Designing and Managing Datacenters Powered by Renewable Energy *

Íñigo Goiri, William Katsak, Kien Le, Thu D. Nguyen, and Ricardo Bianchini
Department of Computer Science, Rutgers University
{goiri,wkatsak,lekien,tdnguyen,ricardob}@cs.rutgers.edu

Abstract

On-site renewable energy has the potential to reduce datacenters' carbon footprint and power/energy costs. We built Parasol and GreenSwitch to explore this potential in a controlled, research setting.

1. Introduction

Datacenters range from small numbers of servers in machine rooms to thousands of servers housed in warehouse-size installations [8]. Estimates for 2010 indicate that collectively datacenters consume around 1.5% of the total electricity used world-wide [8]. This translates into high carbon emissions as most of this electricity comes from fossil fuels. A 2008 study estimated that datacenters emit 116 million metric tons of carbon, slightly more than the entire country of Nigeria [12].

With increasing societal demand for cleaner products and services, several companies have announced plans to build "green" datacenters, i.e. datacenters partially or completely powered by renewables such as solar or wind energy. These datacenters will either generate their own renewable energy (self-generation) or draw it directly from an existing nearby plant (co-location). For example, Apple and McGraw-Hill have built large solar arrays for their datacenters, whereas Green House Data is a small cloud provider that operates entirely on renewables. Although there are other approaches, these examples suggest that many datacenters that seek to lower emissions will prefer co-location or self-generation. In [5], we discuss the current and (expected) future cost and space needs of on-site solar and wind generation.

Co-location and self-generation pose an interesting research challenge: solar and wind energy are intermittent, which requires approaches for tackling the energy supply variability. One approach is to use batteries and/or the electrical grid as a backup for the renewable energy. It may also be possible to adapt the workload (the energy demand) to match the renewable energy supply [1, 4, 6, 9, 11]. For highest benefits, green datacenter operators must intelligently manage their workloads and the sources of energy at their disposal. For example, when the workload is deferrable (i.e., it can be delayed within a time bound), it may be appropriate to delay some of the load, and store the freed-up renewable energy in the batteries for later use (e.g., to shave an expected load peak when the renewable energy is not available). As far as we know, current green datacenter operators do not manage their energy sources and workloads in this manner.

We set out to build software and hardware to explore these issues. This paper overviews two of our main efforts: Parasol and GreenSwitch.

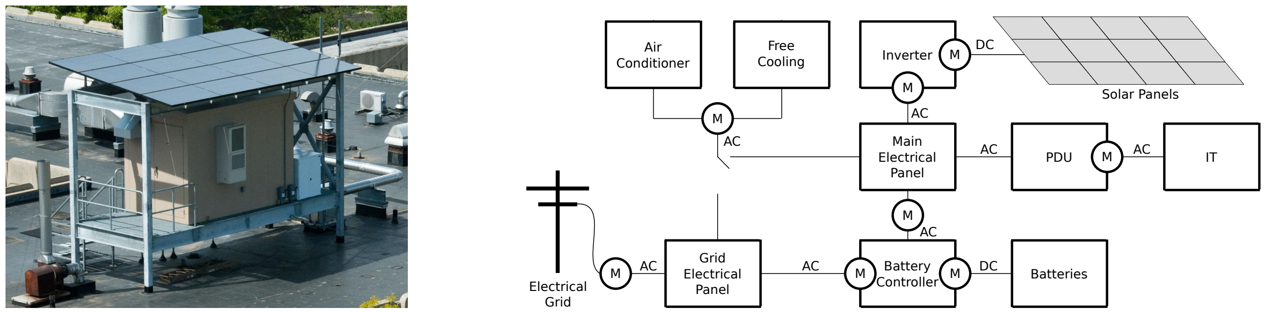
2. Parasol

Overview. Figure 1(a) shows Parasol, a solar-powered datacenter that we have built as a research platform to study co-location and self-generation. Parasol comprises a steel structure, a small custom container housing 2 racks of servers and networking equipment, an air-side economizer ("free") cooling unit and a direct-expansion air conditioner, 16 solar panels (producing up to 3.2kW AC), 2 DC/AC inverters, 16 lead-acid batteries (storing up to 32kWh), 2 charge controllers, and an electricity grid tie. Parasol currently houses 64 half-U Atom-based servers (consuming at most 30W each), but it is large enough to house 150 of them. It uses free cooling whenever outside temperatures and humidity are low enough and air conditioning otherwise. Parasol can use solar energy directly, store it in its batteries, or feed it to the grid for credit ("net metering"). We thought about adding a wind turbine to Parasol, but historical weather data shows that our location (Piscataway, NJ) is not windy enough.

Figure 1(b) shows Parasol's power distribution and monitoring infrastructure. Because Parasol was built as a research instrument for studying power management in green datacenters, it is critical that we understand the power usage of each component, as well as power losses. Thus, note that we have power meters (labeled "M" in the figure), either internal to components (e.g., the DC/AC inverters) or added-on externally (e.g., the meter for the cooling system), for measuring the power flowing into/out of every component. Parasol also includes a switch that allows the cooling system to be powered from the main electrical panel or only from the grid. This allows experimentation with or without the cooling system loading the solar system and batteries.

We describe our rationale for the Parasol design and the mistakes we made while building it over 16 months (at a total cost of $300K) in [5]. In the remainder of this section,

* Parasol and GreenSwitch: Managing Datacenters Powered by Renewable Energy. Proceedings of the 18th Int. Conf. on Architectural Support for Programming Languages and Operating Systems, pp. 51-64, March 2013.



(a) **(b)**

Figure 1. (a) Outside view of Parasol. (b) Parasol's power distribution and monitoring infrastructure.

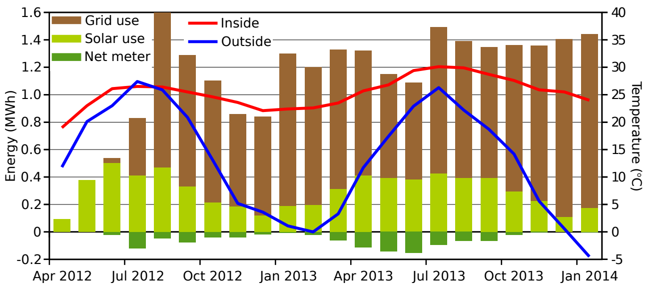


Figure 2. Energy consumption, net metering, and temperatures from April 2012 to January 2014.

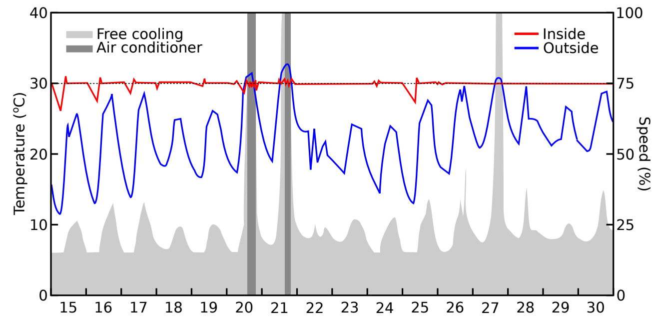


Figure 3. Cooling operation from August 15-30, 2013.

we report on data gathered from operating Parasol over 22 months. Specifically, solar generation and the IT equipment became operational in April 2012, and Parasol became fully operational in June 2012.

Energy production and usage. Figure 2 shows energy usage, energy net metered, and the average inside and outside temperatures from April 2012 to January 2014. We computed a power usage effectiveness (PUE) of 1.06 to 1.08, depending on the computing load, because of losses due to various conversions. April through June 2012 show little or no grid energy consumption because the external meters did not become operational until the end of June 2012. Note that total solar energy production is the sum of solar energy consumed and solar energy net metered. This data shows that during the summer months, Parasol produces more than 500kWh every month, whereas during the winter this production reduces to less than half. For the year spanning July 2012 through June 2013, we computed an average solar capacity factor of 16%. During this time, Parasol supported workloads used for studying GreenSwitch and six other research projects.

Interestingly, grid energy consumption in July 2012 was significantly lower than in other months because we were experimenting with GreenSwitch, transitioning machines to sleep and using batteries (charged with solar energy) to reduce brown energy consumption. Starting in November 2012, we raised the internal setpoint temperature from 27°C to 30°C.

Cooling. Figure 3 shows the operation of the cooling system in Parasol during the second half of August 2012. In this time period, the setpoint for internal temperature was 30°C; the red line shows the actual internal temperature, whereas the blue line shows the outside temperature. The light gray area shows the operation of the free cooling unit, whereas the dark gray area shows the operation of the air conditioner. Observe that even though this time period is in the summer, the air conditioner only ran during two days when the outside temperatures exceeded 30°C. Much of the time, the free cooling unit ran below 25% fan speed.

The average PUE when including both conversion losses and cooling overheads for Parasol has been lower than 1.13, showing that free cooling is very effective at keeping cooling overheads low. The air conditioner has run for less than 20 days in a year, and less than 1% of the total time. Most of the time, our setpoint has been 30°C and the typical temperatures inside Parasol ($> 95\%$) have ranged between 22°C and 30°C. We have also been experimenting with novel cooling policies and pushing the limits of Parasol. During these experiments, the internal temperature at the control sensor has ranged between 15 and 36°C.

Thus far, we have replaced 5 HDDs, 2 SSDs, and one mother board. Although this data is not statistically significant, it is possible that our experiments have decreased the reliability of the IT equipment.

Off-grid operation: Hurricane Sandy. In late October 2012, the East Coast was hit by hurricane Sandy. The storm

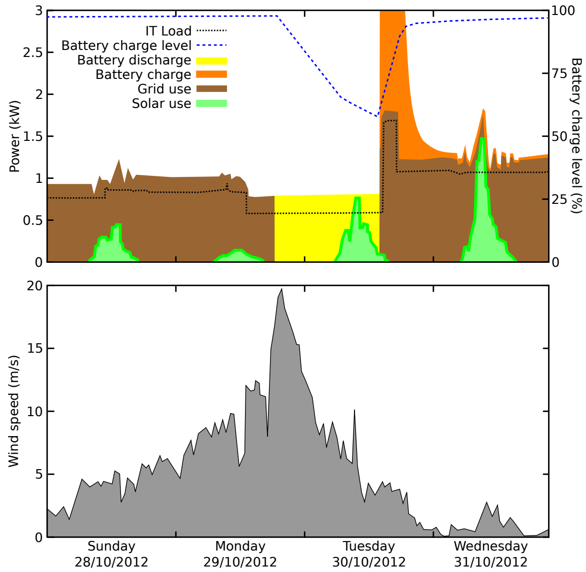


Figure 4. Parasol's operation during hurricane Sandy.

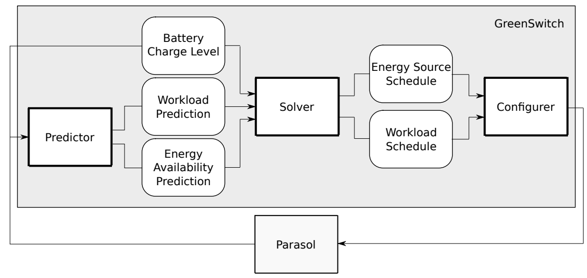


Figure 5. GreenSwitch architecture. Rectangles with round edges are data structures. Rectangles with square borders are processes.

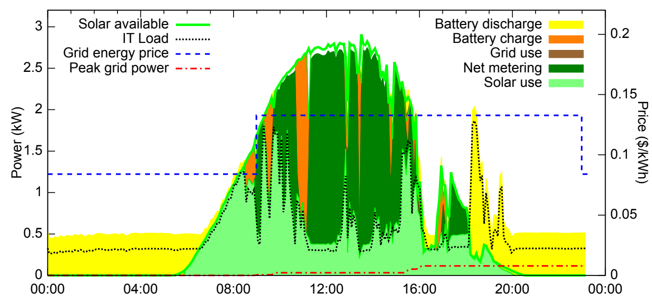


Figure 6. GreenSwitch on deferrable Facebook workload.

reached Rutgers on October 29th and the grid power and network suffered outages for more than 20 hours. Figure 4 shows the behavior of Parasol and the wind speed at our location from October 28th to November 1st. Rutgers lost power in the afternoon of Monday, at the height of the measured wind speed (>70km/h), and it did not come back until the afternoon of the next day. During this time, Parasol used its batteries and solar energy to operate normally (although we did transition half of the machines to sleep as they were not being used). This experience demonstrates the potential for green datacenters to operate through power outages (or in remote locations without a reliable grid power source).

3. GreenSwitch

Overview. We now discuss our research on managing Parasol. Specifically, we describe GreenSwitch a system for scheduling workloads, selecting the source of energy to use (renewable, battery, and/or grid), and/or selecting the renewable energy storage medium (battery or grid) at each point in time. GreenSwitch seeks to minimize the overall cost of grid electricity (including both grid energy and peak grid power), while respecting the characteristics of the workload and battery lifetime constraints. It can also manage workloads and energy sources during grid outages.

Figure 5 illustrates the GreenSwitch architecture. The *predictor* predicts the workload and the renewable energy production one day into the future at the granularity of one hour. The *solver* takes these predictions and the current battery charge level as input, and outputs a workload schedule and an energy source/storage schedule. To compute these schedules, the solver uses analytical models of workload behavior, battery use, and grid electricity cost. The *configurer* effects the changes prescribed by the solver. The changes may involve transitioning some servers between power states and/or changing the configuration of the energy sources.[1]

A full iteration of GreenSwitch occurs every 15 minutes, which enables it to properly control peak grid power use. (Utilities typically compute peak grid power use in windows of 15 minutes.) However, it checks the production of solar energy every 3 minutes. During each of these checks, it runs a full iteration if there has been an unexpected change in production.

GreenSwitch evaluation on Parasol. We perform day-long experiments with Parasol and an implementation of GreenSwitch for the Hadoop MapReduce framework. We study two widely different Hadoop traces, called "Facebook" and "Nutch". The former derives from a larger batch-job trace from Facebook [2], whereas the latter is the indexing part of a Web search system [3]. We instantiate our models with the on-peak/off-peak grid energy prices and the peak grid power charges at our location. We assume the utility pays the wholesale price of electricity for net metering.

In the Facebook trace, jobs arrive throughout the day [2]. Figure 6 shows the GreenSwitch behavior when the jobs in the trace are deferrable (each job can be delayed by up to 1 day) on July 1, 2012. The fill colors represent the use of the different energy sources, whereas the lines are the solar energy production (full), the IT load (dots), the grid energy price (dashes, Y-axis on the right), and the current peak grid power draw (dashes-dots). The white fill represents solar

[1] We have identified configuration parameters to the inverters and charge controllers that allow us (nearly) full dynamic control of every source of energy available to Parasol.

energy that was produced but lost due to inefficiency. The figure shows that GreenSwitch transitioned many servers to sleep in the early hours of the day and deferred some of the load until solar energy was available. When there was no solar energy, GreenSwitch drew energy from the batteries, since they stored enough capacity for the load that was not deferred. We also see that the solar energy was enough to power the workload, charge the batteries, and feed energy to the grid. Compared to a grid-only datacenter, GreenSwitch produced a *profit* of 9% in grid electricity cost. Given this profit, GreenSwitch would amortize the cost of the solar setup and batteries in only 7.6 years.

Despite seeking primarily to minimize grid electricity cost, GreenSwitch is also successful at reducing carbon footprints. It achieves reductions in grid energy use between 36% and 100% in our experiments with Facebook and Nutch, compared to a grid-only datacenter.

4. Main Lessons

We have learned many important lessons in building Parasol and GreenSwitch. First, we learned that engineering contractors are unfamiliar with the state-of-the-art in datacenter design or with research prototypes. Our inability to bridge this knowledge gap quickly (or at all) caused delays. Organizations that want to build datacenters but lack the expertise are at the mercy of these engineers. Second, as Parasol was a major undertaking, its design needed to enable research on many topics (e.g., solar energy, free cooling, wimpy servers). However, as we had not yet started to research every topic, we ended up designing more features and flexibility into Parasol than we may eventually need. This increased costs. Third, the need to collect fine-grained power measurements and accurately estimate energy losses led to extra design complexity. Fourth, placing Parasol on the roof of a building (instead of on the ground) prevented shading from other buildings. Moreover, the cost of the roof placement was roughly the same as that of extending networking and power to ground locations far enough away from buildings. Fifth, the wimpy fans in wimpy servers can generate non-trivial temperature differences across a free-cooled datacenter. Sixth and most importantly, we learned that building a real prototype is critical for completely understanding green datacenters. For example, in designing GreenSwitch, we detected instability in our charge controllers when switching power sources. As a result, GreenSwitch performs these switches in steps, with some idle time in between the steps. Such effects would have been overlooked in simulation.

5. Potential Long-Term Impact

We expect Parasol and GreenSwitch to have a lasting impact on both academia and industry for the following reasons:

1. Increasing interest in leveraging renewable energy: As we mentioned above, several companies are starting to invest in datacenter co-location and self-generation. Regardless of whether they are making these investments for market positioning, public relations, cost, or environmental reasons, the fact is that they are expecting bottom-line benefits from them. Moreover, despite their decreasing-but-still-high capital costs, exploiting renewables in datacenters may reduce overall energy costs, peak grid power costs, or both, as our paper demonstrates. We expect that an increasing number of companies will see benefits in exploiting renewables.

Some research groups have also started studying co-located and self-generating datacenters, e.g. [1, 4, 9, 10, 13]. These studies have been attracting the attention of a growing community with publications in venues such as ISCA and ASPLOS. We expect that our design and experience with Parasol will accelerate this growth, as researchers realize that they can build non-trivial prototypes at relatively low cost. Moreover, the analysis of solar/wind cost and space requirements in [5] suggests that green datacenters will become increasingly attractive.

More broadly than datacenters, our experience will likely encourage more researchers to consider the implications of external signals (e.g., variable electricity pricing and availability) on computing and communication in general.

2. Parasol is the first green datacenter prototype: There has been a dearth of real platforms for the study of co-located and self-generating green datacenters. Parasol addresses this need and is the first platform of its kind. Prior studies have had to resort to simulations or small implementations. In [5], we list instances in which such alternatives would have hidden important effects. We mentioned instability issues above. Another example is that energy losses (e.g., in power conversion) are highly dependent on load, rather than a fixed percentage as often assumed in simulation. These instances will encourage researchers to build prototypes for their studies. We expect the Parasol design to serve as a model for these future research prototypes. Moreover, it enables research on a variety of important topics, including solar energy and its impact on computing, energy storage and its ability to lower costs, free cooling and its impact on reliability, wimpy servers and their performance/energy tradeoffs, and building distributed storage systems using solid-state drives. These topics are of interest to both industry and academia.

3. Parasol's design is directly usable in practice: In its current form, Parasol is a blue-print for industry to build small-scale, low-density green datacenters for enterprises and educational institutions, for example. Self-generating containers are cheaper and more practical to operate, and may be placed in less valuable locations than in machine rooms inside existing buildings. It is also ideal for remote deployments with poor or no access to electricity (networking may need to take place over satellite in this case).

4. GreenSwitch is the first energy source/store manager for green datacenters: GreenSwitch manages workload demand, multiple energy sources (renewable, battery, and grid), and multiple energy stores (battery and grid), all at the same time. Our results show that it is consistently effective at reducing grid electricity costs and carbon footprints.

5. GreenSwitch is simple and adaptable: Although often overlooked in academia, simplicity and adaptability are key requirements for practical adoption by industry. We designed GreenSwitch to have both properties. Specifically, it uses simple models of solar energy availability, energy demand, and battery behavior. In addition, although our current implementation targets Hadoop, GreenSwitch is modular in that only one component (the configurer) is specific to the underlying computing framework.

6. Rich possibilities for future work: Parasol and GreenSwitch create many new research avenues. For example, Parasol enables the study of the interplay between solar energy and free cooling; interestingly, solar energy is most abundant when outside temperature is hottest, i.e. when standard chiller-based cooling may be necessary in warm climates. As another example, GreenSwitch demonstrates the benefits of aggressive and coordinated management of energy sources/stores and workload execution, as well as the interplay between using batteries for powering the workload and for storing renewable energy. Prior work on aggressive use of batteries (e.g., [7]) did not consider renewables.

6. Conclusions

Datacenters that are partially powered by renewable energy represent an increasingly interesting topic from many perspectives. We hope that our experience with Parasol and GreenSwitch will entice other researchers and practitioners to consider these datacenters.

Acknowledgements

We would like to thank Abhishek Bhattacharjee, David Meisner, Santosh Nagarakatte, Anand Sivasubramaniam, and Thomas F. Wenisch for comments that helped us improve this paper. We are also grateful to our sponsors, NSF grant CSR-1117368 and the Rutgers Green Computing Initiative. Finally, we are also indebted to Joan Stanton, Heidi Szymanski, Jon Tenenbaum, Chuck Depasquale, SMA America, and Michael J. Pazzani for their extensive help in building/funding Parasol.

References

[1] B. Aksanli, J. Venkatesh, L. L. Zhang, and T. Rosing. Utilizing Green Energy Prediction to Schedule Mixed Batch and Service Jobs in Data Centers. In *HotPower*, 2011.

[2] Y. Chen, S. Alspaugh, A. Ganapathi, R. Griffith, and R. Katz. Statistical Workload Injector for MapReduce, 2012.

[3] EPFL. CloudSuite, 2012. `http://parsa.epfl.ch/cloudsuite/cloudsuite.html`.

[4] I. Goiri, R. Beauchea, K. Le, T. Nguyen, M. Haque, J. Guitart, J. Torres, and R. Bianchini. GreenSlot: Scheduling Energy Consumption in Green Datacenters. In *Supercomputing*, 2011.

[5] I. Goiri, W. Katsak, K. Le, T. Nguyen, and R. Bianchini. Parasol and GreenSwitch: Managing Datacenters Powered by Renewable Energy. In *ASPLOS*, 2013.

[6] I. Goiri, K. Le, T. Nguyen, J. Guitart, J. Torres, and R. Bianchini. GreenHadoop: Leveraging Green Energy in Data-Processing Frameworks. In *EuroSys*, 2012.

[7] S. Govindan, D. Wang, A. Sivasubramaniam, and B. Urgaonkar. Leveraging Stored Energy for Handling Power Emergencies in Aggressively Provisioned Datacenters. In *ASPLOS*, 2012.

[8] J. Koomey. Growth in Data Center Electricity Use 2005 to 2010, 2011. Analytic Press.

[9] A. Krioukov, C. Goebel, S. Alspaugh, Y. Chen, D. Culler, and R. Katz. Integrating Renewable Energy Using Data Analytics Systems: Challenges and Opportunities. *Bulletin of the IEEE Computer Society Technical Committee*, 2011.

[10] C. Li, A. Qouneh, and T. Li. iSwitch: Coordinating and Optimizing Renewable Energy Powered Server Clusters. In *ISCA*, 2012.

[11] Z. Liu, Y. Chen, C. Bash, A. Wierman, D. Gmach, Z. Wang, M. Marwah, and C. Hyser. Renewable and Cooling Aware Workload Management for Sustainable Data Centers. In *SIGMETRICS*, 2012.

[12] J. Mankoff, R. Kravets, and E. Blevis. Some Computer Science Issues in Creating a Sustainable World. *IEEE Computer*, 41(8), 2008.

[13] N. Sharma, S. Barker, D. Irwin, and P. Shenoy. Blink: Managing Server Clusters on Intermittent Power. In *ASPLOS*, 2011.