



---

# Evaluating ITIL Help Desk / Service Desk Software Top 10 Considerations

*Vance F. Brown*

*So many choices.... So much confusion ...* Where does an IT professional even begin when trying to evaluate the *Why's? How's? What's? and Who's?* associated with evaluating a *Help Desk / Service Desk / IT Service & Support* or *IT Service Management* software solution? Geez ... it took 4 different phrases to even describe the space, and I didn't even use *Call Center*. Oh, and by the way ... are we logging *calls, tickets, or incidents?* And very important terms for our industry have yet to be used - like *Incident Management, Problem Management, Change Management, Configuration Management, Release Management, etc., etc., etc.* With so much "management" in the terminology, one would think that evaluating software solutions for this industry would be simple ... *but it's not!*

It's hard to even know what word or phrase to type into a *Google* search. And software vendors must use pages of "keywords" for their online advertising. So it is no wonder with this rapidly evolving industry that it can feel overwhelming for an IT consumer to try to fairly compare the apples and oranges of what IT software vendors are offering. With all the changes, it also is difficult to determine what truly matters when trying to implement a world-class Help Desk.

For simplicity's sake, the term "Help Desk" is often used in this article as an umbrella for all of the above terms. Granted, this may not be currently accurate, but I have chosen to use the "Help Desk" phrase with "Service Desk" interchangeably, maybe for the last time in such an article, perhaps out of pure nostalgia .... Oh yes - *the good 'ol days!* But it definitely is time for all of us to embrace and celebrate the "Service Desk," and some of the wonderful changes that have come to this industry via revolutionary new technology, best practices, and standardized terminology.

The confusion in the Help Desk software industry is not surprising given the relative newness of the Internet and the *information technology revolution*. Because of the colossal innovations in the past 10 years, there is a lot of aged and cumbersome technology and exceedingly poor IT practices still embedded in the IT market. Only in the past couple of years has the convergence of this revolutionary technology together with recognized industry best practices become undeniably "disruptive" to our industry – but in this case disruption is for the good of our industry. Few would question the dramatic advances in technology – much of which has arrived because of the impact of the Internet. But today most IT experts also agree that the best practices or standards as set forth via the *Information Technology Infrastructure Library (ITIL)* are here to stay.

As a result of this phenomena, considering a Help Desk software solution that does not embrace ITIL would be imprudent, to say the least. However, most market-leading Help Desk systems were not originally designed with the latest technology - or with ITIL - in mind. They were great in their day, but times have changed, and Help Desk *nostalgia* is not a good reason to subject your organization to the inherent frustrations, inefficiencies, and high maintenance costs associated with aged technology.

Given the associated confusion inherent with the obvious changes in terminology, technology, and best practices, the purpose of this article is to remove the veil and some of the mystification and misconceptions that are commonplace when evaluating a Help Desk software solution. So ... here we go ... the top 10 considerations when evaluating a Help Desk software solution:

## 1. Implementation of Best Practices

ITIL is now the *de facto* standard for IT best practices. ITIL was initiated by the British government in the mid-1980s in response to the growing concern that multiple people - which were part of multiple Help Desks - were managing multiple IT infrastructures - in multiple places with multiple procedures. *Sound familiar?* These people were doing basically the same thing, but they used different terminology with inconsistent guidelines and practices. This resulting chaos created mass inefficiencies and the IT personnel were unable to communicate well between the various Help Desks.

In order to attack the problem, the British government joined forces with IT business experts to begin formalizing terminology and best practices for IT support services and management. One of the logical goals of this initiative was that an IT person in one location could easily move to another location without having to learn new terminology and localized best practices. So instead of having a “Help Desk” person responding to a “call” or “ticket,” under ITIL all “Service Desk” personnel were opening an “incident.” Just like an accountant can be hired by any organization with the confidence that the person already understands GAAP (Generally Accepted Accounting Principles), the IT industry is embracing the best practices and terminology using ITIL.

---

*“Purchasing a software system that does not fully embrace ITIL would be like hiring a CPA that has never heard of GAAP.”*

---

What has evolved over the past 2 decades, and which became more widely accepted in the late ‘90’s, is a library of books written by industry experts which sets forth a framework, procedures, and consistent language for handling the support and delivery of services for everything that impacts the IT Infrastructure (e.g. hardware, software, networks, technical documentation and training, etc.). The “IT Infrastructure Library” books and their respective updates currently are managed by the IT Services Management Forum (“ITSMF”), which has chapters in numerous locations.

According to ITIL (the “IT Infrastructure Library”), the “Service Desk” (not the “Help Desk”) is the part of the IT organization that is responsible for the key activities on a daily basis that are associated with the delivery of IT customer support. These activities include:

- Incident management – responding and communicating to the customer solutions to the IT issues.
- Problem management – finding the root cause of the issues and documenting and communicating the work-arounds.
- Change management – the process for evaluating, prioritizing and approving which problems should be fixed.
- Configuration management – Identifying and understanding the relationship between all the items that make up the IT Infrastructure.
- Release management – the process for rolling out new hardware, software, training, etc.

The ITIL Service Desk is the single point of contact between the customer and IT Service Management. Therefore, quality service support revolves around the successful implementation of the Service Desk. Admittedly, the definitions provided above are simplistic, and it is acknowledged that the ITIL definitions are much more complex. But these basic tasks and activities have been part of the role of the Help Desk for decades. But we in the IT software industry applaud the role ITIL has played in standardization of terminology, framework, procedures and practices.

*Now here's the point:* Purchasing a software system that does not fully embrace ITIL would be like hiring a CPA that has never heard of GAAP. Although many organizations have yet to implement the ITIL framework, most CIO's recognize that they should be heading in this direction. The good news is that today you can find Service Desk software solutions that will implement ITIL best practices - while training your personnel on the relevant ITIL guidelines and terminology - simply by using the system. In other words, you will be implementing ITIL in your organization by using the newer software systems that were designed with ITIL in mind. So the process of adopting and adapting to ITIL does not have to be overwhelming if you select the proper software system. With proper technology that originally was designed with ITIL in mind, the implementation of ITIL can be easy! And the benefits for your organization will be substantial.



## 2. Configurability/Customization and Integration

We must very quickly address a common misconception: *ITIL is a framework; it is not a step-by-step detailed methodology.* Therefore, a software system and its ITIL framework cannot be “hard-coded.” You still must be able to configure and implement your organization's detailed procedures. In other words, ITIL cannot tell you what type of incident should be the highest IT priority in *your* organization. It only sets forth a guideline that you should assign a priority to an incident and act accordingly pursuant to whatever service level agreement (“SLA”) you have established for a given CI (“Configuration Item” – i.e. a computer, workstation, software license, etc.).

Also, ITIL does not dictate the level of detail and granularity by which you track your configuration items, it merely recommends that you track all incidents by the associated configuration item, if applicable, and that all your configuration items together make up your Configuration Management Data Base (“CMDB”). ITIL also advocates, for example, that you match the incidents with the associated root cause, or “problem.” The ability to do “incident matching” to problems, known errors, changes, configuration items, etc. becomes crucial under ITIL.

*This leads to several implications when selecting a software system.* First, the system must be highly configurable (i.e. customizable). You should be able to configure your system to track any information you deem desirable. There should be no limits. And with today's “meta-data” driven technology, you should not have to use a programmer! You should be able to add unlimited fields to a configuration item, a customer, a problem, a change request, an incident,

---

*“Updates from the software manufacturer should be able to handle all of your configurations and customizations automatically – without the need for a programmer!”*

---

etc. *And this next point is crucial:* You should be able to set any relationship between these objects. Using technical terminology, you should be able to set one-to-many, many-to-one, and many-to-many relationships between your business objects (e.g. the “customer” object, or the “incident” object, or the “problem” object). Then, for example, when pulling up a particular “problem,” a user instantly should be able to see all the incidents that are associated with

that problem because this is a “one-to-many” defined relationship. Or when selecting a particular CI, one should easily be able to see all the incidents that have been recorded for that configuration item; or all the configuration items that are being used by a particular employee; or all the incidents and problems that are associated with a particular change request; ... *I think you get the point!* In other words, you should be able to create 360 degree views of all of the inherently related information without having to subject your staff to clumsy navigation caused by systems that added “ITIL Compatibility” by *hacking* it into their existing designs.

Modern systems should allow you to define the relationships between any business object, and you should be able to create any new business object without a programmer. For example, if you decide that you want to add a “purchase request” business object so that you can easily see all purchase requests associated with a given Change Request, this should be trivial with modern technology. You should also have the flexibility of modifying or extending the information you track at any time – not just when you first purchase the system.

*Additionally, any software updates from the software manufacturer should be able to handle all of your configurations and customizations automatically – without the need for a programmer!*

It is absurd to try to separate out the various activities of the Service Desk. The Service Desk activities associated with Incident, Problem, Change, Configuration, and Release Management inherently are seamlessly integrated with respect to the Service Desk – so the software should reflect this fact. The implication that Change Management, for example, is not needed by small or mid-sized companies is *hogwash!* (I originally used another word, but I think “hogwash” delivers the message.) All organizations have performed the tasks associated with Change Management for decades, but only recently have systems been able to seamlessly integrate the technological framework.

For example, when a new employee comes on board (which hopefully happens in both small and large companies), this naturally should create several Change Management tasks which you need to be able to define - including possibly ordering a new workstation, updating LDAP, notifying HR, etc. A new employee impacts people in various departments and impacts the IT Infrastructure. So, to purchase a system without integrated change management, for example, introduces inherent inefficiencies. The Service Desk is all about change, and modern systems can make change easy.

Therefore, software systems that do not seamlessly integrate all the activities that impact the ITIL Service Desk can be described by one of three words: *old, incomplete, or misguided!* If you purchase a non-integrated system, your Service Desk will be performing tasks that are not being recorded - and ultimately your IT organization does not get the credit. This makes it difficult when the IT organization needs additional resources to keep up with the organization's desired level of service and responsiveness. If the CIO does not have a record of all that the Service Desk accomplishes, it will make justifying additional resources problematic – and rightfully so.

### 3. Modern vs. Aged Technology

There truly have been some revolutionary advances with modern technology. For a long time we have been able to accumulate and store vast amounts of information – or data. We collect and store emails, faxes, white papers, documents, pictures, etc.; yet with old systems it is difficult to find, manage, and provide structure to the data. In other words, this ocean of data often is *not useful* to solve future problems.

Google has been given credit (and has made a lot of money!) for taking vast amounts of information stored across the Internet and making it useful to consumers. Why can't this same type of technology be utilized by an IT organization? *It can!* Modern systems use Google-like searching to find, structure, and utilize the data. For example, you should be able to pull up instantly a particular customer, incident, problem, change request, etc. Additionally, Knowledge Management has been revolutionized by this Google-like capability. Your Knowledge engine not only should be able to search the Internet or canned knowledge bases for solutions, but from the incident description, it should also be able to search past incidents, known problems, change requests, white papers, etc. We no longer should be storing data for data's sake. We now can use the data to solve problems!

Unfortunately, many organizations feel like slaves to aged systems because of the past huge investments that have been made. Old systems have become deeply embedded into the IT Infrastructures. These systems become like cancers that have developed claws or fingers, making it very difficult to cut it out of an organization. So the companies have to continually feed the cancer with vast resources (both personnel and cash) just to keep it alive. The bottom line is that modern systems do not need to cost the same amount as the old legacy. Newer technology can accomplish *much more* with much less money and human resources. Modern systems are more powerful, easier to configure, easier and less expensive to maintain, and the total cost of ownership is a fraction of the older systems. Yes, removing a cancer with fingers is difficult, but most often it is necessary for the long-term health of an organization. The longer you wait, the harder it will be, and the more resources will have been wasted.



## 4. Web-enabled / Multi-location

No one questions the impact of the Internet. It is responsible for the Information Revolution. It also is becoming the low-cost answer for how to access a single data source from multiple locations. Historically, one of the biggest frustrations and inefficiencies facing organizations was the inability to share a single database from different locations in a cost effective manner. So the solution offered by many software companies was to *synchronize* the data – a so-called “solution” that still sends chills through the hearts of software vendors and consumers alike! But very few small to mid-size companies could afford to establish their own wide area networks so that a common database could be shared ... Until the arrival of the Internet!

So the first “applications” that were developed for the internet to address the multi-location issue were similar to “mainframe” applications of the past. That is, the client computer is used basically a dumb terminal, or “zero-client.” In other words, all the information, computations, business logic, screens, etc. were executed on a central server and then sent via cyberspace to be painted (or displayed) on the local

client computer. A big advantage of these applications is that they did not have to be installed (or deployed) on all the local clients. Again, the client is basically reduced to a dumb terminal and nothing is running locally. On the negative side, however, the local CPU power available with our modern-generation personal computers is basically rendered useless. These early Internet/intranet-based applications are commonly called “browser applications,” because they operate via most modern browsers, including Internet Explorer, Firefox and Safari.

---

*“Most regular users of browser-based apps long for the days of the rich client experience of using a Windows application.”*

---

Anyone who has used a browser application eight-hours a day, five days a week, knows the related frustrations of having to wait on a slow internet connection or limited bandwidth. And most all regular users of these browser-based apps long for the days of the rich client experience using a Windows application. But at least these early browser-based business applications solved the problem of not having to synchronize data when using the application from a remote location away from the central server. And IT organizations applauded the fact that there were no deployment issues when a new version or update of the software became available, for only the central server needed to be updated.

*So what would it be like to take the best of both worlds ... i.e. the use of the Internet to share the data from multi-locations, but with the rich client experience of a Windows application? Enter Microsoft ... of course! After all, there is no way that Microsoft, who revolutionized the concept of having a powerful computing environment on every desk-top and in every home, is going to sit back and allow the world to return to a Mainframe computing model with dumb terminals. So Microsoft developed an entirely new programming environment, called .NET that was designed with the Internet revolution in mind. This new programming environment allows software vendors to take advantage of the best of both worlds. The user can have a rich client experience, but the data is transferred using the same protocols that are used by browser applications (e.g. SOAP and HTTP). These applications are much faster, because some of the business logic can be performed locally on the clients and bandwidth is not wasted by having to transfer screens*

over the Internet. And, of course, a Windows look-and-feel is a much “richer” experience for the user.

But what about the deployment issue? Both .NET 2.0 and .NET 3.0 are focused heavily on *auto-deployment*. Accordingly, relevant updates are automatically installed on the local client without human intervention. Of course, there are security issues with many of today’s business environments, but no doubt these issues will be resolved because of the numerous advantages of the rich client experience. This undoubtedly is the future of web-enabled software applications.

There remains a valid need for some applications to still be run via a common browser. For example, customers should be able to quickly log incidents without having to install a program on the client. After all, logging an incident should only take a couple of minutes out of a customer’s day. Even with web-enabled, rich client software applications, all software vendors should offer a browser-based “Self-Service” module so that the customers can easily log incidents or search the Knowledge Base for solutions - without having to install any application on the client computer. Similar arguments can be made for the ability to pull up basic information from a database on a PDA. But, once again, for these examples we are not talking about using a browser application for hours at a time.

So, does anyone think that the Windows rich-client experience is history? I, for one, am betting my money on Microsoft. You can bet against them at your peril! Accordingly, it would be wise to choose a Service Desk software solution that is written using the modern .NET framework and that is web-enabled - but that takes advantage of the rich-client Windows’ experience – *Yes, truly the best of both worlds!*



## 5. Scalability and the Various IT Software Markets

Historically, there have been three distinct markets for IT Service Desk Software:

1. The small business market – companies typically under \$25 million in revenue.
2. The mid-market – companies typically between \$25 - \$250 million in revenue.
3. The enterprise – companies with over \$250 million in revenue.

Of course these revenue dollar amounts are, for the most part, arbitrary. There is no trumpet that sounds when a company crosses some imaginary threshold and becomes “enterprise.” Additionally, dollar values do not necessarily define the IT needs at each level, but they do have impact when considering what companies can afford. Ten years ago, only the “enterprise” companies could afford systems which were highly scalable, highly customizable, and that could be integrated with other systems. And small to mid-size software vendors were left with the tagline: *easy to implement and use - low cost solution*.

Software vendors have been very aware of the fact that larger companies can afford to pay more. For many software vendors, why try to close a \$40,000 deal when you can play in the “enterprise” space and close a \$400,000 deal? It is not uncommon for an “enterprise” deal to cost ten times a “mid-market” deal when one includes both license fees and professional

services. This is why some IT software vendors who historically focused entirely on the small to mid-size business markets have looked to the “greener pastures” of the enterprise. One of the ways vendors have chosen to differentiate the higher prices is to imply that Change Management, for example, is only needed with “enterprise” solutions. Again, this is *hogwash!* But this is a common way to differentiate the marketplace from a pricing perspective. In the marketplace today, because of the technological revolution, spending more money does not always translate to getting a better solution for your organization.

---

*“There is no trumpet that sounds when a company crosses some imaginary threshold and becomes ‘enterprise.’”*

---

So today, is it really true that one has to choose between *low cost and easy to use* vs. *highly scalable and highly customizable*? Not any more! Yes, you now can “have your cake and eat it too.” The Internet can make applications (whether browser based or web-enabled rich client) inherently more scalable. And, as previously discussed, modern systems utilizing a *meta-data*

programming model can be infinitely customizable and less expensive to maintain because programmers are no longer necessary to deploy the required configurations.

*So what is left to justify the higher prices for “enterprise” software vendors?* Differentiate the pricing for the software by taking out needed ITIL Service Desk features and functionality. *Now for the good news:* Younger and more innovative software companies who do not carry the burden of older software companies with high-cost infrastructures are able to offer systems that are:

- Easy to install
- Easy and inexpensive to customize
- Easily maintainable
- Inclusive of the entire ITIL framework for the Service Desk
- Affordable to all companies.

Yes, there truly is a revolution in our industry, and some of the older software companies built on high fees for licensing, maintenance, and support will not like it .... But you will!



## 6. Company Focus and Product Stability

Focus ... focus ... focus.... Every good business book talks about it. Is anything more important than *company focus* to the success of a company and to the satisfaction of their customers?

I am fond of the scene in the classic movie *City Slickers* where Curly (the older and burly cowboy) has a conversation with Mitch (a city man searching for the meaning of life):

“Do you know what the secret of life is?” asks Curly.

“No. what?” Mitch responds.

“This!” says Curly as he holds up one finger.

“Your finger?” asks Mitch.

“One thing ... just one thing. You stick to that and everything else don’t mean SH\_T!”

“That’s great, but ... *What’s the one thing?*” asks Mitch.

“That’s what you’ve got to figure out!” responds Curly.

Unfortunately, many companies in this industry are choosing the wrong “one thing.” In the “good ‘ol days,” the founders and CEO’s of the leading software companies for the IT software industry were some of the pioneers of the industry. They were passionate about the IT space, and much their focus was on providing a world-class software solution for the IT consumer. They were focused on the customer.

More recently, it appears that the “one thing” in the IT software space has been on a company’s “exit strategy” – how to make the most money by growing a customer base and then selling the product and customer base to another company. Accordingly, the current industry for the IT Service Desk software market can be described by one word: *CONFUSION!*

Almost all of the market-leading companies for this industry have changed ownership during the past seven years, and some, multiple times. One company, which has a market-leading product for the mid-market, is currently owned by a venture capital firm, which makes long term focus very difficult. Another player, because of acquisition, has two products that both were originally designed for the same enterprise space. Which one should you buy?

Other large companies have acquired small companies with Help Desk products that have been around from the early days of the industry. They now pitch to the market that they have one product for “the enterprise” space and another product for “the mid-market.” Yet another company, which had the leading product for the small business market, recently bought a mid-market player. It’s reasonable to assume that they will try to move their “small business” customers to their new, more expensive, product line – of course at a cost to their customers!

Speaking from experience, maintaining two code bases and keeping up with the needed features and functionality are hard enough for one product line – but it is extremely problematic and confusing to maintain two product lines for the same industry. A common strategy is to attempt to move the customer base with the less expensive product line into the more expensive product line and to ultimately phase out, or at a minimum, decrease the R&D investment in the less expensive product line. As previously discussed, because of the recent technological revolution, the once-clear lines between the markets are dimming, at least from a technological perspective. So maintaining two technologies in the modern era for the same industry is devoid of focus.

---

*“More recently, it appears that the ‘one thing’ in the IT software space has been a company’s ‘exit strategy.’”*

---

*So what is the implication about this CONFUSED industry when shopping for a Service Desk software product? Just because a company has financial viability, and has been around for a long time, does not mean the Service Desk software solution you purchase from them is viable.*

Because of the rampant trend of company acquisition and merger, you may be working with another company within months – or the new company may decide to phase out your product entirely. Consider the age of each companies' product lines and code bases; consider how many times the product line you are evaluating has changed company ownership; consider how many product lines they carry for this industry; and pay close attention to whether the senior company leaders have vision, passion, and expertise in this space. Remember to ask the question, “*What’s the one thing?*”



## 7. Ease of Use

ITIL requires that *all* incidents be logged. That’s right – ALL OF THEM! But this is for good reason. So whenever a customer makes a communication with the Help Desk in any way, it should be tracked and recorded. However, many common Help Desk requests - like resetting a password – historically have not been recorded because it would take more time to log the incident than to solve it. The user can configure *today’s technology* (1) to run the program to reset the password; (2) to create and populate the incident fields; and (3) to send a confirmation email to the customer - *all with the click of one button!* It is even possible to perform all of these tasks, without human intervention, simply as an automated response from an incoming employee email. Of course security issues need to be considered, but the point is that modern software systems today can be configured to automate almost any series of mundane manual tasks. There should never again be a reason that the Service Desk does not get the credit for everything it does! This is not just to get a pat on the back ... the Service Desk is doing a disservice to the company when it cannot justify needed resources because it has not recorded all that it accomplishes on a daily basis.

Second, incidents or service requests that are not recorded and tracked easily fall through the cracks. With modern systems, you can set up “business processes” so that no recorded task or assignment ever gets lost or forgotten. If a task (such as ordering a computer for a new employee) has been overlooked, then the system could launch a reminder email to the person, or the issue automatically could be escalated to a manager. If an incident or service request is not recorded, you may forget about it until the spark becomes a fire that needs to be put out and that sends people into panic mode.

Additionally, often times the Service Desk is inundated with calls about issues they already know about. In other words, there is a “known problem” using the ITIL terminology. But the Service Desk may not log all the calls because the issue already has been reported, and besides, it takes too much effort to enter the new incident. Systems today should be able to create an incident for a known problem in seconds. And when there is a work-around to the problem, then everyone who called about the problem should automatically receive an email regarding how to solve the known error.

The point is that the modern systems should be easy to use and make it MUCH easier to do your job and to stay compliant to the ITIL standards. Do not purchase a software system that cannot easily automate the mundane and redundant tasks of the Service Desk.

## 8. Customer Service

C.S. Lewis once said, “*Put first things first and we get second things thrown in: put second things first and we lose both first and second things.*” So what’s the “one thing” – i.e. the “first thing” – to a company? The “one thing” and the “first thing” better consist of listening to and serving the customer. If a company cares for the customer, listens to their needs, and is responsive - then it will be able to build a great product and the company should be financially viable. If a company merely builds a great product but does not care for the customer, then the company will not be world-class – and the customers will be very frustrated. One of the best indicators of a company’s success is whether or not they follow the Golden Rule: “*Do unto others as you would have them do unto you.*”

For customer service to be successful in an organization it must be part of that company’s core culture – from the receptionist, to the sales person, to the customer service personnel. Customer service is not just a department - it should be part of the company’s internal fabric.

*So how can you discern whether a company is customer focused?* Consider some of the following questions: Is the receptionist polite? Is the sales consultant responsive to your needs, questions, and concerns – or do you just feel “sold to?” Is the company willing to guarantee that the product will work in your organization and that you will be satisfied? Does the company proactively communicate with you, or do you only hear from them when it is time to renew your support and maintenance contract, or when they want to sell you another product? If you have a problem with the product or company, and it is not being resolved in a timely manner, would the CEO be willing to take your call if you have followed the escalation process but the issue remained unresolved? If you are the “*first thing,*” then why wouldn’t he or she take your call?

## 9. Software Pricing Model and TCO

What are you really paying for an IT Service Desk solution? It is very difficult in today’s market to really compare the proverbial apples and oranges. But here is a general rule of thumb: *The more complex the pricing model, the more there are hidden costs.*

For example: Is there a separate “server fee?” Are you getting all the ITIL related activities of the Service Desk, including Incident Management, Problem Management, Change Management, Configuration Management, and Release Management? If not, how much will these “modules”

---

*“The more complex the pricing model, the more there are hidden costs.”*

---

cost you to add in the future? How much will it cost to configure the system? Is a programmer needed for customizations? How much will it cost for yearly maintenance, updates and support? How many people will it take to administer the system? Is there a separate fee for Self Service? Is there a

separate fee for LDAP integration? The list goes on ....

Understanding and comparing the “Total Cost of Ownership” for acquiring a system will be challenging, but this is an area where it is well worth the time spent. Just remember this: *Just because you are part of a large company, this does not mean that you need an “enterprise” solution.* You may be surprised by the power, ease of use, scalability, and affordability of a “mid-market,” lower-priced solution. Again, the lines between these markets, from a technological and best practices perspective, are growing very dim.



## 10. Intangibles

After the painful task of evaluating the different software companies and considering all the questions posed in this article, it is a good idea to trust your instincts – your gut! This may sound corny and a bit difficult for process-minded people. But it is so important! Who in the evaluation process did you feel most comfortable with? Which company do you believe is most trustworthy? These are important to consider, because you are not just selecting a Service Desk software solution, but you are choosing an ongoing business relationship. And just like any relationship, there will be problems. But which company do you feel will best respond to the problems in an appropriate, timely, and professional manner? At the end of the day, you will be working very closely with the company you select. So it is a healthy choice if you actually like and respect the people involved.



## Summary

There you have it! Hopefully some of the confusion has been addressed and you are ready to evaluate the various Help Desk products and make the best decision for your organization. You are shopping at a good time, because never before has there been more of a dramatic convergence of technological advances together with known best practices. Good luck in the process of finding a Help Desk (or should it be Service Desk?) software solution. Maybe it is time for me to fully embrace the “Service Desk!”



**Vance F. Brown** has been involved in the IT Service industry for over 10 years. From 1996 - 2000, Vance was President and CEO of *GoldMine Software Corporation* (formerly Bendata, Inc. and currently FrontRange Solutions - the makers of *HEAT* and *ITSM* Service Desk solutions, and *GoldMine* contact manager). Under Vance’s leadership, the *HEAT* product went from a rating by the Gartner Group as a “niche” player to the “market leader” in both “ability to execute” and “vision.” Vance currently is CEO of Cherwell Software, the developer of Cherwell Service Desk ([www.CherwellSoftware.com](http://www.CherwellSoftware.com)). Vance graduated from *Wake Forest University*, *summa cum laude*, with degrees in Economics and Computer Science. He graduated from law school, *with honors*, from the *University of North Carolina*, finishing as a member of the Order of the Coif and the Law Review. Vance would love to get your feedback about this article at [Vance.Brown@CherwellSoftware.com](mailto:Vance.Brown@CherwellSoftware.com).