Ener-G: A Generic Approach for Modeling Energy Consumption

Gergö Lovász
University of Passau
Innstrasse 43, D-94032 Passau
gergoe.lovasz@uni-passau.de

Florian Niedermeier
University of Passau
Innstrasse 43, D-94032 Passau
florian.niedermeier@uni-passau.de

Michel Steichen
University of Kaiserslautern
Paul-Ehrlich-Straße 34, 67663 Kaiserslautern
steichen@informatik.uni-kl.de

Bernd Reuther
University of Kaiserslautern
Paul-Ehrlich-Straße 34, 67663 Kaiserslautern
reuther@informatik.uni-kl.de

Hermann de Meer
University of Passau
Innstrasse 43, D-94032 Passau
demeer@uni-passau.de

Paul Müller
University of Kaiserslautern
Paul-Ehrlich-Straße 34, 67663 Kaiserslautern
pmueller@informatik.uni-kl.de

I. Introduction

For the last couple of years, due to ecological as well as economical reasons, energy efficiency of ICT has become a major issue. Leading manufacturers of IT equipment advertise with "green" and energy-efficient products. Steadily increasing energy consumption in data centers [1], as well as high electricity costs prompt organizations, companies, and administration to decrease the energy consumption of their IT infrastructures. With technologies like virtualization [2]–[5] and live migration of virtual machines [6], tools are given that allow a dynamic and automatic resource allocation. Combined with load-based energy consumption models of infrastructure-components (e.g. servers, storage, routers, switches, etc.) and the monitoring of their energy-related properties, especially their load, such techniques can be used to increase the energy efficiency of IT infrastructures.

The monitoring of the load on certain infrastructure-components provides information whether the component is able to provide a required quality-of-service (QoS) to a service which is planned to be deployed on the component or not. The energy consumption models allow to estimate the energy consumption of the components for a specific service deployment scenario. Live migration enables the relocation of virtualized services based on the monitoring information and estimated load-dependent energy consumption of the components in a way that the overall energy consumption of the IT infrastructure is decreasing.

In Section II a generic approach for modeling energy consumption is presented, using a server as an example for a component. Section III concludes this document.

II. Generic Approach for modeling energy consumption

In this section a generic approach for modeling the energy consumption in a complex infrastructure with complex components is described.

The approach consists of three phases:

- Modular description of the infrastructure components by identifying their less-complex subcomponents in order to decrease the complexity.
- Functions $f_1, ..., f_n$ have to be found that describe the energy consumption of the subcomponents $s_1, ..., s_n$ of a component c based on the load and the static and dynamic properties of each subcomponent. A function F has to be developed that estimates the energy consumption of c, based on the static and dynamic properties of c as well as on $f_1, ..., f_n$.
- For applications with certain QoS requirements (and therefore resource requirements) the energy consumption has to be determined when deployed on a specific component. This allows comparing different application deployments and choosing the most energy-efficient one.

A. Modular description of components

A component (e.g. server) has static properties that do not change during runtime, dynamic properties that can and normally do change at runtime, and subcomponents which are interacting with each other and contribute to the overall power consumption of the component.

Figure 1 shows a graphical representation of a server subcomponent tree. The root of the tree is the component itself for which the energy consumption should be modeled later on. The nodes of the tree represent the subcomponents whereas the leaves are the static and dynamic properties of the component and its subcomponents. For a server, several subcomponents can be identified (e.g. HDD, RAM, power supply unit (PSU), CPU, NIC). Some of them have further subcomponents themselves. E.g. the CPU could have several Cores with certain properties like the L1-cache size.

A compromise has to be found between the granularity of the component model on the one hand and the simplicity of the model and the measurability of dynamic properties on the other hand. The higher the granularity of the model, the higher the precision of the estimated energy consumption. But a very detailed model leads to a longer computation

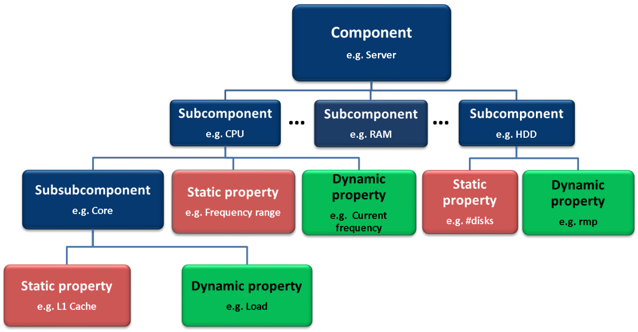

Fig. 1. Subcomponent tree of a server

time of the energy consumption estimation and requires more measurement and monitoring capabilities. Therefore during the development process of the energy consumption model of a component the subcomponents have to be identified that mainly contribute to its energy consumption.

B. Estimated energy consumption of a component

Through the modular description of a component c the estimation of c's energy consumption is simplified because in a first step the energy consumption of the subcomponents can be investigated. Based on these results a more complex energy-consumption model for c can be developed considering the correlation of the energy consumption of the subcomponents.

For a subcomponent s_i of c, $1 \leq i \leq n$ a function f_i has to be found that approximates the energy consumption of s_i based on its dynamic parameters, static parameters, and if existing, on the estimated energy consumptions of its own subcomponents. As in most cases the load is most important dynamic property, therefore, it has to be investigated how the energy consumption of a subcomponent changes with respect to changing load. Since in most cases only the energy consumption of the whole component (e.g. server) can be measured directly and the energy consumption of a subcomponent (e.g. CPU) not, a solution has to be found to determine the influence of the subcomponent on the overall energy consumption.

To solve this problem, benchmarks are required that stress certain subcomponents with different load levels without generating load on other subcomponents. When the energy consumption of the whole component is measured the impact of the different loads on the subcomponent on the overall energy consumption can be determined since the load on other subcomponents is negligible. For a server, benchmarks have to be found that stress single subcomponents like CPU, NIC, HDD, or RAM. For example if the CPU is stressed, load on HDD, RAM, and NIC should be avoided.

Correlations between subcomponents can be found by stressing more than one subcomponent at the same time. The energy consumption of a component c is then finally calculated by a recursive function F defined as follows:

$$F(\vec{c_{sp}}, \vec{c_{dp}}, c) = \begin{cases} f_n(\vec{s_{sp}}, \vec{s_{dp}}) & \text{if } SC{=}\emptyset \\ \sum_{s \in SC} F(\vec{s_{sp}}\vec{s_{dp}}, s), & \text{else} \end{cases}$$

The dynamic and static parameters of a component or subcomponent are identified by the indices sp and dp respectively. Both parameters are declared as vectors containing an arbitrary number k of static or dynamic properties ($k \in \mathbf{N}$). The set SC contains all subcomponents s of a given component or subcomponent. If the set SC is empty the recursion stops and the function f_n returns the energy consumption of subcomponent n regarding its static and dynamic properties.

C. Estimated energy consumption of an application

The energy consumption of an application can be described by a characteristic resource usage profile. This usage profile is time-dependent as applications are normally not continuously using the same resources. For instance runtime and idle time of an application are both depending on the frequency of user requests. Such periodicities should be considered in the generated profile.

Furthermore dependencies and resource races between many applications can occur. Two different scenarios are possible:

- There can exist a dependency between logically independent applications which are competing for the same resources.
- In many cases applications can be logically dependent from each other. For example the services provided by a database and a web server are closely tied together.

For an arbitrary application the characteristic functions $f_1, ..., f_n$, as defined in section II, have to be determined every time it is executed on a different architecture. Besides this, the platform dependent static and dynamic properties have to be identified. Finally the energy consumption of an application can be computed with the recursive function F which permits to choose the most energy-efficient deployment.

III. Conclusion

The model presented in this paper can be used to approximate the energy consumption of applications running on a particular hardware platform. Models for more complex infrastructures can be created by combining the models of the underlying components. By adapting the parameters of function F the model can be used to estimate the energy consumption of any application on any platform and thus optimize the overall power consumption of any given infrastructure.

References

[1] Koomey. *Estimating regional power consumption by servers: A technical note*, 2007, http://www.amd.com/us-en/assets/content_type/DownloadableAssets/Koomey_Study-v7.pdf

[2] Siering. *Realittsverschiebung - Virtualisierungstechniken im Vergleich, Heft 13, Magazin für Computer Technik*, 2006

[3] http://www.vmware.com/

[4] http://www.xen.org/

[5] http://www.virtualbox.org/

[6] *VMware* VMotion für Live-Migration virtueller Maschinen http://www.vmware.com/files/de/pdf/vmotion_datasheet_de.pdf