http://doi.org/10.7941/S97H0V LBNL-2001053

Lawrence Berkeley National Laboratory

Accessing Wi-Fi Data for Occupancy Sensing

Marco Pritoni, Bruce Nordman and Mary Ann Piette

Lawrence Berkeley National Laboratory

Energy Technologies Area
March 2017

Disclaimer

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor The Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or The Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or The Regents of the University of California.

Acknowledgments

This work was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Building Technologies Office, of the U.S. Department of Energy under Contract No. DEAC02-05CH11231. We would like to thank, for their great help and assistance, the following (in alphabetic order): Sweta Agarwal, Rich Brown, Daniel Colvin, Stephen Czarnecki, Christian Kohler, Aveek Kumar Das, Joshua Morejohn, Janie Page, Steve Ray, Travis Schick, Michael Smitasin, and Gary Thomas.

Table of Contents

Disclaimer i
Acknowledgments i
Table of Contents ii
Executive Summary 4
1. Background and Introduction 5
2. Roles and processes 7
2.1 Data users 8
2.2 IT department 8
2.3 IRB committee and Security processes 9
3. Network system topology and architecture 10
3.1 Small Building with Single Access Point 10
3.2 Large Building with Autonomous Access Points 11
3.3 Large Building or Group of Buildings with Controller 11
3.4 Large Campus with Multiple Controllers 13
4 Mechanisms to access and store Wi-Fi data 15
4.1 Use SNMP (with custom MIBs) 16
Command-Line Example 19
Output Example 20
4.2 Use Command Line Interface on Controller or AP 20
Command Example 20
Output Example 21
4.3 Run Reports using a Web GUI on controller or AP 21
GUI Example 21
Output Example 22
4.4 Use APIs with other NMS tools 22
API Script Example 23
Output Example 24

4.5 Post-process and store the data 24
Post-Processing 24
Data Storage 25
5. Using the data 26
5.1 Offline applications 26
5.1.1 Benchmarking and performance tracking 26
5.1.2 Measurement and verification 27
5.1.3 Other applications based on baseline models 28
5.2 Online applications (dynamic building controls) 30
5.2.1 Occupancy-based HVAC scheduling (start/stop) 30
5.2.2 Demand-controlled ventilation 34
6. Conclusions and Future Work 38
8. References 39
9. Appendix 44
Cisco 44
Create a report from controller/overseer GUI 44
Get summary from controller command line interface (CLI) 45
From command line: 45
Get summary from AP CLI 46
Query Prime infrastructure API with Python 46
Use SNMP scripts 47
Aruba 48
Query AirWave API for data using Python 48
Use a SMNP script 49
Get summary from CLI 50
Avaya 50
Visualize associated clients using GUI 50
Get summary from CLI 51
Use SNMP scripts 51
Additional Network Management Software (NMS) 52

Executive Summary

A key issue for saving energy in buildings is to assure that delivery of energy services matches building occupancy as closely as possible, to assure that energy is not wasted providing the services to empty rooms or buildings, and to assure that needed services are provided during all times when desired. Accomplishing this to date has been significantly hampered by a lack of inexpensive mechanisms to obtain and use building-wide and more granular data about occupancy. In recent years, the concept of inferential (or implicit) sensing has been proposed and explored to use data from IT systems that already exists in most buildings, to obtain data that are not perfect, but are nearly free to obtain.

Past work by LBNL and others has primarily demonstrated the principle and potential for this, with a primary focus on data from Wi-Fi networks as the best near-term opportunity for inferential sensing from IT networks. This is due to its near-ubiquity, ease of explanation to many audiences, relative uniformness in deployment, low latency of detection, clear ways to mitigate privacy and security, and other benefits (Price et al., 2015). While the potential and value are clear, researchers or building managers who want to obtain the data lack a source of information to understand what data might exist, what devices may have it, and what mechanisms are available to obtain the data. The purpose of this report is to fill that gap.

The report begins with a review of institutional issues in collecting such data, including very real concerns about privacy and IT security. It then describes the various system architectures used in Wi-Fi systems in commercial buildings today, and a variety of mechanisms that can be used to obtain information from them, including examples of how to use them, and sample output. Next is a summary of how to use such data for several different purposes, from retrospective analysis to dynamic building operation, and a conclusion. An appendix provides additional detail on specific mechanisms available from major equipment manufacturers.

The mechanisms specified in this report can be used by building owners and operators to confirm proper operation and uncover any issues or unexpected conditions as a resource for building owners and researchers interesting in utilizing inferential sensing data.

1. Background and Introduction

The number of occupants in a building can have a large impact on its energy use. When present, occupants demand higher levels of service from building systems (lights, computers, appliances), increasing their energy consumption. People also constitute thermal loads that the heating and cooling system must compensate for, to keep the environment comfortable. Hence, occupants also have a passive impact on the energy balance of the building. On the other hand, buildings should reduce their energy use when they are unoccupied. For instance, they should reduce ventilation rates, as suggested by energy codes and standards (CEC 2016, ASHRAE 2016a, 2016b). Despite the importance of occupancy, energy information systems for buildings, traditionally, do not track the occupants' presence and location (Price et al. 2015). The historical reason is that occupancy sensors (e.g. IR, ultrasound or CO_2) were expensive to install and to connect to a central system, especially in an existing building. Recent work in Canada found passive infrared sensors relatively unreliable, with better results coming from use of cameras already installed on PCs and use of keyboard/mouse data from those PCs (Newsham et al, 2016).

If it were readily available, occupancy data could be used in real-time applications, such as building controls, as well as retrospective analysis, to understand past energy use and the factors driving it. In the former case the control system can dynamically adapt static schedules using occupancy data (Balaji et al. 2013, Storey & Montgomery 2014, Henderson 2016, Sensible Building Science 2015). Examples of the latter are measurement and verification (M&V) or fault detection and diagnostic (FDD) (Price et al. 2015). In fact, energy use in a building may increase or decrease after a retrofit and some of the change may be attributed to occupancy. Thus, inclusion of occupancy data in M&V is important for energy analysis. In some instances, we have seen that the energy use does not change when occupancy does. This suggests, for instance, that lights and HVAC equipment may be "on" in some rooms when no one is present. So, the lack of a correlation between energy use and occupancy may indicate opportunities to retrofit controls.

During FY 2015-2016 we identified several potential data sources for inferential occupancy sensing in buildings (also called implicit or virtual sensing), and collected sample data on eight of them from LBNL buildings. Specifically, we acquired data from LBNL's telephone system, its Wi-Fi infrastructure, and several sources from the IP network infrastructure. Since all hardware required is already present in buildings, the implementation cost is close to zero. Inferential sensing extends our traditional sense of occupancy in several dimensions, as shown in Figure 1. A traditional occupancy sensor provides a single yes/no result, for a single location. inferential sensing can extend this for people to give a count, identify individuals, and specify their activity; and can provide visibility across rooms or zones or a whole building rather than just a single location. A key finding from our work was that data from Wi-Fi systems is the best single opportunity for inferential sensing at this time. Among the reasons for this are that it is:

- Easy to understand for people who don't understand network technology
- The most widespread single method available—applicable to nearly any building type
- Simple to implement from an IT perspective

- Served by a modest number of key manufacturers (for the commercial sector at least)
- Notable for a low latency of detecting arrival and departure of devices

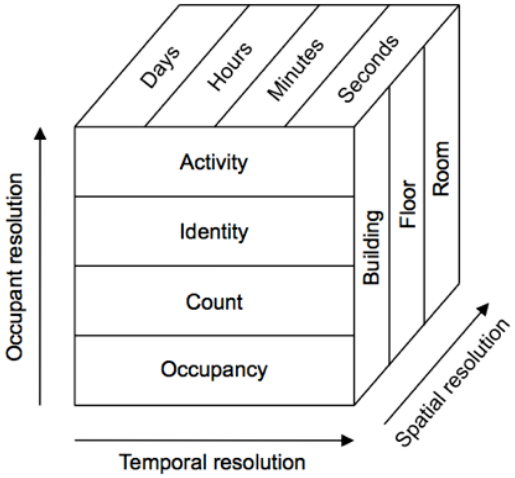


Figure 1: inferential sensing characteristics (source: Melfi et al. 2011)

The underlying principle is that most devices on Wi-Fi systems in commercial buildings are notebook PCs or smartphones that are highly correlated in ownership to individual building occupants, and whose operational pattern in terms of when they are network-connected closely tracks the physical occupancy of the device owner. Thus, the variation of the count of associated devices should be highly correlated to the number of people in the building. While the number of devices per person is almost always zero, one, or two, the figure is likely not usually correlated with the time of day so that the overall pattern of occupancy should be well shown even if the exact number is not specifically tracked. That is, there is likely a fairly constant ratio of occupants to devices, at least for a given building at a point in time.

Our research also identified the need to make the mechanism to extract occupancy traces from Wi-Fi systems simpler, since protocols, and software interfaces are not consistent across manufacturers. That is, while the information is useful and inexpensive to procure, the details of how to do it in practice are not generally known by building owners. This report illustrates how to access this data for the three major network equipment manufacturers of commercial building Wi-Fi systems. We employed a combination of review of manufacturer documentation available online, interviews with key personnel from the manufacturers, and interviews with IT professionals who manage such systems at large installations. The key is that the choice is generally not between Wi-Fi data and better data from dedicated occupancy sensors; as the latter are expensive, the usual choice will be between Wi-Fi data and no data at all.

The report is organized as follows. We start from a review of roles and responsibilities of each stakeholder in a project that involves the IT department and a building/energy team. Then we present the most common network topologies and mechanisms to access Wi-Fi data. Lastly, we show a few applications of the occupancy data, including controls and data analytics.

2. Roles and processes

We begin with a consideration of the personnel and organizational mechanics of research or operational projects of the type that might involve the use of Wi-Fi data. This is significantly based on experience with this topic directly, and other similar research efforts, at LBNL and at UC Davis. Some aspects of this will no doubt be different in process in non-academic commercial buildings, but the underlying issues are present regardless.

Typically, several stakeholders are involved in projects that use Wi-Fi data for energy analytics or controls. These can be broken down into two parts: acquiring the data, and using it; this paper focuses primarily on the former. These projects are usually driven by energy researchers or energy managers who have an application in mind and access to buildings and automation systems but have never previously used Wi-Fi or similar data. The IT department manages the Wi-Fi network and has control over information that flows out of it. In most cases, the IT department has not previously been involved in energy efficiency or management projects. Thus, successfully setting up a system that uses the data includes not just technical issues, but also organizational ones and setting up cooperative systems of a type not usually present previously.

Wi-Fi networks inherently have information that is potentially sensitive (e.g. a device's unique MAC address which can be used to track the device owner), quite unlike traditional occupancy sensors which have no idea who is in a space (or often even how many). For this reason, other entities can get involved to protect privacy and/or security. In academia, an Institutional Review Board (IRB) oversees the enforcement of federal and other regulations to protect the rights and welfare of human subjects involved in research projects. Some organizations have an explicit program to monitor activities that might introduce cybersecurity risks or compromise privacy. The active participation of all these parties is key for a successful project. In the rest of the section we detail the role of each stakeholder and its unique point of view and key responsibilities. Figure 2 shows the three main stakeholders of the process and the different tasks that each performs. The IT department (on the left side) extracts the Wi-Fi data from the network, post-processes it and shares it with the users through a data interface; the users (on the right hand) access the data and run their analyses and applications; on top of the diagram the departments in charge of privacy and security oversee and authorize the process.

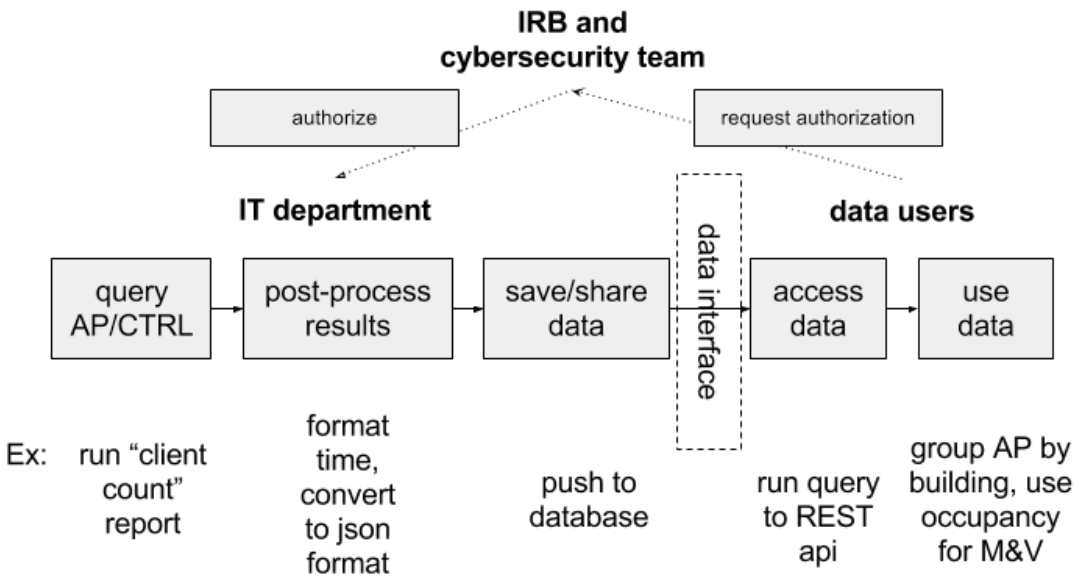


Figure 2: Roles and processes in an energy project using Wi-Fi data

2.1 Data users

Energy researchers as well as energy and facility managers can benefit from more accurate information about building occupancy. A list of applications with examples and a discussion on data required is presented in Section 5.

Although the Wi-Fi network for a building or campus is a promising source of data to estimate occupancy as it does not require additional hardware, those with an energy focus do not have direct access to it; they have to rely on the IT department to provide the data. Since archiving of this type of data is not a priority for network administrators, users also need to build an infrastructure to save and access this data. A shared folder in a local network-accessible location or a site in the cloud are both viable solutions for retrospective analysis, while a database is better suited for real-time applications. We provide some examples and references in Sections 4 and 5, but the detailed implementation of these systems is out of the scope of this document. After the data are extracted from the Wi-Fi system it may be necessary to conduct post-processing, data aggregation, resampling and/or data fusion with other sources of occupancy information. Users need to have access to other building data that they need, which may require collaboration with their facility management personnel.

2.2 IT department

In small companies or organizations, the IT department usually manages the Wi-Fi network. Large campuses typically have a separate branch of the IT department, sometimes called communication resources or network administration, that oversees wired and wireless networks. In some campus environments, multiple separate networks exist that are managed by different groups. Aside from the organizational structure, network managers are responsible for reliability, security, and performance of their Wi-Fi network. They have access to the administrative tools to monitor the network and know how to extract data from the system for their applications. For instance, they periodically analyze the number

of people who connect to different access points to plan system upgrades in crowded areas. They also know the location of each access point and use floor plans to investigate the cause of complaints about slow Wi-Fi or lack of coverage. These data are usually analyzed using spot measurements, stored only for a few months and accessible only using network administrator tools and permissions. Also, these tools allow the IT staff to track the position of a specific person in a building, being able to identify MAC address and authorization credentials of their device(s)[1]. For these and other reasons, the data needs of the IT staff are notably different from those of energy researchers or managers.

To prevent sensitive information from being released to the public, network managers may be reluctant to share any Wi-Fi data with external users. For this reason, the IT department and data users must work closely to create a useful, secure, privacy preserving information interface to expose this data in a way that meets the needs of both parties (Figure 2). Users should carefully define the granularity and frequency of the data they need because different applications may be possible only with certain IT tools. Section 4 defines different methods to extract data from Wi-Fi network systems and the Appendix details these tools for the dominant commercial building access point manufacturers.

2.3 IRB committee and Security processes

In academic campuses, the IRB must review all human subjects research. The major purpose of IRB review is to ensure that the rights and welfare of research subjects are adequately protected. If the data interface between the Wi-Fi network and the researchers inherently preserves privacy, the IRB should consider the project as an exemption or an expedited procedure. However, data users should consider that the needed approvals could require from a few weeks to a few months. Non-academic institutions may not have such a rigid process, but human resources or other departments may want to make sure that the project preserves privacy. For instance, they may require hiding MAC addresses, email addresses and other uniquely identifying personal information, as well as have procedures or requirements for managing and eventually deleting data.

Some organizations, such as LBNL, have an individual or group of individuals tasked with monitoring potential IT security risks. The concern about security increases when users are trying to actively control buildings. The group in charge of IT security and the facility managers should coordinate to make sure the whole network (IT and building control system networks) are secure. The cybersecurity team may also be attentive to privacy, particularly at organizations which lack a formal human subjects process. Principles that are usually applied in managing potentially sensitive data include to only make available to researchers those data that are clearly needed for the activity, and to establish procedures for control of the data. As an example of how to address privacy concerns, at LBNL data were not exported from buildings that had very few occupants (less than 15) as indicated by the laboratory's online phonebook. Fundamentally, at LBNL only device counts were shared with the energy researchers, not any data about the devices themselves.

[1] Insights from interviews with network administrators of 4 campuses and network equipment manufacturers.

3. Network system topology and architecture

In the U.S. and elsewhere, Wi-Fi wireless networks have become a standard building feature, being deployed in most commercial buildings, especially offices and institutional buildings. The hardware needed for this has become a commodity product. The basic building block of each commercial wireless network is the access point (AP). An AP is a device that logically connects wireless client devices to one another and provides access to a local area network and through that, to the Internet (NIST, 2013). Each AP usually connects 10-40 client devices[2] at peak (though may be rated to cover many more) and has a limited range due to ordinary radio signal attenuation and interference from building structural elements. As a result, large buildings need many APs, deployed throughout each floor and area of the building. A client is an end-point device such as a phone, computer, printer, or light bulb using the Wi-Fi network.

When multiple APs are installed in a building, they can be configured as interconnected autonomous APs or they can be managed through a central wireless controller. When used as autonomous units, each AP saves settings and configuration locally and can be configured from the network; however, no central device is required to be active for ongoing AP operation. When a controller is used, configuration information is maintained in the central controller. The controller handles automatic adjustments to RF power, channel selection, authentication, security, and load balancing. Multiple controllers can be set to allow inter-controller roaming to save resources otherwise required to re-associate and re-authenticate (Cisco, a; Aruba a; Avaya, a; Ruckus a). AP vendors developed proprietary operating systems and optimized application protocols between APs or between APs and a controller, so that actual deployments tend to use products from only a single vendor; this is unlike the usual practice in wired networking in which it has been readily feasible to mix equipment from a variety of manufacturers in a building or campus network. APs and controllers usually allow for several different types of interaction to configure and monitor them. Further, additional Network Management Software (NMS), commonly used by network administrators for operation, can be used to provide information about the networked APs and connected clients. Section 4 illustrates several mechanisms to retrieve the data and the Appendix presents vendor-specific details.

The rest of the section describes the four major system architectures in use. Note that the broad classifications used in this Section may vary from vendor to vendor. A more detailed discussion of mechanisms from several large APs manufacturers is included in the Appendix.

3.1 Small Building with Single Access Point

[2] Personal communication with manufacturers.

Small businesses and residences use a single access point that integrates the functions of a router (and often includes an Ethernet switch), as shown if Figure 3. Even though similar concepts apply, vendors frequently sell different hardware and software for these applications in part due to the differing needs for capacity, reliability, and network functionality, as well as ease of use and price considerations. These systems are out of the scope of this document, though they may well have the ability to export device count data.

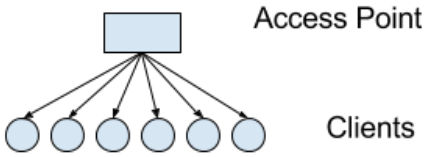


Figure 3: Residential (and small commercial) Single AP Architecture

3.2 Large Building with Autonomous Access Points

Large buildings or campuses, especially those that rely on legacy systems, can have autonomous APs (sometimes called "thick" or "fat"), an architecture fairly common several years ago; see Figure 4. They are connected to each other by a wired network and/or by a wireless mesh network. Autonomous APs require more time for configuration, security and policy settings and firmware updates than do those served by a central controller. In addition, autonomous APs do not provide load balancing, smooth roaming between APs and centralized monitoring.

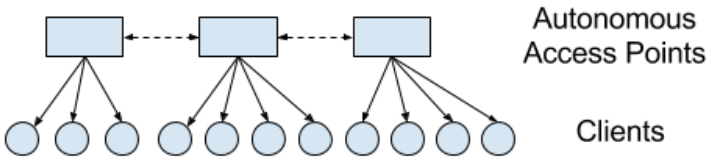


Figure 4: Autonomous AP Architecture

3.3 Large Building or Group of Buildings with Controller

A more modern approach, economically justifiable for mid-size to large networks, uses a controller to coordinate "thin" (or "light") APs as shown in Figure 5. Scalability is greatly improved with this configuration, reducing deployment and management complexities. Centralized configuration is practical necessity when the number of APs is large (i.e. hundreds or more). Central management for all access points allows for automatically shifting clients from one AP to another when one AP begins to fail or is nearing capacity. Central monitoring can produce many types of information useful for the administrator; an example is "heat maps" of estimated radio strength as depicted in Figure 6[3]. These maps are useful for energy purposes because they show the location of the AP in relationship to the building floorplan.

[3] These generally include signal strength degradation from distance as well as that from building structural elements.

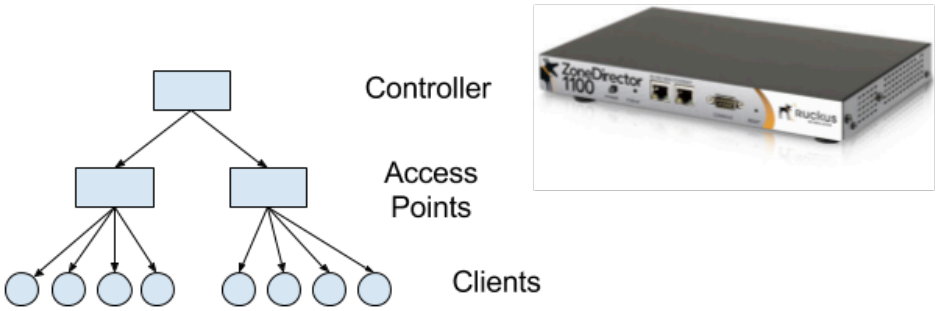


Figure 5: APs and controller Architecture. On the right an example of controller (Ruckus ZoneDirector 1100)

Wireless controllers have been typically rack-mounted hardware components (Figure 5) deployed on-site, but in the last few years new virtualized controllers have emerged. The resulting system, frequently called “controller-less” inherits most of the features of a controller-based architecture, without the need of physical hardware (or rather, no physical hardware on-site). The controller functions are provided by a virtual machine, running on a server in the cloud (Aruba, n.d. b; Cisco, n.d. b; Ruckus, n.d. b) (Figure 7).

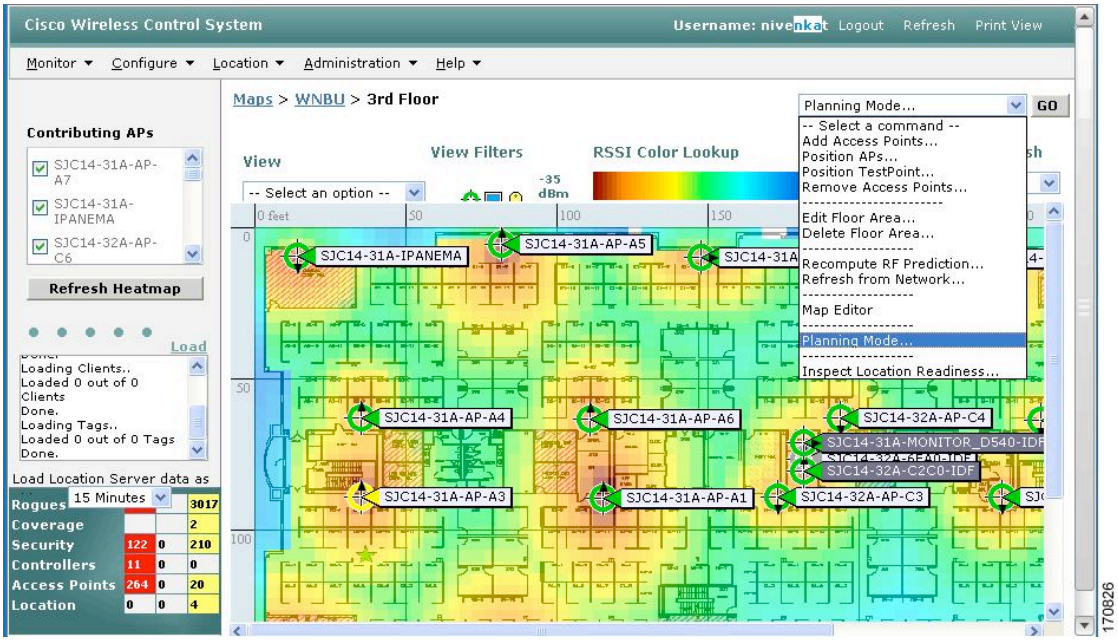


Figure 6: Signal strength map from a Wi-Fi controller system. This tool provides the position of the APs in the building (Cisco, n.d. c)

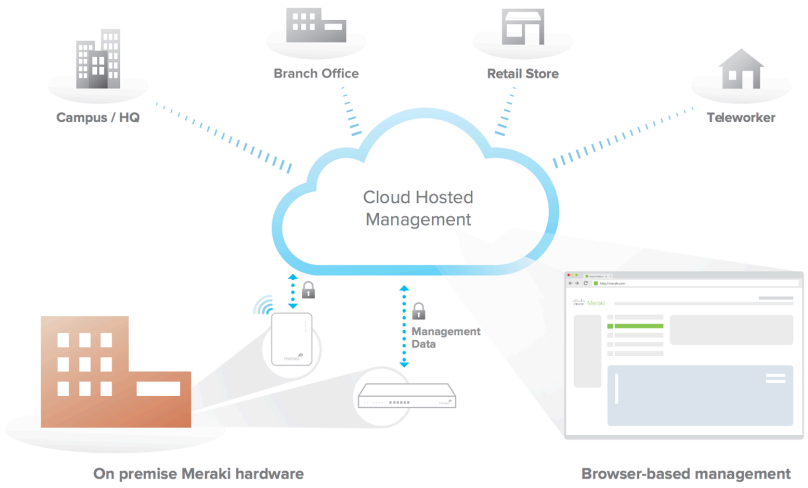


Figure 7: A representation of a cloud-based controller-less system (Cisco, n.c. d)

3.4 Large Campus with Multiple Controllers

To oversee, monitor and diagnose problems in large networks (wired and wireless, in some cases covering thousands of APs, often scattered widely in location), administrators use a variety of software tools called Network Management Software (NMS) (Figure 8). Example offerings from AP and controller manufacturers are Aruba AirWave (Aruba, n.d. c) and Cisco Prime Infrastructure (Cisco, n.d. e).

In addition to software provided by APs and controller manufacturers there are a variety of tools offered by non-AP vendors (e.g. Solarwinds, (Solarwinds, n.d.)) or distributed with open source licenses (e.g. OpenNMS). These provide a variety of features, including real-time analytics, alerts and performance monitoring, and tend to have a web user interface as a primary way of interacting with the data.

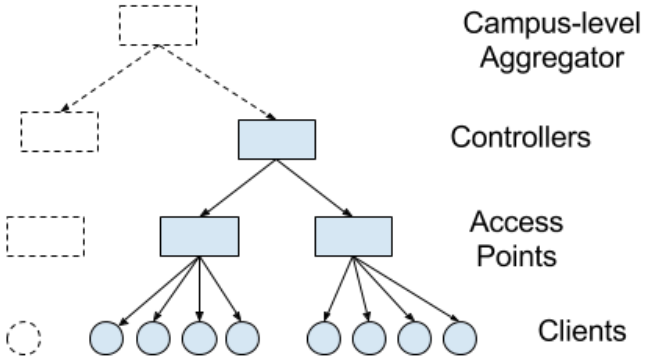


Figure 8: campus-level software aggregator for multiple controllers

Whether AP vendor or independent, these tools usually can display graphs or generate reports based on data accessed via the Simple Network Management Protocol (SNMP), a communication protocol long used to exchange information about network resources. Management of network equipment with SNMP was much more standardized when only wired (usually Ethernet) equipment was used, which enabled independent network management tools to flourish. The use of SNMP to retrieve occupancy data is detailed in Section 4.

Some manufacturers recently started to offer location-based services. Using triangulation (and sometimes Wi-Fi or Bluetooth beacons[4]) they can report the position of a wireless client (e.g. a mobile phone) with an accuracy in the order of a few meters, and sometimes much better (Ruckus, n.d. c; Aruba, n.d. d; Cisco, n.d. f). This new technology development will be briefly described in Section 5. However, such services are usually targeted to high-value applications such as tracking customers in retail stores, and so they come with a significant cost burden. The granularity they provide is much greater than that needed for most energy purposes.

[4] A beacon in this context is a network device that listens for the presence of client devices but does not provide the clients with data communication services as an AP does.

4 Mechanisms to access and store Wi-Fi data

A client (e.g. computer or phone) can be in four states in relationship to a Wi-Fi network:

- Not associated, not authenticated and not probing
- Not authenticated nor associated, but probing
- Authenticated but not associated
- Authenticated and associated
- Authorized (if applicable)

APs periodically broadcast information about the network(s) they support so that new devices in the area will know their availability. This enables devices to automatically rejoin networks they have previously been associated with and for device users to be presented with a list of available networks. Many APs can support more than one network with different privileges.

The process of moving through the five states above starts with the device probing for an AP. When the client decides that it would like to connect to a Wi-Fi network, it starts an authentication process. When the device is authenticated, the client can associate (register) with the AP to gain full access to the network (Figure 9). Many networks also have a password or other registration mechanism for authorization to limit who can access the network, and/or to be able to track network activity by individual.

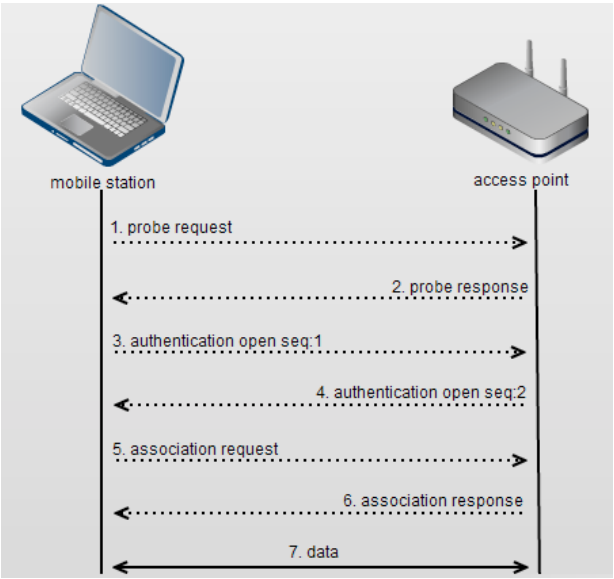


Figure 9: authentication and authorization mechanism for IEEE 802.11 (Intel, 2017)

Management systems are generally most focused on counting devices that are authenticated or associated, as these are ones that are seeking to use the AP for data services. In an office or university environment, most occupants of the building will be those who are there frequently and so will almost certainly have connected to the Wi-Fi network. Only in places like a retail environment would most devices not be associated.

In a network with multiple APs a client can roam from one AP to the next searching for a stronger signal or it can be handed over to a different AP by the Wi-Fi network if the current AP has too many clients or too much traffic. Therefore, association with one AP has a loose relationship with location of the device even though most of the time, a device will be associated with the AP that is physically closest to it.

In this section, we describe general mechanisms to retrieve “Wi-Fi AP client count” data from the Wi-Fi network. Due to lack of standardization of the implementation of these procedures, the practical procedure is different for each vendor. Vendor-specific information is described in the Appendix and applications of this data are presented in Section 5. In section 4.5 we briefly review options to store the extracted data, as some applications require data being accessible in real-time.

The mechanisms are each a combination of the specific IT process used to extract the data and the device/software from which the data are obtained; which devices exist in a particular site depends on the network architecture used. Table 1 shows how these combine and which combinations are known to exist. Access points are most likely to be contacted directly when they are autonomous rather than those closely managed by a controller, and mechanisms will likely differ between the two.

Table 1. Mechanisms for extracting device count data

	SNMP	CLI	GUI report	API
Access Point	A	A	-	-
Controller	A	A	A	C
Management System (NMS)	-	C	A	C

Note: Coding in cells: (A) always exists, (C) can exist, (-) not known to exist

4.1 Use SNMP (with custom MIBs)

The Simple Network Management Protocol (SNMP) provides a "simple" set of operations that creates a standard mechanism to monitor and manage network devices like routers, switches, as well as edge IT equipment such as servers and printers. The range of information that can be monitored with SNMP is wide, from conventional items, like the amount of traffic flowing into an interface, to more esoteric items, such as the air temperature inside a router (RFC 1157, Mauro & Schmidt 2005). SNMP is widely used in network management and many applications build on top of it. SNMP requires a server (manager) application running on a networked machine and one or more clients (agents) running on managed devices. SNMP agents expose information on the managed systems as variables. The variables accessible via SNMP are organized in hierarchies. These hierarchies, and other metadata (such as type and description of the variable), are described by Management Information Bases (MIBs) (RFC 1213).

For wired networking (usually with Ethernet), the mechanisms for network management were developed some time ago, and were well-standardized through the Internet Engineering Task Force (IETF), with a heavy reliance on SNMP. The IETF created standard data models for representing device status and other information in the form of MIB descriptions; for example, two core MIBs used are

called MIB-1 and MIB-2 (RFC 1213). In wired network infrastructure, it is common to have devices of different scales (e.g. low-speed simple switches at edges of networks to complex higher speed routers in the core), from different companies, installed at different times, that all need to interoperate. This required general interoperability. With the wide deployment of Wi-Fi, its deployment pattern has been significantly different from that of wired network infrastructure. In a building or campus, Wi-Fi infrastructure is commonly all from one manufacturer, and acts as a single system with respect to the rest of the network, more akin to a server. As such, it can be managed on a more unitary basis, so the need for cross-vendor interoperability is greatly reduced. As a result, each manufacturer has implemented different protocols and data models. All the major manufacturers of Wi-Fi systems do implement SNMP as well, but use "proprietary" (custom) MIBs. These are documented on-line, so quite useable, but mechanisms for using them therefore need to be customized for each manufacturer.

SNMP itself does not define which variables each system should expose, but it uses an extensible design, where the available information is defined by MIBs. MIBs describe the structure of the management data of a device subsystem; they use a hierarchical namespace containing object identifiers (OID). Each OID identifies a variable that can be read or set via a SNMP request. An example of such hierarchy is presented in Figure 10. For instance, to access the IP value in "MIB-2", one needs to specify the following OID: 1.3.6.1.1.1.4 (Figure 10).

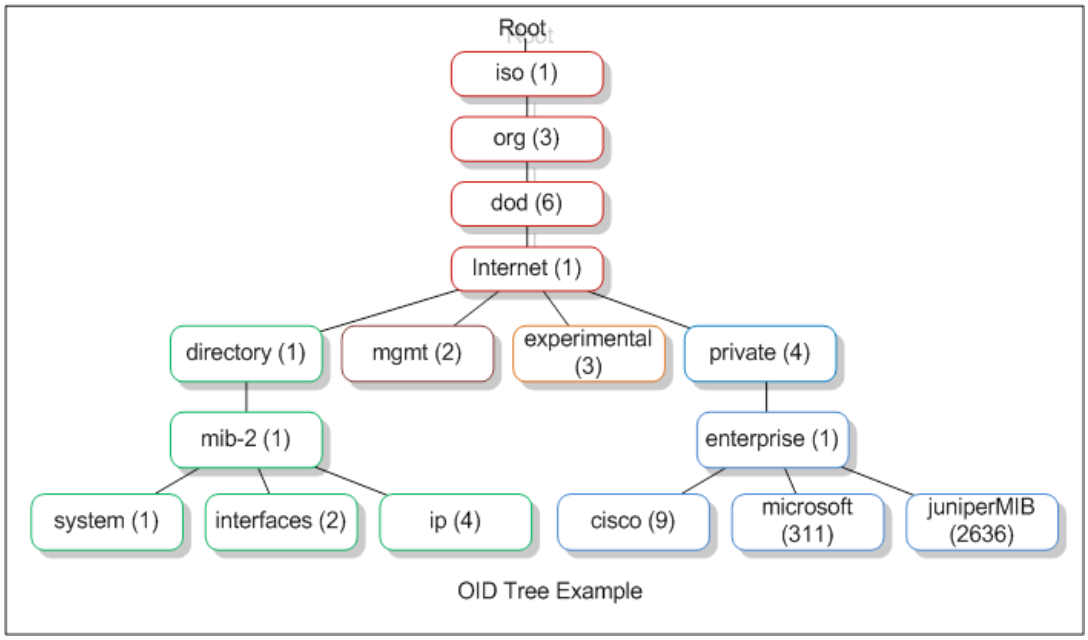


Figure 10: Example of an OID tree structure (Leskiw, 2017)

ile MIB descriptions are human-readable documents, they are meant to be readily interpreted by machines. The process of determining the specific content of an OID number is cumbersome and time consuming. Figure 11 shows an extract of a Cisco custom MIB. To reconstruct the OID, users must walk back the tree structure and extract the numbers at each step (e.g.: `cDot11ParentAddress` (1) -> `cDot11AssociationGlobal` (1) -> ….etc), then rewrite them backwards separated by "." .

```
583  --*********************************************************************
584  --*  dot11 association global parameters
585  --*********************************************************************
586  cDot11ParentAddress OBJECT-TYPE
587          SYNTAX     MacAddress
588          MAX-ACCESS read-only
589          STATUS     current
590          DESCRIPTION
591                  "This is the MAC address of the parent access point
592                  or root bridge for this device.  The value is zero
593                  if this is a root access point or bridge."
594          ::= { cDot11AssociationGlobal 1 }
595
596  cDot11ActiveDevicesTable OBJECT-TYPE
597          SYNTAX     SEQUENCE OF CDot11ActiveDevicesEntry
598          MAX-ACCESS not-accessible
599          STATUS     current
600          DESCRIPTION
601                  "This table contains the list of active devices
602                  currently associated with this device on each of
603                  the IEEE 802.11 interfaces.  This table has a
604                  sparse dependent relationship on the ifTable."
605          ::= { cDot11AssociationGlobal 2 }
606
607  cDot11ActiveDevicesEntry OBJECT-TYPE
608          SYNTAX     CDot11ActiveDevicesEntry
609          MAX-ACCESS not-accessible
```

Figure 11: Extract from CISCO-DOT11-ASSOCIATION-MIB.mib (Cisco, n.d. g)

Several NMS tools and visualizers allow for loading MIBs and displaying them in a more compact format. For example, Figure 12 shows how a free web visualizer (Oidview, n.d.) presents the same Cisco MIB using a folder-like structure.

Statistics for **MIB CISCO-DOT11-ASSOCIATION-MIB:**

Objects: 105	**OIDS: 91**	**Object Groups: 10**
Traps: 0	**Notifications: 0**	**Notification Groups: 0**
Tables: 5	**Tabulars: 56**	**Scalars/Other: 20**

Object Name	Object Identifier
ciscoDot11AssociationMIB	1.3.6.1.4.1.9.9.273
ciscoDot11AssocMIBObjects	1.3.6.1.4.1.9.9.273.1
cDot11AssociationGlobal	1.3.6.1.4.1.9.9.273.1.1
cDot11ParentAddress	1.3.6.1.4.1.9.9.273.1.1.1
cDot11ActiveDevicesTable	1.3.6.1.4.1.9.9.273.1.1.2
cDot11ActiveDevicesEntry	1.3.6.1.4.1.9.9.273.1.1.2.1
cDot11ActiveWirelessClients	1.3.6.1.4.1.9.9.273.1.1.2.1.1
cDot11ActiveBridges	1.3.6.1.4.1.9.9.273.1.1.2.1.2
cDot11ActiveRepeaters	1.3.6.1.4.1.9.9.273.1.1.2.1.3
cDot11AssociationStatsTable	1.3.6.1.4.1.9.9.273.1.1.3
cDot11AssociationStatsEntry	1.3.6.1.4.1.9.9.273.1.1.3.1
cDot11AssStatsAssociated	1.3.6.1.4.1.9.9.273.1.1.3.1.1
cDot11AssStatsAuthenticated	1.3.6.1.4.1.9.9.273.1.1.3.1.2
cDot11AssStatsRoamedIn	1.3.6.1.4.1.9.9.273.1.1.3.1.3
cDot11AssStatsRoamedAway	1.3.6.1.4.1.9.9.273.1.1.3.1.4
cDot11AssStatsDeauthenticated	1.3.6.1.4.1.9.9.273.1.1.3.1.5
cDot11AssStatsDisassociated	1.3.6.1.4.1.9.9.273.1.1.3.1.6

Figure 12: Extract from CISCO-DOT11-ASSOCIATION-MIB.mib presented as folder-like structure (Oidview, n.d. b)

As noted above, the information about clients connected to different APs is stored in vendor-specific custom MIBs. Manufacturers register their custom objects under a section of the tree called "Enterprise" (Figure 10). The common root for this part of the tree is 1.3.6.1.4., therefore we expect each OID to start with that prefix. A list of relevant MIB names and OIDs for each vendor is provided in the Appendix.

Command-Line Example

The following command uses a Linux library, but an equivalent is available for Windows. This command is run on a computer on the same network as the controller, using SNMP to extract data from the controller (note Controller IP address in command), with the data saved on that other computer.

```
snmpwalk -v 2c -c <COMMUNITYSTRING> <CONTROLLERIP> 1.3.6.1.4.1.14823.2.2.1.4.1.2.1.10 |
cut -c48- > /var/$fileOutput
```

A periodic job can be scheduled to run the command on a recurring basis, e.g. every 10 minutes. This example is from an Aruba system but a similar mechanism could be used for systems from other manufacturers.

The “snmpwalk” procedure moves through an SNMP tree structure to extract all data under a specific node. It is a convenient way to extract data as the requesting device does not need to explicitly send requests for each object, or even know the name of each. The “community” string is like a simple (and not very effective) password used in SMNP version 2.

Output Example

The following is sample output from the LBNL Aruba Wi-Fi system. Each of the following lines represents a single device connected to the AP whose name is listed in quotes (this encodes the name of the building and room number where the AP is located). In red and in blue, the IP and MAC addresses of the devices have been anonymized. The results must be aggregated, counting the number of lines in which each AP appears, to get the device count by AP.

```
SNMPv2-SMI:r174":enterprises.14823.2.2.1.4.1.2.1.10.0.4.32.999.999.999.10.0.205.29    =    STRING:
"ap135-50b-2232a-r174"
SNMPv2-SMI::enterprises.14823.2.2.1.4.1.2.1.10.0.11.107.999.999.999.10.0.206.216     =     STRING:
"ap135-50b-2232a-r174"
SNMPv2-SMI::enterprises.14823.2.2.1.4.1.2.1.10.0.14.155.999.999.999.10.0.205.24 = STRING: "ap135-
70a-1112-r174"
SNMPv2-SMI::enterprises.14823.2.2.1.4.1.2.1.10.0.19.239.999.999.999.10.0.211.9 = STRING: "ap135-
90-3111-r174"
SNMPv2-SMI::enterprises.14823.2.2.1.4.1.2.1.10.0.21.153.999.999.999.10.0.194.157     =     STRING:
"ap135-50f-1620b-r174"
SNMPv2-SMI::enterprises.14823.2.2.1.4.1.2.1.10.0.21.153.999.999.999.10.0.199.250     =     STRING:
"ap135-50f-1620b-r174"
SNMPv2-SMI::enterprises.14823.2.2.1.4.1.2.1.10.0.22.207.999.999.999.10.0.196.57 = STRING: "ap135-
70-157-r174"
SNMPv2-SMI::enterprises.14823.2.2.1.4.1.2.1.10.0.22.235.999.999.999.10.0.209.126     =     STRING:
"ap135-90-2002h-
```

4.2 Use Command Line Interface on Controller or AP

Each AP and physical controller runs a proprietary operating system (OS). This system can be accessed by a command line interface (CLI) or a web interface (described in section 4.3).

The network manager can connect to the device (typically via Telnet or SSH[5]), login, and run commands using the CLI. Available commands depend on the privileges of the users and other configuration settings. Each manufacturer uses a different set of commands, but each one has commands to retrieve the number of clients associated with APs or controllers. A periodic job can be scheduled to run the command line on a recurring schedule. Some manufacturers have a scheduler function built into their OS, for the others Unix “cron” or Windows “at” functions can be used to repeat the command on a periodic basis.

Command Example

The following command in Cisco IOS provides a table of associated clients for an AP. To run the command the user must login into an AP iOS. Some iOS have different modes of operation, such as user

[5] Telnet and SSH are two widely used network protocols to login remotely into a computer, and provide command-line access. They are supported by unix/linux, Windows, and MacOS.

execution, privileged execution, interface configuration or global configuration. This command should be available in privileged execution, since it exposes sensitive data (MAC addresses).

```
> show dot11 associations client
```

A periodic job can be scheduled to run the command line on a recurring basis. The simplest implementation is to execute the periodic job on an external server, and capture the output of the command over the network into a file for post-processing on the server. The results must be aggregated, counting all the lines that belongs to an AP.

Output Example

```
SSID                                    [yyyy]                                    :

MAC Address    IP address IPV6 address     Device        Name      Parent          State
xxxx.da1d.1b05 999.20.1.102    ::          unknown       -         self            Assoc
xxxx.2c90.c42f 999.20.1.103    ::          unknown       -         self            Assoc
xxxx.59c0.4ac4 999.20.1.116    ::         unknown       -        self           Assoc
```

4.3 Run Reports using a Web GUI on controller or AP

A web graphical user interface (GUI) is a common feature in modern network controller systems. The manufacturers surveyed all provide a web GUI. The interface is mostly used as an interactive tool to visualize and explore data for all the APs controlled. In addition to visually appealing graphs, these GUIs enable the user to easily generate tabular reports. The reports can be sent to files and dumped into a folder or emailed to a user. We suggest using these reports to extract information from the controller.

GUI Example

Figure 13 shows an example of a GUI interface from a Ruckus system. Reports such as these can be scheduled to be run on a periodic basis.

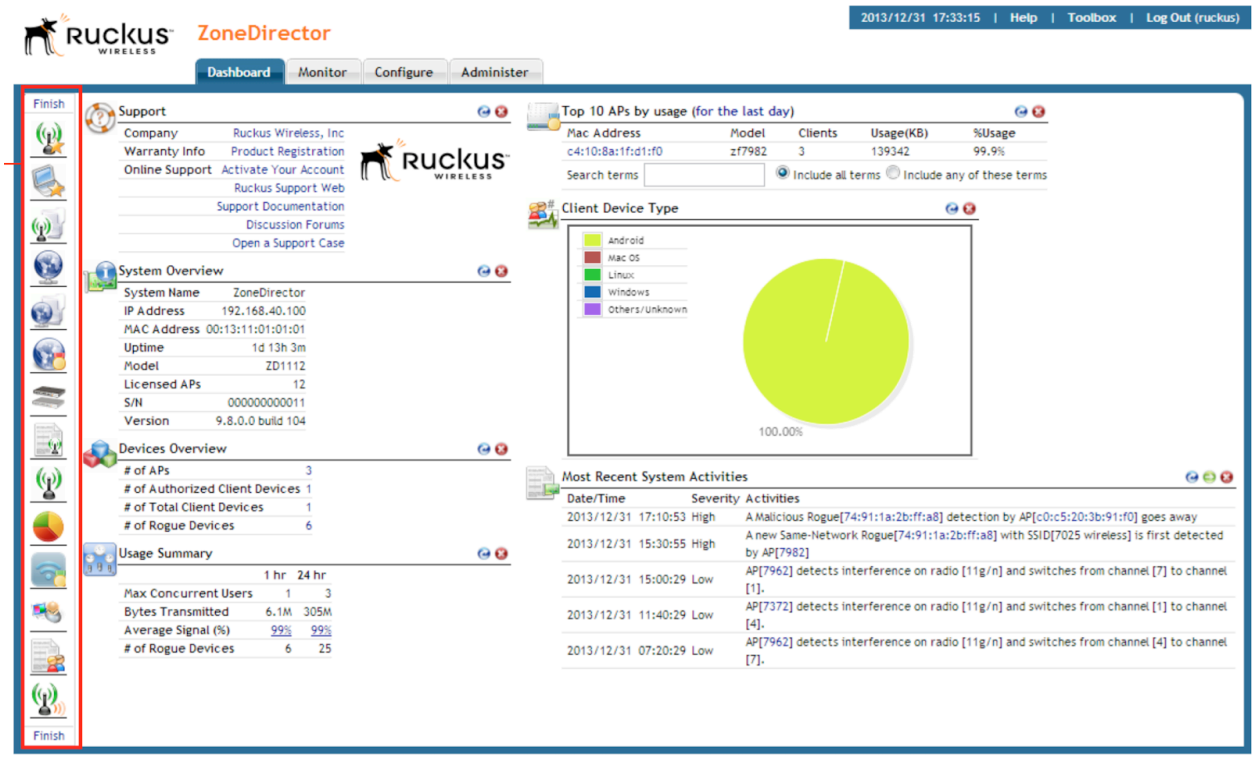


Figure 13: Ruckus ZoneDirector GUI (Ruckus, n.d. d)

Output Example

The text below shows an excerpt of output from the Cisco Prime NMS. The report begins with the total for the requested building, followed by the same time stamps for each individual AP. In this case, the reporting is nominally every 10 minutes but not exactly. In this report "authenticated" refers to the user logging in with an ID and password to the campus network[6].

```
AP Name,Base Radio MAC,Event Time,Associated Client Count,Authenticated Client Count
total,Thu Apr 09 11:24:48 PDT 2015,477,469
total,Thu Apr 09 11:35:18 PDT 2015,500,492
total,Thu Apr 09 11:44:49 PDT 2015,503,492
...
wur100ap,f8:4f:57:3b:4a:80,Thu Apr 09 11:24:48 PDT 2015,6,5
wur100ap,f8:4f:57:3b:4a:80,Thu Apr 09 11:35:18 PDT 2015,4,4
wur100ap,f8:4f:57:3b:4a:80,Thu Apr 09 11:44:49 PDT 2015,2,2
wur100ap,f8:4f:57:3b:4a:80,Thu Apr 09 11:54:53 PDT 2015,3,3
wur100ap,f8:4f:57:3b:4a:80,Thu Apr 09 12:05:46 PDT 2015,6,5
wur100ap,f8:4f:57:3b:4a:80,Thu Apr 09 12:16:22 PDT 2015,5,5
wur100ap,f8:4f:57:3b:4a:80,Thu Apr 09 12:26:34 PDT 2015,7,7
...
```

4.4 Use APIs with other NMS tools

[6] This use of 'authenticated' corresponds to 'authorized' in the description in section 4.0.

Some NMS tools allow for extracting information through application programming interfaces (APIs) which is a more machine-centric mechanism. Common implementations are RESTful[7] APIs that use XML or JSON formats transported over http or https. It is easy to write scripts using high-level programming languages such as Python (with libraries that handle http requests and XML/JSON objects). This approach significantly simplifies post-processing of the data and can lead to more streamlined processes. Campus-wide network management aggregation software (e.g. Aruba AirWave, Cisco Prime Interface) expose such APIs. Figure 14 shows the data flow between Aruba AirWave, controllers and external software.

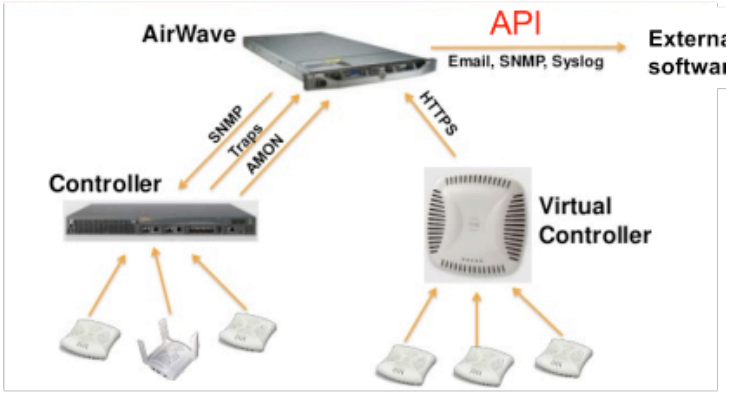


Figure 14: data flow (and protocol used) between Aruba AirWave, virtual and physical controllers and external software (Aruba n.b. e)

API Script Example[8]

```
#!/bin/python
from airwaveapiclient import AirWaveAPIClient
from airwaveapiclient import APList
import hashlib

airwave = AirWaveAPIClient(username='****',password='****',url='https://amp-address')

response = airwave.login()
response = airwave.ap_list()
apList = APList(response.text)
airwave.logout
for ap in apList:
 if 'client_count' in ap.keys():
  clientCount = ap['client_count']
 else:
  clientCount = 0
 print "'%s',%s" % (ap['name'],clientCount)
```

[7] Representational state transfer (REST) or RESTful web services are used to provide interoperability between computer systems on the Internet.

[8] This example of access to Aruba API using a Python script has been provided by one of the network managers interviewed.

A periodic job can be scheduled to run the command line on a recurring basis.

Output Example

mynetwork.noc.ucdavis.edu/wireless/ap-user-count.txt

```
'DV202COUSTEAU-224-714064',7
'HUNT-OS218-735524',3
'KEMPER-OS0064-740242',20
'CHEM-OS131-739966',11
'MILLER-247-712100',0
'UNIVHOUSE-125-729695',5
'VIRIDIAN3-1114-600479',15
'CONF-CENTER-2300-731643',17
'HWC-3006-727613',5
'VIRIDIAN5-1104-600849',2
'EVERSON-OS156-736116',12
'PHY-SCI-LIB-301-700991',9
'THOMPSON-252-712264',1
'TUPPER-1311A-709188',10
'THURMAN-1100-704538',8
'WELLMAN-129-746624',16
'CHEM-ANX-OS1455-740012',2
'LIB-103-726403',28
'HUTCH-176-718495',6
'DATACENTER-1000-651774',11
'NORTH-211-162093',8
'MRAK-14-726581',11
'THOREAU-301B-197151',5
'SHOCKWAVE-LAB-104-548107',1
'RYERSON-309-190854',5
'HART-OS3119-739306',6
'EARTH-PHYS-SCI-1348-723425',5
'DATACENTER-1024-651800',14
'SCI-LEC-123-747095',13
'UNIV-SVCS-270-721743',20
'POTTER-481-732826',3
'CHEM-381-729045',5
'LIB-271-726384',38
'FOOD-SCI-TECH-126-749077',6
'GBSF-5503-740220',11
'THOMPSON-410-712337',1
'ALDER-101-712371',0
'LIB-382-726418',10
'OLSON-141-648508',10
'GILMORE-221-190024',6
'FOOD-SCI-TECH-124-749065',1
'SHREM-1221-601194',3
'FOOD-SCI-TECH-202-166140',0
'MEYER-OS1403-747065',7
'LIB-87C-726334',8
'THOMPSON-221-712257',5
```

Figure 15: An example text file of AP name and device count ('AP', #) resulted from a query to a NMS API

4.5 Post-process and store the data

Post-Processing

In general, data produced by these methods is usually in the form of text files of some regular format (a possible exception may be some API interfaces). Reports are typically intended for a human to read; to be further used in automated procedures, the resulting .csv or .txt files need to be parsed to extract the useful information, possibly aggregate results, and remove any sensitive or unneeded data. These manipulations can be readily done with standard OS shell programming tools.

A convenient approach is to save the first output in a file in a shared folder or send it to a readily accessible location (e.g. to a known URL via https). For some systems, the number of associated clients for each AP is provided directly in tabular form. In this case, the output can be parsed to obtain the desired output (e.g. the count) and hide sensitive information from the users. The output could be pushed to a database, which is especially useful for real-time applications, such as building controls. Data extracted using the methods described above may need to be scraped and cleaned to be in the format desired by the users. IT departments can often accomplish this efficiently through OS scripting or with a standard programming language (e.g. Python), but the users may have to do part or all of the work.

Before using the data, the user team may have to aggregate it in different ways, for instance by summing for an entire building or floor of a building. Users should make sure that each AP is accurately mapped to the building it resides in and to specific locations within the building. In the example, in

Figure 15, mapping APs to buildings is intuitive as the first part of the name is the name of the building. Other conventions for AP naming may not be as clear. Further, they need to make sure all the APs are accounted for, each time the data is captured. If any of the APs are not working or not reporting to the controller, some of the end-use devices may be lost to the system's accounting, creating discontinuity in the data.

Outdoor APs, common in many buildings, require special consideration, since they can frequently count people outside of the building. Even APs that are intended for interior use may pick up some people outside depending on the circumstances the percentage of people connected outside the building can be large if the building is close to a busy location such as a bus stop, outdoor coffee shop, or park. If the application is intended for counting only people inside the building, these APs may need to be excluded or treated differently (e.g. only counting devices that have been present for a sufficient period of time to avoid counting those just passing by).

Data Storage

For offline applications, the data retrieved can be stored in files (typically .txt, .csv, .json or xml), but for online applications such as direct building control, the data are best pushed to a database for fast access. If the user team uses a database for other applications, such as storing building automation system (BAS) data, they can add these new data streams to their existing system. If no data storage is in place, the users have several options including traditional relational databases (e.g., licensed products such as Oracle (Oracle, n.d.) or Microsoft SQL Server (Microsoft, 2017), or open-source alternatives as PostgreSQL (PostgreSQL, 2017), SQLite (SQLite, 2017)), non-relational databases (e.g., MongoDB (MogoDB, 2017)) and time-series databases. If the application requires large amount of Wi-Fi data, time-series databases are preferable (sometimes called "historians"), because they exhibit dramatically better performances in saving and recovering data. Both open-source (e.g., OpenTSDB (OpenTSDB, 2017), sMAP (Dawson-Haggerty, 2010))), and commercial products (e.g., InfluxDB (Influxdata, 2017), Periscope Data (Periscope Data, 2017) are now available. Some IT skills are required to install and setup these systems. Implementation details are out of the scope of this document, but a few examples of the results will be shown in Section 5.

5. Using the data

Occupancy data can be used for two general types of applications: offline (e.g., M&V analysis) and online (e.g., real-time building control).

5.1 Offline applications

5.1.1 Benchmarking and performance tracking

To compare the energy performance of a building to other buildings or to itself in a different period, energy analysts use a variety of normalization methods. This process considers the effects of important energy consumption drivers to compare two buildings more fairly. For example, to access the performance of the HVAC, it would not be fair to compare the heating use of a building in Florida with one located in North Dakota, where the weather is much colder. Typically, only floor area and weather are considered in such normalizations. While it is recognized that occupancy plays an important role in determining energy use, occupancy data are not commonly available in buildings, so it is not included in the analysis.

As discussed above, the count of associated devices for all the APs of a building can be used as proxy for whole building occupancy. Even though the count may not be accurate, there is a good correlation between that count and the number of people in the building (Storey & Montgomery 2014, Henderson 2016). Typical inferential sensing data used for this application has the following characteristics:

- Spatial resolution: whole building (all APs in the building)
- Time resolution: hourly averages (or other frequency that matches the energy data)
- Update frequency: daily updates are usually enough (storage in files is acceptable)

UC Davis Energy conservation office (ECO) has developed a campus-wide educational dashboard for energy data. Figure 16 shows a screenshot of the dashboard. On the right, there is a map of campus with transparent circle of varying size overlaid on each building. The size of each circle is proportional to the energy use intensity (EUI), that is the energy use normalized by square footage. The color of the overlaid circles represents the type of building, i.e., if most space is dedicated to classroom, office, community or lab use. For additional contextual information, the side panel on the left includes more detailed data and average EUI for the building types on the UC Davis campus, which can be used for benchmarking (Salmon et al. 2016). For instance, the building selected uses 424 kBtu/ft^2 of "site" energy, compared to 208 kBtu/ft^2 of the average campus lab. While the comparison does not need weather normalization, because all the buildings in campus are subjected to the same weather, it would be useful to compare energy use normalized by occupancy. Since occupancy sensors are installed only in a fraction of the buildings, UC Davis (UCD), with help of LBNL, is planning on using Wi-Fi data to conduct such normalization.

Figure 16: A screenshot from the UC Davis energy dashboard. (source:http://ceed.ucdavis.edu)

As these graphs are updated periodically and automatically, the Wi-Fi occupancy data needs to be cleaned and checked to identify erroneous data. This step is fundamental as normalizing by zero, for instance, can lead to large errors that will be clearly visible in the graphs. For data storage UC Davis plans on using a time-series database that can reliably store Wi-Fi data for the whole campus. The same data platform is already used to store energy and building automation system data.

5.1.2 Measurement and verification

Historically, measurement and verification (M&V) models have considered building or equipment characteristics and weather patterns, but not occupancy. LBNL has developed an extended statistical M&V model, that includes occupancy obtained through Wi-Fi connections as the predicting variable. The model was tested on several buildings and shown to outperform the regular model in some of them (Price et al. 2015).

The count of associated devices for all the APs of a building is sufficient for this application. Typical inferential sensing data used for this application has the following characteristics:

- Spatial resolution: whole building (all APs in the building)
- Time resolution: hourly averages (or other frequency that matches the energy data)
- Update frequency: daily updates are usually enough (storage in files is acceptable)

An example of this application was shown in Price et al. (Price et al. 2015). A building at LBNL with both energy and Wi-Fi device count data available was selected for testing. To show the impact of occupancy on energy use we selected the end-of-year as a period when we knew occupancy would be low. The idea was to create a statistical model based only on load data from the non-holiday period, and then use

that model to predict the load during the holiday. Results are presented in Figure 17. The black line represents the real load, the blue line is the model that includes Wi-Fi occupancy, while the red line is the model without occupancy. The upper panel shows the model predictions during the training period; the lower panel shows the fit to the prediction period. As is evident in the plots, both models do about equally well at fitting the load during the training period, but the model that includes the Wi-Fi data performs far better during the holiday period. The model that does not use Wi-Fi data over-predicts the total energy used during the holiday period by 23%; the model that uses the Wi-Fi data over-predicts by only 7% (Price et al. 2015).

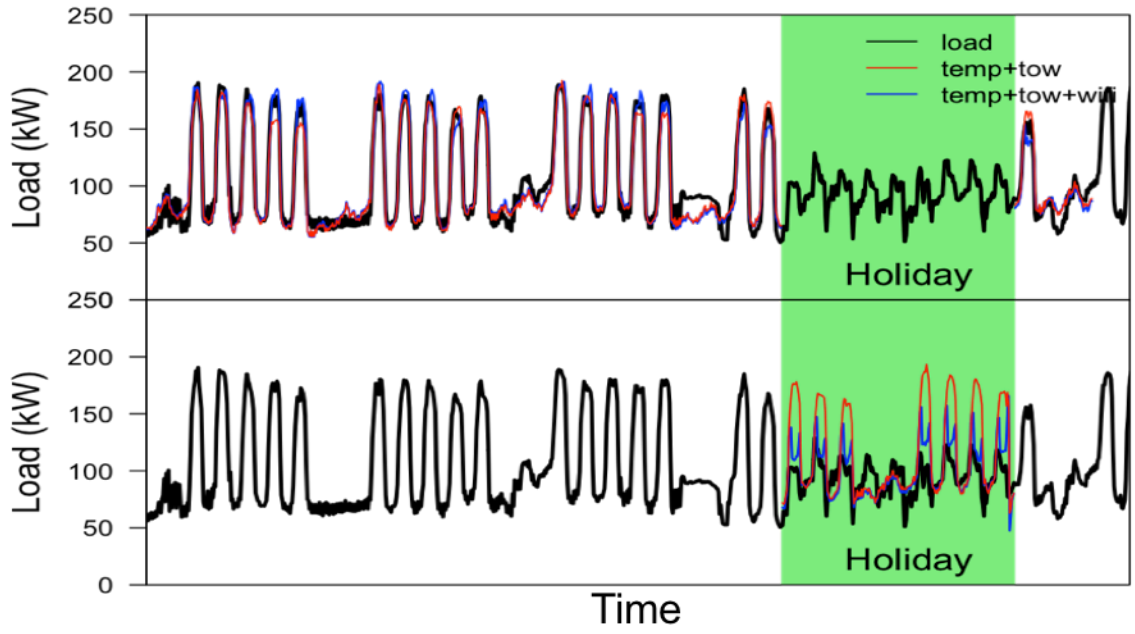


Figure 17: comparison of occupancy-based model (in red) and model without occupancy (in blue). The upper panel represents the training period (non-holiday) and the lower panel represents the validation period (holiday). The blue model overperforms the red one in the validation period. The x-axis covers just over five weeks of time.

Considerations

As in the previous example data cleaning is very important here, as outliers or missing data can seriously compromise the accuracy of the model. We recommend to create a procedure for data cleaning and to inspect the raw data as well as the model fit visually to confirm that the data is reliable. In certain cases, a bad model fit can be a sign of some data quality issue.

5.1.3 Other applications based on baseline models

Baseline models augmented by Wi-Fi occupancy data can be used for other applications too. For instance, they can be used to identify buildings that have little variation in load regardless of changes in occupancy patterns. Since most of the energy in a building is used to provide services to the occupants (with the exception of process loads), a low sensitivity to occupancy may indicate that the building needs to be tuned or could benefit from energy retrofits. Useful strategies to reduce such loads include

occupancy-based lighting, conditioning and ventilation. This approach, together with benchmarking can be used to prioritize building retrofits for a portfolio of buildings, such as a campus.

Another application of baseline models is anomaly detection. These models describe the expected behavior of the building, therefore large variations of actual energy use compared to these models can reveal anomalous operation. An example is provided in Figure 18. The panels show the actual load (black) and associated baseline model (blue) during six weeks in October. The model was trained on data from September of the same year. While the model performs fairly well in the upper panel (RMSE[9] is 1.6 kW), it gets worse in the lower panel (RMSE is 3.2 kW). The model error is caused by a change in the operation of the building. Data in the lower panel shows the effect of the "winter heating mode" where resistance heat in the morning causes a very high spike in the load. This fast heating strategy was not necessary, and the baseline model was able to identify the abnormal behavior. More details about this example are provided in Price et al. (Price et al. 2015).

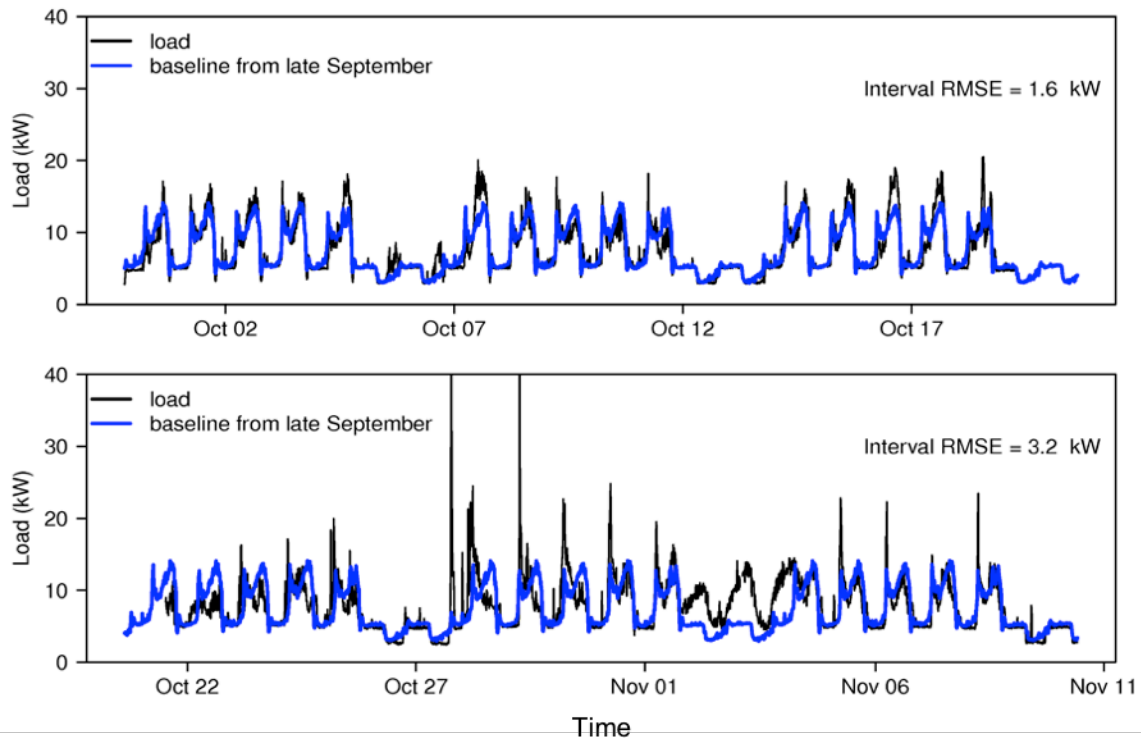


Figure 18: use of baseline models to identify energy anomalies. In the lower panel the inefficient morning resistance heat startup was identified by the model

Baseline models including occupancy data can also be used for energy forecasting. The prediction can inform the planning of generation resources in a grid or microgrid. The forecast can also be useful in case of demand response (DR), as the baseline behavior and the expected response to a DR event is influenced by occupancy.

[9] Root mean squared error, a measurement of the goodness of fit of the model

5.2 Online applications (dynamic building controls)

5.2.1 Occupancy-based HVAC scheduling (start/stop)

The other major use of inferential sensing data for occupancy is in directly controlling building operation. The largest opportunity is HVAC, so that buildings can be run on the basis of actual occupancy, rather than fixed schedules of expected occupancy (Balaji et al. 2013, Storey & Montgomery 2014, Henderson 2016). These applications require data to be delivered in real-time (or at least close to real time). The sampling frequency is likely to be between 1 minute and 20 minutes. Frequent data means that changes can be more quickly reflected in building operation, and so provide a higher quality result, but do present a higher burden on network system resources, and there is a point of diminishing return for higher and higher time granularity. For these applications, the data also needs to be pushed to a more dynamic storage system that can be queried frequently by the control application (e.g.: a time-series database with an API).

The count of associated devices for all the APs of a building is sufficient for this application. Typical inferential sensing data used for this application has the following characteristics:

- Spatial resolution: whole building (all APs in the building)
- Time resolution: 1-20 minutes
- Update frequency: instantaneous (need a database)

In buildings that have a day/night operation (i.e.., most of the buildings), the simplest strategy to take advantage of occupancy information is to dynamically adapt the start and stop of the building based on real occupancy. In fact, buildings are mostly operated using a fixed and conservative (energy intensive) schedule, to cope with the uncertainty of actual building occupancy. A recent project by University of British Columbia (UBC) has experimented with the use of Wi-Fi data to control the start time of the heating system (Henderson 2016). The research team has determined two heating setpoints: a pre-heating setpoint, to avoid a long recovery time from night mode and a comfort heating setpoint, based on a nominal fraction of the building occupants arriving at the building. Preliminary results are shown in Figure 19. The blue line represents the energy use based on actual occupancy data, while the grey line is the energy use with a traditional schedule. The two dashed red lines show the difference in start time. It is clear from the graph that the occupancy-based strategy uses less energy (Henderson 2016).

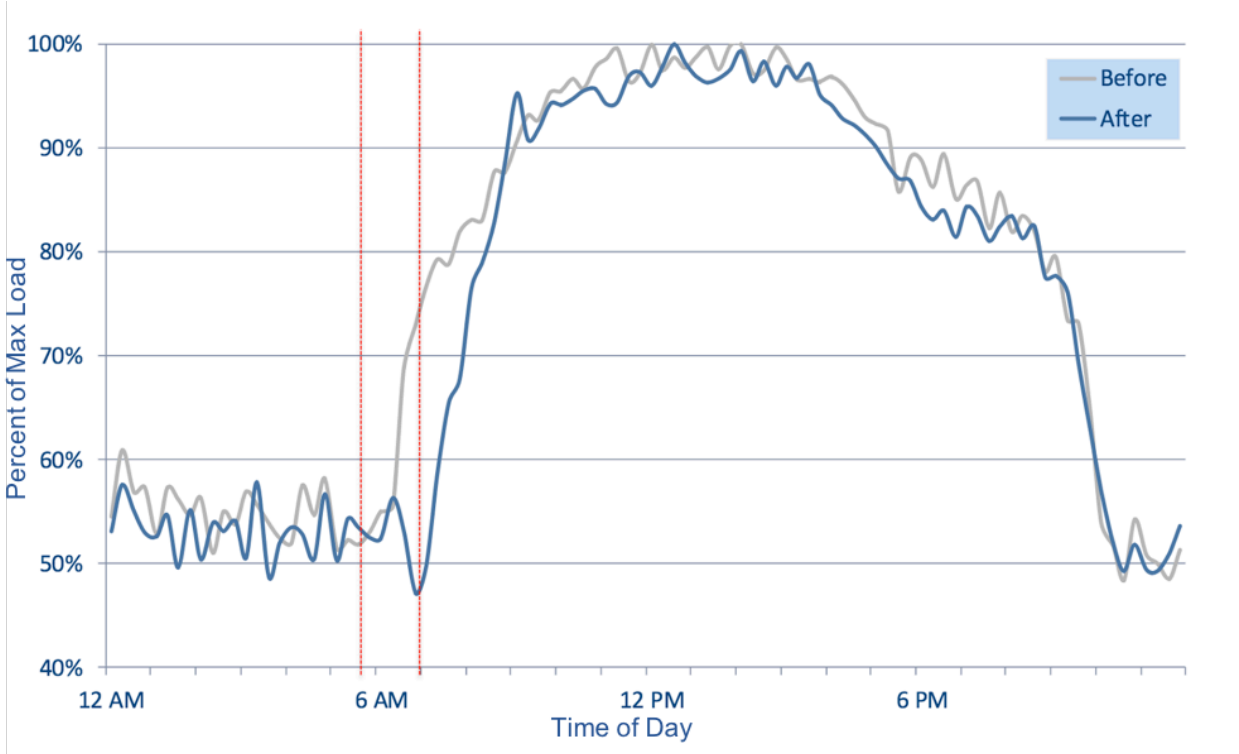


Figure 19: preliminary results from a project at UBC that uses real occupancy data from Wi-Fi to determine the best start-time for building heating (Henderson, 2016)

UC Davis, in collaboration with LBNL, is also planning to use Wi-Fi data to dynamically control a building at the UC Davis campus. The building is a student community center that is home to different student and academic services, meeting rooms, multi-purpose spaces and includes a cafeteria, and it has earned a LEED certification. The building is open for 17 hours every day (from 6am to 11pm), and its use is irregular. As there is no information on actual occupancy (due to lack of occupancy sensors) , the HVAC is operated with fixed schedule; the fixed schedule is the only strategy available to ensure the zones are comfortable. Figure 20 shows the count of Wi-Fi devices in the building during four weeks of May-June 2016. The Wi-Fi data, collected with one of the techniques described in section 4, was aggregated at intervals of 1 hour for this purpose. Weekend and holidays (e.g., Memorial Day) show reduced occupancy. A significant decrease in Wi-Fi counts can be observed at the end of the spring quarter.

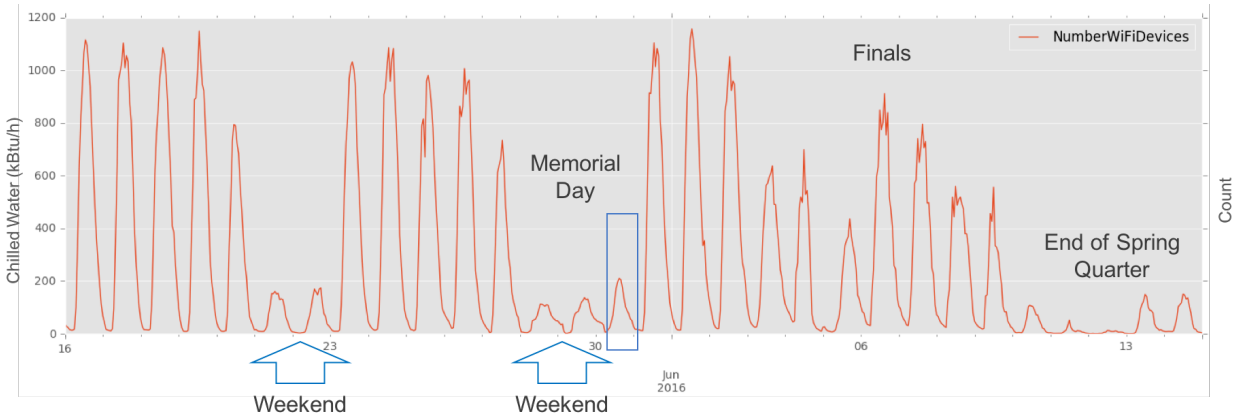


Figure 20: Occupancy patterns in a building in UC Davis target for a control retrofit. The occupancy was measured using Wi-Fi connections.

The HVAC was operated as though the building was fully occupied during the second weekend, including Memorial Day, and at the end of the academic quarter (Figure 21). UC Davis is planning on using occupancy information measured through Wi-Fi connections, to improve the operation of this building.

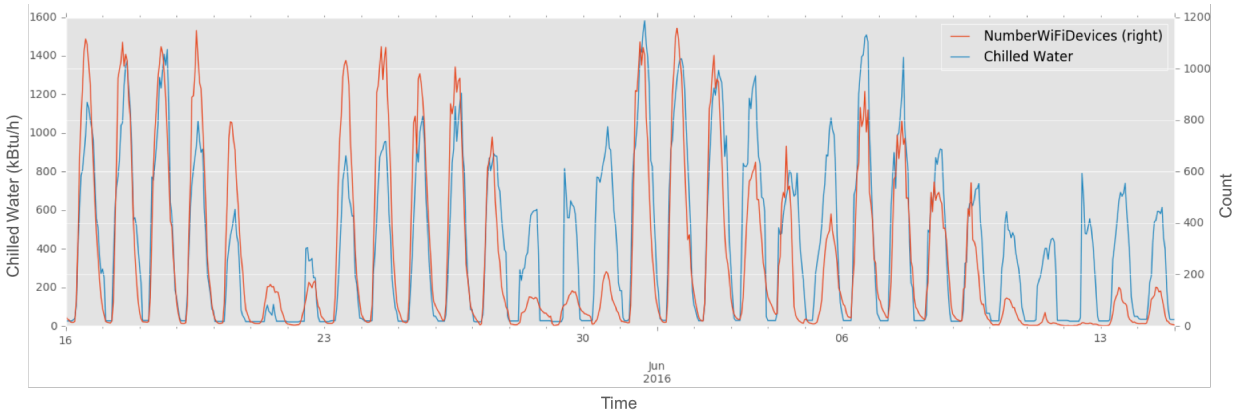


Figure 21: Occupancy (Wi-Fi connections) and chilled water energy use in a building at UC Davis. The operation was not adapted to the reduced occupancy during the second weekend and at the end of the quarter.

Zooming into a single day and using a 10-minute sampling rate we can see the occupancy fluctuating significantly during the day. The rapid oscillations are caused by students who stop at the coffee shop at the end of classes, but remain in the building for only a few minutes (Figure 22).

We also obtained limited data from Pacific Northwest National Laboratory. It is nominal 5-minute data from the "Math" building on the PNNL main site, with counts for the first floor for Wi-Fi associations (a separate count is also available for authenticated devices, but they are quite similar and always within 5% of each other). This data is graphed in Figure 23.

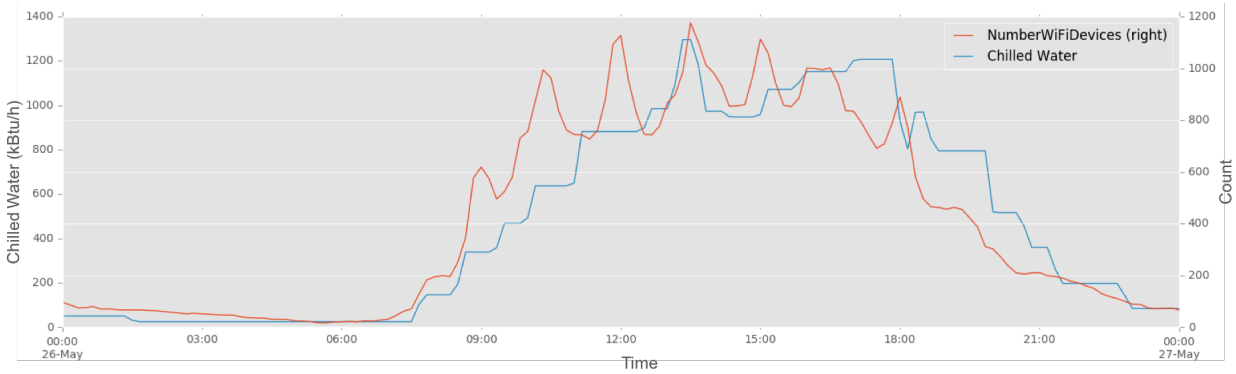


Figure 22: Occupancy oscillations in a building at UC Davis, caused by people stopping briefly at a coffee shop between classes

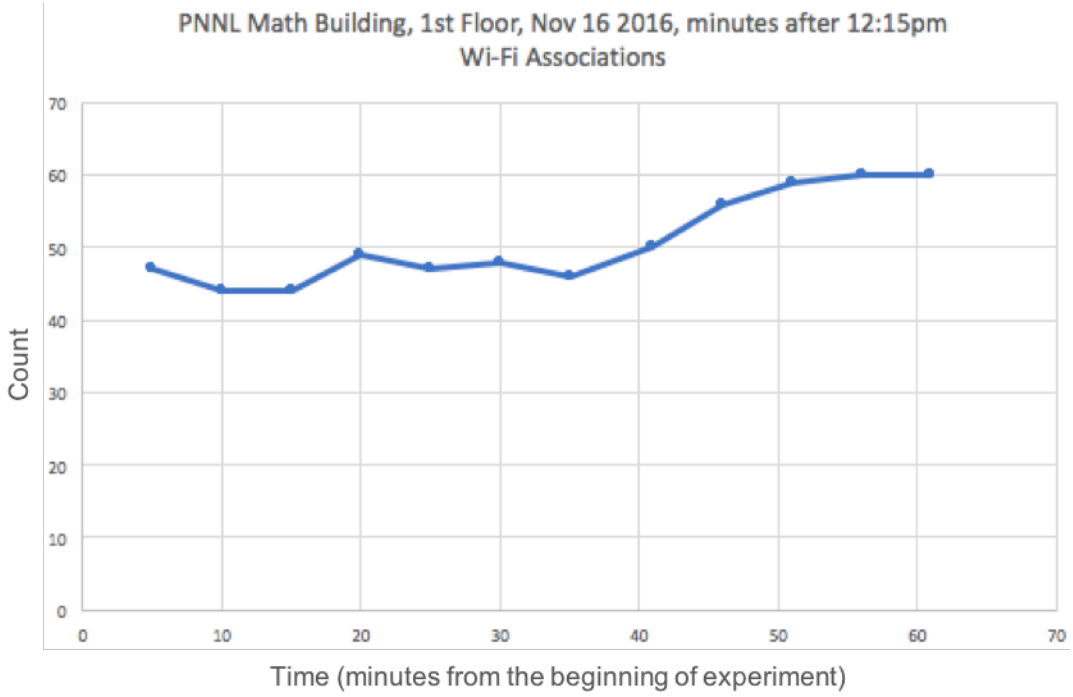


Figure 23. Wi-Fi associations for PNNL, Math Building

High-frequency data

Our basic data collection periodicity for Wi-Fi data has been 10 minute intervals, based on our intuition about how fast occupancy changes. As an experiment, we significantly increased the frequency of requesting data at LBNL for one building (Building 90) for a few days. We obtained data at 7 *second* intervals, and as our system delivers the data in real-time, could observe devices coming onto the network within seconds of that occurring. This demonstrates the real-time and granular capabilities of inferential sensing with Wi-Fi.

Figure 24 shows the count of devices for all of Building 90 (green, right scale), and the count for the AP nearest to our third-floor conference room (blue, left scale)[10]. The conference room hosted a meeting with visitors who had many phones that had not connected to our Wi-Fi system. At 11:23 a.m., all the visitors were asked to connect their phones to the Wi-Fi network and in real time during the meeting we saw it jump by a count of 9 almost immediately (from 35 to 44). The "noise" in the signal would be significantly from people passing through the hallway or otherwise entering and leaving the zone. At 11:39 a.m., the visitors were asked to join a second AP that is not part of the lab-wide network and so we saw a comparable immediate drop as they ceased being connected to the lab network.

In these two cases, the devices were added to and removed from the count for both the individual AP and the whole building. As someone moves around the building under normal circumstances, they will move from AP to AP but remain in the building total until picked up by an AP in

[10] Arranging for the higher resolution data to be provided, and enabling the real-time graphing was accomplished by Christian Kohler and Stephen Czarnecki of LBNL.

a different building, or removed through a time-out if they get and stay out of range of any building AP. Figure 24 shows the potential application for granular and low latency inferential sensing data.

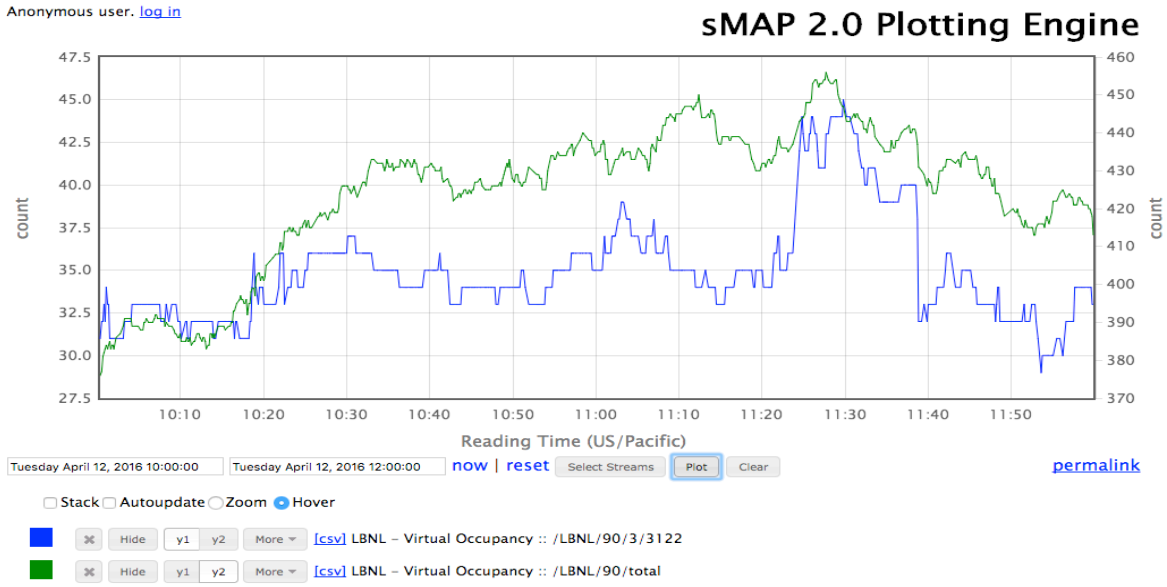


Figure 24: 7-second Wi-Fi device count data for the whole building (green, scale on the right) and for one conference room (blue, scale on right)

5.2.2 Demand-controlled ventilation

In addition to relaxing the start time of the building conditioning based on real occupancy, a useful strategy to save energy is regulating building ventilation depending on the number of occupants. Energy codes require ventilating a building to provide a minimum amount of fresh air per person. When the real occupancy is unknown, building managers often use erroneous and conservative assumptions on minimum ventilation, wasting energy. Previous studies have shown that demand-controlled ventilation can save a significant amount of energy (Liu et al. 2012, Zhang et al. 2013).

The count of associated devices for all the APs of a building is sufficient for this application.

- Spatial resolution: HVAC-zone level.
- Time resolution: 1-20 minutes
- Update frequency: instantaneous (need a database)

For some applications, such as demand-controlled ventilation, the spatial resolution of the occupancy data is important. In fact, the ability to reduce ventilation in a HVAC zone is bound to availability of occupancy information in that zone. While the way the HVAC is zoned frequently follows construction elements (rooms, ducts, walls), the APs usually detect devices in certain radius. In order to understand this problem, we present an example. During out tests with high frequency (7-second) Wi-Fi data at LBNL, we observed the data collected by one AP in a main conference room of Building 90. The HVAC zone of the conference room is separate from the adjacent rooms. Figure 25 shows a typical “noon

seminar" at LBNL. The attendees are almost all people who work in the building, so the AP for the conference room picks up about 25 people, just after 1 pm (this 'noon talk' was at 1), with a corresponding decrease just after 2. The roughly 20 people before (and 25 after) may have been in the room already, though almost certainly some were in offices nearby that are served by the same AP. Without additional information, it is impossible to know exactly how many people were in the HVAC zone of the conference room, therefore it is difficult to estimate the ventilation required. In absence of such information the building managers may use conservative settings, assuming maximum occupancy all the time.

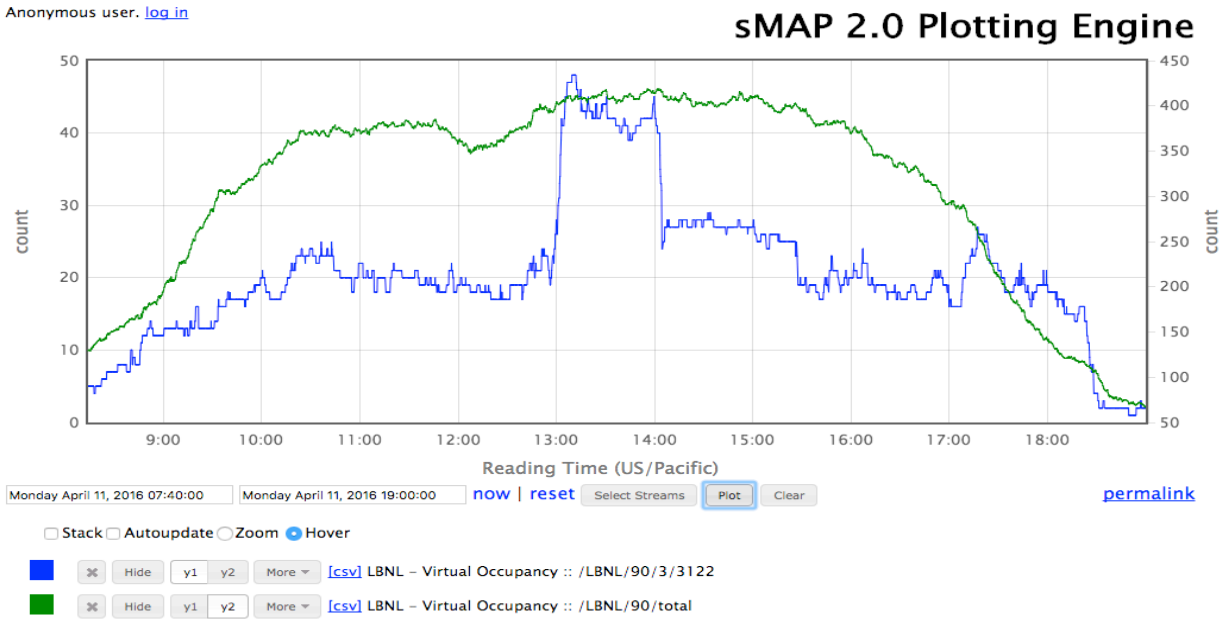


Figure 25: 7-second Wi-Fi device count data for the whole LBNL building 90 (green, scale on the right) and for the conference room (blue, scale on right). The spike around 1pm is caused by a seminar talk. The number of people detected before the talk (~20) and after the talk (~25) may or may not be people that stayed in the room.

Figure 26 shows a day with a morning meeting plus a noon talk. Note that in both cases, the building total does not change particularly as the single AP changes dramatically. This is because people are moving from one AP in the building to another when entering and leaving the conference room, not entering and leaving the building at the same time. Note that in Figure 26 we see lunchtime in the data; we can clearly see people leaving the building for lunch (so the building total dips) but congregating in the conference room for the talk. So, one particular room suddenly becomes crowded even as the total building occupancy decreases. Other APs, or groups of APs will show a commensurate drop.

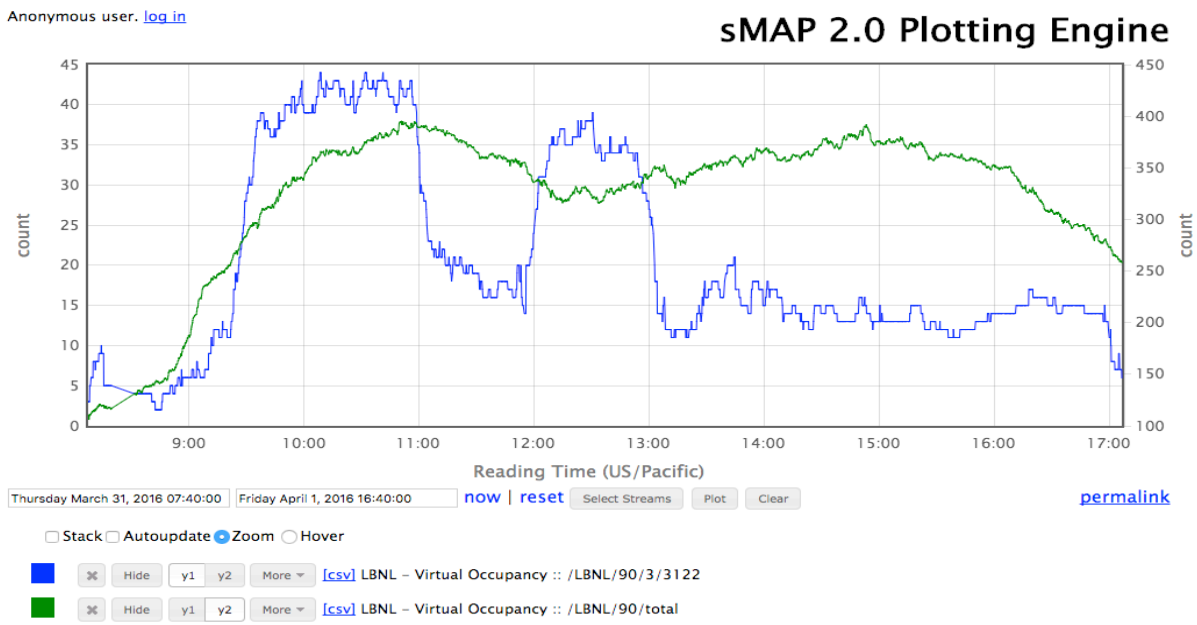


Figure 26: Wi-Fi device count data for the whole LBNL building 90 (green, scale on the right) and for the conference room (blue, scale on right). The two spikes in the blue streams are caused by a morning meeting followed by a noon talk

While the problem of identifying the precise location of people inside a building is still an open question in research (the problem is referred to as indoor localization), recent research suggests that coarse location information can still provide useful information for demand-driven HVAC controls (Balaji et al. 2013). Even though there is no absolute certainty about the location, with a probabilistic approach we can drive ventilation more efficiently than common practice. Instead of using the raw Wi-Fi data, an occupancy estimator can be created using sensor fusion techniques (Balaji et al. 2013). Figure 27 shows a diagram of how different sources of information can be combined to create an estimator for occupancy for a building zone. It is worth mentioning that some manufacturers of network equipment now offer additional hardware and software products to track user location more accurately, mostly for business intelligence and marketing purposes. The additional cost of these products must be weighed against the additional accuracy.

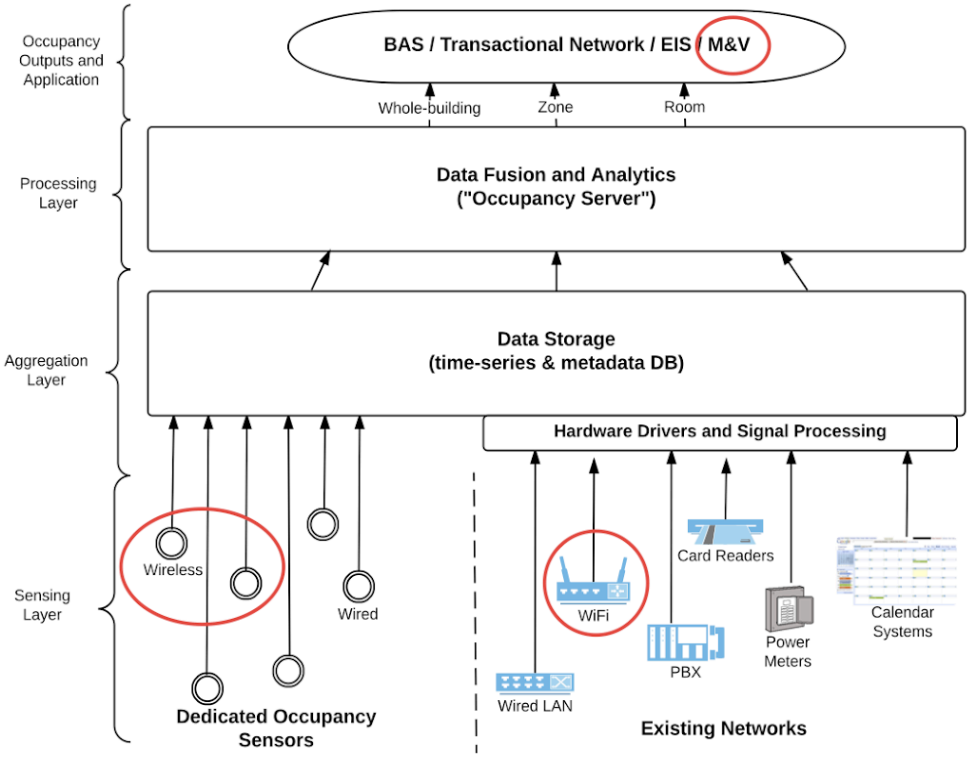


Figure 27: conceptual graph for generalized sensor fusion of occupancy data

6. Conclusions and Future Work

In this report we have presented technical approaches to organize methods and execute retrieval of inferential sensing data, the architecture of common Wi-Fi systems in place today, mechanisms for extracting the data, and a variety of applications for using these data. Inferential sensing, particularly with Wi-Fi device counts, is a readily available method of bringing near real-time occupancy data to buildings systems and analyses. There are a variety of ways to transfer the data from the Wi-Fi system to a building monitoring or control system, or other computer data acquisition system, for further analysis. These are not difficult for IT professionals to implement. Today there are a number of sites around the U.S. and abroad that are collecting such data, with at least one using it for dynamic building operation. With modest effort, nearly any commercial building could use this mechanism to understand occupancy and with additional effort, to connect it to building operation, principally control of HVAC systems.

There are a number of ways that the techniques described in this study could be disseminated and promulgated. One such technique is to work with manufacturers of network equipment to make these capabilities widely available to customers to make inferential sensing data available more simply and consistently. However, the industry is structured such that strategic coordination on topics like this rarely or never occurs. Some effort to ensure coordination like that which occurs in the wired networking industry is needed; if many building owners begin using inferential data from their Wi-Fi systems, that might be a sufficient change to spur industry cooperation to coordinate. That said, all the concepts, and most of the terminology, involved in Wi-Fi device counts is consistent regardless of manufacturer, so that once obtained, utilizing the data is the same regardless of the system present in the building.

Another step in making this data more available would be to work with other partners and organizations to test as many of these specific mechanisms as possible, to confirm that they work as expected and evaluate any outstanding issues. The content of this report could then be expanded and converted into a more comprehensive guide on how to obtain and use the data. This would provide a rich resource for building owners and researchers who would like to make use of inferential sensing data.

Finally, there is a need to disseminate case studies, data, and results of this work to the building energy efficiency community to share the opportunities described in this study. Building owners and operators are unaware of the value of and technical methods to gather these occupancy data.

8. References

American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE). (2016a). "Ventilation for Acceptable Indoor Air Quality". ANSI/ASHARE Standard 62.1-2016. Retrieved Nov 2016 from https://www.ashrae.org/standards-research--technology/standards--guidelines.

American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE). (2016b). "Ventilation for Acceptable Indoor Air Quality". ANSI/ASHARE Standard 62.2-2016. Retrieved November 2016 from https://www.ashrae.org/standards-research--technology/standards--guidelines.

Aruba. (n.d. a). "Aruba controllers webpage". Webpage. Retrieved December 2016 from http://www.arubanetworks.com/products/networking/controllers/

Aruba. (n.d. b). "Aruba Central management and services in the cloud". Retrieved December 2016 from http://www.arubanetworks.com/products/networking/management/central/

Aruba. (n.d. c). "Aruba AirWave data sheet". Webpage. Retrieved December 2016 from: http://www.arubanetworks.com/assets/ds/DS_AW.pdf

Aruba. (n.d. d). "Aruba Location-Based Services data sheet". Webpage. Retrieved December 2016 from: http://www.arubanetworks.com/assets/ds/DS_LocationServices.pdf

Aruba. (n.d. e). "Network management with Aruba airwave". Webpage. Retrieved January 2016 from: https://www.slideshare.net/ArubaNetworks/network-management-with-aruba-airwave

Avaya. (n.d. a). "Avaya wireless LAN". Webpage. Retrieved December 2016 from : http://www.avaya.com/en/product/wireless-lan-9100-series/

Balaji Bharathan, Jian Xu, Anthony Nwokafor, Rajesh Gupta, and Yuvraj Agarwal. (2013). "Sentinel: occupancy based HVAC actuation using existing WiFi infrastructure within commercial buildings". In Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems (SenSys '13). ACM, New York, NY, USA, , Article 17 , 14 pages. DOI=http://dx.doi.org/10.1145/2517351.2517370; available at http://cseweb.ucsd.edu/~jix024/papers/Sentinel.pdf

California Energy Commission (CEC). (2016) "Title 24: 2016 Building Energy Efficiency Standards". Retrieved November 2016 from http://www.energy.ca.gov/title24/2016standards/index.html .

Cisco, (n.d. a). “Cisco Wireless Selector Tool”. Webpage. Retrieved December 2016 from http://www.cisco.com/c/en/us/products/wireless/wireless-lan-controller/index.html

Cisco, (n.d. b). “Cisco Meraki wireless system”. Webpage. Retrieved November 2016 from https://meraki.cisco.com/products/wireless

Cisco, (n.d. c). “Cisco Wireless Control System Configuration Guide, Release 4.1”. Webpage. Retrieved December 2016 from http://www.cisco.com/c/en/us/td/docs/wireless/wcs/4-1/configuration/guide/wcscfg41/wcsmaps.html

Cisco, (n.d. d). “Cisco Meraki EU Cloud Configuration guide”. Webpage. Retrieved December 2016 from https://documentation.meraki.com/zGeneral_Administration/Privacy_and_Security/EU_Cloud_Configuration_Guide

Cisco, (n.d. e). “Cisco Prime Infrastructure”. Webpage. Retrieved December 2016 from http://www.cisco.com/c/en/us/products/cloud-systems-management/prime-infrastructure/index.html

Cisco, (n.d. f). “Cisco Unified Wireless Location-Based Services data sheet”. Webpage. Retrieved December 2016 from http://www.cisco.com/en/US/docs/solutions/Enterprise/Mobility/emob30dg/Locatn.pdf

Cisco, (n.d. g). “Cisco Mib locator”. Webpage. Retrieved November 2016 from http://tools.cisco.com/ITDIT/MIBS/AdvancedSearch?MibSel=250297&SUBMIT1=Submit

Dawson-Haggerty S., Jiang X., Tolle G., Ortiz J., and Culler D. (2010). sMAP: a simple measurement and actuation profile for physical information. In Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems (SenSys '10). ACM, New York, NY, USA, 197-210. DOI=http://dx.doi.org/10.1145/1869983.1870003

Henderson Orion (2016). “Innovative Use of WiFi Technology for Energy Savings at the University of British Columbia”. Presentation at UC Davis Facility Analytics workshop, August 2016. Davis, CA.

Influxdata. (2017). “InfluxDB, the modern engine for metrics and events”. Webpage. Retrieved January 2017 from https://www.influxdata.com/

Intel. (2017). “Understanding IEEE* 802.11 Authentication and Association”. Webpage. Retrieved January 2017 from http://www.intel.com/content/www/us/en/support/network-and-i-o/wireless-networking/000006508.html

Leskiw, A. (2017). “SNMP Tutorial Part 2: Rounding Out the Basics”. Webpage. Retrieved Jan 2017 from http://www.networkmanagementsoftware.com/snmp-tutorial-part-2-rounding-out-the-basics/

Liu G., J. Zhang, A. Dasu. (2012). “Review of Literature on Terminal Box Control, Occupancy Sensing Technology and Multi-zone Demand Control Ventilation (DCV)”. PNNL report 21281, March 2012. Retrieved from http://www.pnnl.gov/main/publications/external/technical_reports/PNNL-21281.pdf Accessed Jan 2017

Mauro Douglas, Kevin Schmidt. (2005). *Essential SNMP*, 2nd Edition O'Reilly Media September 2005 Retrieved from http://www.reedbushey.com/124Essential%20SNMP%202nd%20Edition.pdf

Melfi R., B. Rosenblum, B. Nordman, and K. Christensen. (2011). "Measuring Building Occupancy Using Existing Network Infrastructure," Proceedings of the International Green Computing Conference, July 2011.

MongoDB. (2017). “mongoDB – For Giant Ideas”. Webpage. Retrieved January 2017 from https://www.microsoft.com/en-us/sql-server/

Microsoft. (2017). “Microsoft Data Platform”. Webpage. Retrieved January 2017 from https://www.microsoft.com/en-us/sql-server/

Newsham, Guy R. Henry Xue, Chantal Arsenault, Julio J. Valdes, Greg J. Burns, Elizabeth Scarlett, Steve G. Kruithof, Weiming Shen. (2017). Testing the Accuracy of Low-Cost Data Streams for Determining Office Occupancy and Their Use for Energy Reduction of Building Services, Energy and Buildings, Volume 135, 2017, Pages 137-147, ISSN 0378-7788, http://dx.doi.org/10.1016/j.enbuild.2016.11.029

NIST. (2013). “NISTIR 7298 Revision 2 Glossary of Key Information Security Terms”. Retrieved from http://ws680.nist.gov/publication/get_pdf.cfm?pub_id=913810 Accessed Jan 2017.

Nordman B., K. Christensen, R. Melfi, B. Rosenblum, and R. Viera. (2014). "Using Existing Network Infrastructure to Estimate Building Occupancy and Control Plugged-in Devices in User Workspaces," International Journal of Communication Networks and Distributed Systems. (2014). Vol. 12, No. 1, pp. 4-29, January 2014.

Oidview. (n.d. a). “Oidview. Simple. Flexible. Powerful.”. Webpage. Retrieved December 2016 from http://www.oidview.com/

Oidview. (n.d. b). “MIB CISCO-DOT11-ASSOCIATION-MIB”. Webpage. Retrieved December 2016 from http://www.oidview.com/mibs/9/CISCO-DOT11-ASSOCIATION-MIB.html

OpenNMS. (n.d.). “OpenNMs, the network management platform, developed under open source model” Webpage. Retrieved December 2016 from https://www.opennms.org/en.

OpenTSDB. (2017). “OpenTSDB: The Scalable Time Series Database”. Webpage. Retrieved January 2017 from http://opentsdb.net/

Oracle. (n.d.). “Oracle Database” Webpage. Retrieved December 2016 from https://www.oracle.com/database/index.html

Periscope Data. (2017). “Periscope Data, from SQL query to analysis in seconts” Webpage. Retrieved January 2017 from https://www.periscopedata.com/

PostgreSQL. (2017). “PostgreSQL 10 Beta 2 Released!” Webpage. Retrieved January 2017 from https://www.postgresql.org/

Price Phillip, Bruce Nordman, Mary Ann Piette, Rich Brown, Janie Page, Steven Lanzisera and Jessica Granderson. (2015). “Automated Measurement and Verification and Innovative Occupancy Detection Technologies” September 20, 2015. LBNL LBNL-1007182. Available at https://cbs.lbl.gov/publications/automated-measurement-and-1.

[RFC1157] Case J., M. Fedor, M. Schoffstall, J. Davin. (1990). “A Simple Network Management Protocol (SNMP)”. Internet Engineering Task Force RFC 1157. Retrieved from https://tools.ietf.org/html/rfc1157 Accessed Jan 2017

[RFC1213] McCloghrie. K., M. Rose. (1991). “Management Information Base for Network Management of TCP/IP-based internets: MIB-II” Internet Engineering Task Force RFC 1213. Retrieved from https://tools.ietf.org/html/rfc1213 Accessed Jan 2017

Ruckus, (n.d. a). “Ruckus, Zonedirector controller”. Webpage. Retrieved December 2016 from https://www.ruckuswireless.com/products/system-management-control/zonedirector-controllers

Ruckus, (n.d. b). “Ruckus-wireless-simplifies-Wi-Fi-small-business-launches-new-controller-less”. Press release on Webpage. Retrieved Jan 2017 from https://www.ruckuswireless.com/products/system-management-control/zonedirector-controllers

Ruckus, (n.d. c). “Ruckus location-based services solution”. Webpage. Retrieved January 2017 from https://www.ruckuswireless.com/products/smart-wireless-services/location-services

Ruckus, (n.d. d). “Ruckus Wireless ZoneDirector Release 9.8 User Guide”. Webpage. Retrieved Jan 2017 from https://support.ruckuswireless.com/documents/454-zonedirector-9-8-ga-user-guide

Salmon K., Morejohn J., Sanguinetti A., Pritoni M. (2016). “How to design an energy dashboard that helps people drive their buildings”. ACEEE Summer Study on Energy Efficiency in Buildings, Asilomar, CA. August 2016.

Sensible Building Science. (2015). Case Study: Real-time Occupancy-based Building Controls. Consultant Report for University of British Columbia.

Solariwinds. (n.d). “Solarwinds. Solve your toughest IT management problem, today ”. Website. Retrieved Jan 2017 from http://www.solarwinds.com/

SQLite. (2017) “SQLite is a self-contained, high-reliability, embedded, full-featured, public-domain, SQL database engine”. Website. Retrieved Jan 2017 from https://sqlite.org/

Storey Stefan, James Montgomery. (2014). “UBC Wi-Fi AP Data for Energy Conservation”. Personal conversation with Stefan Storer about a report for University of British Columbia Campus Sustainability Office.

Zhang J., R.G. Lutes, G. Liu, M.R. Brambley. (2013). “Energy Savings for Occupancy-Based Control (OBC) of Variable-Air-Volume (VAV) Systems” PNNL report 22072, January 2013. Retrieved from http://www.pnnl.gov/main/publications/external/technical_reports/PNNL-22072.pdf Accessed Jan 2017

9. Appendix

This appendix provides detailed information about specific mechanisms for each vendor

Cisco

The table below shows how the mechanisms and devices in general map onto those for Cisco Wi-Fi systems.

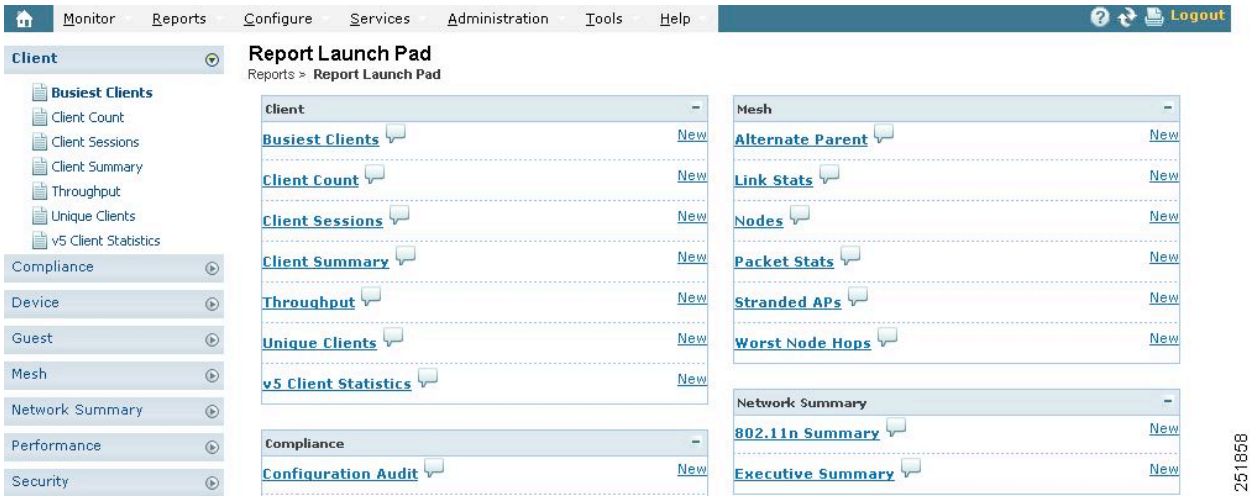
	SNMP	CLI	GUI report	API
Access Point	Y	Y	-	-
Controller	Y	Y	Y	-
Management System (NMS)	-	-	Y	Y

Note: Coding in cells: (Y) exists, (-) not known to exist

Create a report from controller/overseer GUI

(WCS/Prime Infrastructure) - used for offline applications

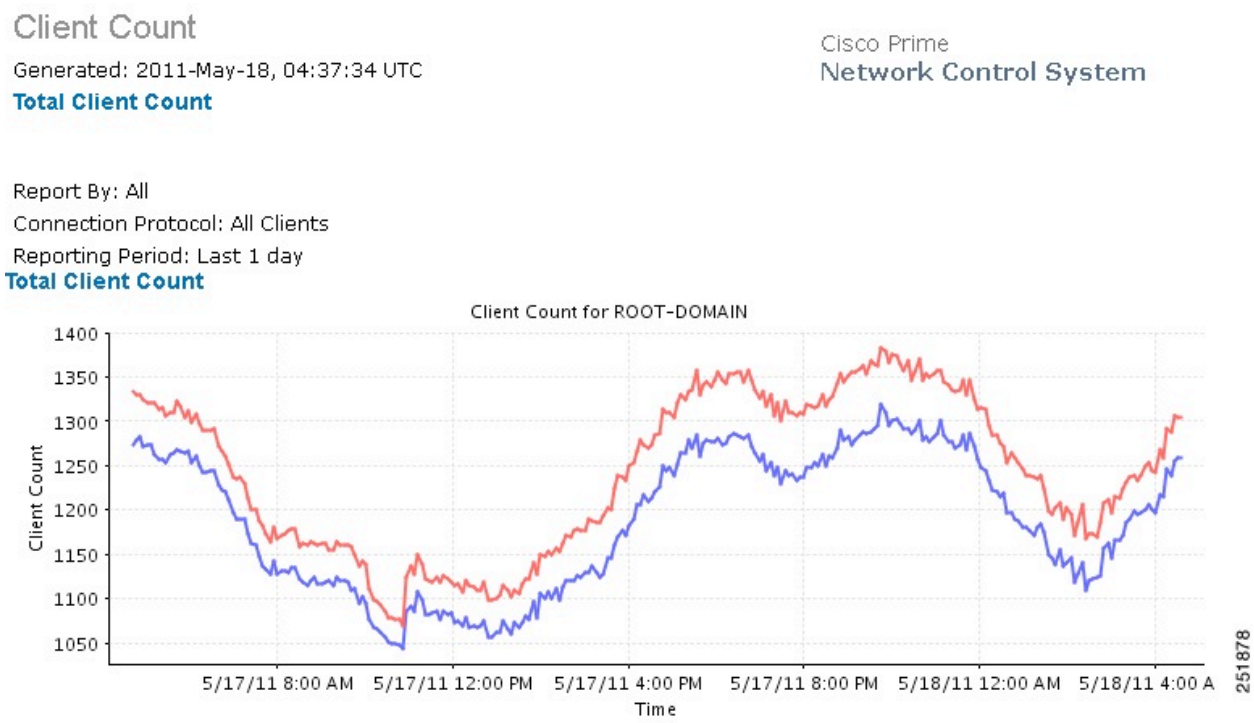
Go to Reports > Report Launch Pad



Client Count - New (hyperlink) Report by: AP by Controller Report Criteria: All Controllers > All Access Points (this will change after the next edit fields are set) Edit: Controller: [Controller you select] Access Point: [Select all the APs you want data from, You can use

Shift+Click for a range]
Reporting Period: (Last 6 Hours)
Recurrence: 'Hourly radio button', Every '6' Hours

Results visualized



Links to documentation

Running Reports: http://www.cisco.com/c/en/us/td/docs/wireless/wcs/7-0/configuration/guide/WCS70cg/7_0reps.html#wp1135759
Client Count Section: http://www.cisco.com/c/en/us/td/docs/wireless/wcs/7-0/configuration/guide/WCS70cg/7_0reps.html#wp1135255
How to use the GUI: http://www.cisco.com/c/en/us/td/docs/wireless/controller/7-4/configuration/guides/consolidated/b_cg74_CONSOLIDATED/b_cg74_CONSOLIDATED_chapter_010000000.html

Get summary from controller command line interface (CLI)

From command line:
http://www.cisco.com/c/en/us/td/docs/wireless/controller/7-0/command/reference/cli70bk/cli70commands.html#wp1318963

> show client summary

```
Number of Clients.............................. 24
MAC Address  AP Name   Status   WLAN Auth Protocol Port
----------------- ----------------- ------------- ---- ---- -------- ----
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11a 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11a 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11b 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11a 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11b 1
xx:xx:xx:xx:xx:xx AP02    Associated  2  Yes 802.11b 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11b 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11b 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11b 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11a 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11a 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11b 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11a 1
xx:xx:xx:xx:xx:xx AP02    Probing  N/A No 802.11a 1
Number            of            Clients.............................            2
```

Alternatively can run:

> **show capwap reap association**

Links to documentation

CLI documentation: http://www.cisco.com/c/en/us/td/docs/wireless/controller/7-0/command/reference/cli70bk/cli70commands.html#wp1318963

Get summary from AP CLI

```
> show dot11 associations client
```

Links to documentation

How to schedule this to run using kron: http://www.techrepublic.com/article/schedule-commands-with-cisco-ios-kron/

Query Prime infrastructure API with Python

Prime Infrastructure exposes an API that can be used to extract information. See documentation in the following links.

Links to documentation

Prime Infrastructure API documentation: https://developer.cisco.com/site/prime-infrastructure/documents/api-reference/rest-api-v3-0/

PI API python module documentation:
https://supportforums.cisco.com/announcement/12690226/cisco-prime-infrastructure-rest-api-python

Use SNMP scripts

The proprietary MIB can be found online in the CISCO MIB Locator:
Cisco MIB locator: http://tools.cisco.com/ITDIT/MIBS/MainServlet
Search for: CISCO-DOT11-ASSOCIATION-MIB.

Links to documentation

SNMP OID for Total Associations on a WLC:
https://supportforums.cisco.com/discussion/10341801/snmp-oid-total-associations-wlc

MIB Compilers and Loading MIBs: http://www.cisco.com/c/en/us/support/docs/ip/simple-network-management-protocol-snmp/26015-mibcompilers.html

Additional links to documentation

Cacti discussion: # of users per AP:
http://forums.cacti.net/viewtopic.php?t=12863&highlight=4400
WLC OID (snmp) for authenticated clients
https://supportforums.cisco.com/discussion/10531976/wlc-oid-snmp-authenticated-clients
Cisco WLC AP count over SNMP
https://supportforums.cisco.com/discussion/11394301/cisco-wlc-ap-count-over-snmp
SNMP MIBs and Traps on the ASA - Additional Information
https://supportforums.cisco.com/document/7336/snmp-mibs-and-traps-asa-additional-information
Cisco WLC SNMP Historical User Statistics Monitoring (w/ Syslog or Splunk)
https://supportforums.cisco.com/document/9869811/cisco-wlc-snmp-historical-user-statistics-monitoring-w-syslog-or-splunk
Cisco IOS Command Reference for Cisco Aironet Access Points and Bridges
http://www.cisco.com/c/en/us/td/docs/wireless/access_point/12-4_10b_JA/command/reference/cr2410b.pdf
Cisco supported MIBs
http://www.cisco.com/c/en/us/td/docs/wireless/access_point/12-3_7_JA/configuration/guide/i1237sc/s37mib.html
Cisco SNMP Counters: Frequently Asked Questions

http://www.cisco.com/c/en/us/support/docs/ip/simple-network-management-protocol-snmp/26007-faq-snmpcounter.html
Cisco Prime Infrastructure 3.0 User Guide
http://www.cisco.com/c/en/us/td/docs/net_mgmt/prime/infrastructure/3-0/user/guide/pi_ug/reps.html

Aruba

	SNMP	CLI	GUI report	API
Access Point	Y	Y	-	-
Controller	Y	Y	Y	-
Management System (NMS)	-	-	Y	Y

Note: Coding in cells: (Y) exists, (-) not known to exist

Query AirWave API for data using Python

>> script by Travis Schick (UCD)

```
#!/bin/python
from airwaveapiclient import AirWaveAPIClient
from airwaveapiclient import APList

import hashlib

airwave =
AirWaveAPIClient(username='****',password='****',url='https://amp-
address')

response = airwave.login()

response = airwave.ap_list()

apList = APList(response.text)
airwave.logout

for ap in apList:
 if 'client_count' in ap.keys():
  clientCount = ap['client_count']
 else:
  clientCount = 0
```

```
 print "'%s',%s" % (ap['name'],clientCount)
>>
```

Can use “cron” to call the script every 5-10 minutes.

To publish to a rest API the IT department can include this generated file into curl command. This can also be periodically run using “cron”.

Results visualized

http://mynetwork.noc.ucdavis.edu/wireless/ap-user-count.txt (need UCD campus ID)

mynetwork.noc.ucdavis.edu/wireless/ap-user-count.txt

```
'DV202COUSTEAU-224-714064',9
'HUNT-OS218-735524',5
'KEMPER-OS0064-740242',0
'CHEM-OS131-739966',11
'MILLER-247-712100',1
'VIRIDIAN3-1114-600479',2
'CONF-CENTER-2300-731643',11
'HWC-3006-727613',8
'VIRIDIAN5-1104-600849',9
'UNIVHOUSE-125-729695',0
'EVERSON-OS156-736116',8
'PHY-SCI-LIB-301-700991',2
'THOMPSON-252-712264',1
'TUPPER-1311A-709188',11
```

Links to documentation

AirWave API python library documentation:
https://airwaveapiclient.readthedocs.io/en/latest/index.html

Use a SMNP script

>> script by Michael S (LBNL)

```
# query Aruba controllers
/usr/local/bin/snmpwalk    -v    2c    -c    COMMUNITY    128.3.133.40
1.3.6.1.4.1.14823.2.2.1.4.1.2.1.37      |      cut      -c48-      >
/var/db/aruba_polling/$devz1
/usr/local/bin/snmpwalk    -v    2c    -c    COMMUNITY    128.3.133.70
1.3.6.1.4.1.14823.2.2.1.4.1.2.1.37      |      cut      -c48-      >
/var/db/aruba_polling/$devz2
```

Before and after this step, some data manipulation is required to setup files, merge them, remove sensitive data. This need to be performed by the IT department.

Links to documentation

Aruba MIB 5.0: https://community.arubanetworks.com/aruba/attachments/aruba/unified-wired-wireless-access/7568/1/ArubaOS_5%200MG.pdf

Get summary from CLI

(Aruba3200)

```
> #show ap association ap-name ap1
```

Links to documentation

Aruba useful CLI commands:
https://community.arubanetworks.com/aruba/attachments/aruba/tkb@tkb/245/1/Useful%20CLI%20commands-v1.pdf

Automate it using unix and cron:
https://community.arubanetworks.com/t5/tkb/articleprintpage/tkb-id/MonitoringManagementLocationTracking/article-id/561

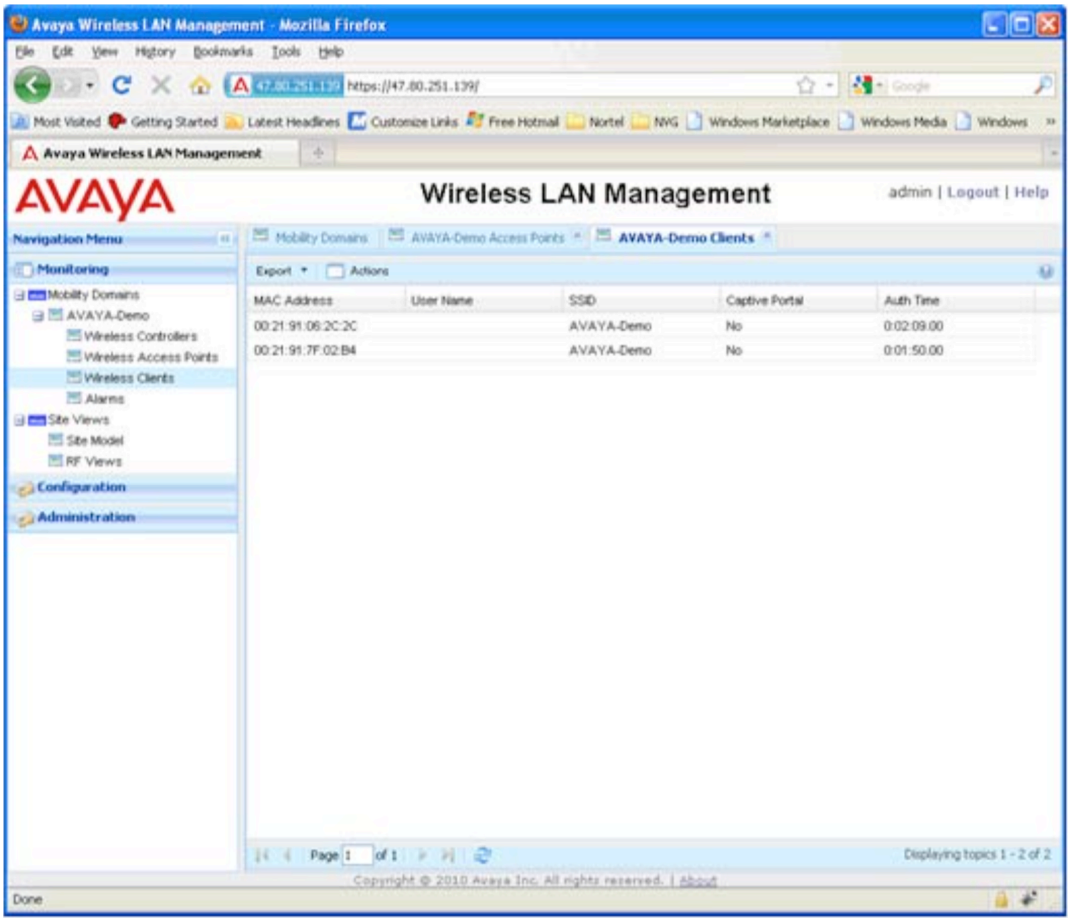
Avaya

	SNMP	CLI	GUI report	API
Access Point	Y	Y	-	-
Controller	Y	Y	?	-
Management System (NMS)	-	-	-	-

Note: Coding in cells: (Y) exists, (-) not known to exist

Visualize associated clients using GUI

Avaya offers a dashboard to visualize associated clients that can be accessed from the drop-down list clicking “Show Associated Client Dashboard”



Links to documentation

Avaya WLAN 8100 Quick Start Guide (page 43):
https://downloads.avaya.com/css/P8/documents/100110216

Get summary from CLI

```
> show users
```

Links to documentation

Avaya WLAN 8100 Quick Start Guide: https://downloads.avaya.com/css/P8/documents/100110216

Using the Command Line Interface:
https://downloads.avaya.com/css/P8/documents/100107948

Use SNMP scripts

The proprietary MIB can be found online:

https://support.avaya.com/downloads/download-details.action?contentId=C2013211435285270_7&productId=P0533
https://support.avaya.com/downloads/download-details.action?contentId=C20090710163411465875766&productId=P0001
http://www.oidview.com/mibs/6889/md-6889-1.html

Additional Network Management Software (NMS)

Links to documentation

NetBrain Dynamic Mapping:
http://info.netbraintech.com/dynamic-map-private-demo-cisco4.html?utm_source=Banner&utm_medium=Cisco%20Support&utm_campaign=Network%20Diagram
Solarwinds:
http://www.solarwinds.com/?&CMP=KNC-TAD-GGL-SW_NA_US_PP_CPC_LD_EN_BRD_DWA-XPIL-X_X_X_X-X&kwid=reTZA7vF&gclid=Cj0KEQjwx96-BRDyzY3GqcqZgcgBEiQANHd-ni9ZHD_RiXIsT6S5whSa79irAwcLMP7uiQS93uxlKEUaAmHE8P8HAQ
Comparison of Network Management software:
http://www.networkmanagementsoftware.com/network-management-software-smackdown/