A Case Study: Implementing Supportworks Professional Helpdesk at Drew University

Betsy Black
Drew University
36 Madison Avenue
Madison, New Jersey 07940
1-973-408-3860
eblack@drew.edu

E. Axel Larsson
Drew University
36 Madison Avenue
Madison, New Jersey 07940
1-973-408-3048
elarsson@drew.edu

ABSTRACT

Drew University, a leader in ubiquitous computing, has provided computers for all incoming freshman since 1984. The support structure has evolved with the technology, and now crosses several technology departments including Administrative Computing (AC), Telecommunications (Telecom), Instructional Technology Services (ITS), and Computing & Network Services (CNS).

CNS, home to the university Helpdesk, worked for a few years with two home-grown ticket tracking services, Helpdesk.drew.edu[1] and Beacon.drew.edu. These systems were not sufficient for escalating calls within and outside of the department. In February 2003, a committee was formed with representatives from all the technology departments at Drew University, and Enterprise Application Specialist Axel Larsson was charged with leading the search for the solution to the university's technology ticket tracking system.

Packages that were considered were Front Range's Heat, Blue Ocean's Track-It!, and Hornbill LTD's Supportworks Helpdesk Professional. Supportworks was selected for its ease of customization, shared mailbox features that would allow students to send e-mail from the department account, full-featured inventory tracking, knowledgebase, web-based self-service, and problem and resolution profiles. It also had the most attractive pricing for the number of licenses we needed.

This paper will describe the planning, selection, design, and implementation of Drew University's helpdesk solution.

Categories and Subject Descriptors

K.6.3 [**Computing Milieux**] Software Management - *Software selection*

General Terms: Management, Software

Keywords: Helpdesk, analysts, Supportworks, Helpdesk Professional, Microsoft Access, database, knowledgebase, call tracking, service level

1. INTRODUCTION

Drew University, founded in 1867, became the first liberal arts college in the United States to have a ubiquitous computing program when it began the Computer Initiative in 1984. That year, incoming freshmen at Drew were given an Epson QX-10 running CPM-80.

The program evolved over the years, and in 1988, the first portable computer (the Zenith 181) was issued. The next year saw the introduction of systems with modems, and the campus was connected through a data port on the students' phones. In 1996, fiber optic cable was installed and the campus network was born. Drew now offers port-per-pillow 100 Mbps network ports in all dormitory rooms and has a substantial wireless network.

All of these improvements—plus the advent of Windows—led to increased support demands, which are somewhat alleviated by the fact that support is only provided for Drew-issued computers. The CNS Helpdesk (then called the Computer Aide Station) was forced to begin taking in student computers for diagnostics and repair in 1998. At first, computers were tracked with paper forms. The form had the student's name and extension, as well as the CNS Helpdesk student employee name, comments, and steps taken.

In 1999, we developed a homegrown web-based tracking system called "Helpdesk.drew.edu" which allowed us to get rid of the paper trail and track comments and procedures online. It also allowed customers to view their tickets to see what had been done and what their position was in the queue.

This system was upgraded to an Oracle-based application in 2000, Beacon.drew.edu. This upgrade was intended to tie in the inventory information for each model of computer we supported. Though this worked well for a few years, it became clear that we needed a product that would allow us to escalate calls to other technology organizations at Drew (only Academic Technology had access to Beacon) as well as a better inventory control procedure.

In 2002, E. Axel Larsson graduated from Drew and his position as a student CNS Helpdesk manager transitioned to a full-time position as Enterprise Integration Specialist. One of the first

projects Larsson developed was uTrack, a Microsoft SQL Server 2000 database that tracked inventory from procurement through deployment.

In the Spring of 2003, the Department of Academic Technology at Drew split into Computing & Network Services (CNS) and Instructional Technology Services (ITS). This compounded the need for a product that would work within several departments. The Oracle license was expiring in the summer of 2003, so a search for a replacement package began in earnest.

2. THE SEARCH PROCESS

Larsson was charged with forming a committee consisting of members from all the technology departments at Drew: Administrative Computing (including Telecommunications); Computing & Network Services (including the Helpdesk, the Computer Store, and the systems/network group); and Instructional Technology Services (including the Faculty Development Lab, Multimedia Resource Center, the Staff Development Lab, and the Training Resource Center).

Committee members first described their organizations and business practices currently in place. Most departments were using paper-based tracking or no tracking systems at all. Of particular concern during this process were scenarios that would cause a call to be escalated to another department such as password reset requests, non-working network ports, Blackboard issues, etc.

The committee decided that critical elements for any program would be a FAQ and Knowledgebase feature, the ability for customers to open and track their own tickets, full text searching, and the ability to integrate with the current inventory database. The package also needed to be easy to use for casual users of the software. While the software would be most intensely used by Helpdesk operators and User Support Services staff, who would be trained on the software, it also needed to be accessible to more infrequent users such as department directors.

Three packages were initially considered: FrontRange's Heat, Blue Ocean's Track-it!, and Hornbill's Supportworks Helpdesk Professional. Track-it! did not have the required functionality and was discarded from the selection process. HEAT was much more costly than Supportworks, had a steeper learning curve, and charged separately for FAQ and Knowledgebase functionality, so the committee decided to select Hornbill's Supportworks Helpdesk Professional.

3. DESIGN AND IMPLEMENTATION

An implementation schedule was devised by Larsson based upon the vendor's recommended procedures and focused on defining support groups and analysts, classifying calls (service level agreements, call conditions, problem & resolution profiles), custom form design, customer and asset databases, and shared mailbox access.

3.1 Support Groups

Supportworks uses a model of call assignment that classifies support analysts into groups. Calls may be assigned to groups or individual analysts. For our implementation, it was decided that we should keep full-time staff and students in separate groups, (e.g., there is one group for CNS Helpdesk full time staff and another for CNS Helpdesk student staff). This allowed us to delegate calls to the student groups, which could be picked up by any student who was on shift. The remaining support groups were set up by department, leaving one group for Administrative Computing, one for Computing and Network Services, one for Instructional Technology Services, and one for Telecommunications. Supportworks does not allow a user to exist in more than one group, which presented a problem for us as we have some student employees who work for more than one department. The committee decided that rather than try to define a "primary group" for each student, we would instead create separate logins in each group for which a student was employed. Student employee Supportworks usernames consisted of their standard Drew uLogin usernames, prefixed by a two character identifier for their support group. This allowed us to accommodate the multiple logins that some students would have.
We also created two generic analysts in the Helpdesk group, chcontact and chpickup, to use for calls that need customer contact and computers that are ready for pickup. These allow us to quickly identify calls that do not require immediate action.

3.2 Service Level Agreements

Supportworks has a rather flexible system for Service Level Agreements that allows for multiple configurable response and fix time triggers. SLAs can be associated with customers, customer departments (charge centers), sites, asset items, and problem profiles. Configurable fix and response time triggers allow the system to take any number of actions in the event an SLA is in danger of being violated, or when a timer has already passed. SLA actions can include: notifying a helpdesk manager, transferring the call to a different person or support group, or changing the call's condition code. Condition codes use colors to represent the severity of a call. Committee members were asked to define the service level and escalation requirements for their areas. We chose to assign SLAs by problem profile, giving priority to those issues that would affect the greatest number of users.

3.3 Call & Resolution Profiles

After defining basic service level agreements, the committee began the process of creating a classification system for calls entering the Helpdesk. Supportworks has a hierarchical system of call classification, known as Problem Profiles. The principal advantage of the hierarchical classification model is its application in "drill-down" style reporting of helpdesk activity. Also, unlike the call classification systems in other products, which are usually restricted to a single level of grouping, the support representative is not at a loss if the call he or she is trying to log does not fit into one of the predefined call profiles. With Supportworks, the support analyst may profile a call to the highest level of specificity that is appropriate. For instance, under the general "Report Problem" profile, there may be a profile for Microsoft Word related issues, and then under that, several common issues with Microsoft Word. If the support analyst is attempting to log a call about Microsoft Word that does not fit into one of the predefined profiles, the analyst may still profile the call as "Report Problem > Microsoft Word," rather than having to resort to selecting a generic "Other" or catch-all category.

When designing the call classification system, the committee decided to profile calls no more than three levels deep for

simplicity, and decided on a top-level classification that consisted of the categories:

- Report Problem
- Service Requests
- How-To Questions / Training
- Maintenance
- Comments/Suggestions

Report Problem – This call profile and the hierarchical classifications include a broad range covering computer hardware and software, telephones, televisions, technology enhanced classrooms, and enterprise application problems. Many of the call profiles for software classifications contain instructions for analysts to use while troubleshooting the problem over the phone or in person.

Service Requests – Customers can request additional network or e-mail space quotas, a network printer queue installation, loaner laptops, voice or data jack installation, OS upgrades, and specialized software access.

How-To Questions/Training – This is reserved for software training requests.

Maintenance – Used internally for installing security patches or configuration changes.

Comments/Suggestions – For customer feedback relating to any area of technology.

The second level of the hierarchy consists of categories for each application, system, or service supported, classified by technology department as appropriate. The third level of classification consists of specific problems or tasks associated with each application, system, or service. These classification categories were designed to aid in resource tracking, so that it would be possible to determine how much of the support organization's resources were being used solving technical problems, training, configuring or deploying new services, and performing routine maintenance activities.

3.4 Asset & Customer Data

After designing the problem classification system, the committee next focused on customer and asset information. The Supportworks customer database is populated automatically from the University's administrative database. The committee decided what fields needed to be included from the administrative system, such as phone extensions, dorm rooms, department, campus mailbox, etc. and a nightly batch job was set up to load the data into Supportworks' SQL Server database. For asset information, the committee decided to continue relying upon Drew's homegrown uTrack inventory application, which was already designed to accommodate our equipment leasing and procurement workflows. As the uTrack application used SQL Server as well, integration was easy. A set of database triggers was set up in the uTrack database so that the Supportworks database would be automatically updated whenever any asset information was changed in uTrack. To ensure that proper inventory workflow processes were being enforced by uTrack, we decided to make uTrack authoritative for all asset information, configuring the Supportworks asset forms to disallow changes to asset information from within Supportworks.

As compared with other packages of this type, Hornbill does not supply custom database loading tools for Supportworks. Integration is accomplished at the database level, and customers are expected to be familiar with their chosen database, and use database tools to populate the database with information from other systems. This increases the learning curve for an initial Supportworks implementation, but is overall more flexible than other vendors solutions who have a strict policy of no support for local modifications to the database.

3.5 Forms & Templates

In order to ease the transition from each department's existing call logging procedures, we used Supportworks' capability to design custom forms extensively. Supportworks offers a graphical form designer that enabled us to duplicate the look of existing paper or electronic forms for logging calls. We were also able to extend the schema of the Supportworks SQL Server database to support additional fields we required on our custom log call forms. Custom forms designed for the initial launch of Supportworks included a form for reporting incidents in mediated classrooms, which included a set of checkboxes for each piece of classroom equipment, a form for customer computers taken in at the helpdesk, and a form for calls that involve site-visits which included fields for building and room number. In the future, we hope to integrate the Supportworks system with a campus wiring database, and make this information accessible from forms for reporting bad network or telephone ports.

3.6 Issues

Another useful feature of Supportworks is the ability to open "Hot" or "Known" Issues. Each time an analyst logs into the system, the Supportworks Today screen notifies him or her if there are any known problems. Issues are also visible to "self service" customers if the issue is tagged as public.

We've used Issues for virus outbreaks; service outages (such as NDPS printing); client issues during the VPN rollout; scheduled network maintenance; and building or switch outages.

Issues are valuable when there are systemic outages or virus outbreaks that affect a large user population. Once an Issue is open, you can easily add calls with the pre-defined problem profile and at resolution, all calls can be closed en masse and e-mail notification will be sent. The Issues can contain specific instructions, e.g., how to remove a virus or run system updates, and are an effective tool for managing large numbers of calls related to the same problem.

Issues can be created and calls added later, or an existing call can trigger an Issue if a larger problem is identified by several customers. Other existing calls can easily be linked from the ticket interface

3.7 Self-service

The final stage of the design process was the customer self-service area. Supportworks includes a very functional customer self-service site that can be used as-is and allows customers to view a list of open issues, log new incidents, review and update their incidents, and search the Supportworks knowledgebase. The web interface is written in PHP and source code is provided, so we were able to customize it to match the appearance of other Drew

technology pages. We also ported some of our custom log calls forms, such as the classroom incident reporting form, to the web interface so that customers could log these calls themselves. The self-service interface supports a variety of authentication methods, including Active Directory, generic LDAP, and Windows NT 4 domain. Using this feature, we were able to easily integrate the Supportworks site with our Novell iChain single-sign-on system, allowing customers to log in to the support site using their regular Drew network passwords.

4. PRODUCTION

The areas of Technology at Drew University have adopted different approaches to the use of Supportworks in their organization. For instance, Telecom's administrative assistant accepts calls for the technicians and then closes them when they're complete. However, the CNS Helpdesk full-time and student staff rely most heavily on Supportworks during their day-to-day operations.

Using their Student CNS Helpdesk Operator logins, students are expected to log telephone calls and walk-in contacts and either close/resolve the problem or escalate it to the appropriate staff support group.

There are 4 common call types that CNS uses most regularly: the Helpdesk Intake (which includes a form to note each component received with a computer, e.g., power adapter, network cable, bag), the Call without Asset, the Call with Asset, and the Loaner Request.

The Helpdesk "Intake" is used when a customer is physically leaving equipment with the Helpdesk, and logging a call generates a printed receipt that the customer signs acknowledging that we've correctly listed the components and agreeing to the terms of the manufacturer's warranty. Once the call is logged and a ticket number generated, the computer and its components are labeled with the ticket number to ensure nothing is lost. When the customer picks up their computer, closing the call generates another receipt acknowledging that they picked up the computer and all components and accepting the repair work performed.

The Call without Asset encompasses software problems, network jack problems, increased email and/or network file storage requests, service outage reports, forgotten passwords, etc.

The Call with Asset is used for computers/printers that are not physically received by the Helpdesk, and are most commonly staff/faculty desktops that require a house call or printers that are to be networked.

Loaner requests are for students who need a computer while theirs is at the Helpdesk for repair.

4.1 A Typical Helpdesk Call

Typical calls might be handled as follows:

A customer calls the desk and requests a password change. Our policy dictates that CNS Helpdesk students can change student passwords, but only if the customer comes to the desk and presents identification. A call is opened, but the password cannot be changed, and the call is resolved. If the customer is a student, they are asked to bring their ID to the helpdesk to change the password. If the requestor is faculty/staff, they are asked to bring their ID to either Telecom or the Faculty Support lab, where the password can be changed, and the call is closed. Finally, if the requestor is off-campus and at too great a distance to bring their identification to the Helpdesk, the call is logged and assigned to Telecom, where the password is changed and mailed to the customer address on file.

A customer calls the helpdesk and is having difficulty connecting to the network. The student employee logs the call, and after selecting the problem profile, is prompted to ask a series of questions to determine whether the problem is in the computer or with the network. Once this is ascertained, the problem is dealt with by the student over the phone or at the desk, or else the call is assigned to a different support group to check the networking equipment. At this point, the call will either be assigned to full-time networking staff or Telecom

4.2 Pitfalls and How to Avoid Them

The major pitfall of the new software was the call profiles. For example, I originally created call profiles for hardware problems, and named them Warranty Hardware Repairs and Non-warranty Hardware Repairs. There were choices under each, such as bad hard drive (warranty repair) and cracked screen (non-warranty repair). The CI computer last year was an IBM model that included an accidental damage claim, so the warranty/non-warranty demarcation did not apply. During the January break, these call profiles were recreated with a profile for each type of component that might fail, e.g., "Report Problem > Hardware > Hard drive" or "Report Problem > Hardware > Cracked Screen. "

Setting call conditions is a feature that we did not put into practice during planning but began using while in production. Call conditions are color-coded flags that provide an instant visual of the call's status. For example, a blue circle is used when the call is on hold and ready for pickup; an amber circle means a computer is at the manufacturer for repair; a red octagon indicates a staff/faculty desktop is infected with a virus and must be disinfected.

The first year of implementation has been a success, with nearly 9,000 calls logged thus far. Customer reaction has been positive, evidenced by instances of self-service tickets increasing. Internally, the software has been favorably received in most areas. We plan to reconvene the committee to identify successes and failures within the system and fine-tune the software based on a year's experience.

5. REFERENCES

[1] Black, B., Saul, J., and Larsson, E. Helpdesk.drew.edu: Home growing a helpdesk solution using open-source technology *Proceedings of the 28th annual ACM SIGUCCS conference on User services*, Richmond, Virginia, Pages: 289 – 298