BLUE ANGEL

The German Ecolabel

Resource and Energy-Efficient Software Products

DE-UZ 215

Basic Award Criteria

Edition June 2024

Version 4

The Environmental Label is supported by the following four institutions:

The Federal Ministry for the Environment, Nature Conservation, Nuclear Safety and Consumer Protection is the owner of the label. It regularly provides information on the decisions taken by the Environmental Label Jury.

The German Environmental Agency with its specialist department for "Ecodesign, Eco-Labelling and Environmentally friendly Procurement" acts as office of the Environmental Label Jury and develops the technical criteria of the Basic Criteria for Award of the Blue Angel.

The Environmental Label Jury is the independent, decision-making body for the Blue Angel and includes representatives from environmental and consumer associations, trade unions, industry, the trade, crafts, local authorities, academia, the media, churches, young people and the German federal states.

The RAL gGmbH is the awarding body for the Environmental Label. It organises the process for developing the relevant award criteria in independent expert hearings – which involve all relevant interest groups.

Version 1 (01/2020): First edition, term of validity until 31/12/2023
Version 2 (01/2023): Extended by 1 year without changes until 31/12/2024
Version 3 (01/2024): Extended without changes, term until 31/12/2025
Version 4 (06/2024): Revision, term of validity until 31/12/2027

Table of contents

This document is a translation of a German original. In case of dispute, the original document should be taken as authoritative.

1 Introduction

1.1 Preface

In cooperation with the Federal Ministry for the Environment, Nature Conservation, Nuclear Safety and Consumer Protection, the German Environmental Agency and considering the results of the expert hearings conducted by RAL gGmbH, the Environmental Label Jury has set up these Basic Criteria for the Award of the Environmental Label. RAL gGmbH has been tasked with awarding the Environmental Label.

Upon application to RAL gGmbH and on the basis of a Contract on the Use of the Environmental Label to be concluded with RAL gGmbH, the permission to use the Environmental Label may be granted to all products, provided that they comply with the requirements as specified hereinafter.

The product must comply with all the legal requirements in the country in which it is to be marketed. The applicant[1] shall declare that the product meets this requirement.

1.2 Background

Software is an important driver of the consumption of energy and resources by digital infrastructure. Its use results in the purchase of digital end devices and the continual expansion of data center capacities and also requires continuous improvements in the performance of communication networks. Increasingly sophisticated software also makes still functioning hardware obsolete and means it has to be replaced at an early stage.

When this Blue Angel ecolabel was published as the first ecolabel for software in 2020, these interrelationships were still something that was primarily discussed in specialist circles. Software tended to be viewed as an incidental feature of the hardware by the general public. However, there is now also greater awareness among the public for the role it plays.

There are now numerous initiatives in the digital economy and other research activities that aim to raise awareness for the environmental impact of software (e.g. German Environment Agency, Environmental Campus Birkenfeld, KDE Eco, Green Software Foundation, SDIA, Green Coding Berlin, Ecodigit, HPI clean-IT-initiative, carbontracker).

According to a recent study by Fraunhofer IZM (Stobbe et al. 2024), greenhouse gas emissions attributable to digital technology have risen continuously in Germany since 2020 – from around 20 million tonnes of CO_2 equivalents in 2020 (3% of national emissions) to about 24 million tonnes by 2023. These emissions are expected to increase further to 30 million tonnes by 2030. Improvements in the efficiency of hardware have obviously not made software more efficient but instead have been accompanied by a continuous rise in energy and hardware demand.

Therefore, this Blue Angel ecolabel focus directly on the software and formulates requirements for greater transparency and frugality. In line with the traditions of the Blue Angel ecolabel, software is viewed as a product for which criteria promoting environmental excellence can be defined. In comparison to the ecolabel from 2020, the scope of validity has now been expanded from purely desktop computer apps to include programs for mobile end devices and server client software.

---

[1] The applicant and licence holder in this document are a legal person.

1.3 Objectives of the Environmental Label

The design of the software and its programming have an influence on the hardware specifications and also energy consumption due to the use of the software.

The Blue Angel environmental label for "Resource and Energy-Efficient Software Products" (DE-UZ 215) may be awarded to products that use hardware resources in a particularly efficient manner and consume a low amount of energy during their use. Due to their lower performance requirements, longer operating lives for the hardware are possible. In addition, these software products stand out due to their high level of transparency and give users greater freedom in their use of the software. The Blue Angel will be awarded to the best products that are currently available on the market and the label will not make any comparative assessment between the products.

The aim of the environmental label is to reduce the total energy consumed by information and communication technology and improve resource efficiency. In particular, the Blue Angel will highlight those products whose manufacturers provide information about their products, especially with respect to their energy and resource efficiency, to recognise this transparency. In addition, the environmental label will also highlight those products whose manufacturers are actively engaged in improving the resource and energy efficiency of their products.

Therefore, the following benefits for the environment and health are stated in the explanatory box:

1.4 Definitions

Backward compatibility system: A hardware system whose most important capacities (e.g. working memory, processor performance) were considered customary on the market in a particular calendar year. This hardware system can be used to verify that a software product can still be run on an older hardware system or operating system.

Application software: Software products that provide a "benefit" for end users. They usually have a user interface (keyboard, mouse, screen output, etc.). Examples of application software are: image processing applications, e-mail programs, web browsers, multimedia applications, text processing or spreadsheet applications.

End of sale: The time at which the software product is no longer provided for sale by the applicant. It is still possible that support is provided for the product.

Desktop computer: A computer whose main unit is set up in a fixed location, is not designed for mobile use and is operated with an external display as well as external peripheral devices such as a keyboard and mouse.

Energy efficiency: Generally, the amount of "useful work" divided by the amount of energy it requires. In the context of this document, "useful work" is understood to mean the successful execution of a representative load (e.g. in the form of standard usage scenarios).

Hardware: All of the technical equipment required to run programs and to store or transport data.

Hardware capacity: Quantifiable characteristic of a hardware system which represents its performance limit on a given dimension of performance (e.g. working memory capacity, computing power, bandwidth).

Hardware system: Delimitable unit of hardware that performs defined functions.

Indicator: An empirically determinable quantity that provides insight into a matter that cannot be measured directly.

Scenario test: Measurement of the energy consumption, utilisation of hardware capacities and network activities when running a standard user scenario repeatedly.

Long-term test: Measurement of the energy consumption, utilisation of hardware capacities and network activities in live operation of a server or end device under a representative working load over a longer period of time.

Idle mode: A state in which the software product has been started but no user interactions are being carried out. In the case of software products for mobile end devices, idle mode describes a state after starting the program so that it is visible on the screen (app in foreground) but without any further user interactions.

Test system: A hardware system used for installing the software product and completing the measurements. The test system comprises not only the hardware but also any other software required for operating the software product (e.g. operating system, runtime environment, etc.) and their configuration. Another name for the test system is "system under test".

Measurement setup: All of the hardware and software components including the measurement equipment that is used to complete the measurements.

Native: A software product is described as "native" when it can be installed as a stand-alone program. A native app can share lower software layers (e.g. runtime environment, interpreter, operating system). In contrast, a software products is described as "non-native" when it requires a higher level application software. For example, a software product is non-native if it is run on a web browser (e.g. Javascript) that uses a database application software or an office software (e.g. VBA) or has been programmed as a macro, module or plug-in for a special application software.

Resource: A natural resource, in particular a raw material, a form of energy, or also the capacity of an environmental medium to absorb emissions. To differentiate natural resources from technical ones, especially hardware resources, the more precise term "hardware capacities" is used here for the latter. Since the use of hardware capacities always results in the use of natural resources, this distinction (which ultimately requires a difficult to define differentiation between the ecosphere and the technosphere) is not of decisive importance here.

Resource efficiency: Generally, the amount of "useful work" divided by the amount of resources it requires. In the context of this document, "useful work" is understood to mean the successful execution of standard usage scenarios.

Software: Programs and associated data in digital form.

Software product: A licensed copy of software that is designed for carrying out the required services, which are implemented and processed on a hardware system.

Software stack: Software components that build on one another to form the software product or which are required for its execution.

Standard usage scenario: A standard usage scenario is a repeatable sequence of interactions with a software product that is as representative as possible for the typical use of the product. The standard usage scenario is different for every software product and utilises the user and data interfaces offered by the software (e.g. keyboard inputs, pointing devices, data flows, Internet requests). The software product can also be used successively or in parallel in different user roles (e.g. user and administrator). The standard usage scenario is created by the applicant themselves and individually defined for each software product. It is used to test the software product under repeatable load states within a scenario test and to measure its consumption of resources. When testing the software product again at a later point in time, e.g. following an update, the same usage scenario should be used or the usage scenario should be amended to reflect any change in the functionality of the software product.

End of support: Time at which the applicant neither makes the software product itself available for sale (end of sale) nor provides support for the product.

Spreadsheet file for recording the criteria: This file is a spreadsheet in which information about the software product, the test system used for the measurements, the standard usage scenario and the results of the assessment of the criteria for the software product are recorded.

Update: An updated and, where relevant, improved version of a software product. The term "security update" is also used in these Basic Award Criteria. This describes an update that is designed to close security loopholes in the software product and make it more secure from an IT perspective in comparison to previous versions.

Sales platform: An online portal or another sales channel used by the supplier to offer the software product and any updates to customers.

1.5 Information on submitting an application and compliance verification

The requirements for energy and resource efficient software products formulated in these Basic Award Criteria are valid at both the time of application and also during the use of the environmental label. Compliance with the requirements must be verified at the time of application before the environmental label is awarded (Paragraph 3.1). If new versions of the software are developed or updates are issued during the use of the environmental label, the requirements must also be complied with during the term of the contract (Paragraph 3.2).

When submitting an application, all of the compliance verifications stated in Paragraph 3.1 must be provided. The central verification document at the time of application is the spreadsheet file for recording the criteria, in which information about the software product, the test system, the measurement method and the results of the measurements are recorded (Annex 2).

The documents submitted with the application must be examined and confirmed by an auditor. RAL gGmbH, with the support of the Federal Environmental Agency, will approve auditors. The list of approved auditors is published on the Blue Angel website.

1.6 Information on the role of the auditor

The use of auditors guarantees that the application is tested independently on the basis of a high level of professional qualifications. The applicant can freely select an auditor and also freely negotiate the contractual conditions for the audit. However, any remuneration must not be dependent on the results of the audit. The auditor must be independent of the applicant, both economically and institutionally, to ensure that the audit is objective.

The auditor can request to see other documents alongside the application documents, such as log files, and request a demonstration of the measurement tests and the tests for verifying backward compatibility. Furthermore, the auditor can request improvements to the measurements or a repeat of the measurement tests.

By signing Annex 1, the auditor confirms the completeness and plausibility of the application documents and the submitted verification documents, the appropriateness of the measurement setup and the usage scenario, compliance with the fault tolerances in the measurement results and the plausibility of all other information provided in the application. If the auditor has any doubts about compliance with all of the requirements, he or she can refuse to sign Annex 1. In case of disputes, the applicant can request that the German Environment Agency acts as arbitrator.

2 Scope

These Basic Award Criteria apply to software products that can be installed and run on fixed or mobile end devices as well as on servers.

Only those software products where at least 90% of their total ICT-related electric power consumption in operation is accounted for by one of the following computer platforms or several of these platforms can be certified with the ecolabel.

- Personal computers that are primarily designed for stationary use, such as desktop PCs, workstations, laptops or convertibles with a desktop operating system (Windows, MacOS, Linux or comparable), hereinafter referred to as a "**PC device**".

- Battery-powered, portable end devices such as smartphones, tablets and smartwatches with a mobile operating system (Android, iOS or comparable), hereinafter referred to as a "**mobile device**".
- Server hardware for the central processing of data, usually operated in a data center or server room, hereinafter referred to as a "**server device**". The server has a network connection and can share data with multiple clients simultaneously. A server does not usually have input and output devices (Human Interface Devices). Instead, users communicate with the server via the network interface.
-

The scope of this ecolabel does not cover software products where:

- more than 10% of the total ICT-related electric power consumption during the usage phase of the software is **outside** of the above-mentioned **computer platforms** (e.g. software that uses computationally intensive cloud and data services, software used in routers, network components, Internet of Things devices, printers, household devices, consumer electronics),
- the energy consumption during operation cannot be measured due to a **lack of available data** (e.g. software that is run in cloud data centers to which the applicant has no access for measurement purposes, peer-to-peer architectures, edge computing, the use of computationally intensive external cloud and data services),
- the ICT-related electric power consumption to **develop the software** (e.g. generate databases, machine learning, artificial intelligence applications) is higher than the energy consumption in the usage phase of the software, calculated based on the assumed use of the software product over a period of one year and a representative number of software products used.
-

If the applicant is able to demonstrate using suitable calculation or assessment methods that the grounds for exclusion described above do not apply, even if the application is for a software product named in the examples for excluded products, the software product also lies within the scope of this ecolabel.

In addition, software products that comes under the scope of these Basic Award Criteria must comply with the requirements of the currently valid version of the Interstate Treaty on the Protection of Minors (JMStV)[2] and must not display any of the characteristics stated in "§ 4 Illegal Content" of the JMStV.

---

[2] Interstate Treaty on the Protection of Human Dignity and the Protection of Minors in Broadcasting and in Telemedia (Jugendmedienschutz-Staatsvertrag – JMStV) in the version of the Second Interstate Treaty amending the Interstate

Media Treaty (Zweiter Medienänderungsstaatsvertrag) of 14 December 2021, https://www.kjm-online.de/fileadmin/user_upload/Rechtsgrundlagen/Gesetze_Staatsvertraege/JMStV/JMStV_english_version.pdf, last checked on 27/06/2025

3 Requirements

Software products to be certified with the ecolabel must comply with all of the requirements in these Basic Award Criteria.

Some of the requirements differ based on which computer platform is used to run the software product. Applicants must only provide verifications for these requirements for the stated computer platforms.

In order to determine which requirements must be complied with by the software product, the applicant must first define which computer platforms are utilised individually or jointly by the software. The following table assigns typical software categories to the possible computer platforms.

Table 1: Assignment of software categories to computer platforms

Software category	Mobile device	PC device	Server device
Desktop computer software		X	
Server client software with a native desktop computer app		X	X
App for mobile devices (purely offline)	X		
Server client software with a native app for mobile devices	X		X
Server client software with a web browser as a desktop computer app		X	X
Server client software with a web browser as a mobile device app	X		X
Server application			X

The terms "mobile device", "PC device" and "server device" are used for the computer platforms in Table 1 above. These terms are used in the Basic Award Criteria to indicate whether each criterion must be tested and complied with by the software product (only obligatory for: ...". If a software is assigned to several computer platforms, the applicant must provide verifications for **each of the platforms separately**. If the criteria do not specify a particular computer platform, they apply to all of the computer platforms used.

Exemption: If the software to be certified with the ecolabel cannot be clearly assigned to one of the software categories in the table above or it is assumed that a computer platform will only play a subordinate role in the total measurements, the applicant must examine for themselves which computer platforms are relevant and select these platforms. A computer platform is not relevant if the software product to be certified with the ecolabel cannot be run on it or the energy consumption on this platform does not account for more than 10% of the total energy consumption.

When calculating the total consumption, a proportion of the energy consumption of the server (server device) should be assigned to the software product in question based on a representative number of software products being used on the server.

Note: If a software product consists of several components that are run on different computer platforms (e.g. centrally on a server and locally on a desktop PC), the ecolabel can only be used on those components that were tested and complied with the criteria in these Basic Award Criteria. If a component of the software product is labelled with the ecolabel, even if the ICT-related electric power consumption on the respective computer platform is less than 10% of the total consumption, it is also possible to voluntarily include other platforms in the tests.

Compliance verification

The applicant shall state the name of the software product to be labelled with the ecolabel, the software category and the computer platforms used to verify compliance with the criteria in Annex 1. If the software product cannot be clearly allocated to one of the software categories stated in Table 1, the applicant shall provide the auditor with a calculation to show which computer platforms are relevant.

3.1 Requirements at the time of application

Verification of compliance with the requirements stated in Paragraph 3.1 must be submitted at the time of application. Verification must be provided for the current or explicitly stated version of the software product. The Contract on the Use of the Environmental Label will be issued on this basis. If this version of the software is updated during the term of the contract, verification of compliance with the requirements in Paragraph 3.2 must also be provided.

The requirements in Paragraph 3.1 are split into three sections:
* 3.1.1 Resource and energy efficiency
* 3.1.2 Potential hardware operating life
* 3.1.3 User autonomy

3.1.1 Resource and energy efficiency

A software product must deliver its functions with a minimal use of resources and a minimal energy demand. The resource and energy efficiency of the software product should be optimised. To operationalise the resource efficiency, the hardware resources and energy demands will be used as reference parameters.

3.1.1.1 Minimum system requirements

Only obligatory for: **PC device, server device**
The minimum system requirements for the operation of the software product must be stated.
* Minimum processor architecture incl. generation (e.g. Intel i5-3570k 3.4 GHz 4 Core (64-Bit), Intel Atom x7- Z8700, Intel Xeon E-2100/2200)
* Minimum local working memory required (MByte)
* Minimum local permanent storage required (MByte)

- Requirements for other software (operating system, middleware, auxiliary applications, container environment: software stack) (e.g. Windows 7, .NET framework and browser version XY)
- The external services required to operate the software that are not available on the test system (e.g. cloud services, storage services, API usage, …)
- The required additional hardware (e.g. graphics card, peripheral devices such as a camera connected via USB). The additional hardware must be directly integrated into the test system.
- For software for *server devices*: The typical number of users, instances, workloads or requests that can be simultaneously handled by the software with the stated system requirements.
-

Only obligatory for: **mobile device**

The minimum system requirements for the operation of the software product must be stated.
- Minimum processor architecture incl. generation (e.g. Kirin 970- HiSilicon, Octa-Core)
- Minimum local working memory required (MByte)
- Minimum local permanent storage required (MByte)
- Requirements for other software (operating system, user interface, …)
- The required external services that are not available on the test system (e.g. cloud services, storage services, API usage, …)
- The required permissions for accessing functions on the end device (e.g. microphone, camera, Bluetooth, contacts, locations services, etc.).
- Any additional external hardware required (e.g. sensors, actuators, smart home devices, ...). The additional hardware must be directly integrated into the test system.
-

Compliance verification

The applicant shall confirm compliance with the requirements in Annex 1 to the contract and state the minimum system requirements in the spreadsheet file for recording the criteria (Annex 2 to the contract). The information must be published in accordance with Paragraph 3.1.3.7 and kept up to date in accordance with Paragraph 3.2.1.

3.1.1.2 Documentation of the measurement setup

The software product must be installed on a test system and the measurement values required in these Basic Award Criteria must be determined in accordance with the quality requirements described in Appendix A for the measurement setup and the completion of the measurements. The applicant is permitted to select the relevant test system and a suitable measurement setup. If the product to be certified is installed on several computer platforms (client server system), the measurements and documentation must be carried out for each of these platforms.

The applicant must confirm that the measurement setup for the completion of the measurements in Paragraphs 3.1.1.3 and 3.1.1.4 complies with the quality requirements **according to Appendix A**.

The applicant must provide the following information in this context:
- Specifications of the test system
- Type of measurement (scenario test, long-term test)

- Name and brief description of the measurement method (designation, measurement setup, e.g. use of virtualisation)
- Description of the measurement equipment
- Name of the automation software, if used
- Confirmation of the tolerance requirements
- Description of the measurements (base load, idle load, scenario test, long-term test)
- Data on the measurements (duration of the scenario test, number of measurement cycles, duration of the long-term test, measurement interval)
- Description of the preparatory work (e.g. new installation of the operating system, time synchronisation, changes made to the standard configuration of the software stack)

Compliance verification

The applicant shall confirm compliance with the quality requirements according to Appendix A in Annex 1 to the contract and enter the required information listed above in the spreadsheet file for recording the criteria (Annex 2 to the contract). If requested by the auditor, the applicant must demonstrate the measurement setup. The information must be published in accordance with Paragraph 3.1.3.7 and kept up to date in accordance with Paragraph 3.2.1.

3.1.1.3 Measurement of the base load and the load of the software product in idle mode

*Only obligatory for: **PC device, server device***
If the measurements are carried out using a **scenario test** (see Paragraph 3.1.1.2), the hardware utilisation and the electrical power consumption for the base load and the idle mode of the software product must be stated. In addition, the applicant must first measure the average base load of the test system without the installed software product and then the additional load on the hardware when running the software in idle mode.

If the measurements are carried out using a **long-term test**, the measurements described here in Paragraph 3.1.1.3 are not required.

The following information must be documented for the base load and the idle mode of the software product (**PC device** or **server device**):
- Average processor utilisation (%)
- Average working memory utilisation (MByte)
- Average permanent storage utilisation (MByte/s)
- Average bandwidth utilisation for data transmission (Mbit/s)
- Average electrical power consumption (W) (calculated if necessary)
- List of all Internet addresses accessed (IP address or domain name), their location (country code according to ISO 3166-1), information on the owner (own or third-party service), frequency the Internet address is accessed (frequency or number per unit of time)
-

*Only obligatory for: **mobile device***
The applicant must measure the average bandwidth utilisation for data transmission for both the base load and the idle more of the software product. For this purpose, the applicant must first measure the average bandwidth utilisation of the test system (mobile device) without the

installed software product and then the additional average bandwidth utilisation when running the software in idle mode. The average bandwidth utilisation can be determined by, for example, measuring the amount of transmitted and received data (megabits) divided by the measurement period used (seconds). In addition, all of the Internet addresses accessed in idle mode must be documented.

The following information must be documented for the base load and the idle mode of the software product (mobile device):

- Average bandwidth utilisation for data transmission (Mbit/s)
- List of all Internet addresses accessed (IP address or domain name), their location (country code according to ISO 3166-1), information on the owner (own or third-party service), frequency the Internet address is accessed (frequency or number per unit of time)

-

Compliance verification

The applicant shall confirm compliance with the requirements in Annex 1 to the contract and state the measurement values in the spreadsheet file for recording the criteria (Annex 2 to the contract). In addition, the applicant shall submit the measurement report (see Appendix A) as Annex 3 to the contract. The auditor must confirm the plausibility of the measurements.
The measurement results must be published in accordance with Paragraph 3.1.3.7 and kept up to date in accordance with Paragraph 3.2.1.

3.1.1.4   Measurement of the software product during use

In order to carry out the following measurements of the representative use of the software product, the applicant must define which type of test will be used for each computer platform in Paragraph 3.1.1.2 (scenario test or long-term test).
In the case of a **scenario test**, a representative standard usage scenario must be formulated and then repeated multiple times (at least 10 times) on the test system. The results/average values of the individual measurement series must be documented and then averaged across all measurement series. In the case of a **long-term test**, a productive system (e.g. a server that can be simultaneously used by a number of users) must be measured during use over a longer period of time (at least 1 week) and the results must be documented by service unit (e.g. per user and week).
Different types of tests can also be used to carry out the measurements separately on different devices (e.g. a long-term test for the server and a scenario test for the client). The type of test used for each device must be documented in the spreadsheet file for recording the criteria (Annex 2).

*Only obligatory for: **PC device, server device***
The applicant must measure and document the hardware utilisation and energy demand during the representative use of the software product.
The basis for the calculations are measurement values and log files that are recorded with the aid of the measurement setup when running the software product. The hardware utilisation includes both the additional load caused by the use of the software product and also a percentage share of the base load during the completion of the scenario test. In the case of the long-term test, the applicant can choose whether to determine the base load separately and calculate the

proportion applicable to the software product or to determine the total load ("gross load") of the test system comprising the base load and the additional hardware utilisation. A separate measurement of the base load is always recommended when the productive system is also running other software applications in parallel that make a relevant contribution to the measured hardware utilisation.

The measurement units for hardware utilisation are units for work, such as %*s (processor utilisation), MByte*s (working memory utilisation), MByte/s*s = MByte (permanent storage utilisation, reading and writing) and MBit/s*s = MBit (volume of data transferred in the network).

Alongside the hardware utilisation, the electrical energy demand of the test system must also be determined. The required electrical energy is determined by integrating the electrical power consumption [W] over the time [s] taken to run the standard usages scenario or the duration of the long-term test (see Appendix B). Alternatively, a measurement device that directly records the electricity consumption [Ws] can also be used. Contrary to the specifications for the hardware utilisation, the measurements for the electrical energy only take into account the values that go beyond the base electrical load for the test system. In the case of a scenario test, the value for the electrical base load should be deducted from the gross value of the measurements to produce a net value.

The following information must be documented for the representative use of the software for each measurement cycle of the standard usage scenario (scenario test) or per service unit (long-term test, e.g. per user and week):

- Processor utilisation (%*s)
- Working memory utilisation (MByte*s)
- Permanent storage utilisation (reading and writing) (MByte/s*s)
- Volume of data transferred outside of the local network (Mbit/s*s)
- Energy consumption (Wh) (net for the scenario test, gross for the long-term test)
- List of all Internet addresses accessed (IP address or domain name), their location (country code according to ISO 3166-1), information on the owner (own or third-party service), frequency the Internet address is accessed (frequency or number per unit of time)

-

*Only obligatory for: **mobile device***

The applicant must measure and document the volume of data transferred during the representative use of the software product.

The volume of data transferred during the standard usage scenario (scenario test) or long-term use (long-term test) must be measured and recorded. The volume of data transferred by the test system in its base load can then be subtracted from these measurement values (see Paragraph 3.1.1.3). In addition, all of the Internet addresses accessed during the completion of the standard usage scenario (scenario test) or during long-term use (long-term test) must be documented.

The following information must be documented for the representative use of the software for each measurement cycle of the standard usage scenario (scenario test) or per service unit (long-term test, e.g. per week):

- Volume of data transferred during representative use (Mbit)
- List of all Internet addresses accessed (IP address or domain name), their location (country code according to ISO 3166-1), information on the owner (own or third-party service), frequency the Internet address is accessed (frequency or number per unit of time)

-

The applicant shall confirm compliance with the requirements in Annex 1 to the contract and state the measurement values in the spreadsheet file for recording the criteria (Annex 2 to the contract). In addition, the applicant shall submit the measurement report (see Appendix A) as Annex 3 to the contract. The auditor must confirm the plausibility of the measurements.
The measurement results must be published in accordance with Paragraph 3.1.3.7 and kept up to date in accordance with Paragraph 3.2.1.

3.1.1.5 Support for the energy management system

The software product must not require that an energy management system (e.g. standby mode, idle mode, clock frequency control) provided by the operating system, BIOS or hardware itself is deactivated or uninstalled for the software to run smoothly. The energy management system on the device must not be negatively influenced by the software product.

In addition, the functionality of the software product must not be negatively influenced by an existing energy management system (e.g. loss of data, impaired usability). There must be no restriction to the functionality of the software product when an energy management system on the underlying system layers or the connected client system has been activated. The need to reregister (without any loss of data) with a server-based software or the time required to reactivate the computer system from an energy saving mode are not considered impairments to the usability or functionality of the software.

Exemption: The software can deviate from the requirement described in the first paragraph if it temporarily carries out a function that can only be sensibly completed without interruption (e.g. video playback without deactivating the screen, saving or transferring data before activating standby mode). The energy saving function may be temporarily deactivated in these cases. The applicant must provide the auditor with plausible justification for utilising this exemption and demonstrate that the energy management system is independently activated again after completion of the relevant function.

Compliance verification

The applicant shall declare compliance with the requirement in Annex 1 to the contract. If the applicant utilises the exemption, he or she must inform the auditor and demonstrate that the energy management system is only deactivated temporarily.

3.1.2 Potential hardware operating life

The software must not contribute to the early replacement of existing hardware with more powerful hardware because the existing hardware no longer meets the performance requirements of the software. In addition, software updates must not result in the need for a hardware update. Instead, users must be able to make mutually independent decisions on the purchase of new software and hardware. Therefore, the requirements in this section address the potential hardware operating life.

3.1.2.1 Backward compatibility

It must be possible to run the software product on hardware for the respective computer platform (PC, smartphone, server) from a calendar year that is at least five years before the time of

application. If a standard usage scenario (scenario test) is used for the measurement of the software product during use (see Paragraph 3.1.1.4), this scenario test must also run on the stated hardware.

The supplier of the software product must make a binding declaration about which computer system from which calendar year can still be used to run the software and **publish** this information in its product information. The required software stack (e.g. operating system, framework, application software) and the configuration of the software product and software stack must be taken into account when determining the relevant year.

As an alternative to naming the computer system and its calendar year, the applicant can state that the software product can be run on an operating system that was released in a calendar year at least five years before the application. The supplier of the software product must make a binding declaration about which operating system (incl. the version number) from which calendar year can still be used to run the software and publish this information in its product information. Further information and example reference systems can be found in Appendix C.

Compliance verification

The applicant shall declare compliance with the requirements in Annex 1 to the contract and document the backward compatibility by naming the computer system including the required software stack and its calendar year or by stating the operating system and its calendar year in the spreadsheet file for recording the criteria (Annex 2 to the contract) under "Backward compatibility system". In addition, the applicant shall mark the corresponding sections of the product information and submit the relevant pages of the product information as Annex 6 to the contract or alternatively shall state in Annex 1 the Internet link (URL) to its own website where this information is published. If requested by the auditor, the applicant shall demonstrate that the software can run with the stated system requirements.

3.1.3   User autonomy

The software product should not restrict user autonomy with respect to the product and should not create any dependencies. The requirements described below contribute to different aspects of user autonomy.

3.1.3.1   Data formats

It should be possible for other software products to process the data generated by the software product and it must not create a dependency on specific software developers (vendor lock-in). For this purpose, the formats of the data processed and generated by the software or alternatively the exportable data must be so well documented (syntax and semantics) that it is possible for the data to be processed by other software products without any loss of important information.

The data formats must be documented using one of the following methods:
- Submitting the manuals or technical data sheets in which the data formats are semantically and syntactically documented or

- Providing examples of other software products (from other suppliers) that can process these data formats or
- Stating the data formats and assigning them to a general standard.
-

Compliance verification

The applicant shall declare compliance with the requirements in Annex 1 to the contract using one of the options described above and submit corresponding verification documentation as Annex 4 to the contract. Alternatively, the applicant shall state in Annex 1 the Internet link (URL) to its own website where the data formats are documented. The auditor must confirm that this documentation is sufficient and understandable.

3.1.3.2 Transparency of the software product

In order to enhance the software product with additional functions, the application programming interfaces (APIs) must be clearly documented. The APIs should correspond to open standards. In the interests of long-term and comprehensive use of the product, the source code should also ideally be fully or partially published, at the latest when the product is no longer being updated, and permission to change the product granted.

The following information must be provided:
- If APIs are used: Template for the interface documentation.
- Information on the licence under which the software product will be sold and the rights that the licence provides including information on whether third parties are permitted to develop the software product.
- Information on the licences for all software components, insofar as they differ from the licence under which the software product is sold.
- Either:
  - Information on the extent to which program components with documented source code have been published or whether the entire software product has been published as open source code. If the source code for the software product has been published, the applicant must provide information on how the source code can be accessed (e.g. via standard source code management platforms such as GitHub, GitLab, Bitbucket, Sourceforge, etc.).
  - Or: If the source code has not been published: An explanation of what will happen to the software product after the end of support, e.g. when the product or the certified version of the product can no longer be purchased and/or support is no longer available for the use of the product or the version of the product (e.g. publication and free licensing of the supplier's own source code, further support for users via third parties, escrow contract, etc.).
  -

Compliance verification

The applicant shall declare compliance with the requirements in Annex 1 to the contract and submit the corresponding verification documents (interface documentation, permalink to the software source code on a source code management platform, software licences or similar) as Annex 5 to the contract. The auditor must check the completeness of these documents.

3.1.3.3　Continuity of the software product

It must be possible to use the software product over a long period of time without any serious disadvantages arising (especially problems related to IT security). In addition, the software manufacturer must offer a function that can be used to keep the software product up to date. Security updates must be provided free of charge. A fee may be charged for updates that exclusively include new functionalities (and no security updates).

The requirements for the further development and update of the product in Paragraph 3.2.1 must be complied with in the provision of the updates.

*Only obligatory for: **PC device**, **server device***
The applicant undertakes to supply security updates for the labelled product free of charge for at least 5 years after the end of sale.

*Only obligatory for: **mobile device***
The applicant undertakes to ensure that the product remains secure for at least 3 years after the award of the ecolabel by providing security updates free of charge as necessary and offering the updated version of the product on the relevant sales platform.

Compliance verification

The applicant shall declare compliance with the requirements in Annex 1 to the contract, mark the corresponding sections of the product information and submit the relevant pages of the product information as Annex 6 to the contract or alternatively shall state in Annex 1 the Internet link (URL) to its own website where information on the security updates is published. The auditor must confirm that this information is sufficient and understandable.

3.1.3.4　Uninstallability

It must be possible to completely remove the software product from the computer platform after the end of its operating life without leaving any unnecessary traces of data. Data that has been created and processed by the users with the software product is excluded from this rule and must not be automatically deleted when uninstalling the software product. It must be possible for the user (e.g. a system administrator) to easily uninstall the software product, including any additional components or libraries that may be installed by the software, within a short period of time. The product information must include a description of the process for uninstalling the product.

Compliance verification

The applicant shall declare compliance with the requirements in Annex 1 to the contract, mark the corresponding sections of the product information and submit the relevant pages of the product information as Annex 6 to the contract or alternatively shall state in Annex 1 the Internet link (URL) to its own website where a description of the process for uninstalling the product is published. The auditor must confirm that this description is understandable.

3.1.3.5   Modularity

Software products often offer a variety of different functions that are only rarely used or only used by a few users. As a result, the installation of the software or the activation of parts of the software product can require an excessive amount of storage space, computing power and energy. In order to avoid this issue, the software product should have a modular design, i.e. basic functionality that can be enhanced where necessary with the installation of additional modules.

*Only obligatory for: **PC device, server device***

Users of the software product should be given the option of limiting the functions to the desired range of functions.

The product information must include the following information in this context:

- Information on whether users are given the option during the installation process of not installing certain functionalities that go above and beyond the basic functionality of the software product. If applicable, this procedure must be described.
- Information on whether users can deactivate individual modules or functionalities during the use of the software product. If applicable, this procedure must be described.
-

Compliance verification

The applicant shall declare compliance with the requirements in Annex 1 to the contract, mark the corresponding sections of the product information and submit the relevant pages of the product information as Annex 6 to the contract. Alternatively, the applicant shall state in Annex 1 the Internet link (URL) to its own website where the options for limiting the functional scope of the software during installation or operation are described. The auditor must confirm that this description is understandable.

3.1.3.6   Freedom from advertising and avoidance of tracking

Displaying advertising can increase the resource and energy demands of the software product. In particular, displaying advertising that is provided by e.g. an external service provider will result in the transmission of additional volumes of data from the client to the server and vice versa and will also lead to higher energy consumption locally. The same applies to tracking functions, i.e. the continual monitoring of software users by transferring the status, location or other logging data. This additional network traffic is not important for the direct functionality of the software product and should be avoided.

*Only obligatory for: **PC device, mobile device***

- Software products that are awarded this environmental label must be free of advertising. Advertising for the manufacturer's own company, other versions of the software or other software from the same manufacturer are exempt from this rule.
- In addition, the software products must not contain any tracking functions in their default state. An exception applies to tracking functions that are intentionally permitted by users of the software (e.g. access to the location, transmission of crash notifications). It must also be possible to deactivate these tracking functions again after installation. In addition, tracking functions that are required by law are permitted (e.g. those used in certified medical products).
- Permissible tracking functions must be described in the documentation for the software product (see Paragraph 3.1.3.7).

-

Compliance verification

The applicant shall declare compliance with the requirement in Annex 1 to the contract. The auditor shall confirm that the software is free of advertising and tracking functions or, if the exemption is utilised, confirm that they are approved and documented.

3.1.3.7 Documentation of the software product

The software product must be provided with documentation that enables long-term and re-source-conserving use of the product. The applicant must provide corresponding information about the software product both publicly (e.g. on its own website) and also in combination with the distribution of the software product itself in the form of files (text documents, graphics, generally readable data formats, spreadsheet file for recording the criteria).

The following user information must be included in the documentation:
a) Minimum system requirements (see Paragraph 3.1.1.1)
b) Documentation of the test system (see Paragraph 3.1.1.2)
c) Measurement results in idle mode (see Paragraph 3.1.1.3)
d) Measurement results during use (see Paragraph 3.1.1.4)
e) Calendar year and data on backward compatibility (see Paragraph 3.1.2.1)
f) Documentation of the data formats (see Paragraph 3.1.3.1)
g) Interface documentation, i.e. software source, software licences (see Paragraph 3.1.3.2)
h) Information on software updates (see Paragraph 3.1.3.3)
i) Description of the process for uninstalling the software (see Paragraph 3.1.3.4)
j) Information on modularity and reducing the use of resources (see Paragraph 3.1.3.5)
k) Description of the permissible tracking functions and, if relevant, information on how to deactivate them (see Paragraph 3.1.3.6)

The applicant may freely select a suitable documentation format for publishing and distributing the information described above. However, the applicant must also document the required information for points a) to e) in a single PDF file that will be made available on the Blue Angel website by RAL gGmbH. This PDF document can be created, for example, by exporting the information in the spreadsheet file for recording the criteria (Annex 2).

Compliance verification

The applicant shall declare compliance with the requirements in Annex 1 to the contract, mark the corresponding sections of the product information and submit the relevant pages of the product information as Annex 6 to the contract. Alternatively, the applicant shall state in Annex 1 the Internet link (URL) to its own website where this documentation is made available during the entire term of the contract on the use of the ecolabel. The applicant shall submit a PDF document as Annex 7 containing the information for points a) to e) to RAL gGmbH and agree to the publication of this document on the website of the Blue Angel ecolabel. The auditor must confirm that all of the information has been provided in full.

3.2 Requirements during the term of the contract

3.2.1 Requirements for the further development and update of the product

If the product is changed (e.g. due to an update) during the term of the contract on the use of the environmental label, the applicant must ensure that the software product still complies with all of the criteria in these Basic Award Criteria. In particular, the changes must not result in a need for the hardware to be replaced in order to continue using the software (software induced obsolescence). If the label holder does not comply with the requirements during the term of the contract, permission to use the label can be withdrawn by RAL gGmbH.

If changes are made to the product within this period, the label user must complete the following steps at the latest 12 months after concluding the contract on the use of the environmental label and then at regular intervals after a maximum of 12 months in each case to check compliance with this requirement,

- Carry out new measurements of the energy consumption (PC device and server device) and network activity (mobile device) (see Paragraphs 3.1.1.3 and 3.1.1.4).
- These measurements must be carried out on the same test system or a comparable test system to the one used at the time of the application (see Paragraph 3.1.1.2).
- The applicant must also confirm that the product is still backward compatible (see Paragraph 3.1.2.1) by verifying that the software product can still be run on a computer system or operating system that was placed on the market 5 calendar years before the current measurement.
- The documentation for the software product (see Paragraph 3.1.3.7) must be updated to include the new measurement values and product characteristics and then published.
- The energy consumption values from the new measurements must not increase by more than 10% compared to the values from the previous measurement, otherwise the reasons for the change must be published in the documentation for the software product (see Paragraph 3.1.3.7).
- If the standard usage scenario and/or measurement method have been changed, this must also be described and published in the documentation for the software product.
- The earlier information on the previously completed measurements of the software product must still be available so that any changes can be understood (version history).
-

Compliance verification

The applicant shall declare compliance with the requirements in Annex 1 to the contract.

4 Applicants and Parties Involved

Manufacturers or distributors of final products according to Paragraph 2 shall be eligible for application.

Parties involved in the award process are:
*   RAL gGmbH to award the Blue Angel Environmental Label,
*   the federal state being home to the applicant's production site,
*   Umweltbundesamt (German Environmental Agency) which after the signing of the contract receives all data and documents submitted in applications for the Blue Angel in order to be able to further develop the Basic Award Criteria.

5 Use of the Environmental Label

The use of the Environmental Label by the applicant is governed by a contract on the use of the Environmental Label concluded with RAL gGmbH.

Within the scope of such contract, the applicant undertakes to comply with the requirements under Paragraph 3 while using the Environmental Label.

Contracts on the Use of the Environmental Label are concluded to fix the terms for the certification of products under Paragraph 2. Such contracts shall run until December 31, 2027.
They shall be extended by periods of one year each, unless terminated in writing by March 31, 2027 or March 31 of the respective year of extension.
After the expiry of the contract, the Environmental Label may neither be used for labelling nor for advertising purposes. This regulation shall not affect products being still in the market.

The applicant (manufacturer) shall be entitled to apply to RAL gGmbH for an extension of the right to use the ecolabel on the product entitled to the label if it is to be marketed under another brand/trade name and/or other marketing organisations.

The Contract on the Use of the Environmental Label shall specify:
*   Applicant (manufacturer/distributor)
*   Brand/trade name, product description
*   Distributor (label user), i.e. the above-mentioned marketing organisations.

Appendix A   Quality requirements for the measurement setup and the completion of the measurements

The main quality requirement is the plausible, traceable and repeatable collection of the required measurement values using a suitable measurement set up and a representative usage scenario (scenario or long-term test). The components of the measurement setup and the method used must be documented in accordance with Paragraph 3.1.1.2.

To verify the measurement method used to determine the resource and energy efficiency (Paragraphs 3.1.1.3 and 3.1.1.4), the applicant must confirm compliance with the requirements for the measurement setup and test system (see Paragraph 3.1.1.2). The measurements at the time of application and those during the term of the contract (e.g. in the event of a change to the product, see Paragraph 3.2.1) must be carried out on the selected test system and documented.

At least those computer platforms that are components of the test system selected for the certification of the software product must be measured. The various platforms can be tested using either a scenario test or a long-term test.

Depending on the software product, the measurements include the collection of measurement values (energy consumption data and hardware utilisation, bandwidth utilisation and listing all the Internet addresses accessed according to Paragraphs 3.1.1.3 and 3.1.1.4) for client server systems (PC device, mobile device and server device), desktop software (PC device) and apps for mobile end devices (mobile device). The quality requirements and the procedure for the actual measurements can be found in the sections below.

1   Scenario test for PC and server hardware

In order to determine the energy consumption, hardware utilisation and data transmission of the software product as part of a scenario test for personal computers (PC device) and server hardware (server device), the following measurements must be carried out:
a) Base load (also see Appendix A, Paragraph 1.3): Measurement of the energy consumption and hardware utilisation of the test system prior to the installation of the software product over a period of at least 60 minutes.
b) Idle mode measurement (also see Appendix A, Paragraph 1.4): Measurement of the energy consumption and hardware utilisation of the test system after starting the software product but with no additional user interactions over a period of at least 60 minutes.
c) Measurement under load (also see Appendix A, Paragraph 1.5): Measurement of the energy consumption and hardware utilisation of the test system when carrying out the standard usage scenario **at least 10 times**, although 30 times is recommended.

These three measurement results can be used to calculate the net load based on the proportion of the energy consumption in each case. Document the base load, idle load and net load in the spreadsheet file for recording the criteria (Annex 2). The definitions of the required terms and formulas are listed in Appendix A, Section 1.2.

1.1   Requirements for the statistical significance of the measurements

In order to produce statistically valid results in the scenario test measurements, it is necessary to carry out the same measurements multiple time in succession. Ideally, this should be achieved using a separate software that initially records the commands and actions in the scenario and then automatically repeats them on the test system.

These repeat tests will show whether the various measurement results differ significantly from one another. The measurements can also provide useful information for software developers on how different procedures or architectures impact the consumption of energy and resources or the bandwidth utilisation. To minimise the risk of random outliers in the measurements, the measurements must be repeated at least 10 times, although we recommend that they are repeated 30 times. If using a scenario test, the measurement results for energy consumption in the individual measurement cycles must not exceed a relative standard deviation of 5%.

The standard deviation σ is defined as:

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

n is the number of measurements, x_i is the measurement result for an individual measurement cycle i and $\bar{x}$ is the average value for all measurements:

$$\bar{x} = \frac{1}{n}(x_1 + x_2 + \cdots + x_n) = \frac{1}{n}\sum_{i=1}^{n}x_i$$

The following applies for the relative standard deviation of the energy consumption measurements if using a scenario test:

$$\frac{\sigma}{\bar{x}} \leq 5\%$$

1.2   Notes on the measurement values

The following definitions are valid for measuring the indicators in Paragraphs 3.1.1.3 and 3.1.1.4. In principle, it is possible to measure different measurement values if plausible justification is provided to the auditor.
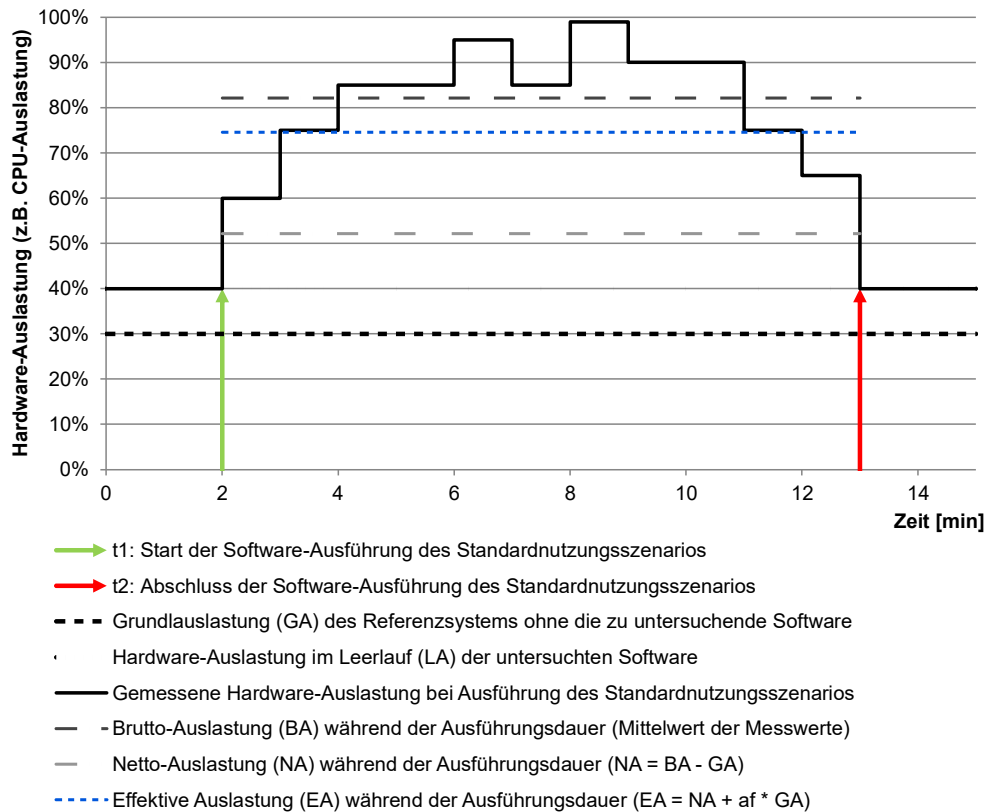
Table 2: Notes on the measurement values for measuring the software product

Identifier	Name	Definition	Comment
FL_i	Full load	Upper limit of the capacity i for the test system	For processing power, the full load is 100%, for working memory the sum of the installed RAM capacities, for network bandwidth the maximum transmission speed, etc.
BL_i	Base load	Average load of the capacity i for the test system when the software product is not installed.	This value represents a constant value for the test system.
IL_i	Idle load	Average load of the capacity i for the test system when the software product is installed but in idle mode	It includes both the base load (BL_i) and also the additional hardware load when the software is in idle mode.
NIL_i	Net idle load	$NIL_i = IL_i - BL_i$	It describes the idle load of the software without the proportion of the base load (BL_i).
t	Time	The time needed by the software product to execute the standard usage scenario on the test system	It begins when the standard usage scenario starts and ends when all required actions have been executed, including follow-up processes (such as releasing memory, deletion of temporary files).
GL_i	Gross load	Average load of the test system when running a standard usage scenario over the time taken to execute it	It is calculated as a time-weighted average of the measured value over t.
NL_i	Net load	$NL_i = GL_i - BL_i$	It describes the average hardware load of the software product after deducting the base load of the test system.
AF_i	Allocation factor	$AF_i = NL_i / (FL_i - BL_i)$ (allocation factor for the execution of the standard usage scenario)	Ratio between the net load of the software product and the maximum load available. It is important to note here that the maximum load available is the difference between the full load (FL_i) and the base load (BL_i). This factor is required for calculating the effective load (EL_i).
AFI_i	Allocation factor idle	$AFI_i = NIL_i / (FL_i - BL_i)$ (allocation factor for the software in idle mode)	Ratio of the net idle load (NIL_i) to the total available load ($FL_i - BL_i$). This factor is required for calculating the effective idle load (EIL_i).

Identifier	Name	Definition	Comment
EL_i	Effective load	$EL_i = NL_i + AF_i * BL_i$	The effective load is the sum of the net load (NL_i) and a proportion of the base load (BL_i), which is determined using the allocation factor (AF_i).
EIL_i	Effective idle load	$EIL_i = NIL_i + AFI_i * BL_i$	The effective idle load is the sum of the net idle load (NIL_i) and a proportion of the base load (BL_i), which is determined using the allocation factor idle (AFI_i).
HD_i	Hardware utilisation	$HD_i = EL_i * t$	The hardware utilisation is the product of the effective load (EL_i) and the time (t). Its units are, for example, percent per second (%*s), megabyte per second (MByte*s), megabytes (MByte/s*s) and megabits (MBit/s*s).

The following diagram shows an example measurement cycle for a computer platform and illustrates the different loads and their descriptions.

Figure 1: Example measurement cycle to determine the hardware utilisation

t1: Start der Software-Ausführung des Standardnutzungsszenarios

t2: Abschluss der Software-Ausführung des Standardnutzungsszenarios

- - - Grundlauslastung (GA) des Referenzsystems ohne die zu untersuchende Software

·　　Hardware-Auslastung im Leerlauf (LA) der untersuchten Software

——— Gemessene Hardware-Auslastung bei Ausführung des Standardnutzungsszenarios

— - Brutto-Auslastung (BA) während der Ausführungsdauer (Mittelwert der Messwerte)

— 　Netto-Auslastung (NA) während der Ausführungsdauer (NA = BA - GA)

- - - - Effektive Auslastung (EA) während der Ausführungsdauer (EA = NA + af * GA)

Source: Final report for the project "Entwicklung und Anwendung von Bewertungsgrundlagen für ressourceneffiziente Software unter Berücksichtigung bestehender Methodik" (Development and application of assessment criteria for resource efficient software taking into account existing methodology) (UFOPLAN-SSD-2015)[3]

1.3 Measurement of the base load

The hardware utilisation and electrical power consumption during operation of the system without the software product installed (base load) is required for the scenario tests in order to determine the indicators in Paragraphs 3.1.1.3 and 3.1.1.4. To measure the base load, the software (operating system, software stack) required to operate the software product is installed on the test system but not the actual software product being tested itself or any software components that are supplied with the software product. No other applications should be started except for the operating system, any other required software and, if necessary, any required automation software. The base load for the hardware utilisation, as well as the electrical consumption, bandwidth utilisation and the list of all Internet addresses accessed must then be determined over a period of at least 60 minutes.

---

[3] https://www.umweltbundesamt.de/sites/default/files/medien/1410/publikationen/2018-12-12_texte_105-2018_ressourceneffiziente-software_0.pdf

1.4 Measurement of the hardware utilisation in idle mode

The hardware utilisation when the software product is in idle mode must be stated. For this purpose, the software product and all of the other software required for the operation of the software product (e.g. runtime environment, databases, etc.) must be installed on a test system and the hardware load must then be determined in idle mode. The system used must be stated in the spreadsheet file for recording the criteria (Annex 2).

The hardware utilisation in idle mode includes the additional load caused by activating the software product, as a percentage share of the base load. The spreadsheet file for recording the criteria (Excel template, Annex 2) can be used for the calculations. The average electrical power consumption (in watts) is calculated by taking the average of all measurements of the power consumption in idle mode. The software product is in idle mode but there is no user interaction. In the case of a scenario test, the software product being measured is installed on a test system. The selected test system, the software stack being used and any configuration (of the software product and software stack) that deviates from the delivered state must be documented (see compliance verification for Paragraph 3.1.1.2).

The software product is then started and the measurement of the hardware load on the test system is carried out. The measurement must last for at least 60 minutes. It is recommended that a measurement value is recorded every second and that you add a waiting time of 1 minute to the process until the system is fully loaded.

Once all of the values have been recorded, the average (idle load) is determined using the formula:

$$\text{LA}_i = \frac{1}{n} \cdot \sum_{k=1}^{n} \bar{x}_k$$

$\bar{x}_k$ is the arithmetic mean for measurement cycle k in % (CPU utilisation), MByte (RAM utilisation and hard disk load) or MBit/s (network) and n is the number of measurement cycles.

The net idle load, allocation factor idle and effective idle load are then calculated. The supplied spreadsheet for recording the criteria can be used for this purpose.

1.5 Measurement of the software product during use

It is assumed that the software product being tested and all of the other software required for operating the software product (e.g. runtime environment, databases, etc.) have been installed on the test system. Starting from this state, the standard usage scenario (see Appendix B) is automatically run and the measurements of the hardware load, bandwidth utilisation and the Internet addresses accessed by the test system are carried out. For scenario tests, the measurements must be repeated at least 10 times, although 30 measurement cycles are recommended. It is recommended that a measurement value is recorded every second and that you add a waiting time of 1 minute to the process until the system is fully loaded.

Once all of the values have been recorded, the average (gross load) is determined using the formula:

$$BA_i = \frac{1}{n} \cdot \sum_{k=1}^{n} \bar{x}_k$$

$\overline{x}_k$ is the arithmetic mean for measurement cycle k in % (CPU utilisation), in MByte (RAM utilisation and hard disk load) or in MBit/s (network) and n is the number of measurement cycles. The net load ($NL_i = GL_i - BL_i$), allocation factor ($AF_i = NL_i / (FL_i - BL_i)$) and effective load ($EL_i = NL_i + AF_i * BL_i$) are then calculated and the hardware utilisation is calculated as an integral of the hardware load over the time taken for the scenario (or alternatively based on the average value: $HD_i = EL_i * t$). The measurement units for hardware utilisation are units for work, over the time taken [s] (i.e. %*s, MByte*s, MBit/s*s=MBit). The supplied spreadsheet for recording the criteria can be used to calculate them.

The verifications must be accompanied by information on the system used for the measurements, information on the standard usage scenario (e.g. which functions of the software were executed?) and information on the configuration of the software product and the software stack – if these differ from the delivered state of the software.

The *energy consumption* is recorded in the same way as the hardware utilisation.
Start measuring the energy consumption of the test system while the standard usage scenario is run automatically.
After the measurement has been completed, the average energy consumption is calculated as the integral of the power consumption. If using measurement equipment that measures the power consumption as an average value per second Pi, the following formula is used

$$E = \frac{1}{3600 \cdot n} \cdot \sum_{k=1}^{n} \sum_{i=1}^{m} P_i$$

whereby a measurement cycle lasts *m* seconds and *n* cycles are evaluated. The electrical work in Wh is derived by dividing this figure by 3600 s/h. If the measurement is not aggregated every second, it is necessary to derive the integral over the measurement period t and calculate the electrical work in Wh using suitable means.

2   Scenario test for battery-operated portable end devices

In order to determine the bandwidth utilisation and the URLs accessed by the software product in a scenario test for battery-operated portable end devices (mobile device), the following measurements must be carried out:

d) Base load: Measurement of the volume of data transferred by the test system prior to the installation of the software product over a period of at least 60 minutes with the display switched off.

e) Idle mode measurement: Measurement of the volume of data transferred by the test system after starting the software product but with no additional user interactions, over a period of at least 60 minutes with the display switched off.

f) Measurement under load: Measurement of the volume of data transferred by the test system when carrying out the standard usage scenario at least 10 times.

g)

The following must be documented for points a) to c):

•   Average bandwidth utilisation for data transmission (Mbit/s); for c) "Measurement under load": the individual measurement values for the volumes of data.

- List of all Internet addresses accessed (IP address or domain name), their location (country code according to ISO 3166-1), information on the owner (own or third-party service), frequency the Internet address is accessed (frequency or number per unit of time).

-

As the recorded values are not dependent on the duration of the scenario, the scenario can be carried out manually on the mobile device and this process does not necessarily require an automation tool.

The measurements of the base load and hardware utilisation in idle mode and the measurement of the software product during use should be carried out using the same process described in Appendix A Section 1 for personal computers and server hardware.

3   Long-term test

As an alternative to the scenario test, a long-term test of the software product can also be carried out. In a long-term test, the software is tested **in real operation with a representative number of users** over a sufficiently long period of time. In particular, this type of test is recommended for server applications in live operation.

In order to determine the energy consumption and hardware utilisation of the software product on a server in a long-term test, the following measurements must be carried out:

Measurement under load: Measurement of the energy consumption, hardware utilisation, bandwidth utilisation and the Internet addresses accessed by the test system during the continuous use of the software product by a representative number of users over a period of at least 1 week, although 1 month is recommended. The length of the measurement period will depend on the software product being certified and a plausible explanation must be provided to the auditor. Any anomalies during the measurement period must be documented (e.g. system crash, switching components, unusually high or low use, etc.).

Recording the hardware utilisation: It is recommended that a measurement value is recorded every second. Once all of the values for the hardware utilisation have been recorded, the gross load is calculated as the arithmetic mean of the measurement values in % (CPU utilisation), in MByte (RAM utilisation and hard disk load) or MBit/s (network). The hardware utilisation during use (Paragraph 3.1.1.4) should be stated per service unit (e.g. per user and week). The average utilisation value is first divided by the number of users and then multiplied by the reference period (e.g. 604,800 seconds in a week). The measurement units for hardware utilisation are units for work, over the reference period [s] (i.e. %*s, MByte*s, MBit/s*s=MBit).

Recording the energy consumption: The energy consumption is recorded in the same way as the hardware utilisation. After the measurement period has ended, the average energy consumption is calculated as the integral of the power consumption. If using measurement equipment that measures the power consumption as an average value per second Pi, the following formula is used

$$E = \frac{1}{3600} \cdot \sum_{i=1}^{m} P_i$$

whereby the measurement period is m seconds. The electrical work in Wh is derived by dividing this figure by 3600 s/h. If the measurement is not aggregated every second, it is necessary to derive the integral over the measurement period t and calculate the electrical work in Wh using suitable means.

The results for the hardware utilisation and energy consumption are then broken down for a period of one hour by dividing the integrated results by the number of hours in the measurement period. In addition, the number of users during the long-term test should also be measured and an energy consumption value per user and week (or another sensible time period) should be given.

The verifications must be accompanied by information on the system used for the measurements and information on the configuration of the software product and the software stack – if these differ from the delivered state of the software.

Appendix B Recommendations and information on completing the measurements

1 Recommendations for completing the measurements

If a software product consists of several components that are run on different computer platforms, measurements must be carried out on all platforms. This process can be carried out simultaneously or successively, as long as the same scenario is used.

1.1 Preparatory work

Before starting the measurements of the software, the following preparatory measures are recommended:

- Time synchronisation: To ensure that the measurement results can be accurately combined, the system times on the components used for the measurements, such as the power measurement device, test system, load generator, network traffic measurement device and, if relevant, other hardware, must be synchronised with one another.
- Deactivate the energy management system on the operating system so that the measurements can be carried out without any interruptions.
- To measure the base load, the operating system and the required software stack must be installed on the system. Software components that are supplied together with the software product (e.g. Java runtime, database or similar that are supplied with the software product) are excluded from this process. If required, an automation software (see below) is also installed.
- For measuring the idle load in a scenario or long-term test, it is also necessary to install the software product to be certified with the Blue Angel and its software components.

1.2 Completing the measurements

After completing the preparatory work, start the energy and performance measurements:

- To measure the base load, the operating system and the required software stack must be installed on the system. Software components that are supplied together with the software product (e.g. Java runtime, database or similar that are supplied with the software product) are excluded from this process. If required, an automation software (see below) is also installed.
- Except for the operating system and, if relevant, any required automation software (see above), no other software should be started and unused sensors (e.g. Bluetooth) on the battery-operated portable end devices (mobile device) must be switched off.
- Start the equipment used to measure the performance, electrical consumption, bandwidth utilisation and list of Internet addresses accessed.
- For scenario tests, the standard usage scenario on the automation software or the automation script is started and run automatically. The hardware utilisation, energy consumption, bandwidth utilisation and all of the accessed Internet addresses are measured during the test. For long-term tests, the hardware utilisation, energy consumption, bandwidth utilisation and all of the Internet addresses accessed are also measured during the test, whereby the start and end points are also documented. On battery-operated portable end devices (mobile device), it is necessary to measure the bandwidth utilisation and in parallel list the

Internet address accessed, whereby the standard usage scenario can also be carried out manually.

- No interventions in the test system are permitted until the measurement has been completed so as not to falsify the measurement results.
- Stop the monitoring of the performance data after completing the standard usage scenario or after the end of the measurement period.
- Reading the data: The measurement has now been completed. At this point in time, it is recommended that the raw data from the measurements of each computer platform (PC device, mobile device and server device) is saved together in one storage location.
- The next step is to individually evaluate the generated log files for each computer platform.
- Information on measuring the hardware utilisation and energy consumption when running a standard usage scenario

2 Information on measuring the hardware utilisation and energy consumption when running a standard usage scenario

2.1 Recommendations for measurement devices and software-based tools

To measure the electrical power consumption, a measurement device (e.g. Janitza UMG 604, Gude Expert Power Control 1202) can be used to record the electrical power consumption during the measurements. It is recommended that the applicant selects a measurement device that aggregates the measured power consumption over one second and saves it together with a time stamp in a log file. For this purpose, (another) software that can record performance data and the current system status can be used (e.g. Windows Performance Monitor[4], Collectl[5]). It is also recommended here that the applicant creates a log file using the software that contains the measured data and creates a time stamp. In addition, it is recommended that the measurements are taken at least every second.

Software-based measurement methods can also be used to determine the electrical power consumption (e.g. by accessing the internal chips RAPL or IPMI).

2.2 Recommendations for the standard usage scenario for the scenario test and long-term test

To automatically record manual user inputs on the computer (PC device), macro recorders (e.g. Pulover's Macro Creator[6], Power Automate[7], Actiona[8]) can be used that are able to record a sequence of actions that a user would typically carry out using mouse clicks and keyboard entries on the software to be certified with the Blue Angel. These actions can be recorded either by directly starting a recording mode or by manually entering and editing them. Whether only the option of using the described recording mode is available or whether both options can be used is dependent on the relevant software.

Alternatively to this type of recording software, the automated standard usage scenario can also be created using scripts or programs developed in-house. It is important that the same software

---

[4] https://techcommunity.microsoft.com/blog/askperf/windows-performance-monitor-overview/375481
[5] https://collectl.sourceforge.net/
[6] https://www.macrocreator.com/
[7] https://www.microsoft.com/en-us/power-platform/products/power-automate/
[8] https://wiki.actiona.tools/doku.php?id=en:start

is used for every measurement (base load, idle mode and standard usage scenario) so that any measurement errors caused by the automated software are minimised.

The result of the recording process is e.g. an executable file or a script that collects together the user inputs so that the standard usage scenario can be run automatically. It is recommended that a log file is also created by the automation software, in which a time stamp for the start and end of the individual measurements is logged.

The main basis for measuring some indicators is the completion of a standard usage scenario, which must firstly be defined for every software product group being tested. It simulates the realistic use of the software during the scenario test (also see Appendix A, Section 1). The standard usage scenario includes the completion of tasks for which the software was developed and, if relevant, the interaction with the user of the software.

2.3   Recommendations for developing a standard usage scenario

Develop a standard usage scenario that reflects typical functions of the software product. The frequency at which these functions are used should be taken into account.
Describe the standard usage scenario in the spreadsheet for recording the criteria (Annex 2).

The duration of the standard usage scenario can be freely selected. However, the scenario used for the scenario test must cover the functional scope of the product and this will be checked by the auditor.

2.4   Examples of standard usage scenarios: Word processing software for a desk-top computer

The standard usage scenario "word processing" in the scenario test is described in detail below. This is consistent with the description in the spreadsheet file for recording the criteria:

Software: Product name
Product group: XX
Version: Date
Designation: XX
Authors: Name(s)
Test system: Year
Load generator: Description

General specifications:

Wait for a period of 60 seconds before starting the measurement.

Requirements:

The files created by the automated process do not exist

Zoom the document to 125% in the program

Rulers are switched on

Control characters are switched on

Copy the text to be added to the clipboard

Font "XX" is installed

Process:

No.	Time	Description of action	Name of action
1	0:00	Open a link to a document on the desktop (double click)	start
2	0:02	Wait until the document has opened	
3	0:08	Change the entire text	ChangeText
3.1	0:08	Click in the text	
3.2	0:09	Select all: Ctrl+a	
3.3	0:11	Change font to "XX"	
3.4	0:17	Change the format to justified text	
3.5	0:19	Set the font size to 9	
3.6	0:26	Click in the text to cancel the selection	
4	0:28	Save as a new document	Save
5	0:35	Add a table of contents	Add contents
…	…	…	…

After end of each measurement:

Delete both of the generated files to ensure the consistency of the process.

2.5 Example of a standard usage scenario for server client software

The example standard usage scenario "Cloud software for self-hosting and cloud computing" for a scenario test comprising two PC devices and one server device can be found below. This is an open source cloud service on a self-hosted server. The software enables users to save and share files and also to communicate using a chat function. The collaborative processing of files is also included and the standard usage scenario thus covers the essential functions of the software product.

Process:

No.	Time	Description of action	Name of action
1	0:00	Start the automation	Start cloud software automation
2	0:00	Open the browser and access the cloud software locally	Open e.g. Firefox
3	0:15	Login with the user name and password	Log in
4	0:20	Upload files:	Start upload
4.1	0:26	Start text file (754 kB)	Start upload text
4.2	0:30	Stop text file (754 kB)	Stop upload text
4.3	0:35	Start PDF file (1.5 MB)	Start upload PDF
4.4	0:39	Stop PDF file (1.5 MB)	Stop upload PDF
4.5	0:44	Start JPG file (214 kB)	Start upload JPG
4.6	0:48	Stop JPG file (214 kB)	Stop upload JPG
4.7	0:53	Start PNG file (125 kB)	Start upload PNG
4.8	0:57	Stop PNG file (125 kB)	Stop upload PNG
4.9	1:02	Start ZIP file (3.8 GB)	Start upload 4GB
4.10	3:13	Stop ZIP file (3.8 GB)	Stop upload 4GB
5	3:17	End upload	Stop upload
6	3:19	Edit text document:	Edit text
7	3:22	Scroll down and enter a test sentence in a paragraph	Scroll text
8	3:35	Save using Ctrl + S	Save text
9	3:41	Search for "Sustainability" using Ctrl + F and also enter a text sentence there	Search text
10	3:49	Save the file again with Ctrl + S and close the text editor	Save Text2
11	3:51	Open cloud chat window	Start chat
11.1		Send "Hello"	
11.2		Receive "Hello" (Client 2)	
11.3	4:10	Upload and send JPG file (214kB) to Client 2	Start sending JPG
11.4	4:54	Stop sending JPG file	Stop sending JPG
11.5	5:06	Small text conversation between Client 1 and Client 2	Stop chat
12	5:12	Select and download all files on the cloud	Start downloading files
13	6:01	All files downloaded	Stop downloading files
14	6:04	Delete all files on the cloud	Start deleting files on cloud
15	6:14	All files on the cloud deleted	Stop deleting files on cloud
16	6:18	Log out user	Log out
17	6:21	Close browser	Close browser
18	6:23	Delete local files	Start deleting local files

No.	Time	Description of action	Name of action
19	7:21	Local files deleted	Stop deleting local files
20	7:29	End scenario	End scenario
21	8:15	Pause (45s)	Pause

This scenario is repeated 30 times in succession, with a break of 45 seconds between each cycle. Excluding these breaks, the scenario last 449s. The breaks between the cycles give the system time to end any background processes that may be running. The breaks also make it easier to differentiate between the individual cycles.

3   Tools to record the bandwidth utilisation and the URLs

To monitor and evaluate the network activities of the software product on mobile end devices (mobile device), it is recommended that external data loggers are used to record and evaluate the data transmitted and received via the network interface (usually WLAN). For example, the tools Wireshark[9] or Pi-Hole[10] can be used for this purpose in combination with the data recorded by the respective WiFi router. It is also recommended here that the applicant creates a log file using the software that contains the measured data and creates a time stamp. It should record all of the network traffic during the test.

---

[9]   https://www.wireshark.org/
[10]   https://pi-hole.net/

Appendix C Information on verifiying backward compatibility

To verify backward compatibility, the applicant must name a representative computer system that was placed onto the market in a previous calendar year on which the software product can still be run (see Paragraph 3.1.2.1). Alternatively, the applicant can name an operating system on which the software product still runs.

The following tables listing the technical configuration of different (example) desktop computers and different operating systems based on their calendar year have been produced to provide assistance with this step. The applicant must select a suitable computer system for the relevant calendar year required for the verification or for an earlier calendar year and if requested by the auditor demonstrate its compatibility with the software product. The applicant can also select a computer system or operating system that is not listed in the table if he or she can plausibly demonstrate the calendar year it was placed onto the market. The named computer or operating system for verifying the backward compatibility of the software must be documented in the spreadsheet file for recording the criteria as the "Backward compatibility system". The test system and computer system for verifying backward compatibility can also be identical, i.e. the measurements can also be carried out on a test system that also complies with the requirements in Paragraph 3.1.2.1.

Table 3: Examples of desktop computer systems and their technical parameters for determining the calendar year they were available on the market, for the years 2018 and 2019

Technical parameters	2018	2019
Manufacturer	Fujitsu	Fujitsu
Model	Esprimo P957[11]	Esprimo P958[12]
Processor	Intel i5-7500	Intel i5-8500
Cores	4	6
Clock speed	3.4 GHz	3.0 GHz
RAM	16 GB (DDR4, 2400 MHz)	16 GB (DDR4, 2666 MHz)
Permanent storage	SSD M2 SATA, 256 GB (6 Gbit/s)	SSD M2 SATA, 512 GB (6 Gbit/s)
Graphics card	Intel HD Graphics 630	Intel UHD Graphics 630
Network	GigaBit LAN	GigaBit LAN

---

[11] Data sheet: https://objects.icecat.biz/objects/mmo_35992891_1487253016_6892_6046.pdf
[12] Data sheet: https://sp.ts.fujitsu.com/dmsp/Publications/public/ds-ESPRIMO-P958.pdf

Table 4: Examples of computer systems for Mac OS and their technical parameters for determining the calendar year they were available on the market, for the years 2015 and 2019

Technical parameters	2015	2019
Manufacturer	Apple	Apple
Model	Mac mini "Core i5" 2.6 (Late 2014)[13]	Mac mini "Core i5" 3.0 (Late 2018)[14]
Processor	Intel i5-4278U	Intel i5-8500B
Cores	2	6
Clock speed	2.6 GHz	3.0 GHz
RAM	8 GB (LPDDR3 SDRAM, 1600 MHz)	8 GB (PC4-21300 DDR4, 2666 MHz)
Permanent storage	1 TB HDD	256 GB SSD
Graphics card	Intel Iris 5100	Intel UHD Graphics 630
Network	GigaBit LAN	GigaBit LAN

Table 5: Examples of operating systems for desktop/server/mobile systems, for the years 2019 – 2023

Year	Windows	Debian	Ubuntu	Apple macOS	Android	Apple iOS	Ubuntu Server	Windows Server
2019	Windows 10	10.0 Buster	19.04, 19.10	Catalina 10.15	10: Android Q	13	18:04	Windows Server 2019
2020			20.04, 20.10	Big Sur 11	11: Android R	14		
2021	Windows 11	11.0 Bullseye	21.04, 21.10	Monterey 12	12: Android S	15	20:04	Windows Server 2022
2022		12 Bookworm	22.04, 22.10	Ventura 13	12L, 13	16	22:04	
2023			23.04, 23.10		14		23:04	

---

[13] https://everymac.com/systems/apple/mac_mini/specs/mac-mini-core-i5-2.6-late-2014-specs.html
[14] https://everymac.com/systems/apple/mac_mini/specs/mac-mini-core-i5-3.0-late-2018-specs.html