bzw. die zukünftig geplanten Systemumgebungen positiv zu berücksichtigen.

Anforderung:	Bewertungskriterium
Nachweis:	Dokumentation des Anbieters
Verweis:	Kriterienkatalog nachhaltige Software, Kriterium 2.2

### 3.3 Nutzungsautonomie

Ein Softwareprodukt soll die Autonomie des Nutzens im Umgang mit dem Produkt nicht einschränken. Das angestrebte Ideal ist ein Softwareprodukt, das die Entscheidungsfreiheit der Nutzenden über die Nutzungsdauer der Software und die Beanspruchung von Hardware-Kapazitäten (bspw. nur den Funktionsumfang zu installieren, der auch benötigt wird) und damit natürlicher Ressourcen weitestgehend ermöglicht. Die nachfolgend genannten Anforderungen tragen zu unterschiedlichen Aspekten der Nutzungsautonomie bei.

#### 3.3.1 Transparenz der Datenformate

Damit die Interoperabilität des beschafften Softwareproduktes mit bereits vorhandener oder zukünftig beschaffter Software ermöglicht wird, müssen die Datenformate (Datei- und Datenstromformate), die das Softwareprodukt zur Ablage der von Nutzenden erzeugten Daten oder zum Austausch von Daten mit anderen Programmen verwendet, ausreichend dokumentiert sein. Die Datenformate müssen als offene Standards angelegt sein, so dass eine Weiternutzung der Daten mit einem anderen Softwareprodukt möglich ist.

Zur Anwendung dieses Kriteriums muss durch die ausschreibende Stelle für das Unterkriterium b) angegeben werden, welche Standards zum Zeitpunkt der Vergabe zu den offenen Standards gezählt werden bzw., wie die Offenheit des Standards nachgewiesen werden kann.<sup>2</sup>

Folgende Angaben sind zu machen:

- a) Vorlage der Handbücher oder technischer Datenblätter, in denen die Datenformate dokumentiert sind.
- b) Benennung der Datenformate und Zuordnung zu einem offenen Standard.
- c) Beispielhafte Nennung weiterer Softwareprodukte (anderer Anbieter), die diese Datenformate verarbeiten können.

Die Nichterfüllung dieses Kriteriums führt zu einem Ausschluss des Angebots.

<sup>2</sup> Siehe auch „European Interoperability Framework“, <http://ec.europa.eu/idabc/en/document/3473/5585.html>

Anforderung:	Ausschlusskriterium
Nachweis:	Dokumentation des Anbieters
Verweis:	Kriterienkatalog nachhaltige Software, Kriterium 3.1.1

### 3.3.2 Transparenz des Programmcodes

Sofern es beabsichtigt ist, das beschaffte Softwareprodukt um zusätzliche Funktionalitäten zu erweitern, müssen die Anwendungs-Programmier-Schnittstellen (APIs) klar dokumentiert sein. Die API-Schnittstellen sollen offenen Standards entsprechen.

Wenn zusätzlich durch den Beschaffer beabsichtigt ist, das Produkt unabhängig vom Anbieter weiterzuentwickeln, muss außerdem der Quellcode ganz oder teilweise offengelegt werden sowie die Erlaubnis dazu erteilt werden, das Produkt zu verändern. Dies sollte durch die Lizenzierung der Software mit einer für den Anwendungsfall passenden Softwarelizenz geschehen.

Folgende Angaben sind zu machen:

- a) Sofern Anwendungs-Programmier-Schnittstellen vorhanden sind: Vorlage der Schnittstellendokumentation.
- b) Angabe, in welchem Umfang Programmbestandteile mit dokumentiertem Quellcode offen gelegt werden, bzw. ob das gesamte Softwareprodukt als offener Quellcode bereitgestellt wird.
- c) Nennung der Lizenz unter der das Softwareprodukt vertrieben wird und Angabe dazu, ob die Software durch Dritte weiterentwickelt werden darf.

Die genannten Bewertungskriterien sind optional zu verwenden und je nach Anwendungsfall unterschiedlich stark zu gewichten oder ggf. als Ausschlusskriterien (d.h. als verpflichtend) zu setzen. Bei der Bewertung verschiedener Produkte sind solche zu bevorzugen, die einen höheren Grad der Transparenz und Interoperabilität aufweisen.

Anforderung:	Wahlweise pro Indikator Bewertungskriterium oder Ausschlusskriterium
Nachweis:	Dokumentation des Anbieters
Verweis:	Kriterienkatalog nachhaltige Software, Kriterium 3.1.2

### 3.3.3 Kontinuität des Softwareproduktes

Es soll möglich sein, das Softwareprodukt über einen längeren Zeitraum zu nutzen, ohne dass schwerwiegende Nachteile (insbesondere Probleme der IT-Sicherheit) auftreten. Bei der Bereitstellung von Software-Updates durch den Anbieter soll der Anwender die Wahl haben, ob nur Sicherheitsupdates oder auch sonstigen (z.B. funktionelle) Updates installiert werden sollen

Folgende Angaben sind zu machen:

- a) Nennung des Zeitraums, für den der Anbieter die Unterstützung des Produkts mit Sicherheitsupdates garantiert (Enddatum).
- b) Angabe einer Reaktionszeit bis zur Bereitstellung von Sicherheitsupdates nach Bekanntwerden von Sicherheitslücken (Tage).
- c) Angabe darüber, ob zwischen Sicherheitsupdates und sonstigen Updates unterschieden wird und ob der Anwender sonstige Updates überspringen kann.

Für die genannten Kriterien sollten durch den Bedarfsträger sowohl Mindestanforderungen als Ausschlusskriterium (Mindestzeitraum für Sicherheitsupdates, maximale Reaktionszeit) definiert werden, als auch deren Übererfüllung als Bewertungskriterium positiv berücksichtigt werden. Bei der Bewertung anhand der Bewertungskriterien sind solche Angebote zu bevorzugen, die eine höhere Kontinuität des Softwareproduktes ermöglichen.

Anforderung:	Ausschlusskriterium und Bewertungskriterium
Nachweis:	Dokumentation des Anbieters
Verweis:	Kriterienkatalog nachhaltige Software, Kriterium 3.1.3

### 3.3.4 Deinstallierbarkeit

Ein Softwareprodukt soll am Ende seiner Nutzungsdauer möglichst rückstandsfrei von dem Computersystem entfernt werden können und keine unnötigen Datenspuren hinterlassen. Die mit der Software erstellten und bearbeiteten Daten sind hiervon ausgenommen und dürfen auch bei Deinstallation der Software nicht automatisch gelöscht werden. Hierzu muss die Software inklusive der ggf. durch die Software zusätzlich installierten Komponenten oder Bibliotheken durch den Nutzenden (bzw. den Systemadministrator) einfach und ohne großen zeitlichen Aufwand deinstallierbar sein.

Folgende Angaben sind zu machen:

- a) Beschreibung des Procedere der Deinstallation des Softwareproduktes.
- b) Nennung der Anzahl an verbliebenen Dateien, Ordnern, Datenmengen und ggf. Registry-Einträgen o.ä. nach Deinstallation des Softwareproduktes.

Bei der Bewertung verschiedener Produkte sind solche zu bevorzugen, die besonders einfach und rückstandsfrei deinstallierbar sind.

Anforderung:	Ausschlusskriterium oder Bewertungskriterium
Nachweis:	Messprotokoll an einem Referenzsystem. Wurde kein Referenzsystem festgelegt, kann diese Messung alternativ auch an einem von Anbieter benannten Computersystem durchgeführt werden.
Verweis:	Kriterienkatalog nachhaltige Software, Kriterium 3.2.1

### 3.3.5 Offlinefähigkeit

Grundsätzlich soll die Funktionalität und Verfügbarkeit einer Software nicht durch externe Faktoren, wie beispielsweise die Verfügbarkeit eines Lizenzservers, beeinträchtigt werden. Das Softwareprodukt soll weitgehend auch ohne eine Netzwerkanbindung (offline) funktionsfähig sein, sofern diese nicht explizit für dessen Funktionserfüllung erforderlich sind.

Folgende Angaben sind zu machen:

- a) Bewertung der Offlinefähigkeit mit folgender Klassifikation
  - a. Offline-Betrieb möglich
  - b. Offline-Betrieb mit Einschränkungen möglich  
(Nennung der Einschränkungen)
  - c. Offline-Betrieb unmöglich

Bei der Bewertung verschiedener Produkte sind solche zu bevorzugen, die eine höhere Offlinefähigkeit aufweisen.

Anforderung:	Bewertungskriterium
Nachweis:	Eigenerklärung des Anbieters
Verweis:	Kriterienkatalog nachhaltige Software, Kriterium 3.4.1

### 3.3.6 Dokumentation des Softwareprodukts, der Lizenz- und Nutzungsbedingungen

Das Softwareprodukt muss mit einer Dokumentation versehen sein, die eine langfristige und ressourcenschonende Nutzung des Produktes ermöglicht. Insbesondere soll darin die Installation, Deinstallation, Datenimport und -export beschrieben sowie Möglichkeiten zur Reduzierung von Ressourceninanspruchnahme aufgezeigt werden (z.B. Aktivierung von Energiesparmodi, Deaktivierung nicht genutzter Module, Beeinflussung der Größe von Auslagerungsdateien, Bereinigung von Datenresten). Weiterhin sollen in der Dokumentation Angaben zu den Lizenz- und Nutzungsbedingungen gemacht werden, sodass ggf. eine rechtssichere Weiterentwicklung des Softwareproduktes ermöglicht wird. Die Dokumentation kann als separate Textdatei oder innerhalb eines Hilfe-Systems in der Software integriert sein.

Folgende Angaben sind zu machen:

- a) Vorlage der Produktdokumentation

Bei der Bewertung verschiedener Produkte sind solche zu bevorzugen, die mit einer besser verständlichen Dokumentation versehen sind.

Soll die Software erst noch entwickelt werden, so ist die Vorlage einer Dokumentation als Vertragsbestandteil aufzunehmen.

Anforderung: Ausschlusskriterium oder Bewertungskriterium

Nachweis: Vorlage der Produktdokumentation

Verweis: Kriterienkatalog nachhaltige Software, Kriterium 3.5.1

## 4 Erläuterungen zur Durchführung von Messungen

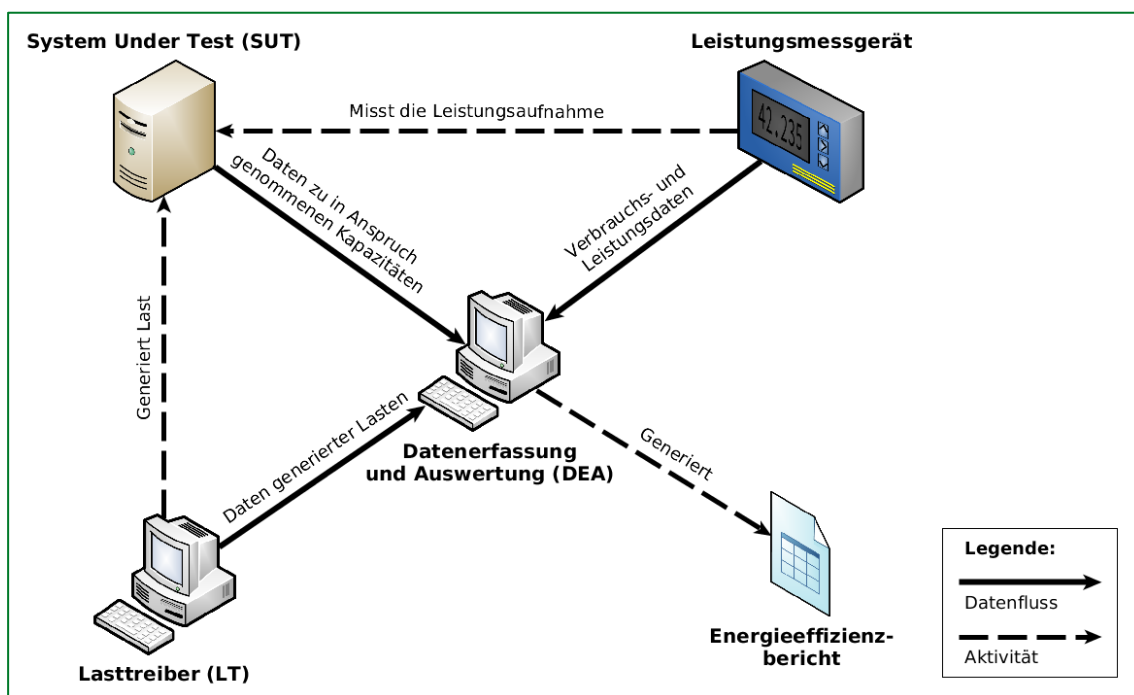
Die Einhaltung einiger der oben genannten Anforderungen muss durch Vorlage von Messprotokollen an einem Referenzsystem entsprechend den hier beschriebenen Messvorschriften nachgewiesen werden. Nur so ist es möglich, die Angaben unterschiedlicher Softwareanbieter zu vergleichen.

Dies betrifft folgende Anforderungen:

- ▶ Hardware-Auslastung im Leerlauf der Software
- ▶ Hardware-Inanspruchnahme und Energiebedarf bei Ausführung eines Standardnutzungsszenarios
- ▶ Abwärtskompatibilität
- ▶ Deinstallierbarkeit

Als Voraussetzung für die Durchführung von Messungen muss zum einen ein Referenzsystem festgelegt werden, also ein System Under Test (SUT), auf dem die Software lauffähig installiert wird. Das Referenzsystem muss mit geeigneten Energiemessgeräten ausgestattet und die Hardware-Auslastung durch Datenlogger erfasst werden. Zum zweiten muss ein Standardnutzungsszenario festgelegt werden, das die typische Nutzung der Software beschreibt und damit eine reproduzierbare Abfolge an Befehlen und Nutzerinteraktionen als generierte Last vorgibt. In der nachfolgenden Abbildung ist der Messaufbau exemplarisch dargestellt.

**Abbildung 1: Exemplarischer Messaufbau Energie- und Ressourceneffizienzmessung von Software**



Quelle: UBA 2018

In den folgenden Abschnitten werden Regeln zur Festlegung des Referenzsystems und des Standardnutzungsszenarios aus dem Forschungsprojekt „Entwicklung und Anwendung von Bewertungsgrundlagen für ressourceneffiziente Software unter Berücksichtigung bestehender Metho-



dik“ (UBA 2018) vorgestellt. Für die Durchführung von Ausschreibungen oder die Markterkundung müssen durch die ausschreibende Stelle das Referenzsystem und das Standardnutzungsszenario vorgegeben werden.

Die Messungen müssen entweder durch geeignete Prüflabore oder durch die Anbieter selbst erfolgen, sofern diese über die erforderliche technische Ausstattung verfügen. Als Hilfestellung stellt das Umweltbundesamt eine Auswertungssoftware (OSCAR 2018, siehe Literaturverzeichnis) zur Verfügung, mit der die Log-Dateien der Hardware-Auslastung und elektrischen Leistungsaufnahme des Referenzsystems ausgewertet werden können. Die Auswertungssoftware beinhaltet außerdem eine ausführliche Anleitung zur Durchführung von Messungen von Energieverbrauch und Hardwarekapazitäten von Software mittels Standard-Benchmarking-Werkzeugen. Weitere Informationen zur Anwendung der Kriterien enthält auch der Anhang 3 des Forschungsberichtes (UBA 2018), in dem die „Operationalisierung für die Aufnahme der Indikatoren“ beschrieben wird.

## 4.1 Referenzsystem

Als Referenzsystem ist durch die ausschreibende Stelle in der Leistungsbeschreibung ein Computersystem vorzugeben, das den tatsächlichen Anwendungsfall möglichst gut repräsentiert. Bei Softwareprodukten zur lokalen Anwendung im Büro ist dies beispielsweise ein Desktop-Computer mit einer behördentypischen Hardwareausstattung und einem verbreiteten Betriebssystem. Soll das Softwareprodukt dagegen für eine Server-Anwendung oder beispielsweise ein Smartphone entwickelt bzw. beschafft werden, muss als Referenzsystem ein anderes, geeignetes Computersystem vorgegeben werden.

Ein Referenzsystem muss zur Durchführung von Messungen bei folgenden Anforderungen vorgegeben werden:

- ▶ Hardware-Auslastung im Leerlauf der Software,
- ▶ Hardware-Inanspruchnahme und Energiebedarf bei Ausführung eines Standardnutzungsszenarios und
- ▶ Deinstallierbarkeit.

Die Messungen der Deinstallierbarkeit kann, sofern kein Referenzsystem festgelegt wurde, alternativ auch an einem anderen Computersystem durchgeführt werden.

Als Referenzsysteme für „Arbeitsplatzcomputer für Büro-Software“ wurden in dem oben genannten Forschungsprojekt (UBA 2018) die in Tabelle 2 dargestellte Computersysteme<sup>3</sup> festgelegt. Für eine aktuelle Ausschreibung sollte beispielsweise ein Referenzsystem aus dem Jahr 2017 gewählt werden oder bei veralteter Hardware-Ausstattung auch ein älteres System, beispielsweise aus dem Jahr 2014. Bei zukünftigen Ausschreibungen können auch neuere Computersysteme mit aktuelleren Spezifikationen festgelegt werden.

Im Forschungsvorhaben (UBA 2018) sind darüber hinaus auch ältere Referenzsysteme ab dem Kalenderjahr 2010 genannt, die insbesondere für die Anforderung

- ▶ Abwärtskompatibilität
- eingesetzt werden können.

---

<sup>3</sup> Sie beschreiben die typische technische Ausstattung von Computern in Behörden.

**Tabelle 2: Referenzsystems als Zeitreihe 2014 – 2017**

Technische Parameter	2014 – 2015	2016	2017
<b>Hersteller</b>	Fujitsu	Fujitsu	Fujitsu
<b>Modell</b>	Esprimo P920 <sup>4</sup>	Esprimo P956 <sup>5</sup>	Esprimo P957 <sup>6</sup>
<b>Prozessor</b>	Intel i5-4590	Intel i5-6500	Intel i5-6500
<b>Cores</b>	4	4	4
<b>Taktfrequenz</b>	3,3 GHz	3,2 GHz	3,2 GHz
<b>RAM</b>	4 GB (DDR4, 2133MHz)	4 GB (DDR4, 2133MHz)	8 GB (DDR4, 2133MHz)
<b>Festplatte (HDD / SSD)</b>	HDD SATA III 500GB (6 Gbit/s)	HDD SATA III 500GB (6 Gbit/s)	SSD SATA III 128GB (6 Gbit/s)
<b>Grafikkarte</b>	Intel HD Graphics 4600	Intel HD Graphics 530	Intel HD Graphics 530
<b>Netzwerk</b>	LAN GigaBit	LAN GigaBit	LAN GigaBit
<b>Betriebssystem</b>	Win 8.1	Win 8.1	Win 10
<b>Wirkungsgrad Netzteil</b>	91% - 94%	84% - 92%	84% - 92%

Quelle: UBA 2018

## 4.2 Standardnutzungsszenario

Das Standardnutzungsszenario simuliert die realitätsnahe Nutzung der Software. Das Szenario beinhaltet die Abarbeitung von Aufgaben, für die die Software entwickelt bzw. beschafft wurde und ggf. die Interaktion mit dem Anwender der Software. Durch die Verwendung von einheitlichen Standardnutzungsszenarien ist es bspw. möglich, den Energieverbrauch und die Hardware-Inanspruchnahme verschiedener Softwareprodukte der gleichen Produktgruppe miteinander zu vergleichen.

Die Entwicklung von Standardnutzungsszenarien sollte einem transparenten Verfahren unterzogen werden, das interessierte Kreise einbezieht. Interessierte Kreise sind dabei insbesondere die potenziellen Anwenderinnen und Anwender der Software, aber auch Software-Entwickler. Im Fall einer Ausschreibung kann das Standardnutzungsszenario auch von der ausschreibenden Stelle vorgegeben werden.

Zur Sicherstellung der Vergleichbarkeit von Softwareprodukten einer Produktgruppe wird mit allen Produkten der Gruppe dasselbe Standardnutzungsszenario ausgeführt. Folgende Aspekte kommen bei der Erstellung von Standardnutzungsszenarien zum Tragen:

<sup>4</sup> Datenblatt: <https://sp.ts.fujitsu.com/dmsp/Publications/public/ds-ESPRIMO-P920-0Watt-de.pdf>  
<sup>5</sup> Datenblatt: <https://sp.ts.fujitsu.com/dmsp/Publications/public/ds-ESPRIMO-P956-E90-de.pdf>  
<sup>6</sup> Datenblatt: <https://sp.ts.fujitsu.com/dmsp/Publications/public/ds-ESPRIMO-P957-E90-de.pdf>

- ▶ Standardnutzungsszenarios werden mittels einer Automatisierungssoftware wie z. B. Win-Automation erstellt und enthalten Standardaufgaben, die mit der Software bearbeitet werden. Dabei wird darauf geachtet, dass sich vor jeder Aktion eine Wartezeit befindet, welche den Nutzer simuliert. Des Weiteren wird darauf geachtet, dass möglichst viele verschiedene Standardaktionen ausgeführt werden, die beim Nutzenden in einem typischen Arbeitsablauf vorkommen.
- ▶ Ein wichtiger Faktor bei der Erstellung eines Standardnutzungsszenarios ist die Reproduzierbarkeit der Testabläufe für alle zu vergleichenden Softwareprodukte. Dazu muss im Vorfeld der Erstellung des Szenarios geprüft werden, dass alle Aktionen, die in dem Szenario enthalten sein sollen, auch mit allen zu vergleichenden Softwareprodukten möglichst in Standardkonfiguration, also ohne zusätzliche Pakete oder Plugins durchgeführt werden können.
- ▶ Es muss darauf geachtet werden, dass bei der wiederholten Durchführung (beispielsweise 30 Testdurchläufe) alle in einem Durchlauf im Referenzsystem persistierten Änderungen vor Ablauf des darauffolgenden Durchlaufs wieder rückgängig gemacht werden müssen, um einen vergleichbaren Durchlauf zu gewährleisten. Systemeigene Prozesse, die unregelmäßig auftreten und die Messergebnisse verfälschen, wie bspw. Virens Scanner, müssen deaktiviert werden.
- ▶ Die Synchronisierung der Messungen erfolgt über Zeitstempel. Sämtliche Messpunkte der Messung der Hardwarekapazitäten, wie auch der Messung der elektrischen Leistung, werden mit einem Zeitstempel versehen. Zusätzlich erzeugt die Automatisierungssoftware eine Log-Datei, in der für jeden Messdurchlauf eine Start- und Endzeit sowie Zeitstempel für Beginn und Ende der durchgeführten Aktionen gespeichert wird. So ist es möglich, den einzelnen Aktionen genau die Inanspruchnahme von Hardwarekapazitäten und Energieverbrauch zuzuordnen.
- ▶ Zur Verarbeitung und späteren Reproduzierbarkeit werden die Messwerte, die Log-Datei und das Automatisierungsskript an zentraler Stelle gesichert.

Tabelle 3 gibt einen beispielhaften Überblick über die durchgeführten Aktionen innerhalb der Standardnutzungsszenarios, wie sie beim Forschungsprojekt (UBA 2018) festgelegt wurden.

**Tabelle 3: Überblick der durchgeführten Aktionen für ausgewählte Softwareprodukte**

Textverarbeitung	Browser
Gesamten Text bearbeiten	E-Mail lesen/schreiben
Inhaltsverzeichnis einfügen & aktualisieren	Web-Videostream anschauen
Ansicht anpassen	Online-Shop besuchen
Inhalte hinzufügen & bearbeiten	Lesezeichen setzen
PDF erzeugen	Add-on installieren
Speichern	Datei downloaden

Content Management Systeme	Datenbanken
Kommentare beantworten Neue Seite erstellen Alle Seiten veröffentlichen PDF-Dateien hochladen PDF-Dateien verlinken Seite betrachten Zusätzlich: Lastgenerierung zur Simulation von Besuchern	Schema bereits vorhanden Daten eintragen Daten lesen Daten verändern Daten löschen 230 Durchgänge je Funktion, 120.000 Zugriffe je Durchgang

Quelle: UBA 2018

### 4.3 Grundlegende Definitionen für die Messung und Berechnung von Softwareeigenschaften

Tabelle 4: Grundlegende Definitionen

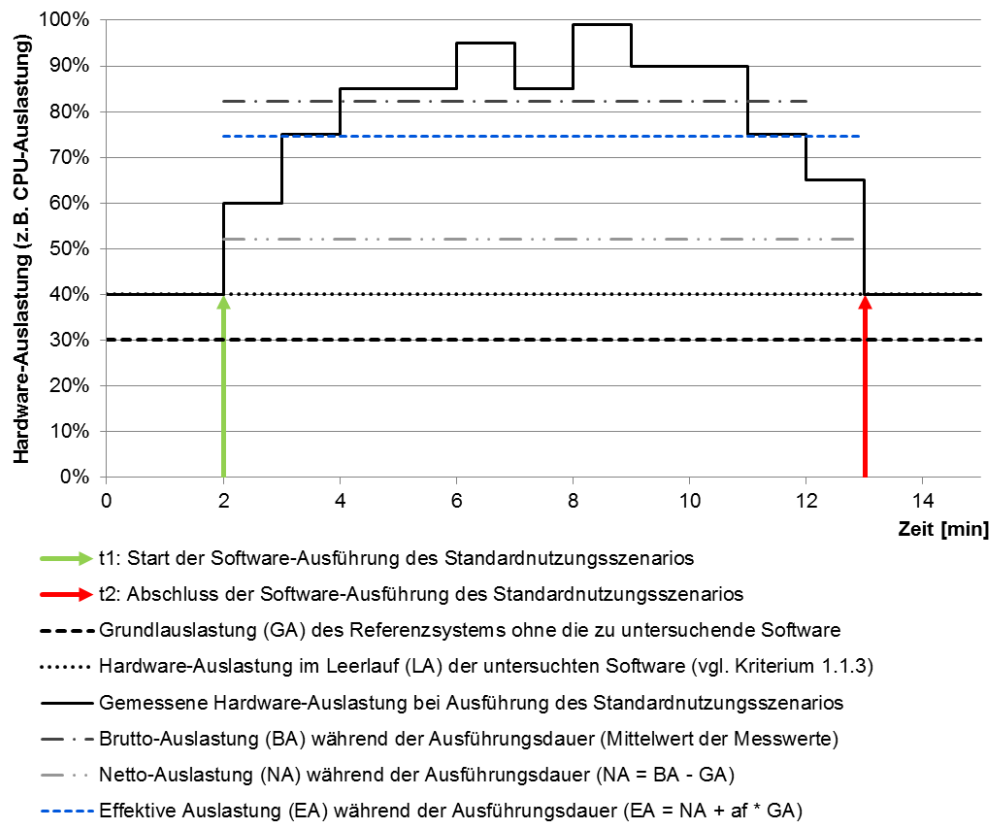
Bezeichner	Name	Definition	Anmerkung
VA <sub>i</sub>	Vollauslastung	Obere Grenze der Auslastung i des Referenzsystems	Bei Prozessorleistungen beträgt die Vollauslastung 100%, bei Arbeitsspeicher die Summe der installierten RAM-Kapazitäten, beim Netzwerk die maximale Übertragungsgeschwindigkeit usw.
GA <sub>i</sub>	Grundauslastung	Messwert der mittleren Auslastung i des Referenzsystems ohne die zu untersuchende Software.	
LA <sub>i</sub>	Leerlauf-Auslastung	Messwert der mittleren Auslastung i des Referenzsystems im Leerlauf der untersuchten Software	Sie beinhaltet sowohl die Grundauslastung (GA <sub>i</sub> ) als auch die zusätzliche durch den Leerlaufbetrieb der Software verursachte Hardware-Auslastung.
NLA <sub>i</sub>	Netto-Leerlauf-Auslastung	$NLA_i = LA_i - GA_i$	Sie beschreibt die durch den Leerlaufbetrieb der Software über die Grundauslastung hinausgehende beanspruchte Hardware-Auslastung
t	Ausführungsdauer	Laufzeit der untersuchten Software zur Ausführung des Standardnutzungsszenarios auf dem Referenzsystem	Sie beginnt mit dem Start des Standardnutzungsszenarios und endet mit der Erledigung aller dort vorgesehenen Aktionen, inklusive nachgelagerter Prozesse (z.B. Freigeben von Arbeitsspeicher, Löschung temporärer Dateien).

Bezeichner	Name	Definition	Anmerkung
BA <sub>i</sub>	Brutto-Auslastung	Mittlere Auslastung des Referenzsystems bei Ausführung eines Standardnutzungsszenarios über dessen Ausführungsdauer	Sie wird als zeitgewichteter Mittelwert der Messwerte über die Ausführungsdauer berechnet.
NA <sub>i</sub>	Netto-Auslastung	$NA_i = BA_i - GA_i$	Sie bezeichnet die durch die Software verursachte mittlere Hardware-Auslastung nach Abzug der Grundauslastung des Referenzsystems.
AF <sub>i</sub>	Allokationsfaktor	$AF_i = NA_i / (VA_i - GA_i)$ (Allokationsfaktor für die Ausführung des Standardnutzungsszenarios)	Verhältnis der durch die Software beanspruchten Netto-Auslastung zu der maximal zur Verfügung stehenden Auslastung. Dabei ist berücksichtigt, dass maximal nur die Differenz zwischen der Vollauslastung (VA) und der Grundauslastung (GA) zur Verfügung steht.
AFL <sub>i</sub>	Leerlauf-Allokationsfaktor	$AFL_i = NLA_i / (VA_i - GA_i)$ (Allokationsfaktor für den Leerlauf der Software)	
EA <sub>i</sub>	Effektive Auslastung	$EA_i = NA_i + AF_i * GA_i$	
ELA <sub>i</sub>	Effektive Leerlauf-Auslastung	$ELA_i = NLA_i + AFL_i * GA_i$	Genutzt um die Indikatoren zur Berechnung der Hardwareansprüche von Kriterium 1.1.3 zu berechnen
H <sub>i</sub>	Hardware-Inanspruchnahme	$H_i = EA_i * t$	Genutzt um die Indikatoren zur Berechnung der Hardwareansprüche von Kriterium 1.1.4 zu berechnen

Quelle: UBA 2018

Die nachfolgende Abbildung stellt den Messzyklus beispielhaft dar und illustriert die verschiedenen Auslastungen und deren Bezeichnung.

**Abbildung 2: Beispielhafter Messzyklus zur Bestimmung der Hardware-Auslastung**



Quelle: UBA 2018

## 5 Literaturverzeichnis

- OSCAR 2018: Open Source Software Consumption Analysis; Auswertungssoftware zu Hardwareauslastung und Energieverbrauch; Birkenfeld; Dezember 2018; <https://green-software-engineering.de/downloads/ssd-tools.html>
- SSD-Kriterien: Hilty, L.; Naumann, S.; Maksimov, Y.; Kern, E.; Filler, A.; Guldner, A.; Gröger, J.; Kriterienkatalog für nachhaltige Software in der jeweils aktuellen Fassung; <https://www.umwelt-campus.de/forschung/projekte/green-software-engineering/kriterienkatalog>; Fassung von Mai 2017 in: UBA 2018 (dort Anhang 1, Seite 92 ff)
- UBA 2018: Gröger, J.; Naumann, S.; Hilty, L.; Filler, A.; Guldner, A.; Kern, E.; Köhler, A. R.; Maksimov, Y.; Entwicklung und Anwendung von Bewertungsgrundlagen für ressourceneffiziente Software unter Berücksichtigung bestehender Methodik; Öko-Institut e.V. in Kooperation mit Hochschule Trier, Umwelt-Campus Birkenfeld und Universität Zürich, Institut für Informatik im Auftrag des Umweltbundesamtes; Dessau-Roßlau; Dezember 2018; <https://www.umweltbundesamt.de/publikationen/entwicklung-anwendung-von-bewertungsgrundlagen-fuer>