CLOUD-BASED SOFTWARE PLATFORM FOR BIG DATA ANALYTICS IN SMART GRIDS

Yogesh Simmhan, Saima Aman, Alok Kumbhare, Rongyang Liu, Sam Stevens, Qunzhi Zhou and Viktor Prasanna, *University of Southern California, Los Angeles, USA*

ABSTRACT

There is a global effort to incorporate pervasive sensors, actuators and data networks into national power grids. This *Smart Grid* offers deep monitoring and controls, but needs advanced analytics over millions of data streams for efficient and reliable operational decisions. This article focuses on a scalable software platform for the Smart Grid Cyber-Physical System using Cloud technologies. Dynamic Demand Response (D^2R) is a challenge application that we target on the USC campus microgrid to perform intelligent demand-side management and relieve peak load. Our platform offers an adaptive *information integration pipeline* to ingest dynamic data; a *secure repository* for researchers to share knowledge; *scalable machine-learning models* trained over massive datasets for agile *demand forecasting*; and a *portal* to visualize consumption patterns. Our design incorporates hybrid Clouds, including IaaS, PaaS, public and private, which suit the unique component needs for on-demand provisioning, massive scaling, and manageability, and helps us expand from the microgrid to the Los Angeles power grid.

KEYWORDS: CLOUD COMPUTING, SOFTWARE PLATFORM, DATA ANALYTICS, CYBER PHYSICAL SYSTEMS, SMART GRID, WORKFLOWS, MACHINE LEARNING

1. THE SMART GRID CYBER-PHYSICAL SYSTEM

Energy security and environmental sustainability are global concerns with immense societal impact. Significant energy assets go toward electricity generation, and the power sector is expected to grow worldwide [1]. Electricity is a pervasive service whose *reliable* supply is essential for the modern civilization, and its *efficient* generation and consumption is becoming equally important. Belying its vital importance, the power grid's infrastructure improvements have not kept pace with time – the US Power Grid is the largest interconnected "machine" in the world, and nearly a century old.

However, technology changes are starting to permeate through the entire grid, from generation to transmission to distribution (Figure 1). *Renewables* like wind and geo-thermal are being included into the generation mix, not just by the power utilities but also by their consumers through rooftop solar panels. Long distance transmission networks are being instrumented with *Phasor Measurement Units* to detect and prevent cascading failures. *Smart meters* are being deployed at consumer premises to monitor real-time energy consumption and securely communicate them back to the utility over communication networks. These same meters can also receive signals from the utility with information on dynamic power pricing and incentives for reducing load during peak periods. *Building and Home Area Networks* can sense and control "smart" home appliances like washing machines, air conditioning units and electric vehicles' charging to balance convenience against energy efficiency. Microgrids push such instrumentation and control to encompass large institutional campuses with co-generation, with the aim of reducing energy costs and encouraging sustainable practices.

Digital Object Indentifier 10.1109/MCSE.2013.39 1070-9924/$26.00 2013 IEEE

As a consequence, the modern power grid is transforming into a ***Cyber-Physical System (CPS)***, where *physical* infrastructure and *computational* cyber-infrastructure must coordinate to ensure an efficient and reliable power grid [2]. However, this transformation is not without challenges. Existing grid operations require a human-in-the-loop to a surprising degree. Renewables such as wind and solar are inherently unreliable and cause the electricity supply to be susceptible to the vagaries of nature. On the demand-side, intelligent appliances, electric vehicles adoption, and rooftop solar panels make the consumer load profile variable. Any demand-supply mismatch causes grid instability unless rapidly rectified. *In the absence of computational and analytics support for automated decisions, the human grid operators are ill equipped to examine and utilize millions of data and control points for managing the dynamism in energy usage patterns.*

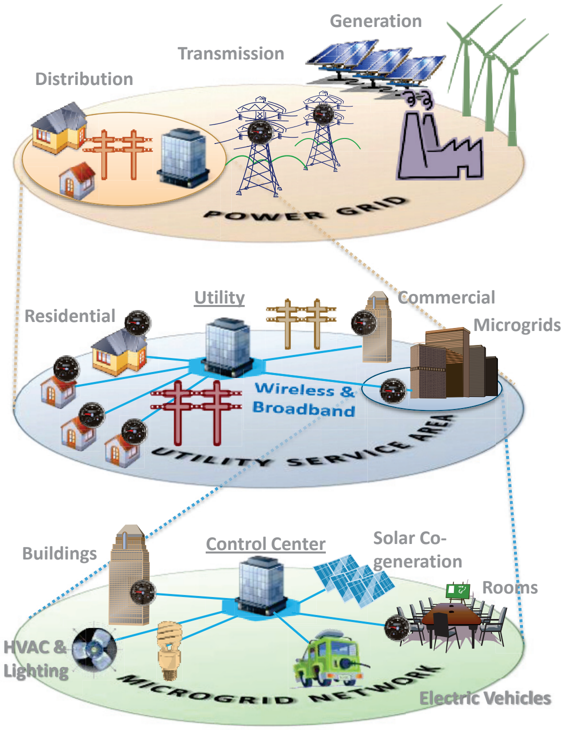

FIGURE 1. The Smart Grid Cyber Physical System. From generation to transmission to distribution, with monitoring capability. Microgrids in the utility's service area enhance sensing and actuation over inter/intranets.

Making the power grid *smart* [3] depends on the ability to wrangle the unprecedented influx of sensing data to draw insights into the system behavior and automate the available controls. This is, interestingly, a **"Big Data" challenge** that requires advanced informatics techniques and cyber-infrastructure. Energy use events streaming from millions of smart meters, sampled every 15 mins, need to be collected and correlated with a consumer's historical profile. Data mining and pattern matching are necessary for online detection of critical situations and their correction with low latency for grid stability. Analytical and computational models can help predict the power supply and demand to take preemptive actions for curtailing demand by notifying consumers. These efforts are *multi-disciplinary*, and require power engineers, data analysts, behavioral psychologists, and microgrid managers to share knowledge for optimal operations, with the active participation of consumers.

In this article, we describe our experiences building a ***Cloud-based software platform*** for data-driven analytics that takes us a step toward the Smart Grid vision. Our efforts are part of the **Los Angeles Smart Grid Project**, sponsored by the US Department of Energy and the Los Angeles Department of Water and Power, underway since 2010. In particular, we address *dynamic demand-response optimization (D^{2}R)*, a unique *challenge application*, where supply-demand mismatch has to be detected, and pre-emptively corrected by initiating demand-side management from consumers [4]. Our software platform supports D^{2}R activities through a *semantic information integration pipeline* to ingest real-time data from sensors and dynamic data sources; a *secure repository* for

Digital Object Indentifier 10.1109/MCSE.2013.39 1070-9924/$26.00 2013 IEEE

researchers and engineers to collaborate and share data and results; *scalable machine-learning models* that are trained over massive historical datasets to predict demand; and a *web portal and mobile app* to visualize current and historical energy consumption patterns. While the platform is being deployed for demonstration in the University of California's (USC) Campus Microgrid, it is designed to scale to the city of Los Angeles using a host of *Cloud technologies*.

2. D^{2}R in the USC Campus Microgrid

The *Los Angeles Smart Grid Project* is a five year research and demonstration project to transform the largest municipal utility in the US into a Smart Grid. Besides deploying smart meters to 50,000 customer premises, it is exploring, developing and demonstrating effective tools and technology for the power grid. Specifically, our group is investigating the informatics possibilities and software challenges in bringing about this advance.

The Los Angeles Department of Water and Power (LADWP) serves 4.1 million consumers and accounts for nearly 1% of the total US electric energy consumption [5]. LADWP has a net generation capacity of 7,100 MW, of which renewable energy will account for 33% by 2020. Over 60% of this renewables mix is from *intermittent sources* like wind and solar. Wind power plants have unpredictable production variability, and are usually not aligned with the daily consumer usage cycle, where peak loads occur mid-day. Solar photovoltaic generation is more aligned with the daily load, but their output can change rapidly with cloud cover, dropping by 50% within a minute [13].

Demand response optimization (DR) is an approach to reduce customers' consumption, in response to a peak energy signal from the utility, by shifting, shaving and shaping electricity load. It contrasts with energy efficiency by soliciting curtailment on-demand only during periods of supply-demand mismatch. Peak loads may be caused by a drop in the supply from renewable generation or an increase in the demand due to, say, a heat-wave in a region. Current grid technology limits DR to static strategies, such as time-of-use pricing and day-ahead notification based on historical averages. But Smart Grid infrastructure offers instantaneous communication capability between the utility and the customer, and automated controls at residences and buildings that enable ***dynamic demand response optimization (D^{2}R)*** for near real-time detection, notification and response.

However, the key to successful D^{2}R is ***intelligent decision making*** on *when, by how much,* and *whom to target* for reliable and accurate curtailment, and this requires advanced data analytics. The benefits of D^{2}R are considerable. It increases the reliability of the grid by using the customers as a virtual energy source during peak periods (negative demand → positive supply); by lowering the peak, it avoids the need to build power plants for standby capacity; it limits the environmental impact since the cleanest energy is to avoid using energy; and it helps integrate renewables by using demand-side management to address supply fluctuations.

While LADWP upgrades their Smart Grid infrastructure, the *USC campus microgrid* serves as a testbed to develop and validate end-to-end D^{2}R technologies. The microgrid encompasses 100 diverse buildings used by a community of over 50,000 students, staff and faculty, and is the largest power private consumer for LADWP. The microgrid is also highly *instrumented*. Every building has a smart meter, and equipment sensors to monitor, say, Heating, Ventilation and Air Conditioning (HVAC) unit's airflow and set point temperature. The microgrid also has *direct controls* over these equipment from the Energy Control Center to initiate direct load curtailment. These use the campus Ethernet as the communication backbone. This comprehensive microgrid ecosystem, including the infrastructure and the energy "consumers", makes the USC campus a *living laboratory* for power engineers, behavioral scientists, analysts and facility managers to study best practices for D^{2}R, with the goal of scaling to LADWP. In particular, it offers a *real-world environment* for us to develop and evaluate our Cloud-based software platform to support D^{2}R research and operations.

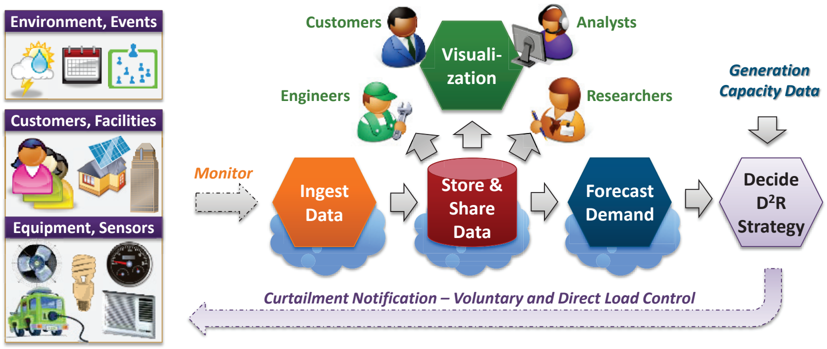

FIGURE 2. D^{2}R Lifecycle in USC Campus Microgrid using our Cloud-based Software Platform. It forms an *observe, orient, decide & act (OODA)* loop.

3. D^{2}R Software Platform on Clouds

The two key ingredients for successful and agile D^{2}R operations are *demand forecasting* and curtailment *strategy selection*, both of which act as a closed loop system on the microgrid. In a data-rich Smart Grid, these decisions are guided by data analytics and mining that must scale with the number of buildings and customers, and the temporal granularity of decision making.

Figure 2 shows the lifecycle of D^{2}R operations within the USC campus microgrid and components of our software platform that support it. Most of these components, shown with a blue cloud backdrop, are hosted on Clouds. We also use different flavors of Clouds, including public and private Clouds, as also Infrastructure as a Service (IaaS) and Platform as a Service (PaaS). The choice of the Cloud flavor for each component is based on their specific needs, which include elastic resource acquisition, platform manageability, data reliability, and scalable programming abstractions. As a consequence, our software platform is seamlessly layered on top of *hybrid Cloud and cluster infrastructure*. Here, we provide an overview of the platform architecture and the D^{2}R lifecycle.

First, data has to be acquired from thousands of equipment sensors and smart meters in campus buildings through the USC energy control system. These include instantaneous data sampled and streamed periodically, as well as historical data aggregated over several years that need to be bulk-loaded. Behavioral research and end-use analysis further require collecting periodic information on ambient weather, organizational structure, building construction, scheduling of facilities, office space assignment, and consumer surveys. These sources themselves may change over time. As a result, an automated data ingest pipeline has to support dynamic data acquisition at variables rates and volumes, and be adaptive to current data sources and operational needs. We have developed the *Floe* continuous dataflow engine, to compose and orchestrate a modular semantic information integration pipeline for the microgrid [6], that uses **private Cloud infrastructure** for *on-demand resource elasticity* (Section 4).

Data acquired by the pipeline has to be stored and shared with different D^{2}R applications. These vary from operational analytics for initiating curtailment, to researchers mining data for

Digital Object Indentifier 10.1109/MCSE.2013.39   1070-9924/$26.00 2013 IEEE

correlations, to consumers visualizing and gaining knowledge on their energy profile. The need for data collaboration has to be balanced against the concerns of data security, since survey data and even fine-grained sensor data may reveal intrusive details about people and their activities. We use **public Cloud storage platforms** that offer a *manageable* and *reliable* data hosting solution for distributed access and *co-location* with compute resources for analytics. Also, we address data sharing and privacy issues posed on untrusted public Clouds through our *Cryptonite* secure data repository, details of which are discussed elsewhere [7].

Data-driven forecasting models are essential for D^{2}R, and there are two key classes of these. Demand forecasting models predict the energy consumed (in KWh) at different spatial and temporal granularities, say, at intervals of 15 mins for individual or aggregate buildings during the next few hours or the day after. Curtailment forecasting models offer predictions on potential energy reduction (negative KWh), at intervals of 15 mins relative to a baseline demand, when using different strategies on buildings or consumers at specific time periods. Once both these predictions are available, D^{2}R strategy selection can be performed as an optimization problem. These data-driven models are trained using historical data on the microgrid behavior, and utilize large dimensions of features that are direct and indirect indicators of energy demand or reduction. In particular, our demand forecasting models use regression tree machine learning and ARIMA[1] time-series to offer high accuracy predictions for D^{2}R. However, model training is computationally costly and data-intensive. Our *OpenPlanet* regression tree learning application [8] uses the Hadoop **MapReduce platform** for performing these analytics, and is optimized to *scale* on **private Cloud infrastructure** (Section 5).

Lastly, there are two types of information dissemination in the microgrid with their individual goals: to *learn* and to *act*. Sharing details of their energy footprint with consumers on campus helps them feel like an active partner in responsible energy usage. We achieve this through a *Smart Grid portal* and a companion *Android mobile app* that provide visualization of current and historical energy use in the campus microgrid. The content for the visualization is served by our data repository. This learning is complemented by "action" notifications that are propagated to individual consumers, using both mailing lists and push notification to the mobile app, during periods of D^{2}R curtailment. The actual customers and buildings to target during a peak load period is determined by the D^{2}R strategy selected using analytics.

4. Dynamic Information Integration Pipeline

Scientific workflows and dataflows provide a convenient abstraction for composing data transformation pipelines and *in silico* experiments using interconnected building-block tasks [9]. These are then run on local workstations or remote cyber-infrastructure using workflow engines that orchestrate the task execution and data exchanges between them. Despite their growing popularity and ease of use, existing workflow engines have limited support for processing continuous data streams with the same flexibility and efficiency as processing files. Support for both bulk data files and dynamic streaming data, which scales to thousands of sensor streams with low latency processing, is essential for composing data acquisition pipelines for the Smart Grid CPS. Further, these pipelines when running in an operational setting are in an "always on" mode. Hence

[1] Auto-Regressive Integrated Moving Average (ARIMA)

Digital Object Indentifier 10.1109/MCSE.2013.39 1070-9924/$26.00 2013 IEEE

any change to the pipeline's composition has to done in-place, without loss of in-flight data. This form of *application dynamism* is, again, not considered by contemporary workflow systems.

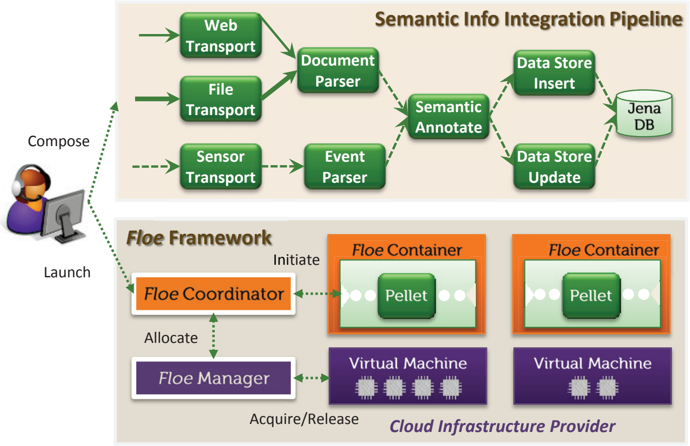

FIGURE 3. *Floe* Framework design on Cloud infrastructure (bottom). Semantic Information Integration Pipeline is composed and executed using it (top).

We have developed the *Floe* dataflow framework that offers novel data abstractions for composing hybrid dataflows that include file and stream processing, and is designed to operate continuously (Figure 3, bottom). Users' dataflows are designed as task graphs whose nodes are the application logic, or *pellets*, and the directed edges represent data transferred between them. Cycles are supported. Floe uses decentralized orchestration. Data is directly exchanged between pellets, without a central coordinator, to reduce the message/file transfer latency. The framework manages pellet-to-pellet message channels and triggers pellet execution. Pellets can iterate over incoming messages or have it pushed to them. Data parallelism is intrinsic; users simply flag pellets to operate sequentially or concurrently on incoming messages.

The Floe *coordinator* instantiates the Floe graph, including pellets and their dataflow wiring, on computational resources allocated by the Floe *resource manager*. The coordinator starts the execution and monitors its status. One of the novel features of Floe is the ability for users to dynamically recompose the dataflow graph at runtime through in-place pellet updates. This allows users to "upgrade" their pellet's application logic without interrupting the continuous dataflow execution. The coordinator silently pauses and resumes the dataflow to perform this.

Floe is designed from the ground-up to operate on IaaS Clouds, besides traditional cluster environments, with support for Cloud providers such as Eucalyptus, Amazon AWS and Windows Azure. The framework can implicitly *scale-out* and *scale-in* on elastic Cloud virtual machines (VMs) based on current processing needs. It also supports multi-core processing, and can *scale-up* and *scale-down* the number of cores used within a VM. The manager acquires and releases VM instances from the Cloud service provider currently configured. Individual pellets have exclusive use of one or more CPU cores on a VM, and pellets are sandboxed from each other using Java 7's `ForkJoinPool`. The framework can dynamically change a pellet's core allocation at runtime to meet user-specified quality of service (QoS) metrics such as message latency time or CPU usage thresholds. For data parallel pellets, the framework can additionally tune the concurrency level by increasing the thread parallelism. These optimizations are *automatic*, and transparent to the user.

The Smart Grid information integration pipeline [6] is composed as a Floe graph (Figure 3, top). The pipeline has four logical stages that handle *data transport* from remote data sources, *data parsing* to interpret the structure and *semantic annotation* to enrich the data context, and *data storage* to persist the information to a data repository. The pipeline is composed modularly, with pellets available for each stage combined together to support different types of data sources. For example, the *sensor transport pellet* periodically pulls events from the building control system, the *document parsing pellet* interprets CSV and Excel files and disaggregates them into individual data items, the *semantic annotation pellet* creates semantic RDF triples[2] that relate sensor events with the buildings containing the sensors and the department housed in the building, and the *SPARQL[3] insert pellet* stores the RDF triples to a semantic database. The pipeline is one homogeneous dataflow and simultaneously supports multiple data sources. So event data that may come from different transport pellets, say real-time sensor events and an FTP file with archived events, will all eventually flow through the same semantic event annotation pellet.

The pipeline is executed by the Floe framework is deployed on a private Eucalyptus IaaS Cloud at USC. The choice of a private Cloud allows collocation with sensor sources and low latency data ingest. The Cloud has 16 nodes with 8 Opeteron CPU Cores connected though gigabit Ethernet. The framework deploys the pipeline on VMs dynamically acquired from Eucalyptus and scales the pellets up and down on a VM to adapt to changing sensor data sampling rates and to bulk data loads that occur every day. The pipeline runs round the clock and is used to load the semantic data repository that support the Smart Grid portal and mobile apps.

Semantic ontologies used in our information architecture help manage the diversity of data sources and types generated and used by multi-disciplinary groups. We integrate multiple ontologies on power and sensor systems, earth and environmental sciences, infrastructure and organizations, and spatio-temporal concepts, along with concepts unique to Smart Grids such as curtailment strategies and prediction models into a single interconnected ontology. Data items from different sources are mapped to concepts in this ontology by the semantic annotation pellets. So an energy consumption event coming from the *D55Watts sensor* is conceptually connected to the *KAP building* in which it is deployed, and further related to the *Civil Engineering department* that uses the building. This offers a powerful and holistic knowledge platform to model system behavior for D^{2}R activities.

5. Scalable Demand Forecasting

Demand forecasting at utilities has typically been done for the entire service area or at the substation level since electrical load data from SCADA[4] monitoring systems were only available at that spatial granularity. These forecasts either use statistical averages over historical demand, which are simple but less accurate, or computational power flow models that have better accuracy but are complex, requiring a thorough understanding of the system interactions. With the advent of smart meters that provide energy use data from millions of consumers at fine temporal resolutions, statistical averages become sensitive to variations and hand-crafting computational models for individual consumers is unsustainable. Figure 4(a) shows the one year aggregate energy use for the USC campus, at 15 min intervals (96 intervals per day). The plot exhibits the temporal variability of energy use, with peaks at midday and base load at nights, and also seasonal highs in late fall and lows during the summer break.

[2] Resource Description Format (RDF), www.w3.org/RDF

[3] SPARQL Protocol and RDF Query Language (SPARQL), www.w3.org/TR/rdf-sparql-query

[4] Supervisory Control And Data Acquisition (SCADA)

Digital Object Indentifier 10.1109/MCSE.2013.39 1070-9924/$26.00 2013 IEEE

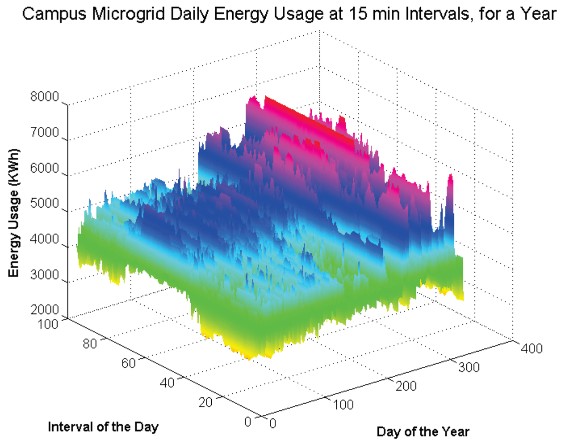

FIGURE 4(a). Surface plot of campus energy use at 15 min intervals in one year. Daily phases & seasonal variations across semesters are seen.

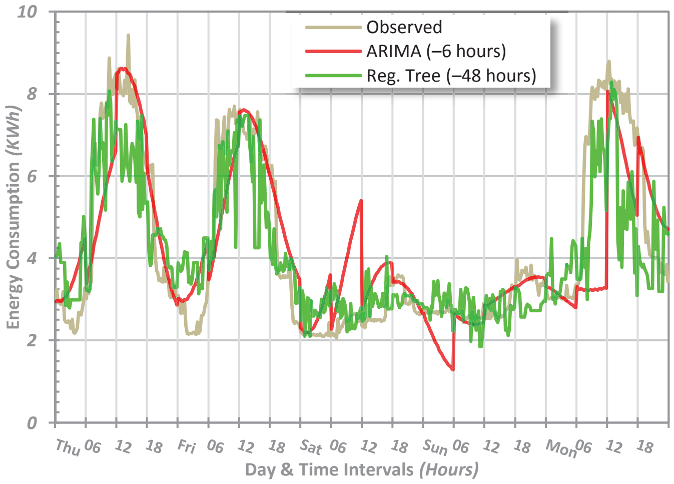

FIGURE 4(b). Energy use observation and forecasts for a campus building for five days. ARIMA forecasts are done 6 hours ahead while Regression Tree is 48 hours ahead.

ARIMA is a popular statistical model for forecasting time-series, such as energy consumption using historical data, and we use it for near term (1-24 hours in advance) predictions. We complement this by using machine-learned models that additionally include features that indirectly influence energy in the campus such as building area, number of occupants, ambient temperature, HVAC setpoint temperature, external weather, and the academic calendar. These models are used for medium (1-7 days in advance) and long term (1-12 months in advance) predictions.

We have adopted *Regression Tree* [11], a machine-learning model that is trained on historical datasets that contain feature vectors corresponding to energy use at 15 min intervals. Upon training, this model gives a decision tree structure where each node is a decision point based on one feature (e.g. *Outside Temperature* > 75'F), and navigating down the tree through different nodes (e.g. *Building Net Area* < 8000, *Day of the Week* ∈ <Mon, Wed>) leads to a leaf which is as a linear regression function that can be evaluated to a numerical KWh value. So given a time period in the future with a set of feature vectors for every 15 min interval it contains, the decision tree can predict an energy demand value for each vector. Figure 4(b) shows energy use predictions made for five days by the regression tree (48 hours in advance) and ARIMA (6 hours in advance) models that we train for a single campus building. When compared against the actual observed demand during the period, we see CV-RMSE errors of 6.0% and 6.7% for the two models. ARIMA is good at following trends but falters when switching between weekdays and weekends. Regression Tree predicts more ahead in time but is less smooth, though its overall error is smaller.

One of the biggest benefits of data-driven models is the convenience of automatically building a model without extensive technical knowledge on the system and keeping them up to date by retraining over new data. This reaffirms the fourth pillar of data-driven science [10]. Further, it allows analysts to run ensemble feature combinations and pick the ones with most impact on the energy demand. This can help scope data collection and provide insights on reducing energy use. These models can also be constructed for different spatial and temporal granularities to trade-off against accuracy, and for different operational needs. We observe that there is no "one size fits all" global model, and instead a collection of models are used for different purposes.

Making predictions using Regression Tree models is quick. But model training is data and compute intensive, and exacerbated by ensemble feature training on a daily or weekly basis. For e.g., 5 years of training data for 100 campus buildings has about 17 million feature vectors, and training one model takes 7 hours on a single quad-core server with 8 GB RAM due to memory pressure. Keeping in mind the need to scale to a city with 1.4 million customers, we use the Hadoop MapReduce platform on a private Cloud infrastructure. *OpenPlanet* is our implementation of the PLANET distributed regression tree algorithm [8]. OpenPlanet is designed as an iterative MapReduce application with up to three concurrent MapReduce jobs per iteration. Each iteration builds one level of the regression tree by splitting the feature vectors at each node at that level into two data partitions based on one feature condition such that it maximizes their variance. The iterations continue till the data partitions reduce to a certain size, after which we switch to the Weka Java machine learning library to complete the remaining subtrees.

OpenPlanet incorporates several optimizations to improve its scalability on elastic Cloud resources. The switch-over point from MapReduce iterations to Weka is important. Weka builds the regression (sub)tree completely in-memory unlike MapReduce which does large scale I/O reads for each level of tree building. So, the more of the tree that is built using Weka, the better the performance. We use this intuition to tune the switch-over condition such that the data partition size is just small enough to fit in the available memory on a machine. The second optimization improves the CPU utilization. MapReduce workers run on a single core and operate on independent blocks of data. The default Hadoop block size creates insufficient blocks to keep all workers busy, and these idle workers limit OpenPlanet's scaling with the number of compute nodes. We address this by tuning the block size to ensure there are enough blocks for all workers. These two optimizations help us achieve a 75% improvement in training time.

OpenPlanet can run on any IaaS Cloud or PaaS Cloud with a Hadoop installation. Due to insufficient resources in our USC private Cloud, we run OpenPlanet for the campus microgrid on the *FutureGrid* cyber-infrastructure, a private IaaS Cloud resource available for academic use. When a data analyst wishes to train a model, our D^{2}R software platform creates a Hadoop environment on FutureGrid using myHadoop, transfers the OpenPlanet binaries and training data from USC to the FutureGrid machines, and initiates the OpenPlanet application. Trained models are moved back to USC and predictions done locally for operations. For commercial use such as LADWP, we envision running OpenPlanet on either an on-site private Cloud or a commercial public Cloud such as Amazon AWS.

6. Discussion & Conclusions

We are now at year three of this five year effort to investigate a software platform for D^{2}R in Smart Grids. Cloud computing has proved essential in our design. However, our experience shows that Cloud abstractions offer different trade-offs for individual components in the architecture. Limitless resources in distributed data centers offered by commercial Cloud providers ease data sharing and on-demand analytics, but come with a real monthly bill. They are worth it when data persistence and sharing across a wide community is important, such as for our Cryptonite data repository that runs on the Microsoft Azure Cloud, or when the computational workloads are variable and do not justify permanent infrastructure. The latter may occur when particular forecast models have been selected for operations in the city-grid, and need to be retrained weekly, say, on Amazon AWS.

Digital Object Indentifier 10.1109/MCSE.2013.39 1070-9924/$26.00 2013 IEEE

Private Clouds offer physical security over data, yet manageability of the hardware remains a concern. But they prove essential when latency for data movement needs to be low, as in the case of our information pipeline. Large datasets on the order of terabytes, expected for city-scale power grids, would also favor private Clouds due to the network and storage cost overhead of public Clouds, as would cases where workloads run often with high resource utilization. During our exploratory research phase using ensemble analytics models, this was the case and FutureGrid offered a suitable platform in comparison with costs on, say, Amazon AWS. However, institutions, such as power utilities, not used to managing large cyber-infrastructure may trade-off the costs of commercial Clouds in return for higher reliability and lower total cost of ownership.

Choices also exist between IaaS and PaaS Clouds. The former offers fine controls over elastic resources, but present a technical challenge (research opportunity?) in designing frameworks that efficiently utilize their capabilities, such as our Floe continuous dataflow engine. IaaS's support for legacy applications through virtual machines is apparent. However, PaaS Clouds truly democratize the building of scalable applications rapidly, as can be seen by the popularity of the Hadoop platform. But not all applications fit the mold of these platforms' programming abstraction, and even if they do, squeezing good performance still requires extensive tuning.

A software architecture on hybrid Clouds, such as ours, can be selective by hand-picking the right abstraction and Cloud provider to balance these trade-offs. However, it does introduce overheads for operating across multiple providers. Tooling to move data and sometimes applications across these providers is necessary, as is provenance tracking and bookkeeping of usage. Our work does not address automating these seamlessly, but the Cloud research community is active in this space.

Our software platform recently won the *IEEE International Scalable Computing Challenge (SCALE) award for 2012*, in recognition of applying scalability principles to the Smart Grid domain with real-world impact. But there are further opportunities for novel Cloud research that Cyber-Physical Systems (CPS) like Smart Grids pose. Application resilience is one such topic that we are pursuing in the context of Floe. CPS requires mission critical applications that must run continuously, reliably, predictably, and at scale. The commodity hardware and multi-tenancy of Clouds means that the infrastructure behavior is non-uniform and suffers from sporadic faults. Programming frameworks that are robust to such dynamism are essential to build and run CPS applications on Clouds. We are also examining additional data analytics and optimization techniques for performing curtailment predictions and strategy selection to close the D^{2}R loop. Scalability and responsiveness will remain a key factor here as we transition into operationalizing the platform within the microgrid.

The impact of our work goes beyond Smart Grids, and demonstrates to the CPS domain – still in its infancy – the value of Cloud computing and data-driven analytics for intelligent and sustainable management of physical systems. In addition, by building the software platform around scalable analytics, we can generalize it to other Big Data domains that exhibit similar characteristics.

ACKNOWLEDGMENTS

This material is based upon work supported by the United States Department of Energy under Award Number DEOE0000192, the Los Angeles Department of Water and Power (LADWP), and the National Science Foundation under Award CCF-1048311. The views and opinions of authors

Digital Object Indentifier 10.1109/MCSE.2013.39 1070-9924/$26.00 2013 IEEE

expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof, the LADWP, nor any of their employees. We thank FutureGrid for resources provided under NSF Award 0910812.

REFERENCES

1. International Energy Outlook Report, *U.S. Energy Information Administration (EIA)*, 2011
2. The New Frontier of Smart Grids, Xinghuo Yu, C. Cecati, T. Dillon, Simões, M.G., *IEEE Industrial Electronics Magazine*, 2011.
3. Putting the 'Smarts' Into the Smart Grid: A Grand Challenge for Artificial Intelligence, Sarvapali D. Ramchurn, Perukrishnen Vytelingum, Alex Rogers, Nicholas R. Jennings, *Communications of the ACM*, Vol. 55, No. 4, April 2012, Pages 86-97
4. Automated Demand Response: The Missing Link in the Electricity Value Chain, A. McKane, I. Rhyne, A. Lekov, L. Thompson, and M.A. Piette, *ACEEE Summer Study on Energy Efficiency in Buildings*, 2008
5. 2012 Power Integrated Resource Plan with Appendices (Draft), *Los Angeles Department of Water and Power*, October 5, 2012.
6. Semantic Information Modeling for Emerging Applications in Smart Grid, Q. Zhou, S. Natarajan, Y. Simmhan and V. Prasanna, in *International Conference on Information Technology : New Generations*, 2012
7. Cryptonite: A Secure and Performant Data Repository on Public Clouds, A. Kumbhare, Y. Simmhan and V. Prasanna, in *IEEE International Cloud Computing Conference*, 2012
8. Scalable Regression Tree Learning on Hadoop using OpenPlanet, W. Yin, Y. Simmhan and V. Prasanna, in *International Workshop on MapReduce and its Applications (MAPREDUCE)*, 2012
9. Workflows and e-Science: An overview of workflow system features and capabilities, E. Deelman, D. Gannon, M. Shields, and I. Taylor, *Future Generation Computer Systems*, Vol. 25, No. 5, May 2009, 528–540.
10. *The Fourth Paradigm: Data-Intensive Scientific Discovery,* Edited by Tony Hey, Stewart Tansley, and Kristin Tolle, Microsoft Research, 2009
11. Improving Energy Use Forecast for Campus Micro-grids using Indirect Indicators, S. Aman, Y. Simmhan and V. K. Prasanna, in *International Workshop on Domain Driven Data Mining*, 2011.
12. Accommodating High Levels of Variable Generation, Special Report, North American Electric Reliability Corporation, April 2009

BIOGRAPHIES

Yogesh Simmhan is a Research Assistant Professor of Electrical Engineering and Associate Director of the Center for Energy Informatics at USC. He leads the research efforts into building a scalable software platform to support Smart Grids and is exploring its broader applicability to cyber-physical systems. His interests are in distributed systems, Cloud Computing, scalable data and metadata management, and dataflow programming abstractions for complex analytics. IEEE and ACM member. Contact him at simmhan@usc.edu.

Saima Aman is a Doctoral student in the Computer Science department at USC. Her interests are in machine learning and data mining as applied to the Smart Grid domain.

Alok Kumbhare is a Doctoral student in the Computer Science department at USC. His interests are in resilient workflow frameworks for mission critical applications on Clouds infrastructure, with application to cyber-physical systems.

Digital Object Indentifier 10.1109/MCSE.2013.39 1070-9924/$26.00 2013 IEEE

Rongyang Liu is a Masters student in the Electrical Engineering department at USC. His interests are in computer architecture and VLSI design.

Sam Stevens is a Masters student in Green Technology at USC. His interests are in portal and mobile applications that can benefit sustainable energy management.

Qunzhi Zhou is a Doctoral student in the Computer Science department at USC. His interests are in semantic web and complex event processing.

Viktor Prasanna is Charles Lee Powell Chair in Engineering, Professor of Electrical Engineering and Professor of Computer Science, and the director of the Center for Energy Informatics at USC. He is the principal investigator leading the demand response optimization efforts of the Smart Grid project at USC. His research interests include High Performance Computing, Parallel and Distributed Systems, Reconfigurable Computing, Cloud Computing and Embedded Systems. Fellow of IEEE, ACM and AAAS.