Scaling IP-Telephony Systems Management

Andrew Hunkins

Use the FCAPS model to segment the challenge, and use the proper tools and techniques to manage each FCAPS component.

Size does matter when it comes to systems management. If you're working for a small organization and you don't have a significant number of management tasks, you can fully leverage the tools that come with your system. So if you're working for a small organization, this article is probably not for you.

The concepts discussed here are intended for those responsible for medium or large enterprises, and those interested in fully optimizing their telephony investments.

In the December issue of *BCR*, Gary Audin and Fiona Lodge explained an important model for systems management: FCAPS. You'll recall that FCAPS is an acronym for the five areas of systems management: Fault, Configuration, Accounting, Performance and Security.

I'd like to offer three concepts that dive deeper into the reasoning behind FCAPS and therefore, the power of it. This information will help you make the right choices about systems management as you migrate or expand your communications network into IP-telephony technology.

The Constituents Of FCAPS

Most professionals, when first exposed to FCAPS, see it as a logical separation of functions. And each area *is* a separate function: different tasks administered by different groups of people, each with different areas of expertise. Without this alignment of people you don't have FCAPS.

First, let's consider fault management. There are people in your organization who focus on keeping things up and running. If equipment breaks, or the network is overloaded, they fix it. They rely on fault management information to identify and trace problems.

Now consider a different area, like configuration management. There are people in your organization tasked with getting all your employees the services they need, provisioned as your business requires. Now realize that these configuration management people are not the same as the ones replacing switches or troubleshooting latency problems—at least not in most medium and larger enterprises.

Similarly, your security professionals are not running call accounting reports. Nor is your call center manager performing moves, adds and changes for all your employees.

Each of these groups of users has specific and separate needs. The FCAPS model is organized to meet the needs of tasks in each area.

This implies that a different management application is needed for each constituent group.

In small organizations, it's relatively easy to use the manufacturer-provided tools, or to find one tool that performs most of the main FCAPS functions. But for the more complex environment of larger organizations, no single product from any one company will sufficiently perform the functions in all areas of the FCAPS model.

Too many organizations have failed in their efforts to combine technologies in hopes of creating one super application to do it all. The user interface requirements are compromised and become ineffective.

Consequently, five applications is the ideal number, one for each FCAPS area. Any fewer and either an organization is unintentionally not following the FCAPS model, or is small enough such that more than one area can be managed by the same person. The latter scenario is fine, but the former must be avoided.

It is also a reality that medium-sized enterprises, and certainly larger ones, will have systems made by diverse manufacturers. This is usually because of merger activity, or because a certain brand excels in a specialized capacity. But as a result of this tendency, any attempt to reduce the number of management applications to the magic five usually hits the obstacle of requiring a single tool in each FCAPS area that's capable of managing systems from multiple manufacturers.

That's a significant obstacle, but still, the goal is to select vendor-agnostic tools within each FCAPS area. The good news is that sticking to the FCAPS model will make this much easier!

Vendor-agnostic systems management is best

Andrew Hunkins is the founder and CTO of Unimax Systems Corp., which provides configuration management software products and solutions for the enterprise. He can be reached at ahunkins@unimax.com.

achieved through the use of standard protocols. I know, many of you probably just sighed or chuckled when you read that, but the standards are reasonably aligned with the FCAPS areas:

- **Fault** management is supported by protocols like Simple Network Management Protocol (SNMP) and the Common Information Model (CIM).
- **Configuration** management is best performed using Simple Object Access Protocol (SOAP) or similar eXtensible Markup Language (XML)-based protocols.
- **Accounting** information is often collected by capturing simple log records or by Structured Query Language (SQL) queries into the managed system.
- **Performance** management is also done through logs, as well as reading specific values used as counters.
- **Security** management is the least developed area, but is evolving quickly and seems to be taking advantage of Service Oriented Architecture (SOA) technologies—specifically Web Services—and directory-based protocols such as Lightweight Directory Access Protocol (LDAP), supporting identity management functions.

Systems management architectures should attempt to narrow the use of protocols along the FCAPS boundaries. Organizations can survey the protocols offered by the systems to be managed, and compare these with the protocols supported by the tools expected to be used. The result will create a set of guidelines which can be presented to vendors for future purchases, as well as facilitating consolidation of an existing network.

Management Is A Two-Way Street

The next concept revolves around the meaning of the term *management*. True management is the ability to gather information, make sound decisions, and then follow through.

You've often heard the phrase, "You can't manage what you can't measure." But the act of managing goes a step beyond just measuring. It's clear that managers—good managers anyway—collect information, make decisions, and then effect change. Input and output—it's a two way street. Imagine a manager who collected information, made a good decision, but didn't follow through to effect the necessary change. That's not management, and you needn't tolerate such a paradigm with respect to systems management tools.

Many management tools only provide reporting. A fault management tool will generate an alarm, an accounting package will print a usage report, or performance software might tell you how many voice mailboxes have not been accessed in six months. This is useful information, and you need it to make good informed decisions. But why stop there?

Look for software tools that let you make changes back to the systems after you've analyzed the reports and made a decision regarding what type of change should occur. For example, a fault management package may allow you to make network adjustments, or a call center performance management package might allow you to assign additional agents to a heavily saturated route. A security management application may let you add a policy and apply it to groups of users, or accounting software will let you change the chargeback rate for certain usage items.

The examples in the previous paragraph illustrate the potential for effecting changes within each FCAPS area when the need for such changes is reported. However, one area of FCAPS—configuration—also lends itself to effecting changes across the FCAPS spectrum.

By integrating your configuration management tool with the other "read only" FCAPS tools, or with your internal business workflow, you can automate many of the high-volume changes that you decide to initiate as a result of the information reported to you.

Here are some examples of actions that could be taken automatically by configuration software:

- Delete the 187 mailboxes that the performance software indicates are no longer being used.
- Reduce class-of-service rights for a phone that is shown by the accounting software to exceed a credit threshold.
- Disable a phone that the security software says is unauthorized.
- Make a new voice mail message distribution list for the North American sales region selling to the pharmaceutical industry, and add 340 sales reps as list members in preparation for an important message from the sales vice president to be delivered 90 minutes from now.

An Interface Is Not An Application

To this point, I've used the term *tool* to communicate use of a software "function" to do the data gathering and change execution work. But not all tools can perform in the scenarios described above. Particular attention should be paid to the architecture of the software.

There is a difference between an interface and an application. The role of each tool must be clearly defined.

All systems have interfaces to support the FCAPS function. Generally these interfaces fall into two groups: human User Interface (UI) and Application Programming Interface (API).

An interface is the user's view, window or console session that shows what is happening on the system. There is little or no processing of information on the PC or terminal where the user is. Examples include: Command Line Interface (CLI); Web page to a server on the system; or menu-driven screen session generated by the system. In almost all cases, no management information is stored locally with the user, and there is no audit or lasting record of configuration changes.

Management standards align reasonably well with the five FCAPS areas

Managing telephony devices really implies managing end users

An application, by contrast, runs on a PC or workstation. It stores and organizes the preferences, settings and rules that are unique to your organization and to the specific user. Applications typically scale to support a central database so many clients can share the same information, creating a global view and centralized control.

Information is gathered from a managed system through an API, processed and analyzed offline, and individual commands are sent back to the API. A central change log is created to support compliance requirements mandated by new corporate governance laws and security policies.

The difference between interface and application is clear when managing components of the data network. No IT manager would manually monitor or configure thousands of switches individually. Systems management applications perform data-gathering and change execution across hundreds or thousands of devices.

For various reasons, the distinction has not been as clear in traditional telephony. This world is composed of users, not just cookie-cutter devices. A PBX may be a single device, but it represents thousands of users. Each user has several services, like a phone, voice mailbox, and any number of applications deployed on the server, such as zone paging, find-me/follow-me, etc. And of course, each service potentially can have hundreds of class of service settings for each user. So the difficult problem of managing large numbers of employees—each with unique needs—has largely been neglected as hardware vendors focus on providing device management tools.

But this is changing. As convergence accelerates, so too, must our understanding of the proper roles of interfaces and management applications for telephony systems.

For small organizations, the system management tool is the native UI of the system to be managed. But larger organizations will experience problems when the console UI is expected to perform complex tasks. I am constantly amazed at how many well-run organizations struggle with systems management because they expect too much from the native UI.

By the way, a special differentiation should be made about the management functions performed via the corporate intranet or an extranet. Enterprises clearly have a need to manage multiple systems from multiple locations, especially remote autonomous locations. Web-based management is the common choice. The problem is that nearly all the Web-based interfaces are just that—the simple user interface into a single remote system.

Isolated changes will compromise any attempt to achieve a centralized network view. System vendors have done a great job with UI features that synchronize global data in the background, such as dial plans, but the system console interface won't handle the unique information and workflows for your business.

For example, if an employee's membership in a distribution list for broadcast voice mail or a zone paging application depends on the reporting hierarchy, a management tool can allow simple organizational changes to ripple down, automating thousands of changes that realign the group memberships as the enterprise requires. But no organization should expect PBX system vendors to accommodate such complex data structures and logic in a native UI console.

For enterprise-caliber management, use the proper remote access technology, one that connects you to your FCAPS management applications. Choices include terminal services, Web sessions, and Web-based access to management applications. In all cases, the remote or Web-based access is to a management application, not the system user interface. This gives remote locations the speed and flexibility of direct localized management (historically provided through access to the local system's native console) along with the global network view, centralized audit logs and federated control needed by headquarters.

When you buy a PC it often comes loaded with free software, like personal finance software. This may be all you need for home use or a small business. In your organization, however, the computers may come with the same free software, but never is there the thought to run the corporation's finances with this software.

The same is true for telephony management. The system may sport a user-friendly Web interface for management, and this is likely perfect for small businesses, but enterprises on a larger scale should look beyond the native interface.

These more extensive management applications are built and sold by the PBX system vendors themselves, as well as by software vendors who specialize in telephony management. In either case, organizations should differentiate between interface and application, and be clear about the roles each plays.

Conclusion

Matching your tools to the constituents of each FCAPS area will allow you to have, at a minimum, five effective tools each designed for a specific group of professionals. Enterprises should make sure that systems management tools include the ability to actually make changes to the network, not just produce reports. Finally, keep expectations modest for a system's native interface and deploy true management applications when the complexity warrants it.

Unless large organizations recognize the complexities inherent in each of these management areas—especially when dealing with large numbers of employees, disparate types of telephony systems and services from multiple vendors, in a telephony environment that may span numerous geographical locations—the associated costs of maintaining such a network will only go up□