

ESUP-Portail Helpdesk: user support at establishment-level

Pascal Aubry



Alexandre Boisseau

IFSIC / University of Rennes 1, France

pascal.aubry@univ-rennes1.fr

CRI / University of Brest, France

alexandre.boisseau@univ-brest.fr

Currently, helpdesks are shackled by a negative reputation that depicts them as staffed by useless incompetents whose management is inept and expensive. This view reflects the faltering debut of user support services, whose objective had not yet evolved to encompass providing real service to users.

Translated from Cyril Roiron, study about the development of the helpdesk's decision-making function and its relationship to training, June 2000, tecfa.unige.ch/~roiron/recherche/, in French.

Abstract

The advent of University portals [1] accelerated a phenomenon that has been observed for a long time, notably the progressive replacement of locally-deployed applications (in entities, for instance) with centralized applications that permit a software development geared to and around a specific institution's Information System.

This transformation, which has already occurred for services (e.g., website and email) and equipment (e.g., for storage and/or backup), has not yet taken place in user support applications. esup-helpdesk [2], the helpdesk developed and distributed by the consortium ESUP-Portail [3], remedies this lack by natively integrating the multi-entity nature of educational and research institutions. This helpdesk has already been adopted by many universities, colleges, and research laboratories in France.

This article is based on the results of a study [4] conducted in 2004 by a French information technology (IT) working group who surveyed opinions about the distribution and use of helpdesks in the French educational/research community. These results provide a general idea of the perceived problems—both technical and human—encountered when implementing a helpdesk tool. In this article, we demonstrate how the helpdesk distributed by the ESUP-Portail consortium employs original solutions in an attempt to avoid these problems. Our goal is to convince the reader that the esup-helpdesk is not just another helpdesk application, but rather is a well-adapted response to the specific needs of the educational/research community. We also intend to promote esup-helpdesk at the European level, and elsewhere in the world.

Keywords: helpdesk, user support, flexibility.

1. Why be a hedgehog?

Some people, when presented with technological progress, curl up in a ball like a hedgehog, hoping that the



newfangled technology will simply go away if ignored. Though this may be reasonable for a hedgehog faced with a dangerous situation, the benefits of such behaviour for the technologically-reticent are more difficult to understand, especially for those who are convinced that technology-use makes sense. In an effort to better comprehend the apparent lack of interest in a technology whose utility is obvious (to its users anyway), a French information technology (IT) working group¹ completed a study in 2004 about the use of helpdesk technology in the French educational/research community. They surveyed 120 people in this community with regard to their opinions about the distribution and use of helpdesks in their institutions. The results of the survey should allow IT providers to calm the fears and address the issues of the hedgehogs in their target communities, and perhaps convince them to unroll from their little balls and explore the IT garden.

The 2004 survey results indicate that, despite the obvious usefulness attributed to helpdesks by those who use them, only 34% of the survey respondents actually use this technology, with only 6% of the rest having even tried to avail themselves of the service (cf figure 1).

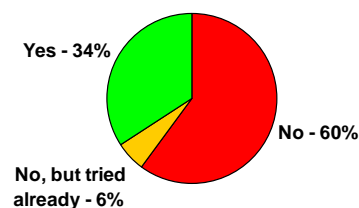


Figure 1 – GT-SSD study: do you use a helpdesk?

The reasons why so few people use this kind of tool are many, and it is rarely a lack of interest, given that only 3% of the respondents chose this response (cf figure 2).

¹ The working group GT-SSD was supported by CRU (Comité Réseau des Universités, www.cru.fr) and UREC (Unité Réseaux du CNRS, www.urec.cnrs.fr).

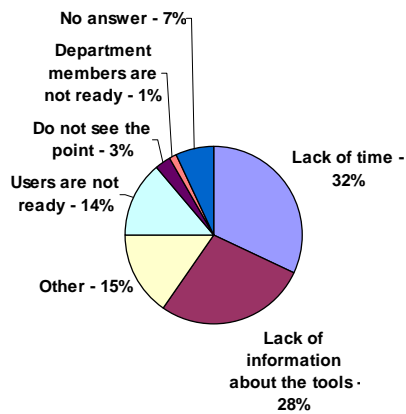


Figure 2 – GT-SSD study: why don't you use a helpdesk?

Of those who use helpdesk technology, more than 50% appear to be globally satisfied (cf figure 3) and would even recommend it to their peers (cf figure 4).

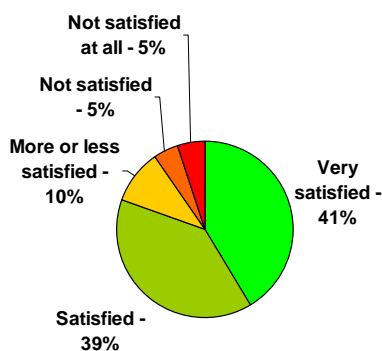


Figure 3 – GT-SSD study: are you satisfied by using a helpdesk?

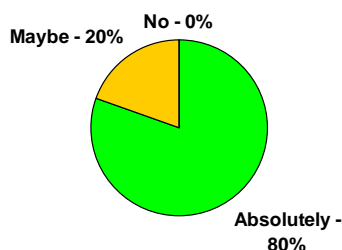


Figure 4 – GT-SSD study: would recommend using a helpdesk?

This positive impression is easy to understand given a closer look at the benefits attributed to helpdesk use (cf figure 5).

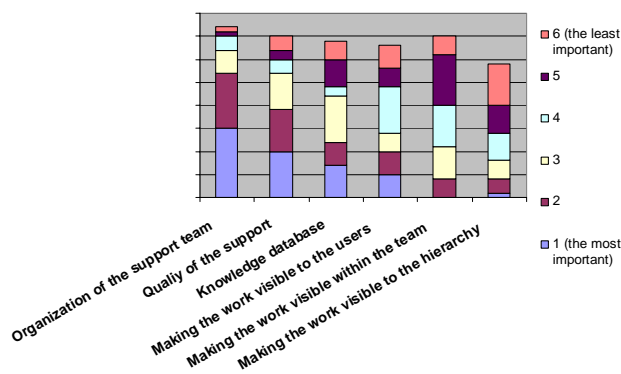


Figure 5 – GT-SSD study: the main benefits of helpdesks

However, introducing a helpdesk in an existing support team may be more difficult than it would first appear, with the path strewn with obstacles that are often more human in nature than technical (cf figure 6). The two principal obstructions appear to be related to the reticence of users, and of the support team itself.

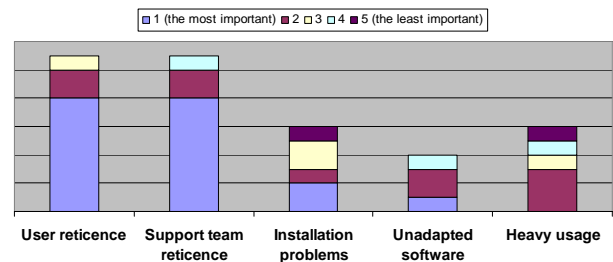


Figure 6 – GT-SSD study: the main issues with helpdesks

Thus, finding a solution is more difficult since human problems are always much harder to solve than technical problems.

Many applications for user support exist, and a number of them were evaluated in the beginning stages of the ESUP-Portail project. However, none of them responded to the specific needs of the target community. Either designed for computer scientists or without a French interface or missing key functions, all the existing solutions were rejected. Of course, the inherently multi-entity nature of the community had encouraged the development of individual solutions, and in fact, a good number were already in use or under development when the ESUP-Portail began, including *OTRS* [5], *Request Tracker* [6] or *Gédéon* [7]. However, none of these were truly appropriate for a universal user-support helpdesk. Based on the evaluations of what worked and didn't work, and modelled on an application running at the IFSIC (University of Rennes 1) since 2001, esup-helpdesk was developed to meet the specific needs of the target community.

The evolution of the esup-helpdesk project has always been driven by the same goal: increasing the comfort of the people that use the application. Though esup-helpdesk has only been implemented in universities over the past two years, it has benefited from five years of feedback about the older IFSIC application. This feedback has helped to provide technical solutions to the organizational and human problems highlighted in the survey discussed above, making us think that esup-helpdesk is the most appropriate response to user support needs in the Research/Education community. Through this article, we hope to convince the reader that esup-helpdesk is not just another helpdesk application, but is instead a versatile and flexible tool that is able to provide complete support for an entire establishment.

2. Internal organization and external view of the helpdesk

For a single tool to handle all the problems of an entire establishment, the tool must be able to deal with the differences between the departments. The only way for esup-helpdesk to achieve this goal was to make the

application extremely flexible, thus allowing it to be adopted by institutions that are very different in size (e.g., large universities, research labs, upper-level schools) and in organizational structure (e.g., a single “entry point” manager who distributes tickets to other managers, autonomous managers, changing roles).

2.1. Dividing the establishment into departments

The division of the establishment into departments must be:

- Flexible** enough to take into account any existing support structures;
- Discrete** enough to respect the natural independence of departments;
- Communicative** enough to allow tickets to pass from one department to another one.

To make esup-helpdesk adaptable to any organization, all methods for dividing establishments into departments are authorized. For example, some institutions chose to divide by entity and others by campus.

2.2. Masking the complexity from users

Universities are complex, often made up of many different departments, and listing all of them for perusal when entering the application would be burdensome. It has proved useful to propose only the departments that really interest the users in the prompt that appears when entering a ticket. Thus, esup-helpdesk makes it possible to hide the departments from (show them to) users depending on the user profile and localization, the latter taken from their web browser's IP address.

Figure 7 presents a simplified version of how the University of Nancy 2 presents the support services to their users: attributes tell the application whether users are students or staff, and at which campus of the university they are located. This way, users see only the two or three departments they may need.

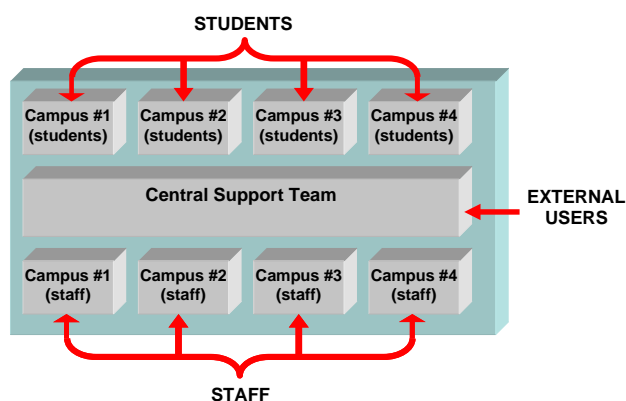


Figure 7 – A simplified schema of support services at the University of Nancy 2

2.3. Using virtual departments

Masking the complexity of the establishments to users is important, but the esup-helpdesk also needed to present users with a view of the helpdesk, though this view would not necessarily correspond to the helpdesk's real internal organization. This was achieved by using what is called

“virtual departments”. Tickets created in such virtual departments are automatically moved to another “non-virtual” department.

Figure 8 shows how this feature is used at University of Rennes 1 to give users the impression of support proximity. The virtual departments are represented in the figure by dotted rectangles; tickets created in these departments are automatically redirected to the other “non-virtual” departments (opaque rectangles).

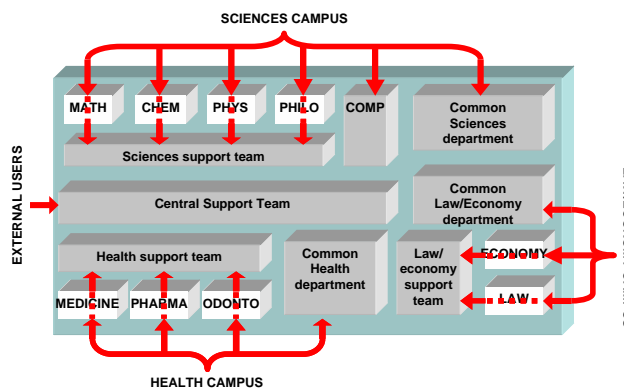


Figure 8 – A simplified schema of support services at the University of Rennes 1

Using virtual departments allows the real internal organization of the support services to be changed with no impact to the users, since the image provides to the users differs from the real organization. For instance, splitting a support team in two may go totally unnoticed by users.

2.4. Dividing departments into categories

All the software distributed by the consortium ESUP-Portail is intended to be used by thousands of users and dozens of department managers, a typical scenario for a French university. In other words, the esup-helpdesk was built to handle thousands of tickets. Since a flat structure was not appropriate for so many tickets, to facilitate the classification of the tickets, each department can be divided into categories. Like the division into departments, the way that categories are created is left totally up to the institution. Some use categories to map competences (e.g., web development, email and storage); others map buildings or operating systems, for example.

These categories can also be virtual, in which case, the tickets created in a virtual category are automatically redirected to another “non-virtual” category. Figure 9 shows how this feature can be used to create an impression of proximity to users, even if the tickets entered for some categories are not processed by the department that the user actually sees (dotted rectangles represent the virtual categories.)

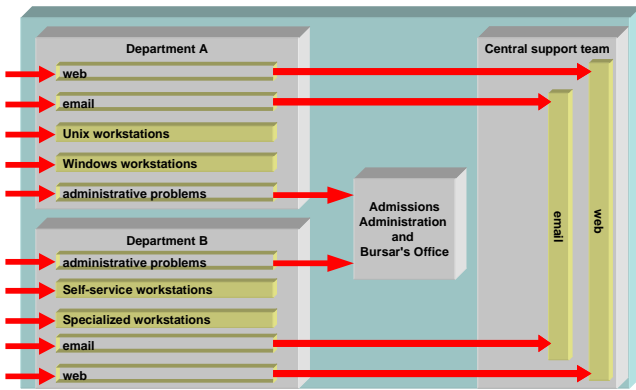


Figure 9 – Virtual categories

2.5. Implementing a customized vision

Departments and categories are created using the web interface and stored in the database. By default, all departments are visible by all users. Restricting the visibility of the departments, and thus creating a customized vision of the helpdesk, can be accomplished by editing an XML configuration. To provide very complex configurations, it is possible to define (named) conditions:

```
<define-conditions>
  <define-condition name="staff_user">
    <or>
      <attribute-eq
        name="employeeType"
        value="teacher"/>
      <attribute-eq
        name="employeeType"
        value="admin"/>
    </or>
  </define-condition>
  <define-condition name="stud_user">
    <not>
      <named-condition name="staff_user"/>
    </not>
  </define-condition>
  <define-condition name="stud_network">
    <fqdn-ends-with
      suffix=".stud.domain.edu"/>
  </define-condition>
  <define-condition name="admin_network">
    <or>
      <fqdn-ends-with
        suffix=".admin.domain.edu"/>
      <ip pattern="148.60.10."/>
    </or>
  </define-condition>
</define-conditions>
```

The above example defines the condition named **staff_user** (resp. **stud_user**), which is true when the user is a staff member (resp. student), and the meeting of this condition requires access to the user's attributes. Two other named conditions (**stud_network** and **admin_network**) are based on the user's client.

The named conditions can be later used to define the departments that user will see based on their profile.

```
<list>
  <list-rule>
    <condition>
      <or>
        <named-condition
          name="stud_network"/>
        <named-condition
          name="stud_user"/>
      </or>
    </condition>
    <actions>
      <add-by-label label="STUDENTS"/>
      <stop/>
    </actions>
  </list-rule>
  <list-rule>
    <condition>
      <named-condition
        name="staff_user"/>
    </condition>
    <actions>
      <add-by-label label="STAFF"/>
    </actions>
  </list-rule>
  <when-empty>
    <add-all/>
  </when-empty>
</list>
```

In this example, students and other people located on the student network will see the departments with the label **STUDENTS**, staff members will see the departments with the label **STAFF**. If no department is shown after the first two rules are evaluated, then all the departments will be presented.

Finalizing the configuration is easy since the rules can be tested before they are effectively used.

Any helpdesk department manager will now ask how long it will take to get a configuration that is well-adapted to an already existing environment. The answer to that question is that it takes time. However, although installing and setting up a helpdesk is a project itself, it is not always the most time-consuming part of the undertaking. Most of the time is not spent on the technical configuration, but rather on laying the groundwork before and making user comfortable with the solution after. Figure 10 presents a schematic of the time flow when introducing a helpdesk.

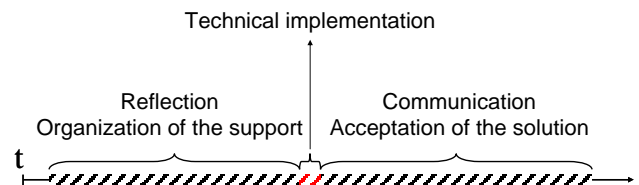


Figure 10 – The time flow when introducing a helpdesk

As Figure 11 shows, once the organization of the support service has been determined, installing and setting up esup-helpdesk to fit the organization typically takes around one week. In this, esup-helpdesk is on a par with other tools. In any case, the time spent setting up a helpdesk is quite insignificant when compared with the benefits accrued.

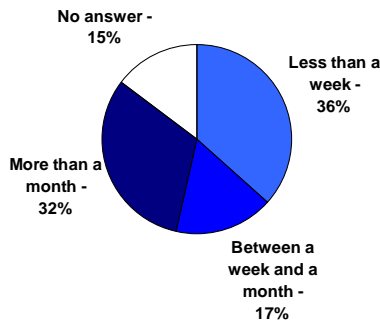


Figure 11 –GT-SSD study: the time spent setting up a helpdesk

2.6. Formalizing the competences

Introducing a helpdesk tool within a support team provides a real opportunity to formalize the competences of the team. Specifically, mapping categories to competences and defining the members of the categories (i.e. determining who will follow up on tickets corresponding to certain competences) offers a chance to detect weaknesses as well as redundancies.

3. The main goal: user satisfaction

The main goal of a helpdesk is user satisfaction. The quality of the service rendered by esup-helpdesk is provided by:

- Quick, low-problem ticket input, by offering pre-completed tickets, automatically assigning tickets to managers, and installing a high-quality knowledge database;
- Good priority management;
- Real-time follow-up of tickets;
- Clear identification of a unique speaker for each ticket;
- A universal and ergonomic interface;
- The possibility of confidential exchanges.

3.1. The input of tickets

From the user's point of view, a well-resolved problem is a quickly resolved problem. Thus, one of the keys to success is having all the elements needed to resolve problems as soon as tickets are created.

To limit the number of ping-pong exchanges following the input of the tickets, esup-helpdesk successively invites the user to choose:

- A department, which is supposed to process the ticket. At this point, only departments visible to users are presented, making the user's choice quite simple.
- A category, corresponding to the competences needed to solve the problem, for instance.
- A pre-completed ticket, which corresponds to a certain ticket pattern.

For example, a user chooses "Library" for the department, "Printing problem" for the category, and "No more paper" for the pattern. With this information, the helpdesk then needs only to ask for the name of the printer. Once this complementary information is supplied, someone can go to the printer and fill it with paper.

3.2. The automatic assignment of tickets to managers

Often the thing that takes the most time is deciding who takes charge of a new ticket. To reduce this delay, esup-helpdesk tries to automatically assign tickets to a manager at the time of creation. Predefined algorithms respond to the most common choices: assignment to the first available manager, cyclical assignment, random assignment, assignment to the manager that has been assigned the smallest number of tickets, to name a few. However, it is also possible to customize the application by adding algorithms, for instance, writing a Java class to link with an external calendar application.

3.3. Priority management

Without a helpdesk application, user support is traditionally done by email; problems are solved via the exchange of emails between users and managers. Many strategies for managing priorities are available, essentially depending on the manager's mood at the time the problems are processed: FIFO, FILO, SISO² or SINO³.

It is commonly agreed that priorities can not be managed correctly if the support service is not completely separated from the email inbox.

3.4. Email alerts

On the other hand, email is quite efficient for informing managers about ticket-related events. Therefore, esup-helpdesk allows users and managers to be contacted by email when something untoward happens with a ticket. To avoid problems due to SPAM, which quickly lead to managers redirecting all email alerts to a folder near */dev/null*, managers can specify in their preferences exactly when they want to receive alerts, for any/all department/s that they manage (cf figure 12). Users can also set their preferences to define when they want to receive emails (cf figure 13).

Ticket monitoring for department IFSIC

- Warn me when a ticket is created in a category of which I am member.
- Warn me when a ticket is created (in any category).
- Warn me when something happens on free tickets.
- Warn me when something happens on tickets I am managing.
- Warn me when something happens on tickets with no category.
- Warn me when something happens on tickets in a category of which I am member.
- Warn me when something happens on any ticket.
- Warn me only if priority is at least: LOW: managers will look into this problem when all else is fine

Figure 12 – Manager preferences for ticket monitoring

Automatic ticket monitoring

Choose the tickets you will automatically monitor:

- Warn me when something happens on a ticket I have created.
- Warn me when something happens on a ticket I am concerned with.

Figure 13 – User preferences for ticket monitoring

² Sometimes In, Sometimes Out.

³ Sometimes In, Never Out.

When correctly and astutely configured, these email alerts permit the real-time follow-up of tickets, without rampant inbox stuffing.

Please notice that email alerts are fully compatible with separating email from the support service itself. Email alerts sent by esup-helpdesk in order to:

- Warn of an event;
- Inform about the content of an event (the history of the ticket is sent along in the body of the message);
- Propose a hypertext link to the application.

It is impossible to reply to the alerts, due to an invalid *Reply-to* header. Thus, all the exchanges are done via the web interface and enrich the knowledge database.

Let us notice that all the email alerts sent to users are memorized in the database, for traceability reasons (a user can not say afterwards that he has not been warned of an event on a ticket).

An “answering machine” feature also exists; esup-helpdesk can be programmed to regularly read an IMAP inbox and to automatically transform received messages into tickets.

3.5. A unique and identified speaker

Ticket follow-up in esup-helpdesk relies on the designation of a unique speaker for each ticket, who is identified as such. The only thing more irritating for users than receiving no response to a request is receiving several conflicting responses. Worse, in addition to irritating the user, such multiple response projects a deplorable image of the support team. Clearly defining the manager who is responsible for a ticket prevents these multiple responses and saves time.

Users are often afraid by the fact that user-manager exchanges are dematerialized (faceless), since they prefer human contacts. This clear identification of a speaker for each ticket counters this natural fear. More generally, all esup-helpdesk actions are nominative and logged; nothing is anonymous.

3.6. A user interface for everyone

When well-written, web applications are wonderful because anybody can use them. If you do not believe me, ask your mother-in-law to order a train ticket on the Internet (Note: I did, and she succeeded).

Yet, designing a good user interface is not easy. Remember how long people had been forgetting their credit cards in cash dispensers before ergonomists reasoned that this would not happen if users were invited to take their money after they retrieved the card.

According to its users, esup-helpdesk is easy to use because it is very intuitive. Though no ergonomist worked on the project, the interface was continually improved in each successive iteration until it reached the current stable version, shown in figure 14. User requests contributed greatly to the application's early evolution; most of the time, these requests focused on details, but it is these details that provide to the application's overall ergonomics.

Initially developed for large institutions, such as universities, esup-helpdesk is now used in smaller institutions by both technical and non- technical people. It is so flexible that it has been adopted by high-level research and educational institutions, in addition to the universities for which it was originally developed.

3.7. Visibility and confidentiality

Case law shows that electronic mail will probably be the last bastion of privacy. Most users know it, and it is often the reason why they hesitate before choosing another tool to resolve their problems. There are several reasons, and the main one is the fear that their problems will be exposed to the community.

To help resolve this crucial point, esup-helpdesk introduces the notion of confidentiality—at several levels. A ticket can be totally private (i.e. visible only to its creator and the managers), partially visible (i.e. it appears on the control panel, but its content is visible only to its creator and the managers), or totally visible (i.e., visible to all users). Similarly, all the actions taken with regard to

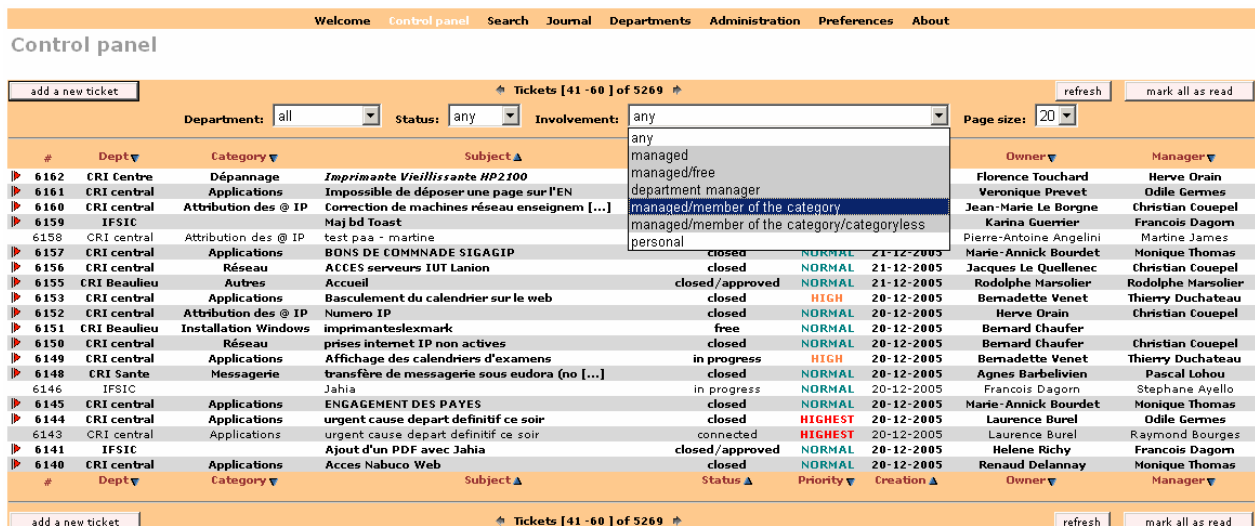


Figure 14 – A user interface for everyone

tickets can be protected so that they are available to all users, to the managers and the ticket creator only, or uniquely to the managers. Given this multi-level confidentiality, even the most critical information, such as passwords, can be entered into the helpdesk. Generally, it is recommended that this confidentiality feature be used with discretion because protected tickets are not shown, which sometimes leads to multiple tickets for a single incident. In addition, only the tickets (and the resulting actions) that are visible to all users are indexed.

3.8. The knowledge database

Because accessing the content of **closed and opened tickets** is an essential helpdesk contribution, esup-helpdesk performs a full-text indexation of all the tickets in the database. A search interface with advanced search criteria is available for all users. This interface allows users to search for a solution to their problem themselves before creating a new ticket. Thus, the knowledge database contributes to user satisfaction, since it allows problems to be solved by the users themselves, even before any ticket is posted.

A FAQ (**Frequently Asked Questions**) manager is also provided by esup-helpdesk, with a fully-integrated FAQ editor (based on the well-known Wysiwyg FckEditor). The main advantages of storing the FAQ contents in the database (instead of using an already existing external FAQ manager) are:

FAQs are displayed in the same web interface; managers can identify which FAQ entry corresponds to a user question in just a few clicks, and close tickets by redirecting users to a FAQ answer.

The visibility of the FAQs can be restricted at any level, according to the role and profile of users (as is done for tickets).

FAQs are indexed close to tickets so users can launch searches on both tickets and FAQs, as shown in Figure 15.

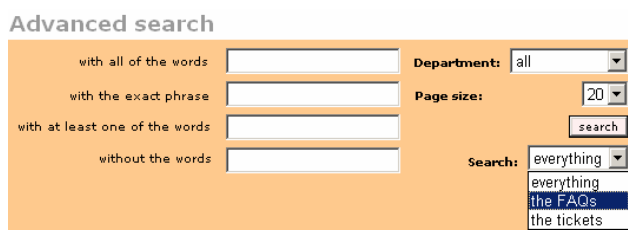


Figure 15 – The advanced search interface

At least, the FAQ contents can be exported (to XML and PDF) through a web service in order to link FAQs with others tools such as a CMS.

4. Other goals

4.1. Making the work visible

It is important to make sure the work being done can be seen by the three constituent communities:

Users usually appreciate being shown the work being done by a support team. This generally results in an overall improvement of the relationship between users and the support team. When they see that managers

are working on problems of general benefit to the community, some users will even decrease the priority of their individual tickets. This disclosure is fully compatible with the confidentiality offered by esup-helpdesk, exposing the work, given that actions containing critical information can be hidden for unauthorized users.

Support teams also benefit from work visibility. Sharing tickets appears to promote communication between managers, who also benefit from the wider knowledge database.

Institutional hierarchy can become more sympathetic to support service request when the key people see the work being done. Increasing visibility can, for example, help to justify request to increase the support team personnel.

4.2. Integrating esup-helpdesk into the Information System

esup-helpdesk is integrated into the Institution's Information System at several levels. CAS (Central Authentication Service [8]) protocol, which has been implemented in almost all the French Universities, is used for authentication, and adopting this SSO mechanism insures the *de facto* integration of esup-helpdesk into the rest of the institution's software. Since esup-helpdesk is a uPortal channel, universities that choose to use uPortal [9] to implement their portal can achieve an exceptionally complete integration. The user database—commonly an LDAP directory—is accessed via uPortal, which allows user attributes to be read and used to create the user profiles (cf 2.2). esup-helpdesk also provides an LDAP gateway that allows user data to be easily collected from the establishment's LDAP directory.

4.3. Supporting local and outside users

As mentioned above, esup-helpdesk is used by local users (i.e. users of the establishment), who are authenticated by CAS. However, esup-helpdesk also allows outside users to be authenticated, via a password sent by email. Sites with no LDAP directory already use this type of authentication for their local users, under production. This esup-helpdesk functionality means that a single tool can effectively manage the entire support effort, including both local and outside users.

5. A tool to support users or to supervise managers?

Some fear that the primary motivation for introducing a helpdesk within a support team is to collect data that will allow the service rendered to be evaluated, and will sometimes be used to justify reorganizing "underperforming" teams. However, though a helpdesk database obviously provides information about a team's activity, our experience shows that it is very hard to use that information to draw conclusions about the efficiency of a team. Many criteria must be taken into account, and the results are very hard to interpret.

The database of esup-helpdesk is very rich. For instance, it is possible to force managers to reveal the time spent on

ticket closure, at department level. Many statistics may be extracted from this data, including the time needed for users to receive an answer after creating a ticket, the time needed to resolve the problem once the ticket has been examined, the time actually spent on a ticket, and the number of tickets opened simultaneously. In response to user requests, the production of statistics is currently being developed in esup-helpdesk. However, no statistics about individuals will be produced, as the developers want esup-helpdesk to remain what it was originally designed to be: a tool to help user support managers.

The risk of transforming a helpdesk into a supervision tool does exist, and managers should be aware of this. The potential for abuse should be clearly discussed with the institution's hierarchy before setting up a helpdesk application. However, we think that it would be silly reject helpdesk technology on this basis, since the risks are really minor compared to the benefits.

6. The esup-helpdesk software

esup-helpdesk is part of the software distributed by the consortium ESUP-Portail, which is a complete, open-source solution for colleges and universities that wish to offer their students and staff integrated access to their information and services.

6.1. History of the project

As mentioned in the introduction, esup-helpdesk was derived from an older project, begun in 2001 at the IFSIC, the computer science department of the University of Rennes 1. Figure 16 presents the time line of this project development.

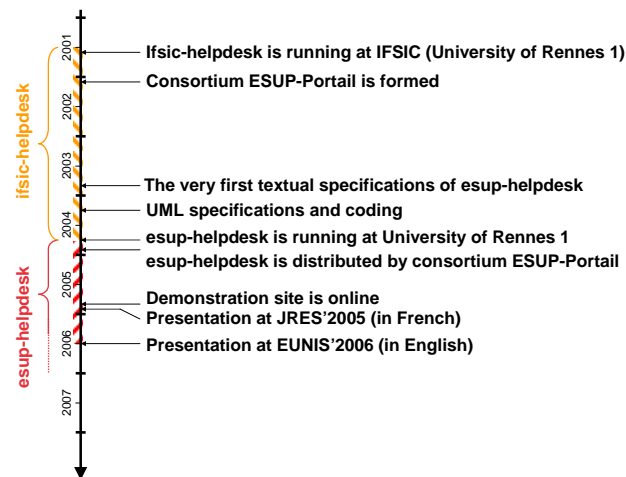


Figure 16 – The history of the project

6.2. An open-source software, using J2EE standards

esup-helpdesk is a UML-specified 100% Java software, distributed under LGPL licence. It uses only open-source standards (e.g., XHTML, XML, XSLT) and J2EE technologies, including:

- Xalan/Xerces** for the XSLT processing of the HTML rendering;
- Lucene** [10] for the full-text indexation of the tickets;
- Axis** for the Web Services.

esup-helpdesk also uses **Hibernate** [11] for all database access, which allows:

- abstraction of the database manager, since Hibernate relies on JDBC;
- abstraction of the data itself, since the code only manipulates objects, whose persistence is handled by Hibernate;
- performance optimization, notably via caching.

The previous version of esup-helpdesk (2001), written in PHP, struggled to deal with the thousands of tickets involved in the system. The new Java version processes the required number of tickets with ease, probably due to the Hibernate software.

Since version 2.0 and the usage of Hibernate3, esup-helpdesk is known to work with Oracle, MySQL and PostgreSQL.

6.3. Architecture

esup-helpdesk is provided as a uPortal channel, which means that it is an application that can be directly integrated into a uPortal instance. uPortal, the portal developed by JA-SIG, is a J2EE servlet, generally served by Tomcat.

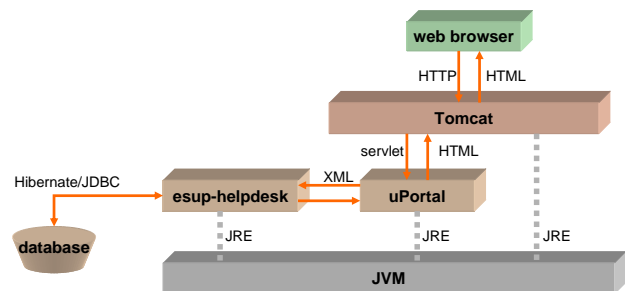


Figure 17 – The global architecture of esup-helpdesk

As shown in figure 17, the architecture is quite complex, notably when compared to applications developed on lighter frameworks such as LAMP (Linux, Apache, MySQL and PHP). It is, however, very robust. Often, establishments deploying uPortal generally use clustered architectures, in which case esup-helpdesk also benefits from the inherent fault tolerance and load-balancing.

Aware that the architecture shown above is complex and, thus, inaccessible to some, esup-helpdesk developers, who want to share their product with as large a community as possible, also offer esup-helpdesk as a stand-alone application: the Quick Start package (cf figure 18). Please note that the Quick Start package is not, as is often thought, a downgraded environment intended for test purposes: several sites have used this package to install the tool without worrying about installing a uPortal architecture, and exploit esup-helpdesk as is.

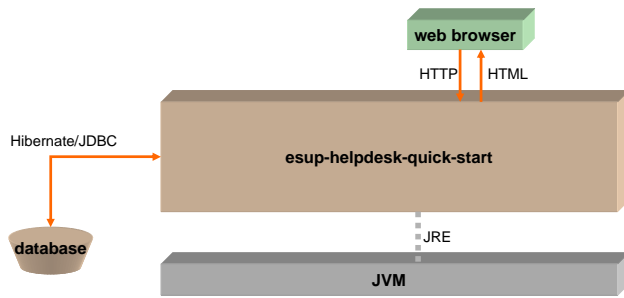


Figure 18 – The architecture of the Quick Start package

This package provides:

- The current esup-helpdesk uPortal channel,
- A uPortal instance to run the channel, and
- A Tomcat instance to deploy the uPortal servlet.

The Quick Start is very easy to use. Thanks to this package, esup-helpdesk is now used by people who did not even know *ant* [12] before, who still do not know either Java or Tomcat. Installing and upgrading esup-helpdesk can be done with a few simple commands. Developers answer support requests sent to the support mailing list of the project helpdesk-support@esup-portail.org.

6.4. Deployment

esup-helpdesk is in use in several research labs, universities, and university-level schools, and is being tested in other places. As the majority of the French universities have now chosen esup-Portail for their portal software, we hope that they will now adopt esup-helpdesk. Figure 19 shows where in France esup-helpdesk has been implemented or is under consideration (on April 2005).

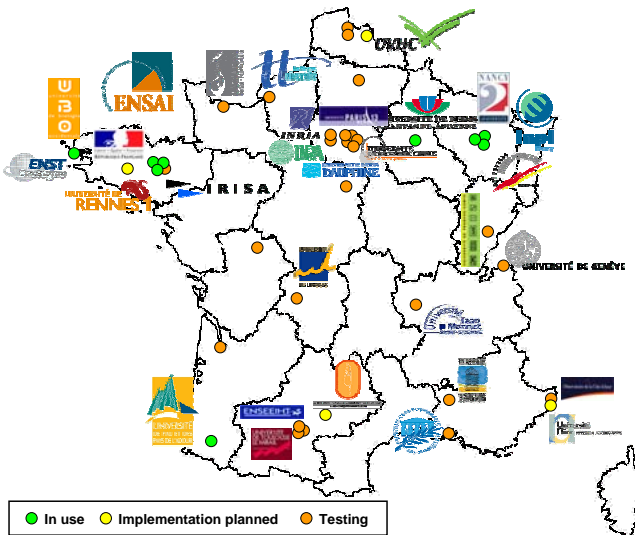


Figure 19 – The deployment of esup-helpdesk

The web interface has been made accessible to an international community, and is currently available in English and French⁴. The project recently affiliated with the JA-SIG ClearingHouse [13]. It is currently being

⁴ Adding a new language is simply achieved by adding a properties file, also called a resource bundle.

tested outside of France, and we hope that it will be distributed and used outside of France in the near future.

7. Future work

esup-helpdesk is a collaborative development that integrates any contribution that is of general interest. Feature requests can be made by contacting the mailing list: helpdesk-users@esup-portal.org. These requests are validated and added/integrated by the developers when possible. There is also a French-language mailing list: helpdesk-utilisateurs@esup-portail.org.

The timeline for the future development of the project, presented in figure 20, can be consulted on the website dedicated to the project: <http://esup-helpdesk.sourceforge.net>.

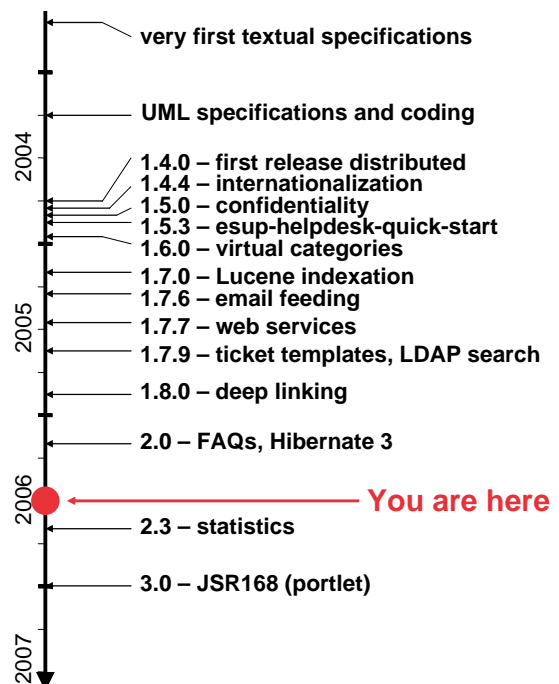


Figure 20 – The future timeline for the project

7.1. A statistics generator

This statistics generator is the one of the most frequently requested by users. It is currently under development and will be available in version 1.10, planned for June 2006. The statistics will be presented on the web interface and access will be given to users according to their role and profile. An XML export is also planned to allow the statistics to be used outside of the application.

7.2. Other changes

Esup-helpdesk version 2 uses the uPortal API to communicate with uPortal. In order to run in any portal, version 3 will comply to **JSR-168** [14] and thus become a *portlet*.

Version 3 will also use higher-level development tools (notably the **Spring framework** [15]) to improve the quality of the software and ease the integration of new contributions. It is planned for the end of 2006.

8. Long is the road (to the helpdesk)

Over the past two years of esup-helpdesk distribution, we have met many colleagues who are trying to promote helpdesks in their establishments. As we draw this article to a close, we can't help but think of these people, whose colleagues roll into a tight little ball at the mere thought of changing their work methods. These human hedgehogs, who on the one side denounce the burden of user support and on the other refuse any new tool, are apparently widespread. Clearly, introducing a helpdesk application becomes much easier when it is encouraged by institutional policy.

There isn't much data about the use of helpdesks. The results of 2004 survey mentioned in the introduction indicate that 66% of the respondents had never used a helpdesk and only 2% had used one for more than 5 years (cf figure 21). However, by dint of talking about helpdesks, the idea is gaining ground despite all the hesitation.

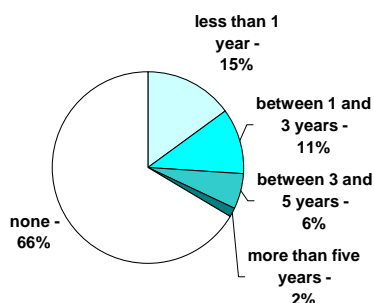


Figure 21 – GT-SSD study: for how long have you been using a helpdesk?

In any case, helpdesks can no longer be ignored; we will all use them in the coming years. We think it would be preferable to choose one ourselves today rather than being forced to use one that is less to our liking tomorrow.

Try esup-helpdesk at the demonstration site: <http://helpdesk-demo.esup-portail.org>!

Acknowledgements

We want to thank Anne Facq and all the members of the working group GT-SSD for completing their survey and for their permission to use the results on which this article is based [4]. Their work provides a much needed wake-up call to the French community with regard to the necessity of using helpdesks.

References

- [1] SDET : Schéma Directeur des Espaces numériques de Travail, www.educnet.education.fr/equip/sdet.htm, in French
- [2] ESUP-Portail Helpdesk, esup-helpdesk.sourceforge.net.
- [3] ESUP-Portail, a complete and open solution for colleges and universities desiring to offer integrated

access to services and information for their students and staff,

www.esup-portail.org.

- [4] Survey on the use of helpdesks, Working group GT-SSD, gt-ssd.crrp-bordeaux.cnrs.fr, in French.
- [5] OTRS, Open-source Ticket Request System, www.otrs.org.
- [6] Request Tracker, www.bestpractical.com/rt/
- [7] Gédéon, a helpdesk application, gedeon.u-bordeaux.fr, in French.
- [8] Pascal Aubry, Vincent Mathieu and Julien Marchal, Open-source Single Sign-On with CAS (Central Authentication Service). In Actes of EUNIS2004, Bled, Slovenia, July 2004, best paper award, published in the journal *Uporabna informatika* (Applied Informatics, ISSN 1318-1882, surveyed by INSPEC), ed. Prof. Andrej Kovacic.
- [9] uPortal by JASIG, Evolving portal implementations from participating universities and partners, www.uportal.org.
- [10] Lucene, a high-performance, full-featured text search engine library, the Apache Software Foundation, lucene.apache.org.
- [11] Hibernate, an object/relational persistence and query service for Java, www.hibernate.org.
- [12] Apache Ant, ant.apache.org.
- [13] JASIG ClearingHouse, sharing ideas and solutions, jasigch.princeton.edu.
- [14] JSR 168: Portlet Specification, www.jcp.org/en/jsr/detail?detail=168.
- [15] The Spring Framework, www.springsource.org.

The authors

Pascal Aubry played with real-time systems at ECP until 1993. In the years that followed, he worked on the distribution of synchronous programs at IRISA, receiving his Ph.D. in Computer Science in 1997. Now at IFSIC (*Institut de Formation Supérieure en Informatique et Communication*) at the University of Rennes 1, he manages web-projects. He has been part of the ESUP-Portail project since its inception in late 2002, working on web security (SSO, authorizations) and data storage. He is the esup-helpdesk project manager.

Alexandre Boisseau graduated from the University of Rennes 1 in 2004 and is now in charge of implementing the ESUP-Portail software at the University of Brest. He has participated in the esup-helpdesk project since it began in 2004, and has written many critical parts of the code.