axians

Automation **Made Easier**

Axians DPCM saves money and increases productivity by leveraging HMC REST APIs to perform automated tasks with less scripting, programming and complexity

VINCI ENERGIES

Manually building and managing IBM Power Systems environments is complicated and resource intensive. While IBM system tools such as the IBM Hardware Management Console (HMC), Network Installation Manager (NIM) and Virtual I/O server (VIOS) provide a place to start, there's a better way.

Axians Dynamic Power Cloud Manager (DPCM), for example, provides a powerful, easy-to-use automation and administration solution for Power Systems environments. DPCM saves time, improves availability and enables significant cost and overhead savings. From an operational perspective, DPCM helps operations teams with compliance, standardization, flexibility and upgradability. For a visual of the cost savings DPCM provides, see Figure 1.

If admins aren't end-to-end knowledgeable about their computing environments or lack strong scripting and programming skills, baked-in IBM tools may be more burdensome than helpful. "The HMC is mandatory for successful systems management, but it's not enough for the full management of logical partitions," says Ilja Franke, senior developer with Axians.

With DPCM—along with other add-on tools such as Red Hat Ansible integration with HMC REST APIs, VIOS and NIM; and IBM PowerVC for zero-day LPAR development—admins can build LPARs nearly on the fly, defining parameters to determine where they're going to be hosted, how much disk and memory they will be allocated, which network they will use and where their data will be stored. They can manage LPARs and other resources with the same amount of ease—enabling complete systems administration. And, they can leverage HMC REST APIs to perform automated tasks with much less scripting and programming.

Baked-in Power Systems Tools

The Power Systems platform is extremely flexible, being capable of, for example, running a variety of operating systems (OSes), including AIX, Linux and IBM I, in any number of virtualized and isolated LPARs. Notably, this flexibility extends even further by providing redundancy to avoid complete outages in cases of hardware failure.

A Power Systems server can manage several network cards with several VIOS to provide network connectivity

for the LPARs on the system, for instance. So, someone can shut down or reboot one VIO for maintenance while the other still manages the network without the LPARs noticing. Similarly, if a network card fails, another can transparently take over. Additionally, most customers have several Power Systems servers among which they can move LPARs to avoid hardware-failure downtime.

But this is also where complexity comes into play, because all this needs to be configured and maintained even though all of the corresponding information is located in different places—a particularly acute pain point for admins. Some of this systems management can be done with the HMC, but other tasks require logging into a VIO with the CLI and making adjustments from there. Indeed, using NIM for backup and installation, VIOS for redundancy for network and storage, and HMC for general systems management can create confusion, even though each is a necessary component of Power Systems environments.

"There are lot of separate parts that you can try to automate as a customer, but you need to have an understanding of each of those parts and how they fit together," says Michæl Buss, Axians ITS Technical Services. "So, yes, you could manage everything by yourself, using baked-in IBM tools, but that's a lot of work."

Complexity shouldn't stop businesses from reaping the benefits of automation. Rather, organizations should understand the benefits and limitations of system tools. Take the HMC and VIOS, which is where

Task	Effort for each LPAR in hours		# / year	Effort in hours / year		Savings in hours / year	Savings in $ / year
	Manual	With DPCM		Manual	DPCM		
Firmware Update	1	0.25	1	55	13.75	41.25	$4,537.50
System Recovery	8	2	0.1	6.4	1.6	4.8	$528.00
Backup Verification	2	0.25	52	104	13	91	$10,010.00
Load Advisor/ Capacity Planning	4	0.25	12	48	3	45	$4,950.00
New SAP HANA Deployment	12	0.5	4	48	2	46	$5,060.00
				261.4	**33.35**	**228.05**	**$25,085.50**

Figure 1. Firmware updates, system recovery, backup verification, load advisor/capacity planning, and new SAP HANA deployment times with DPCM, versus without. DPCM enables exponential time and cost savings.

automation actually begins in admin scenarios, typically with HMC CLIs and VIOS commands. Admins usually start out writing their own scripts for the CLIs and then embed them in schedulers or cronjobs.

"The HMC is the official IBM tool to manage Power Systems and all of the platform's available functions," notes Sebastian Luckau, DPCM Software Architect & Core-Developer–IEM Axians Infrastructure Software. "But you have to be a very skilled Power admin to work with it. So, if you're a normal user who's not aware of what's involved in configuring DLANs, configuring adaptors, configuring SIA, configuring IoT or all those special things you can do with the HMC, you will fail. You have to truly understand the Power Systems environment to take full advantage of the HMC."

Even if a user dœs understand all the ins and outs of a Power Systems environment, they'll still have to develop HMC CLI scripts, VIOS commands and perhaps, depending on task complexity, actual programs that automate tasks to get jobs done both simpler and quicker. This requires a great deal of scripting and/or programming experience—skills many admins may lack.

HMC REST APIs

This is true even when HMC REST APIs, which provide an additional way to deploy and manage LPARs, VIOS, etc., are exploited. REST is a style of web service that manages nouns, which represent known items like LPARs and other managed systems.

"These REST APIs are the officially documented IBM APIs for the standard HMC APIs. The HMC itself uses those APIs for its functionality," Franke says. "Basically, everything you can with the HMC UI, you can with the REST API, like providing access to performance data and things like that."

The wrinkle with the HMC REST APIs, though, is that their use case generally aligns with the use case of CLI scripts. Determining whether to use CLI scripts or HMC REST APIs depends on the specific development goals and the complexity of the task. If the task can be constructed using simple scripting, developing it with HMC REST APIs may not be worth the effort.

However, if the task is too complex for CLI scripting, HMC REST APIs and a "proper" programming language may be the answer. This is in part because the REST API commands and queries are standardized to use readymade libraries to, for example, handle authentication or parsing results. Many CLI commands, on the other hand, have a different way of formatting their output, thereby requiring admins to write a different parser for each command.

Tools such as Ansible, PowerVC and DPCM leverage HMC REST APIs under the covers to perform a variety of automated tasks with much less scripting and programming—and dramatically improved UIs. In the case of DPCM, tapping into these APIs with decipherable automation plug-ins simplifies complex automation routines, allowing admins to focus on administering systems instead of writing scripts or actual programs..

European Airline's IT Department Flies High Thanks to a New Axians Automation Tool

That nobody's perfect is a simple fact of life. Indeed, even experts in their particular fields can make mistakes.

That's in part why a large European airline decided to use Axians' Dynamic Power Cloud Manager (DPCM) to automate many IT tasks that had previously been done manually. This wasn't because the organization's IT staff lacked talent and skills—but more because automated workflows and LPAR creation are likely to experience few or even no poor outcomes, which ultimately saves both time and money.

"Among other reasons, the company wanted to reduce the amount of possible errors that could be made by human operators," an AIX Power professional from an IT vendor that helped with the DPCM installation says. "Mistakes can and do happen when you're keeping LPAR profiles in sync or administering more than 1,000 discarded devices on many managed systems."

The airline also wanted to make sure it could recover crashed LPARs in a short amount of time. Other troubleshooting areas had to do with the standardization of server data and the creation and resource allocation of LPARs.

The traditional way of addressing these concerns was writing HMC CL scripts or programs and developing VIOS commands—cumbersome methods that make it more difficult to build automated routines.

DPCM more than met all of the airline's challenges. "We started with an in-house proof of concept deployed in a test environment," the AIX Power professional recalls. "Based on the results of that, we deployed DPCM to the production environment, which took only around four hours to complete. It nearly works directly out of the box."

DPCM acts as an overlay of HMC functionalities, VIOS configurations and new management tasks, providing admins with a single user-friendly method to build and manage LPARs and associated resources. It can define changes coming through operations, such as the addition of a new customer or system, or changes due to data center incidents.

"If you remove a logical partition from the HMC, it's gone," the AIX Power professional says. "If you remove it in DPCM, it's still there, and you can easily recreate it on any kind of managed system. If you were to lose a single managed system and the definitions on it, you could use DPCM to rebuild those LPARs on any other managed system."

The response from the customer has been nothing but positive. "The technicians say they're spending much less time operating their Power Systems environment with the help of DPCM. This gives them additional time on the other side to deploy additional automation workflows and further enhance the entire environment," the AIX Power professional remarks. "It's a turnkey solution that helps bring down costs, optimize processes, and standardize recoveries and ways to create new systems or LPARs."

"The benefits of using DPCM and Ansible with the HMC REST APIs is that you can automate many things you'd otherwise have do by hand," Franke says. "This allows you to, for example, more easily combine different tasks into single, more complex tasks that are much more capable but don't require in-depth scripting or programming knowledge."

PowerVC has a more powerful and relatable UI than the HMC. It's easier to use and more flexible than the HMC, allowing admins to more simply create zero-day LPARs using HMC REST APIs. As a result, even less experienced can users can build out LPARs and then use DPCM to manage them.

This last point is key, because PowerVC has some limitations, especially when it comes to ongoing LPAR management. In fact, it's best used within green-system environments, not in existing, LPAR-populated computing environments, because it's not capable of, for example, handling LPAR backups, recoveries and modifications.

Additionally, routine maintenance tasks can't be performed with PowerVC, as admins might expect, making additional tools and scripts necessary to manage all-day operations tasks. DPCM compensates for this with robust LPAR-management capabilities that go well beyond what PowerVC offers.

As Luckau notes, "PowerVC is a great tool for new and smaller environments that are introducing new systems without LPARs, where you can start from scratch without restrictions related to storage and network. You can use PowerVC to quickly deploy LPARs in these

straightforward situations, but it's really not suitable for much more beyond that, including for complex jobs such as backing up or increasing the number of LPARs."

This is where a solution such as DPCM comes into play. Companies can use PowerVC in day-one environments without existing LPARs and then manage them over the long run with DPCM. In most cases, however, as Luckau further points out, PowerVC is a one-off tool that dœsn't really match customer requirements. "It can be a good starting point, yes, but it's severely limited beyond that. Because of this, you might as well go with a single long-term solution such as DPCM and/or Ansible to deploy and manage LPARs in both new and existing environments," he says.

Ansible Integration

PowerVC, despite its user-friendly interface, simply wasn't designed for this type of long-term management situation, but Ansible and DPCM have been. They use pre-existing plugins to help admins easily develop automated routines, including for management. Ansible can also be integrated into DPCM for a variety of purposes.

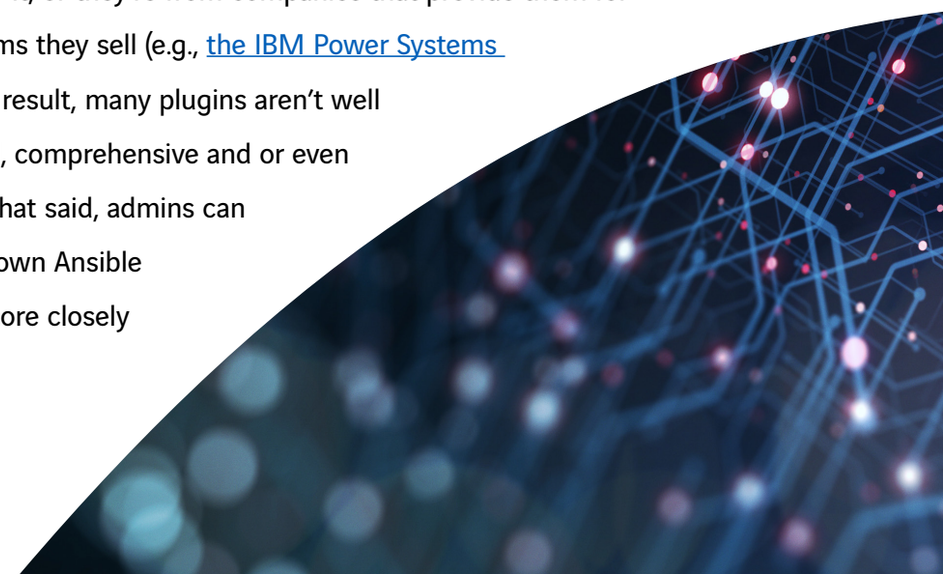
"Because system admins are probably already familiar with Ansible, DPCM provides Ansible plugins so they can basically use DPCM with their existing Ansible setups just like they would use other Ansible plugins that are integrated into specific workflows," Buss says.

"In addition, we provide the ability to configure access details to customers' Ansible servers so the workflows can be triggered via the DPCM UI directly or when the admin uses some DPCM functionality in the UI. We also let the admin write simple task lists in an Ansible 'playbook' notation that runs on the DPCM that enhances some of our own internal functionality."

Ansible, like DPCM, can be used as a compliment to PowerVC to reduce the complexity of API scripting and programming and providing a broader management reach. It offers, for example, a method by which system admins can specify lists of tasks with simple logic ("playbooks") by describing which Ansible module or plugin to use and under which parameters it should be called.

Any complexity is masked and incorporated into the modules/plugins so admins can use them by simply defining parameters so that the list of tasks reads like a simple description of what needs to be done. The plugins themselves, which are typically written in Python, are plentiful, free and open source.

But these plugins are either rather simple, because someone needed to do something quickly and then provided the resulting plugin to whomever wants to use it, or they're from companies that provide them for use in systems they sell (e.g., the IBM Power Systems server). As a result, many plugins aren't well documented, comprehensive and or even functional. That said, admins can create their own Ansible plugins to more closely



fit their specific needs—but this again requires development time and programming skills.

Another approach is to take parts of existing Ansible playbooks (the task list with simple logic) and put them into separate files and folders (called "roles" in Ansible) so they can be reused in other playbooks without having to copy and paste the text itself. But users are limited to only one call per playbook at the same time and have to wait for that call to finish before calling another playbook.

If multiple playbook calls are required, someone could write a script that calls Ansible multiple times, but then they'd probably also have to parse and collect the output that needs to be displayed after all of the playbooks have been called, yet another manual task. To help accommodate for this, Red Hat also offers the Ansible Tower (via the open-source and free AWX Project), which can combine multiple playbooks for a workflow and schedule them.

Information Consolidation With DPCM

Both DPCM and Ansible—and PowerVC in limited situations—sound simple compared to the HMC, which has issues even beyond its scripting and programming requirements and its unfriendly UI that aren't discussed often. For example, IBM tools have different APIs depending on the versions are being used. With each, API parsers must be checked and reviewed to make sure they still function as intended. This isn't a trivial matter for any size organization, which is why many of them already have adopted or are considering adopting tools such as DPCM to tackle that challenge.

"The devil is in the details in the end," Luckau says. "You won't find all of the information in the HMC regarding the different versions of the HMC that are out there. So, if you have a user with an older HMC, you can't update one HMC here but not that one over there. Everything is too well-orchestrated to take that approach. You need to standardize across IBM system tools and confirm proper parser functionality everywhere. This takes time."

Because Power Systems environments tend toward complexity, communicating across separate portions of those environments can be problematic. If a VIOS or NIM is having difficulties and maybe the HMC is down, automations could come to a halt, with, for instance, the Ansible automation responsible for restoring or moving an LPAR failing during a system recovery.

These communication breakdowns happen in part because each tool has only one, unconsolidated piece of the puzzle. So, if one or more of these automation resources is having an issue, a core component of the automation won't be available. Admins could manually gather and log this information, but because this type of environment is so dynamic, that could require perhaps hours out of every day just to keep everything up to date.

These issues are in part why some admins are turning to DPCM, which is designed as a package that installs a set of Docker containers to decouple DPCM from the underlying OS it runs on. Rather than leaving fragmented bits of information—DPCM automatically gathers and consolidates it on a DPCM database.

"Someone can access the DPCM database to get information about the disaster profile of an LPAR, such as which adaptors are connected to which in a certain profile, simply by going to the DPCM database where all of this information has been gathered and combined," Franke says. "That's much easier than having to go to the IBM system tools to complete a profile. And if those systems are down, you won't be able to get any of it anyway."

This is made possible in part by DPCM's tight integration with resources such as system tools. DPCM uses NIM functionality for network installations, updates and migrations, and to fully automate unattended backup and recovery tasks. DPCM also needs to manage the HMC to create, change and delete LPAR profiles and running LPAR configurations, as well as use HMC functionalities such as Logical Partition Manager to migrate LPARs to other systems.

Although full DPCM replication isn't necessary, it's encouraged for admins overseeing environments that include multiple remotely located data centers. By having a DPCM server—each hosting a copy of its database and the consolidated information it contains—situated at each data center, admins can use DPCM functionality to recover LPARs between locations and manage Power Systems environments at any time, even remotely.

If the network connection between the data centers is severed but the data centers themselves are still operational, they form separated "islands" that can't exchange data. DPCM can be useful in such emergency cases because it usually runs on the same hardware at each data center. Should one of the servers fail, DPCM can run redundantly to capture the LPARs on the failed system and get them up and running again on another.

"Because DPCM has its own database for all the information gathered from the Power Systems environment, you still have all of the failover definitions in a separate database that you can use to build up your data center with new hardware," Luckau says. "You don't have to do it from scratch because it will simply restore from DCPM database."

The Automation Safety Net

When tasks are automated, the Power Systems platform is highly resilient and flexible, especially when it comes to resource consolidation, which makes life easier for people not only in the IT department, but also those supporting the business. After all, both groups simply want things to run as smoothly as possible.

Baked-in IBM tools—as mandatory and indispensable as they are—can sometimes be difficult to work with if admins don't have the proper understanding of their computing environments or are unfamiliar with scripting and programming. But tools such as DPCM, Ansible and even PowerVC can help bridge underlying complexities and enable ease of use.

"Customers make comments about efficiencies and how they want to work around resource and time constraints," says Christian Heitkamp, Axians' product manager, DPCM. "So, when there's a complete system outage—which still happen—quickly coming back online is critical. You can't spend enough on a safety net for your IBM Power Systems environments that helps you avoid losing millions due to an outage. You can sleep better at night thanks to automation—from wherever it's derived—and all of the benefits it has to offer."

Axians DPCM can **automate your Power Systems management**

Axians Dynamic Power Cloud Manager (DPCM), a natively built and purpose-driven management solution for IBM Power Systems, is a cornerstone for easy, structured and cost-effective Power Systems administration and automation. This includes everything from the deployment of operating systems to live migrations between data centers—even while the VMs are running—to capacity planning to backup and recovery.

Because companies spend big on hardware, Axians believes the cost of Power Systems infrastructure management should be correspondently lowered, especially in light of a reduction in skilled support staff. In short, DPCM—which has been in continuous development for more than 10 years and has a solid customer base with more than 10,000 of CPU cores—speeds up returns on investment by enabling companies to do more with less.

LEARN MORE