Energy-Efficiency Indicators for e-Services

Jeroen Arnoldus*, Joris Gresnigt†, Kay Grosskop*, and Joost Visser*‡
*Software Improvement Group, Amsterdam, The Netherlands
{j.arnoldus,k.grosskop,j.visser}@sig.eu
†Logius, Ministry of the Interior and Kingdom Relations, Den Haag, The Netherlands
joris.gresnigt@logius.nl
‡Radboud University Nijmegen, Nijmegen, The Netherlands

Abstract—**Great strides have been made to increase the energy efficiency of hardware, data center facilities, and network infrastructure. These Green IT initiatives aim to reduce energy-loss in the supply chain from energy grid to computing devices. However, the demand for computation comes from software applications that perform business services. Therefore, to measure and improve efficiency for entire systems, energy-efficiency indicators are needed at the level of services.**

We have designed an initial set of indicators for energy-efficiency of e-services and we have tested them on two e-government services of the Dutch national government. We explain how these indicators serve as a starting point for energy-optimization initiatives, supported by appropriate contractual agreements between service owners and suppliers.

I. INTRODUCTION

IT systems consume energy, and increasingly so. In 2010, overall electrical energy consumption by IT was estimated between 1.1% and 1.5% [1]. In certain regions, such as the Amsterdam metropole region, electrical energy consumption by data centers alone was 10% in 2011 [2].

The rise of energy consumption by IT systems is caused by the rapid development of modern society into an information society, as witnessed by increasing usage of e-services through desktop internet browsers and mobile platforms. These client systems are typically connected to servers in datacenters, where a large part of the requested computation, communication, and storage takes place [3].

In face of rising energy consumption by IT, great strides have been made to increase the energy efficiency of hardware, data center facilities, and network infrastructure. These Green IT initiatives aim to reduce energy-loss in the chain of supply from energy grid to computing devices [4].

In spite of such advances, IT energy consumption keeps rising steeply, which indicates that rising demand is outpacing efficiency improvement. This need not be surprising, given that development of software applications has in the past decades focused on developer efficiency and functionality increase, not on minimization of resource consumption [5]. In other words, improvements on the level of hardware are cancelled out on the level of software.

Given the unsustainability of IT energy consumption growth, we argue that the scope of efficiency improvement needs to be enlarged from the energy-supply chain (grid, network, datacenter, hardware) to encompass also the end-user software services where the demand for computing resources originates.

As a step towards controlling overall energy consumption of IT, we have developed a small set of energy-efficiency indicators for e-services. The purpose of these indicators is to provide service owners, application programmers, system administrators, and management a shared insight into the energy consumption profile of the e-services for which they are responsible. This enables them to set feasible targets for reduction of energy consumption define a joint course of action.

Section II explains how we derived our energy-efficiency indicators by applying a goal-directed approach for building an energy-efficiency evaluation model. Section III provides additional detail on the definition, measurement, and interpretation of each indicator. In Section IV, we show the application of the indicators to report the energy consumption of two e-government services of the Dutch national government. In Section V, we discuss how these metrics can be used for optimization and reduction of operational energy costs. Related work is discussed in Section VI. Finally, Section VII concludes the paper with a summary of our contributions and an outlook to future work.

II. THREE INDICATORS

We have employed the Goal-Question-Metric (GQM) [6] methodology to design an initial evaluation model for assessing and comparing the energy-efficiency of e-services. GQM starts by defining a measurement *goal*, then proceeds to formulate the *questions* that must be asked to help reaching the goal, and ends up with designing the *metrics* that help answering these questions.

An overview of the goals, questions, and metrics of our model as well as their interrelations is provided in Figure 1.

Our measurement goal is to help stakeholders of a portfolio of e-services to prioritize optimization activities and set targets for these optimizations.

To support this goal, the following questions need to be answered:

Q1 Which service consumes most energy? Answering this questions allows us to focus on those systems where optimizations will potentially have greatest impact.

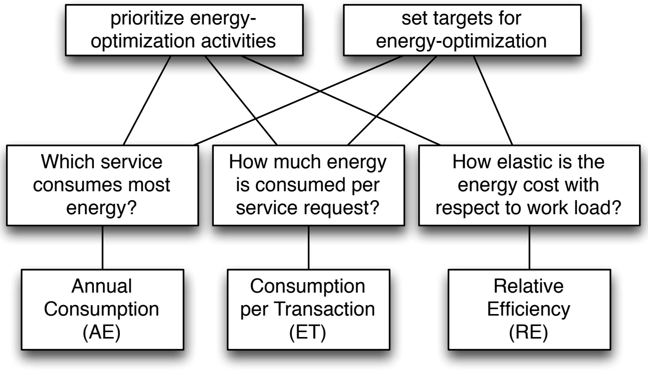

Fig. 1. The goals, questions, and metrics of the energy-efficiency model.

Q2 How much energy is consumed per service request? Answering this question puts the energy cost in perspective of the generated business value. This allows us to develop and intuition the proportionality of energy costs and judge whether energy cost is reasonable.

Q3 How elastic is the energy cost with respect to the variability of the work load? The answer to this question brings out the most imporant reason for current energy waste: systems that do not scale down energy consumption when work load is low, resulting in very low overall efficiency.

These questions bring us to the following three metrics:

AE Annual consumption: The total energy consumption of an e-service on an annual basis.

ET Consumption per transaction: The average energy consumption per executed end-user or business transaction.

RE Relative efficiency: The efficiency of the e-service with respect to its optimal efficiency (typically at maximal load).

Together, this initial set of three energy-efficiency indicators provides an energy-efficiency profile for an e-service. In the upcoming section we will discuss these three indicators in more detail.

III. INDICATORS IN DETAIL

For each of the three indicators, we provide a definition and some guidelines for measurement and interpretation.

A. Annual Consumption

The total energy consumption on a yearly basis is defined by the following equation:

$$AE = AE_{hardware} + AE_{communication} \tag{1}$$

Thus, total consumption of a service is composed of the consumption of hardware nodes (computing and storage devices) and the consumption of communication channels (network devices). These measurements are reported in kilo Watt hours (kWh).

Annual energy consumption does not require continuous measurement over the period of an entire year. Instead, it can be measured during a smaller period (for example a week) and then scaled to a year (multiplication by 52 in case of a 7 day measurement period). Thus, “annual” refers to the base to which the indicator is scaled, not to the period of which it is representative. As a consequence, annual consumption can fluctuate within a given year and one should always specify over which period the indicator is calculated.

When dedicated hardware is used to run a service, the consumption of the various nodes is summed to compute $AE_{hardware}$. In case of shared nodes, part of the consumption of the node must be attributed to each service. For compute nodes, the percentage of CPU usage by each service can be used to determine a fair attribution. In case of other hardware equipment other metrics should be used. In case of storage, tor example, allocation can be done on the basis of percentage of memory use or percentage of read and write operations.

To calculate the energy consumption due to data communication of a service, $AE_{communication}$, a different approach is needed. Typically, communication runs over channels that are not owned or controlled by the service owner which precludes direct measurement of consumption. Instead, an estimation can be made by combining information on the amount of data that is transferred and the type of channel that is used (e.g. optic fiber or wireless) with available benchmarks of consumption of Internet data traffic [7].

Of course, annual energy consumption cannot be used directly to compare energy-efficiency of services. However, this indicator allows to identify the largest consumers within a portfolio of services. Also, when several services of similar complexity or competing functionality are taken into consideration, comparisons on this indicator can be helpful to spot gross ineffeciencies. Finally, this indicator is essential for tracking energy-consumption over time and allows to quickly spot efficiency fluctations, for instance when consumption goes up after a new version without major functional changes is released.

B. Average Energy Consumption per Transaction

When we divide the energy consumption of an e-service by the number of end-user or business transactions that it handles, we obtain the average energy consumption per transaction, defined as follows:

$$ET = \frac{AE}{AT} \tag{2}$$

where AT is the annual number of transactions. Again, the number of transactions does not need to be measured over an entire year, but can be scaled to a yearly number from the measurement over a shorter period. The energy consumption per transaction is reported in Watt hours (Wh).

To determine the number of transactions, it is essential that first a decision is taken about which transactions to count. Here we are interested in transactions that are meaningful to the end-user and/or the service owner, such as processed orders, executed payments, updated dossiers, or permits issued.

Typically, a main transaction type can be identified for each e-service, from which it derives its reason of existence.

A service may perform other transactions as well (such as log-on, user profile change, or status overview), but these are ancilliary to its main transaction, and are not counted. Also, low-level technical transactions, such as database access, page rendering, or input validation, are not taken into account to determine the overall number of transactions.

The average consumption per transaction factors out the transaction volume and can therefore be used to compare and rank services independent of their volume. As such, the indicator is not intended to identify the largest overall consumers, but to identify services that have higher consumption than would be expected in view of the functional complexity or business value of its main transaction.

End-users and service owners typically have an intuition about the functional complexity and/or business value of these transactions. As a result, they can use this indicator to start forming an opinion on the proportionality of energy consumption. A highly complex transaction (e.g. calculating a three-day weather forecast) is expected to consume more energy per transaction than a simpler transaction, such as executing a bank transfer.

As in the case of annual consumption, the average consumption per transaction can not be used to make a direct efficiency comparion between services of just any type. Such comparion only makes sense when services with transactions of similar complexity or similar functionality are taken into consideration.

C. Relative Energy Efficiency

Relative energy efficiency of an e-service is computed as the ratio between its average consumption per transaction and its optimal consumption per transation. This is expressed in the following equation:

$$RE = \frac{ET_{optimal}}{ET} \quad (3)$$

Whereas average energy consumption is measured over an extended period (e.g. a week, a month, or a year), the optimal energy consumption is measured in a small time window when the system performs at its highest efficiency. This is typically a moment of peak load, which can be found by finding the lowest registerd TE throughout the observation period. Since very high load situations rarely occur in a production environment, this is better measured during load testing in a representative testing environment. By stepping up the load, the optimum can be found.

Since optimal consumption is always smaller or equal than average consumption, the relative efficiency can theoretically range between 100% (average consumption coincides with the optimum) and 0% (optimal consumption is negligable with respect to average consumption).

Most services have strong variations in work load. For instance, during office hours the load may be significant, while at night, in weekends, and during holidays the load may dwindle. Or throughout the year only occasional use is made of the system while load peaks towards the end of the year.

Ideally energy consumption of a service rises and falls proportionally to the workload. In reality, however, the computing infrastructure only reduces it energy consumption to a limited degree when workload falls, and still consumes energy even when the system is entirely idle [8]. Energy consumption in idle state is often not lower than about 60% of the energy consumption in fully busy state. As a consequence, most energy is wasted when systems do not run at full load.

To make energy consumption scale better with work load, several strategies can be followed. An effective strategy can be to run software systems on shared hardware. This is possible through virtualisation and consolidation. The ultimate form of such sharing is the use of cloud infrastructure, where services not only share servers with other services but also relinquish entire servers when not needed. Another important strategy is to make use of modern hardware with better scaling behaviour, for instance using CPUs with variable clock speed. Strongly software-related strategies include: avoidance of unnecessary events (e.g. polling, timers, synchronisations) that keep a system awake when idle, workload placement in particular time slots to maximize utilisation over short periods, and avoidance of keeping unnecessary data in memory.

Our indicator for relative energy efficiency quantifies the extent to which the scaling behaviour of a system in terms of energy behaviour matches the variability in its workload. For systems with very constant work load, limited scalability can already result in high values of relative efficiency. For systems that rarely have high workloads and are idle for most of the time, high values of relative efficiency can only be obtained with very flexible scaling behaviour.

The relative energy consumption is an indicator that allows direct comparion between services. Both the type and volume of the transations have been fully factored out.

IV. TWO CASE STUDIES

We have conducted two case studies in which the three indicators of the energy-efficiency evaluation model were calculated for two e-services of the Dutch government. In this section, we describe the context of the case studies and the specific e-services that were involved. Then we present and discuss the measurement results.

A. Context

Logius is an agency of the Dutch national government that is responsible for acquiring application management services and hosting services for e-services developed by or for various ministries and agencies. Included in this responsibility is the initial aquisition, vendor management, but also renewing contracts with vendors and transferring contracts to new vendors.

Logius is interested in instruments for stimulating vendors to improve energy-efficiency of the services they host and maintain. To develop such instruments, Logius asked the Software Improvement Group (SIG) and ManSystems to carry out two case studies. Within these case studies, SIG was responsible for measuring the footprint of the investigated services.

We used a structured process to obtain the energy profiles of Digipoort an DigiD Machtigen. First, we created an energy model for both applications. This model is based on the deployment architecture of the applications. To create this model we interviewed software architects and system administrators. The energy model describes which processes are running on which hardware. Second, we worked with the technical staff of both services to collect measurement data. Over the month of December 2012, the transaction logs, energy consumption data, and the data for attribution of consumption for shared hardware nodes were collected and used to calculate the indicators.

B. *Services under Study*

1) Digipoort: Digipoort is an "electronic post office" of the Dutch government. Companies connected to Digipoort can exchange digital information with the Dutch government. This is useful if a company has to provide data on a regular basis, like information for the customs from companies operating at the harbors or airports. The main transaction of this application is processing messages.

Digipoort is hosted at one primary location and a backup location. The system exists of multiple mail transfer agents and a core process routing the messages. These processes are deployed on a dedicated virtual machine cluster. A separated SAN is used for storage. The dedicated hardware is used for the development, test, acceptance, sandbox, and production environment. We collected the total energy consumption of the dedicated virtual machine cluster, the attribution to the production environment, the percentage of the used storage and the use logs.

2) DigiD Machtigen: DigiD Machtigen is an authorization service, where Dutch people can authorize others to represent them digitally. For example, a tax declaration can be submitted by an accountant on behalf of somebody else if the accountant is authorized by him. The main transaction of this system is the authorization.

DigiD Machtigen is hosted at one primary location and a sleeping instance at a backup location. The system exist of a database server, two application servers and two web servers. These servers are deployed on virtual servers hosted on a private cloud solution. This private cloud solution is also used for other applications. Storage is shared in a SAN solution. We collected the total energy consumption of the private cloud cluster, the attribution to the production environment, storage attribution and the use logs.

C. *Results*

Table I shows the energy profiles of Digipoort and DigiD Machtigen. These results are based on the energy consumption and transactions measurements in December 2012.

TABLE I
ENERGY PROFILE OF TWO DUTCH GOVERNMENTAL E-SERVICES, BASED ON DATA COLLECTED OVER DECEMBER 2012.

Application	AE	ET	RE
Digipoort	153,509 kWh	1.27 Wh	34%
DigiD Machtigen	12,800 kWh	38 Wh	11%

As the results indicate, the annual energy consumption of Digipoort in December was much larger (more than ten-fold) than the annual consumption of DigiD Machtigen in the same period. By contrast, the energy consumption per transaction of Digipoort was almost 30 times lower. Finally, the relative efficiency of Digipoort was substantially higher (34%) than the relative efficiency of DigiD Machtigen (11%).

Precaution should be taken to draw conclusions from these figures. The objective of the case studies was to gain experience with the indicators. The indicator values that we calculated should therefor be seen as illustrative examples of what can be done with the indicators, not as a definitive assessment of the energy-efficiency of these services.

Provisional observations that can be made on the basis of these results are the following:

1) Since the annual consumption of Digipoort seems significantly higher, an investigation into possibilities for energy optimization should probably be directed first at Digipoort.
2) In case of DigiD Machtigen the high use of energy per transaction combined with the low relative efficiency is an indicator that this service has extensive periods of low workload, but limited scaling behaviour.

These provisional observations were confirmed by closer inspection of both applications.

Investigation of optimization possibilities for Digipoort resulted in two important findings. First, the processor scaling feature that is active in a standard factory configuration has been explicitly turned off in the computer equipment. Second, it became clear that an acceptance test environment was up and running and consuming energy, though it was not being used. Both findings were addressed, i.e. the processor scaling feature was turned on, and the acceptance test environment was turned off.

Investigation of DigiD Machtigen revealed that this service was designed and configured for the peak volume of March and April, the period when Dutch citizens are submitting their annual tax statements. In less busy periods, such as December, the serivce does not scale down proportionally, which explains the high consumption per transaction and the low relative efficiency. To resolve this issue, scaling behaviour will need to be built into the system on both software and hardware level.

V. USING THE INDICATORS

In this section, we extrapolate from the case studies to reflect on some scenarios in which the indicators can be used.

A. *Portfolio Optimization*

Organisations that own or operate multiple e-services and are interested in reducing the associated energy footprint, can make use of our indicators for optimizing their services portfolio with respect to energy. Portfolio optimization requires that candidates for improvement are identified and prioritized, and an interative improvement process is conducted:

1) Determine energy indicators for all services.
2) Look for those services that have (a) high annual consumption, (b) higher energy-consumption per transaction than other services that are perceived to be of similar complexity, (c) low relative efficiency.
3) Perform an in-depth inspection of these services in order to identify energy-optimization actions. This implies mapping out the architecture of these services, creating and analyzing their energy models, and estimating savings for several change scenarios [9].
4) Implement the optimizations and re-calculate the energy indicators.

Following this iterative improvement process ensures that optimization efforts are focussed on services where the highest savings can be expected. The indicators help to guide this process with objective metrics rather than by relying on the intuition of system engineers only.

B. *Vendor Management*

An organisation that makes use of external parties to manage and host e-services can make use of our indicators for stimulating its vendors to improve energy-efficiency. In a tendering situation, various bidders on a request for proposals can be challenged to compete on improvements against the indicators. In the contract phase, the indicators can be used to set concrete targets for the selected vendor to meet, and during the execution of the agreement, the work can be evaluated against the targets that were agreed upon.

How to use the three indicators exactly and which specific targets to set is dependent on the situation. For example, for a new service that is taken into production for the first time, there may be uncertainty regarding the number of transactions that are expected to be processed. In this case, target values for the three indicators may be set conditionally on the actual number of transactions. For a service that has been operational for some time with, the expected workload and its distribution over time can be used to set sharper targets. Finally, a dynamic approach can be taken where the vendor commits to increasingly demanding targets throughout the contract period. This stimulates continuous optimization of the service.

C. *Accountability*

An organisation that wants to provide accountability towards its clients, employees, and other stakeholders regarding its use of energy resources can use our indicators to provide transparent reporting on energy usage. For example, the measured values of the indicators can be published in the annual report or in statements on corporate social responsibility to demonstrate that the organisation is making strides towards sustainable IT.

VI. RELATED WORK

An overview of approaches for optimizing energy-efficiency of application software can be found in previous work [10]. Here we limit ourselves to metrics and measurement tools.

FVER [11] is an approach to extend the well established PUE metric for data centers in a way that allows to account for end-to-end energy-efficiency assessment of services and to aggregate those to the data center level. The metric is very similar to our relative efficiency indicator RE. It attempts to account for the energy-elasticity of the whole system stack that supports the application, including the software. Similar to RE, it correlates workload and energy consumption statistics. FVER is calculated by comparing a fixed base consumption (without load) to the portion that fluctuates with changing load. Thus an important difference is that FVER compares the optimum and the worst case, while RE compares optimum and the average behaviour.

Several authors [12], [13] have investigated website energy consumption and have related useful work to an energy impact. These approaches tend to focus on the network and client-side energy dissipation and typically only consider the server side in a trivial way. Moreover these approaches are limited to a specific type of applications.

Intel EnergyChecker [14] is an attempt to systematically relate delivered functionality to units of work. It offers an API that can be used to trace “counters” that are application specific units of useful work and to correlate them in a standardized way with electricity consumption statistics. This works in realtime and can be a very powerful approach, but it has two important drawbacks. First, it requires software instrumentation which makes it impractical for a large part of the installed application base. Second, it is restricted to energy metrics inside a certain facility, ignoring for example energy consumption of internet traffic.

VII. CONCLUSION

A. *Contributions*

We have presented an initial set of three indicators for energy-efficiency of e-services. We explained how we arrived at these indicators throuh the Goal-Question-Metric methodology and how they fit into a structured evaluation model for energy-efficiency. For each indicator we have provided a definition and guidelines for measurement and interpretation. In two case studies, we demonstrated the useability of the indicators and we reflected on their use in scenarios for portfolio optimization, vendor management, and sustainability reporting.

B. *Future Work*

We foresee several avenues for future work.

We would like to gain more experience with the indicators by conducting further case studies. Based on our experiences so far, we intend to develop a measurement toolkit that can be used by service maintainers to collect the necessary information with minimal effort. This allows to scale up the experiment to a larger number of organisations and their services.

By collecting a large number of energy profiles for services across functional domains and with a wide range of workload profiles and technology footprints, we will build up a reference database. This reference can be used to establish acceptable thresholds for each indicator in specific situations. Such threshold-setting is a neccessary precondition for developing a energy-labeling scheme for software services.

We expect that experience of applying the indicators will also lead to a clearer understanding of their limitations, which would naturally lead to the definition of complementary indicators. For instance, it would make sense to complement energy-efficiency indicators with indicators for other sutainability aspects, such as carbon emission. Another possibility is to enlarge the scope from the operational efficiency of the services to the efficiency of the preceeding development process.

ACKNOWLEDGEMENTS

The authors thank Frank Hartkamp (Agentschap NL) for inspiring and supporting this research. We also want to thank Dirk Harryvan (Mansystems), Cap Gemini, Equinix, and KPN for their collaboration.

REFERENCES

[1] J. Koomey, "Growth in data center electricity use 2005 to 2010," 2011.

[2] P. Teunissen and E. Lambregts, "Energiebesparing bij datacenters," 2012.

[3] J. Koomey, "Estimating total power consumption by servers in the U.S. and the world," 2007.

[4] R. Brown, E. Masanet, B. Nordman, B. Tschudi, A. Shehabi, J. Stanley, J. Koomey, D. Sartor, P. Chan, J. Loper *et al.*, "Report to congress on server and data center energy efficiency public law 109-431 [J]," *Public law*, vol. 109, p. 431, 2007.

[5] N. Wirth, "A plea for lean software," *Computer*, vol. 28, no. 2, pp. 64 –68, feb 1995.

[6] V. R. B. G. Caldiera and H. D. Rombach, "The goal question metric approach," *Encyclopedia of software engineering*, vol. 2, pp. 528–532, 1994.

[7] C. L. Weber, J. G. Koomey, and H. S. Matthews, "The energy and climate change implications of different music delivery methods," *Journal of Industrial Ecology*, vol. 14, no. 5, pp. 754–769, 2010.

[8] L. Barroso and U. Holzle, "The case for energy-proportional computing," *Computer*, vol. 40, no. 12, pp. 33–37, 2007.

[9] K. Grosskop and J. Visser, "Identification of application-level energy-optimizations," in *Proceedings of the First International Conference on Information and Communication Technologies for Sustainability (ICT4S)*, L. M. Hilty, B. Aebischer, G. Andersson, and W. Lohmann, Eds. ETH Zurich, 2013.

[10] ——, "Chapter 5 - energy efficiency optimization of application software," in *Green and Sustainable Computing: Part II*, ser. Advances in Computers, A. Hurson, Ed. Elsevier, 2013, vol. 88, pp. 199 – 241.

[11] L. Newcombe, Z. Limbuwala, P. Latham, and V. Smith, "Data centre fixed to variable energy ratio metric (dc-fver)," 2012. [Online]. Available: http://dcsg.bcs.org/groundbreaking-white-paper-new-dc-metric-available-review

[12] J. L. Zapico Lamela, M. Turpeinen, and N. Brandt, "Greenalytics: A tool for mash-up life cycle assessment of websites," in *Proceedings of the 24th International Conference on Informatics for Environmental Protection.* Shaker Verlag, 2010.

[13] T. Kim, Y. Lee, and Y. Lee, "Energy measurement of web service," in *Proceedings of the 3rd International Conference on Future Energy Systems: Where Energy, Computing and Communication Meet*, ser. e-Energy '12. New York, NY, USA: ACM, 2012, pp. 27:1–27:8. [Online]. Available: http://doi.acm.org/10.1145/2208828.2208855

[14] "Intel energy checker software developer kit user guide, revision 2.0, december 15, 2010." [Online]. Available: http://software.intel.com/en-us/articles/intel-energy-checker-sdk