Semi-Automated Data Center Hotspot Diagnosis

S. McIntosh, J. O. Kephart, J. Lenchner
IBM Thomas J. Watson Research Center, USA
{skranjac, kephart, lenchner}@us.ibm.com

M. Feridun, M. Nidd, A. Tanner
IBM Zurich Research Lab, Switzerland
{fer, mni, axs}@zurich.ibm.com

B. Yang
IBM China Research Lab, China
yangbbo@cn.ibm.com

I. Barabasi
IBM Green Innovations Data Center, USA
barabasi@us.ibm.com

***Abstract*— An increasingly important requirement for energy-efficient data center operation is to diagnose and fix thermal anomalies that sometimes occur due to excessive workload or equipment failures. Today, the task of diagnosing thermal anomalies entails expert but tedious analysis of data collected manually from disparate management systems. Our ultimate goal is to substantially reduce the time, tedium and expertise required to diagnose thermal hotspots by developing a system that generates accurate diagnoses automatically. We describe a substantial step towards this goal: a loosely-coupled, semi-automated thermal diagnosis system that integrates IT and facilities data, uses simple heuristics to highlight the most likely culprits, and provides a graphical interface that enables an administrator to narrow the list further by exploring data correlations. Among the challenges addressed by our solution are coping with heterogeneous data types and data access methods, and detecting and managing erroneous sensor readings.**

Keywords-Systems Management Data Integration; Linked Data; Semantic Web; Energy Management; Green products.

I. Introduction

System administrators have long employed a variety of software tools to monitor key physical and virtual machine parameters, such as CPU and memory utilization, in order to maintain acceptable levels of system performance for end-users, trouble-shoot problems as they arise, and detect and eliminate potential threats to system availability.

With IT energy costs under scrutiny, the traditional separation between IT and facilities management is eroding. The role of the system administrator is expanding to encompass responsibilities traditionally held by facility administrators, such as ensuring that the data center is cooled both sufficiently and efficiently. For example, IT and facilities concerns are intertwined when a thermal hotspot emerges due to inadequate cooling of some region of the data center. The hotspot may occur due to problems with air or water cooling systems, when workload rises, or when more equipment is installed. To support this expanded role, administrators need tools that provide an integrated, graphical view of IT and facilities data. Unfortunately, while some commercial tools are starting to address both concerns, none yet support insightful integration, cross-correlation and analysis of IT and facilities data.

A commercial tool that offers *some* of these required capabilities, and which we use as a reasonable platform on which to develop a semi-automated hotspot diagnosis capability, is Maximo for Energy Optimization (MEO). This product accepts readings from temperature, humidity, and power sensors distributed around the data center, and measures other data such as cooling system status. MEO can overlay any of these data (or interpolations of them) graphically on a view of the physical layout of data center assets. It can also infer from these data additional quantities, such as air cooling efficiency, and display these results as well.

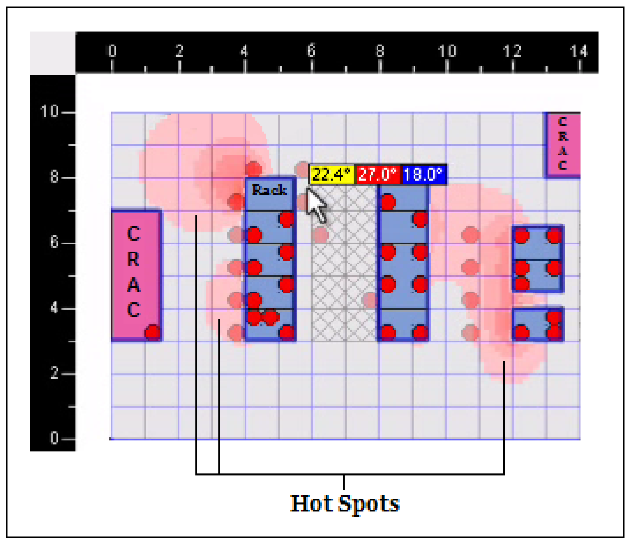

Figure 1. Hot/Cold Map in Maximo for Energy Optimization (MEO).

As an example, Figure 1 shows a portion of the MEO interface with a hot/cold map overlaid on the asset layout of a small data center. The hot/cold map is a more sophisticated version of the traditional temperature map that preferentially highlights anomalously high or low temperatures where they matter most, e.g. near the equipment inlets. To focus the administrator's attention on areas that require remediation, temperatures in excess of local hot thresholds are colored progressively redder, while temperatures which exceed a cold threshold are colored progressively bluer.

While MEO and other visualization tools like it are very useful for detecting hotspots, substantial manual effort may still be required to diagnose and fix hotspots. To diagnose the underlying cause of a threshold violation, the administrator would need to explore a succession of possible explanations, including equipment failures, changes to the cooling controls of nearby CRACs (Computer Room Air Conditioners), recent increases in workload, etc. Unfortunately, since MEO is fundamentally an asset management tool, it has direct access only to the static properties of the data center equipment, not to their dynamic state. For example, to determine whether a workload increase might be the culprit, the administrator must enter the names of dozens of nearby servers into another tool (which might list those servers under a different name) and tediously check their current and historical CPU utilization.

In this paper, we describe a loosely-coupled system that substantially reduces the labor required to diagnose the underlying cause of data center hotspots. The system accomplishes this goal by integrating IT and facilities data, applying simple heuristics to highlight a relatively small number of likely culprits, and providing a graphical interface that enables an administrator to narrow the list further by exploring data correlations. In Section II, we describe the test environment and tools deployed in our research. Section III describes the design and implementation of our semi-automated hotspot diagnosis tool. Section IV describes two sets of experiments: one in which we calibrate the tool, and another in which we test and illustrate the tool in a real data center environment. Section V describes related work, and Section VI concludes with a description of plans for future work.

II. Test Environment

The IBM Green Innovations Data Center (GIDC) is a corporate facility where the latest innovations in energy efficient data center management tools and techniques are deployed. The GIDC actively supports and contributes to research that advances efficient data center management, making them an ideal partner for our project.

A. Physical environment

All of our experiments were conducted in the High Density Room (HDR) of the GIDC. The HDR, shown in Figure 2, consists of about 15 racks that contain a mix of populated blade center chassis and rackable servers. Adjacent racks form rows that split the 48 foot by 20 foot room into alternating hot and cold aisles. There are three Computer Room Air Conditioners (CRACs) that can service the HDR, but typically just one CRAC is in service while the other two are back-ups. Cool air is forced into the cold aisles through conventional perforated floor tiles. Temperature sensors are deployed within racks and throughout the HDR at multiple elevations. Various software tools including MEO render the sensor readings into heat maps and hot/cold maps of the HDR.

The GIDC facility contains several hundred blade servers. For our experiments, we were granted full control over seven blade servers in one rack, spread across three blade center chassis that contained other servers not under our control. This rack is equipped with a sidecar cooling unit that is attached to the side of the rack (possible because our rack is at the end of a row of racks). The sidecar consists of coils through which cold water chilled by the room's chiller flows. It is equipped with a valve for localized control of cold water flow.

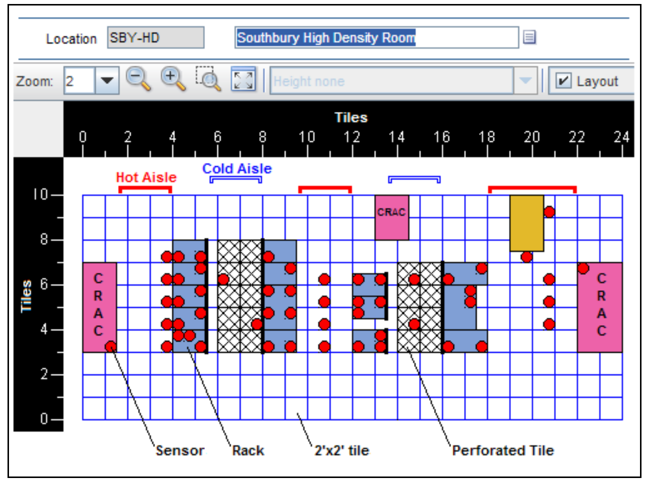

Figure 2. Southbury GIDC High Density Room.

B. Data collection tools

In the HDR, we deployed a variety of commercially available tools to monitor and archive facility information such as temperature sensor readings. We also used system management tools to monitor and archive system parameters for the seven blade servers under our control. Specifically, for each server, we monitored and archived CPU utilization, disk IO, network IO, and memory utilization. The lower part of Figure 3 depicts a high-level view of tools in our tool chain. These include: *IBM Systems Director AEM™ and AutomatedLogic WebCTRL©* to collect sensor readings; *IBM Maximo® for Energy Optimization (MEO) to display* asset, heat, and hot/cold maps; *IBM Tivoli® Monitoring for Energy Management (ITMfEM) to monitor* physical and virtual computing systems performance and archive data into the *Tivoli Data Warehouse (TDW)*; *IBM Tivoli® Application Dependency Discovery Manager (TADDM) to discover* physical resources, applications, and relationships between discovered entities; and *IBM eServer BladeCenter Management Module (Bladecenter)*, to manage the configuration and performance of the blade servers in a chassis.

Using these tools, we collected a rich set of data that characterized both room and server states that existed before, during, and after each of the experiments we conducted. The data for both temperature readings (facility data) and server information (systems data) was collected periodically and stored to TDW, the data warehouse. Data was stored in both raw and summarized form. Raw data are collected and saved every five minutes; summarized data are raw data averaged hourly, daily, weekly and monthly.

III. Semi-Automated Hotspot Diagnosis

In brief, our semi-automated hotspot diagnosis system consists of the following basic steps:

1. MEO detects a hot spot, and displays it graphically.
2. User clicks on hot spot, launching an analytics component that queries a data fusion component that we call *Fusio* about recent and historical values of relevant variables.
3. Fusio returns requested time series data to analytics

component.

4. Analytics component analyzes time series to suggest most likely causes of the hotspot (with rough estimates of temperature and confidence) via a diagnostics interface.
5. Optionally, the user may launch from the diagnostics interface into a hot/cold browser interface that issues queries to Fusio to help explore correlations between temperature and sensor readings associated with the likely cause(s) of the hotspot.

A. Analytics component

The current version of our semi-automated hotspot diagnosis system uses a simplistic analytics component that assumes that hotspot causes tend to be associated with recent changes in one or more operating conditions in the vicinity of the hotspot.

From the point of view of the administrator, the diagnosis is initiated by a simple click on the hot spot shown in a hot/cold map, such as the one in Figure 4. In response to this user request, the diagnostics engine performs its analysis and presents the operator with a list of potential hot spot causes in the format shown in Figure 5.

Using the example in Figure 4, the operator may click on the location indicated by the white arrow. The asset management tool uses the location where the operator clicked as a reference point around which it looks for servers within ten feet. For this calculation it does not use simple Euclidean distance, but rather Euclidean distance around obstacles, since equipment tends to act somewhat like a barrier to heat transfer. Moreover, we are interested only when server air intakes are within the designated region, not just, e.g., their exhausts. This distance from the click point to all nearby server air inlets is computed using the wavefront propagation algorithm of Hershberger and Suri [5].

If the click occurs at the position of the white arrow, thirty-eight servers are identified. The servers' asset numbers are passed to the diagnosis engine, which queries Fusio to identify the subset of servers running at high utilization. The analytics system selects those servers having high utilization coincident with the rise in temperature. This aligns system data with facility data to arrive at a focused set of servers to analyze. The temperature contribution of the servers is computed as

$$\sum_{i=0}^{MaxServers} CPU_i * \beta_{CpuHigh}$$

where CPU_i is the current CPU utilization for server i; *MaxServers* is the number of high utilization servers within a ten-foot radius of a reference point within the hot spot area; and $\beta_{CpuHigh}$ is estimated from calibration experiments to be 0.06˚C.

For the scenario depicted in the Figure 4 map, the diagnosis engine computed the contribution of high CPU utilization to be 0.4˚C. Since the temperature at the hot spot is 27.2˚C, and the starting temperature in the area of the hot spot was 22.4˚C (shown in the map at the left of Figure 4), we know we must account for $Temp_{now} - Temp_{ambient}$, or 4.8˚C. Subtracting the contribution of high CPU utilization leaves us with an unaccounted 4.4˚C, therefore the diagnosis system determines that one or more additional causes are contributing to the hotspot. The system then attempts to identify additional potential causes through data analysis or inference.

Through the calibration experiments (see Section IV.A), we found that the temperature contribution from covering perforated tiles (to block cool air flow) $\beta_{Tile\_Blocked} \approx 1.6$˚C. We also found that the temperature contribution from turning off cold water to the sidecar $\beta_{NoSidecarWater} \approx 3.4$˚C.

Unfortunately, these causes register no time-stamped, machine-readable state, so we are not able to mine data by querying Fusio. The state of the cold water valve can be determined through visual inspection only; the same is true of tile blockage. In this case, the diagnosis system uses inference to determine that tile blockage and/or lack of cold water to the sidecar are additional contributors to the hot spot. Neither $\beta_{Tile\_Blocked}$ nor $\beta_{NoSidecarWater}$ alone account for the 4.4˚C, however, the diagnosis system prevents the operator from taking unnecessary, and ultimately ineffective, remedial action. More importantly, the resulting diagnosis identifies these additional causes as culprits at the outset.

The hot spot pictured in Figure 4 was, in fact, created by pegging seven servers at 100% CPU utilization, while at the same time stopping the flow of cold water to the sidecar. The screenshot was taken when the temperature in the hot spot reached steady state.

B. Data Source Integration with Fusio

Fusio uses a combination of strict and loose coupling to integrate data across a diversity of management systems, making them available to diagnostic and predictive analysis tools or to data visualization tools. It bridges across different naming models and their specific naming schemes, and standardizes the format in which data is provided to the tools, thus obviating the need to understand and use a diverse set of interfaces. For example, given an asset identifier (asset number), the Fusio system can map it to the virtual machines running at the location of the asset, and further provide performance and other management data, obtained from multiple sources.

Figure 3 shows the architecture of the Fusio integration component. Fusio has adaptors specific to the different management systems (shown in the lower part), accessing them via their respective supported APIs. Each adaptor maps the underlying data and metadata to a common format in the form of subject-predicate-object triples specified in the language of the Semantic Web, namely the Resource Description Framework (RDF) [3].

The *Link Information Generator* connects the information for a specific entity from the different sources, e.g., to associate an asset with its physical coordinates as seen by MEO with the corresponding BladeCenter information such as containment in chassis, and with network information such as the fully qualified domain name (FQDN) from TADDM. This *Link Information* allows the component as a whole to serve as an information broker for requests that can be received via an *HTTP Interface* for the most flexible and open access.

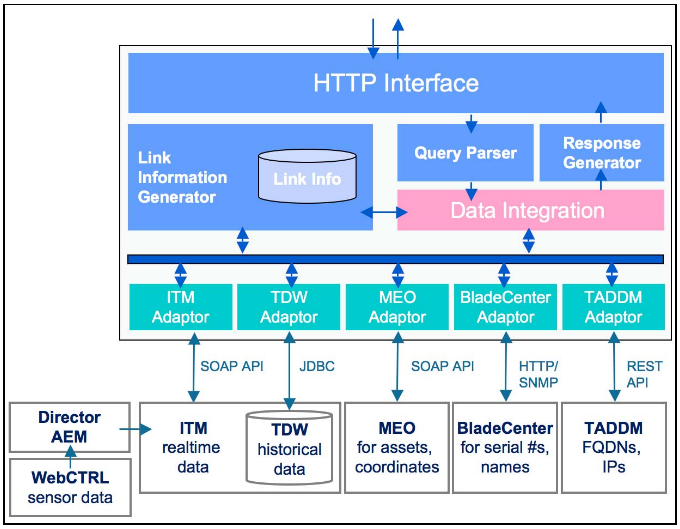

Figure 3. Hot Spot Diagnosis Data Sources and Fusio Architecture

The Fusio query interface supports queries from either the analytics component or the hot/cold interface. The requests specify a desired sensor value or time with given asset numbers, a time interval and a set of requested data categories.

Queries are received in XML format and parsed in the *Query Parser* to fetch the needed data from the sources via the *Data Integration* logic. Responses, e.g., time series of temperature or CPU utilization readings for given assets, will be formatted as XML by the *Response Generator* and returned via HTTP. Additional interfaces can be implemented easily, e.g., the *Link Info* is also made available in the form of Semantic Web Linked Data [4] and consumed in this form by the HC Dashboard (see Section IV.B). More information about Fusio can be found in [1][2].

One challenge addressed by Fusio is linking data that is described differently in different tools. We found that sometimes we had to leverage data from sources that we did not initially expect to use. The best example of this is our discovery that no obvious link existed between an asset known to the asset tracking (facility) tool and data related to that asset but contained in the IT monitoring (system) tool. To resolve this, we used information stored in the blade center management system, which enabled us to establish a link between an asset number (in the facility tool) and an IP address (in the system tool). This provided us with a means to automatically map both physical and virtual machines to physical locations within the data center – a task that is for the most part currently performed by human administrators.

With the data now within reach, we developed a set of standardized, but flexible, queries, e.g., a query to fetch hourly CPU utilization for a specified time interval for a server with a given asset number. The response will in turn provide a time series for the CPU utilization data, as well as some descriptive information about the server.

IV. Example Deployment at GIDC

In this section, we describe first how we calibrated our tool at the GIDC data center by deliberately perturbing operations in various ways, and measuring the responsiveness of the temperature in each case. Then, we detail how, after we subsequently induced a hotspot, we used our tool to infer its likely causes.

A. Calibration Experiments

We performed a series of calibration experiments to gain an understanding of typical temperature responses in the HDR to various stimuli applied singly and in combination. The experiments involved manipulation of server utilization levels, cooling, and air flow with the goal of increasing temperature readings in the cold aisle such that they would exceed the prevailing hot threshold. The hot/cold map, pictured in Figure 4, enabled us to visually verify that a threshold violation had occurred because a red-colored area would appear on the map in the area of the threshold violation.

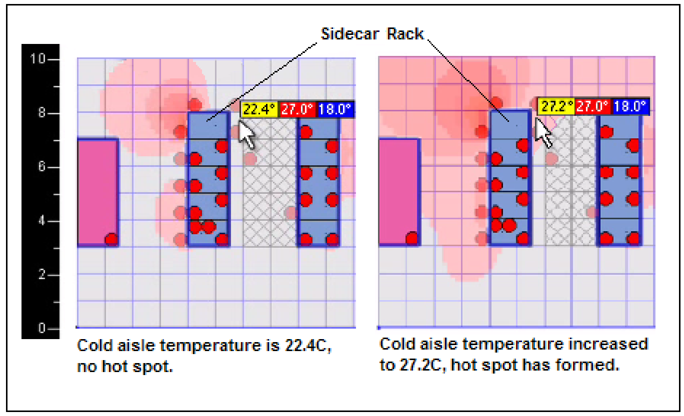

Figure 4. Cold aisle before and after hot spot formed.

Elevated temperatures in the cold aisle are of particular concern to administrators. The cold aisle contains perforated tiles in the flooring, through which cold air flows into the server intake. The temperature of the cool air rises as it passes through the server, and is then exhausted out the back, into the hot aisle. Data centers are commonly organized to minimize the undesirable mixing of hot air with cold air.

In our first experiment, we explored the impact of running a workload generator specifically designed to run the CPU at 100% on each of the seven servers under our control. With the servers at idle, the initial temperature near the rack inlet was 22.4˚C. After turning on the workload generator, we observed a short ramp up of temperature, followed by a brief dip below the original temperature, and then a gradual rise to a steady state temperature of 22.8˚C. (Some exploration revealed that the temporary temperature dip was due to server fans turning on in response to the initial surge in power.) Since all seven servers were identical, and ran identical workload, we inferred that the β value (sensitivity) of the temperature near the inlet of the rack is approximately 0.06˚C for each server running at 100% CPU utilization.

In a subsequent experiment, we measured the impact of turning off the flow of cold water to the sidecar attached to our rack. By turning off the valve through which cold water is directed into the unit, we raised the temperature in the cold aisle by 3.4˚C. In other words, the β value associated with the on/off state of the sidecar is about 3.4˚C.

Next, we experimented with creating a hot spot by combining techniques. We found that by covering two perforated tiles nearest our rack, together with removing rack blanks, we could force a hotspot to form in the cold aisle. A rack blank is used to maintain the vertical division of the inside of a rack. One or more rack blanks are placed next to each other to fill vacant parts of the rack so as to enforce the hot and cold air separation. When the rack blanks are removed, hot and cold air mixes freely within the rack, reducing cooling efficiency within the rack by hindering the removal of hot air to the exhaust side of the rack. When we removed the rack blanks in our rack, and covered two nearby perforated tiles to block all cool air flow from those tiles, the interpolated temperature in the cold aisle increased by 4.2˚C, exceeding the hot threshold and creating a visible hot spot on the hot/cold map.

These experiments show that cold aisle temperature increases appreciably as a result of a physical problem with data center cooling, and the temperature response to high CPU utilization is much smaller.

B. *Test Experiment: Using our Semi-Automated Diagnostics*

Having calibrated the β values, we then tested our hotspot diagnosis system by again creating hotspot conditions in the HDR and seeing whether our system could distinguish among several possible causes, which might be selected from among CPU utilization on one or more blade servers, the sidecar chilled water supply being turned off, one or more perforated tiles being covered, the chiller being broken or turned off, and the dominant CRAC unit being turned off. For the case illustrated here, we deliberately created a hotspot by turning on the workload generator on all seven servers and turning off the cold water flow to the sidecar. After approximately 30 minutes, a hotspot had crept into the cold aisle, as shown in the right-hand portion of Figure 4, which shows a hotspot temperature differential of 4.8˚C.

The semi-automated diagnosis then proceeded as follows. Referring to the example in Figure 4, we clicked on the cold-aisle hotspot, indicated by the white arrow. This action invoked the analytics component, which operated as described in Section III. First, all candidate causes within a ten-foot distance (as measured around obstacles) were identified; this included 38 servers, of which 7 were the ones controlled by us.

The time series returned by Fusio showed that the CPU utilizations of the seven servers under our control had risen to 100%, so they were each estimated to have contributed about 0.06˚C (or 0.4˚C total) to the temperature rise. Thus, simplistic as it may be, the analytics component successfully honed in on the seven servers for which a rise in utilization was coincident with the rise in temperature, and ignored the other 31. All of these calculations took place in a fraction of a second, after which the Diagnostics GUI popped up. A portion of this GUI is depicted in Figure 5; it shows information pertaining to the seven servers that have been flagged as culprits.

Since the analytics component finds that only 0.4˚C of the observed temperature rise above the baseline could be accounted for by the workload increase on the servers, it explores additional likely causes using the algorithm described in Section III.A. The calibration experiments revealed that the temperature contribution from covering perforated tiles (to block cool air flow) was approximately $\beta_{Tile\_Blocked}$ = 1.6˚C. We also found that the temperature contribution from turning off cold water to the sidecar is approximately $\beta_{NoSidecarWater}$ = 3.4˚C. Unfortunately, at present there are no sensors monitoring the states of tiles, so they are not directly accessible to the analytics component. However, the analytics component is able to reason that one possible setting of SideCar variables that would lead to a substantially positive temperature contribution is SidecarWaterFlowing during the time prior to the hotspot and NoSidecarWaterFlowing at present, so it infers a condition of SideCar = NoSidecarWaterFlowing with a confidence level of "Medium". Using similar logic, it also concludes that a nearby tile blockage is also possible, with medium confidence.

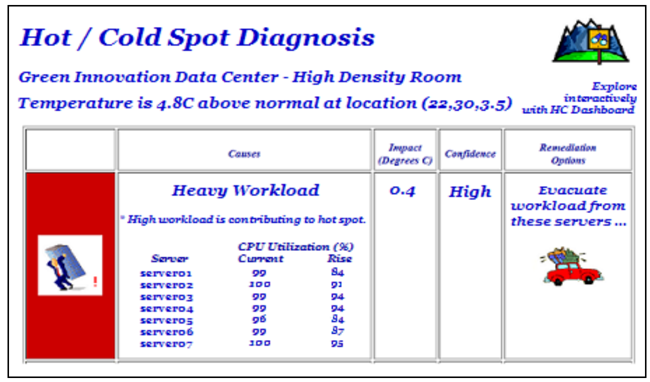

Figure 5. Diagnostics GUI.

Armed with this information, an administrator can then manually investigate the possible causes (sidecar off and tile blockage). They would find in this case that the first diagnosis is correct and the second is not, and take a remedial action (fixing the sidecar cold water flow) that would eliminate the leading cause of the hotspot.

Our tool also provides the administrator with the option of drilling down into possible causes that have been identified by the analytics component and reported in the Diagnostics GUI. Human operators may well want to explore current state and historical development of the environment. For this purpose, a user interface called the *HC Dashboard* can be launched in context from the Diagnostics GUI. The screenshot in Figure 6 shows the different parts of the HC Dashboard. The left hand panel shows a searchable logical tree composed of a hierarchy of data center racks, BladeCenter chassis, power domains, and servers with virtual machines, as created from the linked data of the various systems management tools.

On the top right hand side, a schematic map of the data center is shown, where the physical position and size of the selected entities from the tree are highlighted.

Lastly, the two graph windows on the lower right hand side are used to show time series data of sensors or machine performance. Data is selected by dragging some of the 'time series' entries in the logical tree on the left to one of the graph windows. The graph windows are zoomable, but they maintain time interval synchronization. The dark bands in the graph windows indicate weekends. Moving the cursor across the data shows the values of the closest data points in the legend. This screenshot shows data from one of our heat experiments. Data for three temperature sensors located in different parts of the data center is graphed in the upper graph window and shows a strong spike. A corresponding high CPU load in one of the blade servers can be seen in the lower graph.

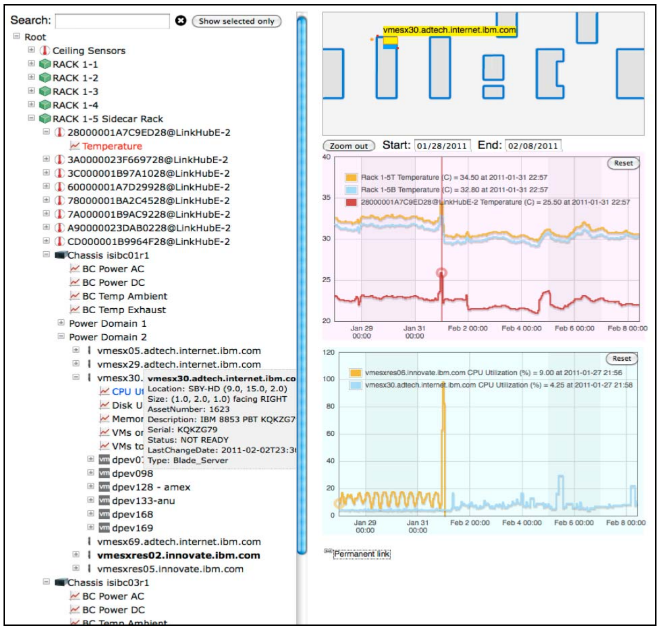

Figure 6. Screenshot of the HC Dashboard.

This shows that data integration and simple analytics can free the system administrator from the tedious tasks of individually extracting data from different tools to identify the cause of the problem.

V. Related Work

Data integration is key to increasing the effectiveness of systems management solutions. Many existing systems provide integration capabilities to support a narrow set of management tasks, often through tight coupling of multiple management systems. There are also platforms or standard models that facilitate integration among different management solutions. One example of such a platform is the IBM Tivoli Integrated Portal (TIP) [6] which allows interaction between applications and enables navigation between components. Standards such as Common Information Model (CIM) [7] provide a model for data integration, as long as mapping from individual models to the common model can be achieved. In general, these solutions are tightly-coupled to a particular solution, and do not provide flexibility through loose coupling of available data.

Similarly, the Tivoli Business Service Manager is another tightly coupled solution. It leverages contextual and relationship information made possible by the Common Data Model standard to provide a central dashboard where state information and metrics can be viewed.

Besides IT assets, cooling systems are another big power consumer. Greenberg et al. show that every 1W of power used to operate a server often requires an additional 0.5-1W for cooling equipment to extract heat at the data center level [8]. Schmidt et al. [9] model Computational Fluid Dynamics simulations of data center air flow and report that careful design and air cooling provisioning is important for sustained operation of a data center. Chaparro et al. [10] present a quantitative model of data center power efficiency and report that reducing supply air temperature provides larger energy savings than running the blowers at high speed. Bash et al. [11] propose an architecture and control mechanism for reducing cooling cost using dynamic thermal management based on a cooling cost model. Sharma et al. [12] report that power consumption for cooling a data center can be reduced significantly by designing the air flow path to prevent mixing of hot and cold air and present non-dimensional parameter based models of the air flow inside aisles. Brooks [13] and Huang et al. [14] investigated more sophisticated Dynamic Thermal Management approaches to degrade performance gracefully by modulating power-intensive chip functions. Finally, Sharma et al. [15] and Moore et al. [16] proposed modulating temperature by migrating workload among servers to thermally balance the load distribution across a data center.

VI. Conclusion

By integrating, mining, and applying analytics to data maintained in IT and facility management systems, we improve the accuracy of hot spot diagnosis, and this leads to a well-targeted selection of appropriate and effective remedial actions.

Our approach leads to efficient remediation of hot spots in data centers by providing the operator with a unified IT and facility view. The operator launches the diagnosis by specifying a hot spot of interest. The diagnostics engine quantifies the contribution of each potential cause to the hot spot, and also identifies clearly the level of confidence assigned to each cause. The resulting diagnosis is targeted at the location of the hot spot, providing the operator with a small, plausible set of diagnoses. Once presented with a diagnosis, the operator may choose to explore further by using the HC Dashboard. This dashboard supports graphical data exploration, enabling pattern search and correlation of time series data. For example, server data can be graphically correlated with temperature readings over the same time period.

In future work, we plan to exploit the integrated IT and facility data to discover methods for hot and cold spot prediction, and suggest remediation options together with our diagnosis. We see potential to expand our system to enable remediation using the data center robot described in [17] and to make actuation capabilities available for launching remedial actions directly from the diagnostics user interface.

VII. Acknowledgment

We are grateful to our colleagues in IBM Research, including Hoi Chan, Rajarshi Das, Doug Dykeman, Hendrik Hamann, Canturk Isci and Liang Liu, and colleagues in IBM CIO Lab Systems including Michael Battaglia, Ed Geraghty, Glendowlyn Howard, Steven Kamalsky, Cristian Manasoiu, and James Thoensen, who supported our research by deploying, configuring, and maintaining tools used to conduct the heat experiments.

References

[1] M. Feridun and A. Tanner, "A search engine for systems management," 12th IEEE/IFIP Network Operations and Management Symposium (NOMS 2010), pp. 697-710, April 2010.

[2] M. Feridun and A. Tanner, "Using linked data for systems management," 12th IEEE/IFIP Network Operations and Management Symposium (NOMS 2010), pp. 926-929, April 2010.

[3] Resource Description Framework, a W3C standard, see http://www.w3.org/RDF/

[4] C. Bizer, T. Heath, and T. Berners-Lee, "Linked Data – the story so far," Int'l J. on Semantic Web and Information Systems (IJSWIS), 2009.

[5] J. Hershberger and S. Suri., "An optimal algorithm for Euclidean shortest paths in the plane." SIAM J. Comput., 28(6):2215-2256, 1999.

[6] IBM Tivoli Integrated Portal (TIP), http://www.ibm.com/developerworks/wikis/display/tivolidoccentral/Tivoli+Integrated+Portal, viewed 4/12/2011.

[7] DMTF Common Information Model (CIM), http://dmtf.org/standards/cim, viewed 4/12/2011.

[8] Greenberg, S., Mills, E., Tschudi, B., Rumsey, P., and Myatt, B., 2006. "Best practices for data centers: Results from benchmarking 22 data centers". In Proceedings of the 2006 ACEEE Summer Study on Energy Efficiency in Buildings.

[9] R. R. Schmidt, E. E. Cruz, and M. K. Iyengar. Challenges of Data Center Thermal Management. IBM Journal of Research and Development, 49(4/5):709–723, 2005.

[10] P. Chaparro, J. Gonzalez, and A. Gonzalez. Thermal-aware Clustered Microarchitectures. In ICCD 2004, pages 48–53.

[11] C. Bash and G. Forman. Cool Job Allocation: Measuring the Power Savings of Placing Jobs at Cooling-efficient Locations in the Data Center. In ATC'07: 2007 USENIX Annual Technical Conference.

[12] R. K. Sharma, C. E. Bash, and R. D. Patel. Dimensionless Parameters For Evaluation Of Thermal Design And Performance Of Large-Scale Data Centers. In 8th ASME/AIAA Joint Thermophysics and Heat Transfer Conference, 2002.

[13] D. Brooks and M. Martonosi, "Dynamic Thermal Management for High-Performance Microprocessors," Proc. 7th Int'l Symp. High-Performance Computer Architecture (HPCA 01), IEEE CS Press, 2001, p. 171.

[14] M. Huang et al.,"A Framework for Dynamic Energy Efficiency and Temperature Management," Proc. 33rd Ann. Int'l Symp. on Microarchitecture, IEEE CS Press, 2000, pp. 202–213.

[15] Ratnesh K. Sharma, Cullen E. Bash, Chandrakant D. Patel, Richard J. Friedrich, Jeffrey S. Chase, "Balance of Power: Dynamic Thermal Management for Internet Data Centers," IEEE Internet Computing, vol. 9, no. 1, pp. 42-49, Jan./Feb. 2005.

[16] Moore, Justin and Chase, Jeff and Ranganathan, Parthasarathy and Sharma, Ratnesh, "Making scheduling "cool": temperature-aware workload placement in data centers", ATEC '05 Proceedings of the annual conference on USENIX Annual Technical Conference, 2005.

[17] C. Mansley, J. Connell, C, Isci, J. Lenchner, J. O. Kephart, S. McIntosh, M. Schappert, "Robotic Mapping and Monitoring of Data Centers," International Conference on Robotics and Automation (ICRA 2011), Proceedings of the International Conference on Robotics and Automation, 2011.