Leading IT Service & Support July/Aug. 2006

HDI Support World

SM

Business Continuity Planning is Going to the Birds:

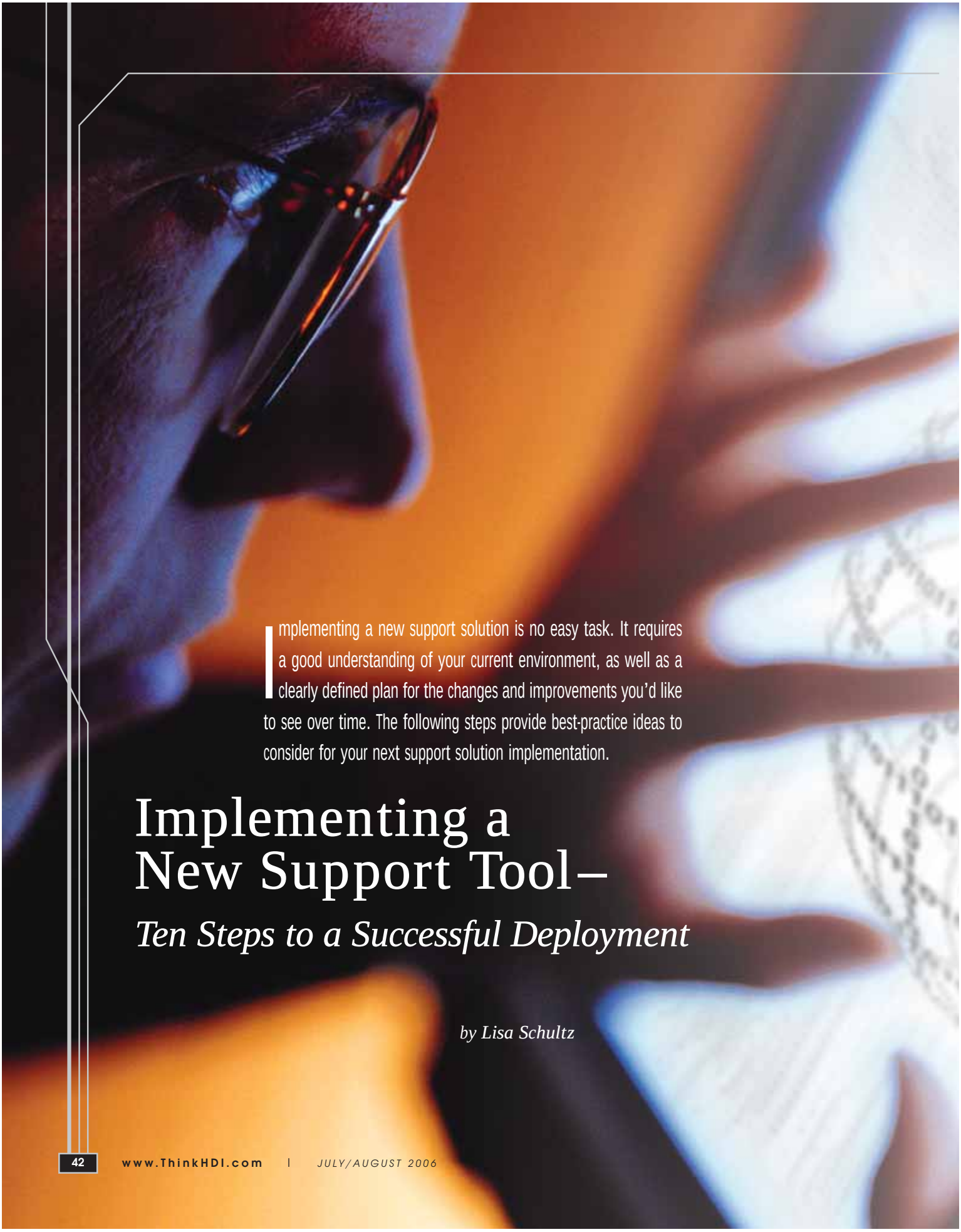
Getting IT Ready for Bird Flu

Good Enough
Knowledge Mgmt.

Uncle Sam's Computer's, Part 2
Strategy

Seven Must-Do ITIL Processes to Successfully Transform Your IT Organization
ITIL

Darth Vader Works at My Help Desk
People Mgmt.



Implementing a new support solution is no easy task. It requires a good understanding of your current environment, as well as a clearly defined plan for the changes and improvements you'd like to see over time. The following steps provide best-practice ideas to consider for your next support solution implementation.

Implementing a New Support Tool—

Ten Steps to a Successful Deployment

by Lisa Schultz

1 Involve your users and get buy-in. Interview users to determine their needs and work patterns. Determine commonalities that will enable you to create templates for repeated calls and correspondence. You may also discover the need for custom fields that can capture information specific to your organization. Work with managers and executives to identify the summary information needed, the reports used in the past, and those available in the new tool. Keep in mind security levels needed for the different types of users.

2 Understand and document how you work and how you want your support representatives to work. Generate a workflow chart of your current call process and then compare it to the expectations of the process when utilizing the new support application. This can help you determine the options to configure and the training issues to address.

3 Don't be afraid to change what isn't working. This can be a good time to change routines; your analysis may prove that you can automate inefficient processes with features such as asset scanning, incident and correspondence templates, and scripts to save time.

4 Spend some time on drafting your category structure. When planning your category structure, decide on a basic set that applies to all incidents. It is useful to chart your category structure to ensure that it is logical and complete; many companies use a large white board when brainstorming the category hierarchy. Involve as many people as possible in category brainstorming, and be sure to consult with your support representatives...they'll utilize categories more effectively if they are involved in the design process. You can come up with a draft list and then have the support staff try to apply the list to current open issues in their existing application; this enables you to target missing items and make adjustments to the hierarchy.

If you have an existing support application, if possible, run a report on incidents by category. Review the incidents assigned to any categories named "Other" so you can create additional categories if necessary. Determine the depth of the structure by examining your reporting requirements, and then look at what might be helpful from a knowledge base usage perspective.

Also consider that there is information that you don't necessarily need to report on. Category based custom fields can be utilized for this type of information. If you have global classification needs, consider using incident level custom fields to segregate your data into fields such as Work Requests/Tasks, Repairs, and Training Issues. Gathering this data can help you to understand where the majority of your service requests are coming from, and where either new equipment or user training might help to minimize calls. It will also give the help desk manager much needed documentation to provide to senior management on how the support team is being utilized.

Further, take a step back and think about what you are monitoring. For example, make sure your categories can track the issues your customers care about, as well as the issues that enable you to efficiently plan company resources. Categories can play a key role in generating reports that track trends and problems.

5 Plan your database conversion carefully. Think about the data you need to keep and move forward for reporting. If you are moving to a different type of database for the storage of your incidents, weigh your options of converting it yourself or contracting it out (which may require a separate project plan). Consider running parallel systems for a period of time; if you have an existing support tool, you may choose to manage all existing incidents in your old system and create new incidents using the new application. This enables you to transition without going through data migration.

Make a maintenance plan that includes a backup and restore plan and reports of your configuration settings. A backup 'paper' plan can be helpful in the event that your application is not available, so incidents can be created in the application program at a later date. Be sure to test the plan.

6 Use your analysis when setting up your Service Level Agreements (SLAs). Consider what your customers expect and what you have promised. Cover the various types of customers with multiple SLAs. When setting up notifications, think in maximums and averages, not the quick phone calls. Build in warnings so the support representatives know how much time they have. Plan for the long-term project related issues that can take up to a month.

7 Take the time to review your knowledge base. Knowledge bases provide a library of issue descriptions and resolutions, and it's important to keep it simple with quality over quantity. Use your incidents to create an informative base of knowledge, and appoint an individual or group of people for the review and editing of incidents submitted for approval to the knowledge base.

8 Implement functionality gradually. Don't implement too much too fast—new tools generally mean lots of new features, but turning them all on at once can spell disaster. People deal with change differently. Focus on core needs first. If you're looking at adding a lot of new capability with the new tool, consider starting with the basic incident functions and then, perhaps thirty days later, add some higher level functions like knowledge management or workflow templates. After another thirty days, look at adding in asset management. Then, when everything is running smoothly for your help desk staff, enable some self-service functionality for your customers.

9 Develop a training plan. Plan to train managers and support representatives on all of the functionality that you choose to implement, and develop a separate plan for training your end users (if you feel that such training is necessary). The configuration setup in the training environment should mimic what will be used in the active support application. Set up some realistic sample data for the trainees to access...understanding is the best path to buy-in. If you've added new procedures with the implementation, make sure that everyone understands what, why, when, and how.

Include information on escalation and notification settings. For example, it will be important for the trainees to know that an e-mail notification will be sent to customers when an incident is opened and possibly again if the incident is not resolved within a specified timeframe. Also include Service Level Agreement information, including hours of operation, types of service being provided, reporting responsibilities, and how customers will be contacting the support group (by telephone, voice mail, e-mail, etc.). This enables you to discuss service expectations for the support staff and how those expectations are configured.

Implement a pilot phase; you may choose to start with a group of support representatives and have them "road test" the configured application and document their procedures in their environment. These users can become departmental user experts so users aren't dependent on too few people for help.

10 Advertise your self-service page. When you roll out to the end user community, it's important to educate users about how to access the self-service page and spread the word about it. Avoid "hallway support." Support representatives need to say, "I'm busy right now, but it's easy to submit an incident ticket using the self-service page," and then follow through...word of mouth ensures success. Make your self-service page easy for your users to use and to access. Be sensitive to the jargon in your environment and use terms that your users are comfortable with. Keep your frequently asked questions and headlines up to date so users will have confidence in the site.

A happy ending. With the right blend of documentation and discussion, introspection and analysis, controlled rollout, and function awareness, your new solution's implementation process can be successful and take your support activities to a new level.

Lisa Schultz has been an integral member in GWI's application deployment team for the past five years, encompassing installation, training, and consulting of its flagship product, c.Support. Lisa has assisted in well over 200 help desk deployments in that time, and is well known for her unflappable education style in any support environment. Prior to her experience with GWI, Lisa managed an IBM Domino development team for a small consulting firm in Atlanta. She has worked in various functions in the customer service industry for more than fifteen years.

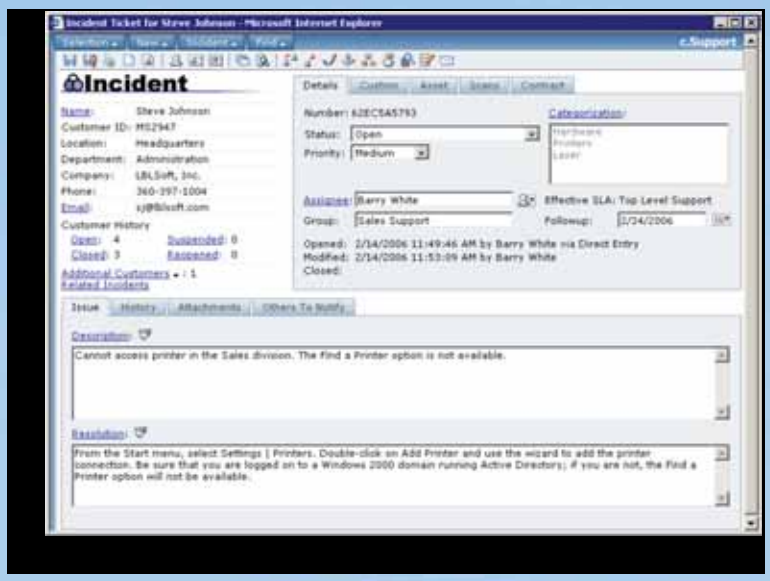
GWISoftware

FINALLY...

A support solutions vendor that gets it!

All-inclusive help desk solution! No hidden costs for the features you need! One license, one installer! Web-based with zero client footprint!

- *Incident Management*
- *Asset Management*
- *Knowledge Management*
- *Customer Self-Support*
- *Workflow Management*
- *Custom Reporting*
- *And so much more!*



Microsoft GOLD CERTIFIED Partner

Front Runner
This application is built on Microsoft® SQL Server™ 2005
www.sqlserverfrontrunner.com

www.gwisoftware.com