

GreenBrowsing: Towards Energy Efficiency in Browsing Experience

Gonçalo Avelar and Luís Veiga

Instituto Superior Técnico - ULisboa
INESC-ID Lisboa, Portugal
luis.veiga@inesc-id.pt,
goncalo.avelar@ist.utl.pt

Abstract. Web 2.0 allowed for the enhancement and revamp of web pages' aesthetics and interaction mechanics. Moreover, current web browsers function almost as a *de facto* operating system: they run “apps”, along with other background plug-ins. All of which have an increasing energy impact, proportional to the rate of appearance of more sophisticated browser mechanisms and web content. We present the architecture of GreenBrowsing. A system that proposes the provision of (i) a Google Chrome extension to monitor, rationalize and reduce the energy consumption of the browsing experience and (ii) a Certification Scheme for dynamic web pages, based on web-page performance counter statistics and analysis, performed on the cloud.

Keywords: web browser, web page certification, green IT, energy efficiency, power consumption.

1 Introduction

The creation of more capable technologies (HTML5, CSS, JavaScript) improved connectivity and content delivery in the last few years. Now, website contents that are sent to web browsers require substantial resource usage, while being processed, in order to perform reasonably and match user expectations. This ultimately leads to increased power consumption, being reasonable to assume that power consumption is increased with resource usage. Users should be aware of the power cost induced by visiting each web page, locally and remotely, relative to the power cost of other web pages, allowing them to choose between two or more functionally equivalent set of web pages the least power hungry.

Hence, we consider web browsers as suitable candidates for the deployment of a power management solution, and present GreenBrowsing: a system that aims at extending Google Chrome [1], in its *Desktop* and *Laptop* versions, in order to decrease the energy costs of browsing, by managing idle tabs regarding resource usage and adapting the content of selected tabs, to perform energy-related optimizations. GreenBrowsing also aims at certification of web pages, to ensure that users become aware of which web domains are the less green and

which pages are more resource hungry. We assume many devices where Chrome will be running are Wi-Fi enabled.

The main challenge of this work is to provide mechanisms that effectively reduce the energy cost when browsing the web, without sacrificing much of the responsiveness and performance that is expected, and (at the same time) to provide means to certify web pages energetically-wise, in order to inform users of the energetic inefficiencies related to different web page visualizations.

2 Related Work

In this section, we address Dynamic Power Management (DPM), since its rationale is very similar to part of what we want to achieve with GreenBrowsing – to reduce the resource usage of idle tabs (unselected, not visualized at some moment). We will also address some of the work done in terms of energy-related certification.

Dynamic Power Management. Dynamic Power Management (DPM) techniques try to achieve power dissipation reductions by employing policies that reduce the performance of system components (typically by inducing them into some sort of sleep state) when they are idle, while under performance constraints [4]. Therefore the greatest challenge in DPM is to know when and what components should be put to sleep, to the end of achieving the minimum power dissipation possible, given the performance constraints imposed. In that case, the policy is said to be optimal.

One pioneering example of DPM is the work by Qiu et al. [8] that follows a stochastic approach. The authors describe a continuous-time Markov Decision Process (MDP) as the system’s power model. Markov Decision Processes can be thought of as state machines whose state transitions depend on actions performed and are done with a certain probability. Each state transition has a score assigned to it, consisting of the product of the probability of that transition occurring, the energy cost of that transition and a certain weight. The idea is to minimize the sum of all scores over time, by iteratively adjusting the weights. After a certain number of iterations, an optimal policy is guaranteed to be found.

Shen et al. propose an approach [10] to dynamic power management using Reinforced Learning, specifically the *Q-Learning* algorithm, in order to learn an optimal Markov Decision Process, after a certain set-up time, as the power manager executes. Each state-action pair of the MDP has a quality value associated with it, that is computed considering the expected average power and request delay caused by the action taken at each state. In this way, the policy can be consistently adjusted, in order to consider states that minimize the delay cost at each state and expected average power wasted, given the decisions taken.

Energy-Related Certification. In terms of Energy-related Certification, there is some work targeting different kinds of computational systems, both mobile devices [11] and software [3]. However, to our knowledge, there is no substantial

volume of work regarding certification of web pages. Camps et al. proposed a solution to do so [5], still, certification is done accounting only for the downloadable content of pages, disregarding other system metrics (e.g. CPU).

3 Architecture

There are two major sub-systems that comprise the GreenBrowsing architecture (Fig. 1): a Google Chrome Extension, and a Web Page Certification Back End.

Chrome Extension Sub-system. The main roles of the Chrome Extension are *to reduce the resource consumption of tabs*, and *to send to the Back End resource-related data*, that can be used to certify web pages in terms of their energy consumption while being accessed.

Chrome Extension modules. The **Observer-Controller-Adapter (OCA)** provides interfaces for gathering performance counters of each running tab and the process(es) it is associated with. It also provides interfaces for issuing commands to tabs and the operating system itself. Moreover, it will be possible to adapt web pages through this module. The **Certification Front End** consists in the necessary code to communicate with the Certification Back End, to the end of sending performance counters. The **Certification Renderer** has the functionality needed to render an energy-related rating of each web page, based on its certification. It serves visualization purposes mostly.

The **Policy Enforcer** includes all the power reduction logic and can use the OCA interface, to gather performance counters and to issue content adaptation and power reduction related commands. In particular, it includes the *Tab management Policy*, which is responsible for choosing when to reduce performance and resource consumption of tabs.

The **Web Page Certifier** fetches performance counters, through the Observer-Controller-Adapter. It also interfaces with the Certification Front End to send the counters gathered to the Back End (for energy-related certification of web pages). Communications with the Certification Renderer are done to visually associate an energetic certification stamp to each domain and page, on each tab.

The **Profile Manager** displays graphical interface the user might use to further tune GreenBrowsing to his/her preferences. Interfacing with the Policy Enforcer is done to communicate user preferences.

Resource Usage & Tab Management. We distinguish *Selected Tabs* (at each time, only one tab is selected in Chrome and therefore only one tab is visualized) from *Idle Tabs* (every tab that is available and open, but that is not selected).

Selected Tabs can have the resources they use reduced by employing adaptation techniques defined by the user. Examples are the removal of images or sound.

Regarding Idle Tabs, they are said to be *Suspended* when they are target of any power management action (other than adaptation), in order to reduce their

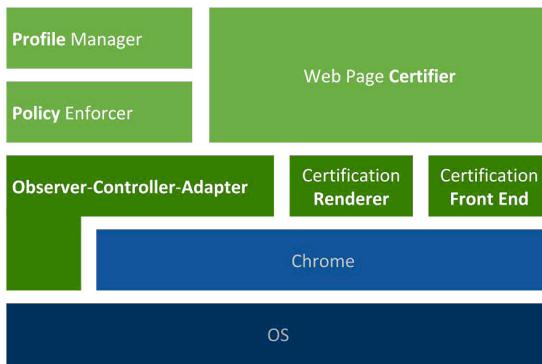


Fig. 1. Layered View of The Chrome Extension

resource usage. In this way, we make only two assumptions regarding general browsing behavior, that will be considered when managing tabs:

- **Last Time Usage.** Tabs that were accessed more recently are more likely to be accessed again and therefore are less likely to be suspended.
- **Distance to other Tabs.** We also assume that tabs closer to the selected tab are more likely to be accessed, and therefore they are also less likely to be suspended.

We can infer power consumption from resource usage. In particular, the CPU intensiveness (in terms of load and number of cycles), and the bandwidth in terms of bits-per-second, all of which on average and over fixed periods of time.

Moreover, we intend to leverage recent work in our research group, addressing resource management and elastic scheduling in cloud infrastructures. It takes into account relative efficiency of scheduled resources usage, with application progress monitoring, factoring in perceived utility depreciation by users when resources are sub-allocated [14,13,12]. We address similar constraints in this work, as the best possible performance is only achievable with high resource usage, and we want users to obtain experiences that combine adequate performance within the intended energy consumption profile.

Analytics and Certification Back End Sub-system. The Certification Back End Sub-System (Fig. 2) is moved to a separate remote sub-system, intended to be deployed in the cloud, as a way to reduce the computational intensiveness that would be required by the extension, in the certification process. The Back End has the objective of *providing a clear and meaningful notion of how much energy web pages consume*, certifying also the domain of each web page, to give an overall notion of how *green* that domain's pages are.

Certification Back End Components.

- A **Network Communication task** that receives energy-related web page certification requests and forwards these requests to specialized workers;
- **Analytics Certifier tasks** that do the work of certifying a given page, according to a specific certification model.
- A **Certification Modeler task** that adjusts the certification model, taking into account all the resource data sent from the extension subsystem. For performance purposes, this design sub-intends the usage of specified **Worker Tasks** to whom parts of the analytical calculations are *mapped* to. The results of processing data at workers are assembled back to the Modeler Task, as soon as they are ready.
- An **Analytics Data Store** that stores the models used in the certification of pages and tuples with information relative to the performance counters of each page;

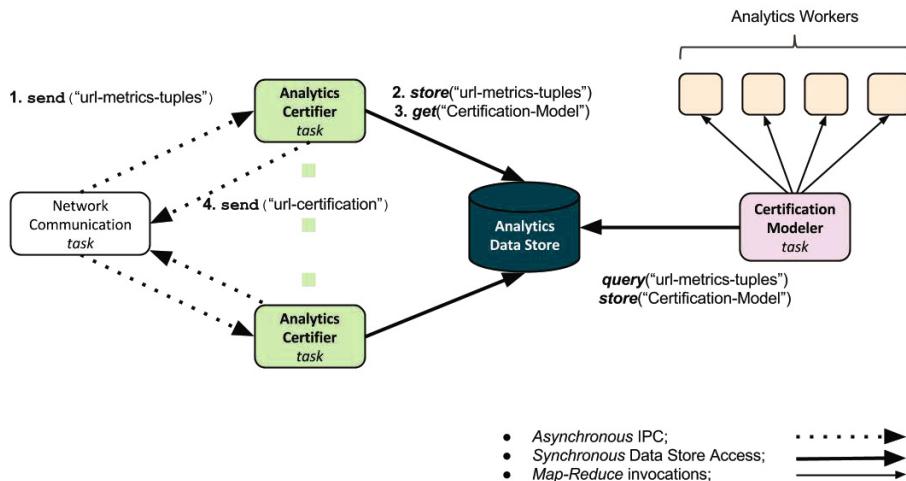


Fig. 2. The Communicating tasks of the Back End

The Certification Process. The power consumption induced by web pages can be indirectly determined by some of the performance counters gathered on the Chrome Extension. For each page, the metrics considered are: (i) the CPU clock cycles, or percentage, because of their relation to power consumption [9] and (ii) the network bandwidth in terms of bit-rate, because wireless communications are one of the major causes for power consumption on Wi-Fi enabled devices [7].

The information sent from the Chrome Extension to the Certification Back End is a 4-tuple $\langle \text{domain}, \text{page-url}, \text{number-of-cycles}, \text{bits/sec} \rangle$.

In order to divide different resource consumption patterns in different ranks, we can employ the clustering algorithm *Expectation-Maximization* (EM) [6] in order to find no fewer than 4 categories of certification (clusters), in a 2-dimensional space that comprehends one dimension for the CPU clock cycles and another

for the network bandwidth bit-rate. If we consider that each dimension has 2 possible qualitative scores (good or bad) depending on the actual value of the measurement obtained for a certain dimension, there are 2^2 possible score combinations. Hence, 4 possible categories or clusters. More scores per dimension can be considered for increased detail.

Each one of the clusters is associated with a vector called the centroid. We assume that given two categories whose clusters C_i and C_j have centroid c_i and c_j , respectively, if $\|c_i\| > \|c_j\|$ (where $\|c_i\|$ denotes the norm of c_i), then the vectors that are members of the cluster C_i are more power consuming than the ones of C_j . In this way and by assessing which cluster an incoming observation is more likely belonging to (given the clusters obtained with previous observations), we are able to classify, and therefore certify, pages regarding power consumption.

To certify domains, the arithmetic mean and the geometric mean of the page classifications (obtained with the method explained previously) are computed.

4 Evaluation Methodology

The evaluation of GreenBrowsing is done with respect to (1) the energy gains by the usage of the extension and (2) perceived latency by the user.

Since it would be difficult to differentiate the actual energy wasted by the browser with idle pages, with and without our extension, from noisy energetic patterns caused by other applications and system activity, energy gains can be measured indirectly through other more direct metrics: (i) *CPU load per tab* and (ii) *bandwidth usage per tab*.

For measuring the perceived latency by the user, the (i) *overall time to load pages* (i.e. fully render pages) will be considered. It will also be considered the (ii) *processing time at the Back End*, in order to account for late certification stamps.

To conduct the evaluation of the resource usage optimization that GreenBrowsing provides, a set of typical web pages can be used. These comprise web pages of news sites, social networks, sports, mail clients and multimedia in order to provide a rich and varied Web page suite to test.

To extract the measurements intended for each metric, we use the Performance Profiling Capabilities along with the Timeline monitor [2], from the Chrome DevTools suite.

5 Conclusions

We observe that in spite of the modern (and resource hungry) web technology, the processing of web pages/applications done by web browsers leads to considerable power consumption, on end-users devices. This hinted us that browsers should be tailored to power management extensions. To that end, we presented the architecture of GreenBrowsing, comprised of a Chrome Extension to manage the power consumption of tabs and a web page energy-related Certification Back End, to be deployed on the cloud. The implementation of GreenBrowsing is ongoing for further evaluation according to the methodology proposed.

References

1. Chrome browser, <https://www.google.com/intl/en/chrome/browser/>
2. Chrome devtools, <https://developers.google.com/chrome-developer-tools/>
3. Amsel, N., Tomlinson, B.: Green Tacker: A Tool for Estimating the Energy Consumption of Software. In: CHI 2010 Extended Abstracts on Human Factors in Computing Systems, CHI EA 2010. ACM, New York (2010)
4. Benini, L., Bogliolo, A., Cavallucci, S., Riccò, B.: Monitoring System Activity for OS-directed Dynamic Power Management. In: Proceedings of the 1998 International Symposium on Low Power Electronics and Design. ACM, New York (1998)
5. Camps, F.: Web browser energy consumption (2010)
6. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. *Journal of The Royal Statistical Society, Series B* 39(1), 1–38 (1977)
7. Perrucci, G.P., Fitzek, F.H.P., Widmer, J.: Survey on Energy Consumption Entities on the Smartphone Platform. In: VTC Spring, pp. 1–6. IEEE (2011)
8. Qiu, Q., Pedram, M.: Dynamic Power Management Based on Continuous-time Markov Decision Processes. In: Proceedings of the 36th Annual ACM/IEEE Design Automation Conference, DAC 1999. ACM, New York (1999)
9. Rodrigues, R., Koren, I.K.S.: A Study on the Use of Performance Counters to Estimate Power in Microprocessors. *IEEE Transactions on Circuits and Systems II: Express Briefs* (2013)
10. Shen, H., Tan, Y., Lu, J., Wu, Q., Qiu, Q.: Achieving Autonomous Power Management Using Reinforcement Learning. *ACM Trans. Des. Autom. Electron. Syst.* 18(2) (2013)
11. de Siebra, C., Costa, P., Marques, R., Santos, A.L.M., da Silva, F.Q.B.: Towards a Green Mobile Development and Certification. IEEE (2011)
12. Silva, J.N., Veiga, L., Ferreira, P.: A2HA - Automatic and Adaptive Host Allocation in Utility Computing for Bag-of-Tasks. *Journal of Internet Services and Applications (JISA)* 2(2), 171–185 (2011)
13. Simão, J., Veiga, L.: QoE-JVM: An Adaptive and Resource-Aware Java Runtime for Cloud Computing. In: Meersman, R., et al. (eds.) OTM 2012, Part II. LNCS, vol. 7566, pp. 566–583. Springer, Heidelberg (2012)
14. Simão, J., Veiga, L.: Flexible SLAs in the Cloud with Partial Utility-driven Scheduling. In: IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom 2013) - Best-Paper Award Runner-up. IEEE (December 2013)