Discover PureApplication System patterns of expertise

Amit Acharya
Steve Curtis

April 11, 2012

This article offers a gateway of understanding into the IBM PureSystems™ family of expert integrated systems by providing a tour of the structure of the PureApplication System and virtual patterns of expertise. Specifically designed for transactional web and database applications, IBM® PureApplication™ System solutions are packaged into patterns built for easy deployment on the cloud. Find resources to help you expand your knowledge of how to develop and deploy using the PureSystems technology.

IBM PureApplication System combines two approaches to manage growing IT complexity and rapid application deployment using patterns of expertise:

- **Cloud approach**: Everything acts "as a Service," allowing users to augment the various services — infrastructure, platform, software, business process, database, storage, etc. — with the ability to incorporate modules that encode best practices in those topic areas into the overall system environment. This is a resource-oriented approach.

- **Workload optimized systems approach**: Similar expert components can be incorporated into the environment to provide proven experience in managing such tasks as workload consolidation, rapid rollout of new and enhanced capabilities, and workload life cycle resolution. This is a task-oriented approach.

The pattern of expertise deployment approach is the foundation of PureApplication System: Use the cloud appliance to build and deploy patterns that represent your completely configured application environments. When you are ready to use a particular application environment, simply pick a pattern and deploy it. PureApplication System automates the deployment, configuration, and integration of the various virtual machines that make up your environment and delivers the completed product in a matter of minutes.

Reducing deployment time, increasing consistency, and fostering agility are benefits you likely expect when exploring cloud-based approaches for your middleware application environments. The PureApplication System solution tackles these issues making the deployment of cloud middleware environments fast, repeatable, and efficient.

Design and deploy applications with patterns of expertise

The PureApplication System architecture supports three segments of middleware, three components that are critical underpinnings of the system:
• **Virtual application patterns**: Used through workload platform services, a virtual application pattern represents a collection of application components, behavioral policies, and their relationships. Core components of the pattern include web applications, databases, queues, connections to existing resources, business process models, batch jobs, mediations, etc. Core policies of the pattern include high availability, SLAs, security, multitenancy, isolation.

• **Virtual systems patterns**: Used through virtualized middleware services, virtual systems patterns are a logical representation of a recurring topology for a given set of deployment requirements. For example: A WebSphere® Application Server Cluster pattern containing IBM Deployment Manager, one or more custom nodes, IBM HTTP Server, and configuration scripts for installing applications to the topology.

• **Virtual appliances**: Addressing key cloud issues such as software licensing and standardization, a virtual machine image file consists of a preconfigured operating system environment and application. The IBM PureSystems family is designed to process virtual appliances rapidly, in a highly automated manner. The IBM Virtual Appliance Factory is a self-enablement toolkit that provides automated tools to help developers create virtual appliances in the Open Virtualization Format (OVF).

Let's look at each in more detail.

**Virtual application patterns**

Virtual application patterns are application-centric in their design. They provide a mechanism to represent middleware applications in a simplified model that abstracts away the underlying middleware infrastructure. These patterns are implemented using virtual application pattern types that integrate the capabilities of multiple middleware software elements into a cohesive, built for purpose solution to allow you to represent your complete, often complex, environments as a single deployable unit.

A virtual application pattern is the critical element that lets you rapidly set up and manage cloud application infrastructure. Expand your knowledge of virtual application patterns with the article, **Manage application services with virtual application patterns**:

... a virtual application pattern describes an application and employs the tools to build the appropriate infrastructure for the application and then deploys the application to that infrastructure. PureSystem virtual application patterns are the captured essence of years of application-infrastructure-deployment experience and best practices. Virtual application patterns encapsulate optimized solutions of multiple middleware elements to host specific types of application workloads. The experts introduce you to virtual application patterns and their place in the ecosystem, describe their components and functions, and provide a basic look at how to create and use a virtual application pattern.

A good example of a virtual application pattern is **IBM Workload Deployer Pattern for Web Applications**. This workload pattern lets you fully automate management of your web application as a single unit, including deployment, life cycle management, elastic scaling, and security.
Virtual systems patterns

A virtual systems pattern is the critical element that lets you rapidly set up and manage the cloud middleware topology. Expand your knowledge on virtual systems patterns with the article, Manage the topology with virtual systems patterns:

... a virtual system pattern describes a middleware topology and employs the tools to automatically build that topology in the cloud. IBM PureApplication System virtual system patterns are the captured essence of years of infrastructure-management experience and best practices. Virtual system patterns encapsulate repeatable topology definitions based on various middleware images and runtime configurations; they give you control over the middleware landscape being deployed. In this article, the authors introduce you to virtual system patterns and their place in the ecosystem, describe their components and functions, and provide a basic look at how to create and use a virtual system pattern.

One of the tasks you'll need expert knowledge on is how to design a virtual systems pattern. In the article, Design a virtual systems pattern, get a basic checklist on how to plan the design of your virtual system pattern and explore virtual system patterns in PureApplication System with experts:

Virtual system patterns in IBM PureApplication System enable fast and repeatable deployments of systems from the virtual machine up to the application. With a virtual system pattern, manual tasks that are needed to bring up your entire topology can be fully automated, which allows an application to be deployed in minutes as opposed to hours or days. Pattern-oriented deployment of middleware eliminates bugs introduced by error-prone, manual configuration processes and allows best practices to be baked into patterns, thereby accelerating and optimizing deployment of solutions. In this article, the authors highlight key points to review when designing and developing a virtual system pattern.

A good example of virtual systems patterns is IBM Database Patterns, which includes the IBM Transactional Database Pattern and IBM Data Mart Pattern. The Transactional Database Pattern lets you accommodate departmental online transaction processing applications that don't require high levels of database customization; it delivers a transactional database infrastructure for information-centric applications. The Data Mart Pattern provides provisioning and management capabilities you can use to manipulate a data mart infrastructure for data-centric applications in a secure, private cloud; this pattern is tuned for the unique I/O throughput required of data mart workloads.

Enabling applications through plug-ins

Using plug-ins is a method to enable existing applications to deploy in the IBM PureApplication System environment; the IBM Workload Plug-in Development Kit is a feature that helps you create those bridges to the system.

Virtual application patterns are realized by supporting a combination of application-centric deployment models, virtual applications (consisting of individual components and policies), the infrastructure and middleware they employ, and workload-specific content — extensions or plug-ins that define components, links, and policies used to describe and deploy virtual application
patterns. In the article, Create and customize virtual application patterns with the IBM Workload Plug-in Development Kit, the experts explain how developers can get started with the toolkit on their way to creating and customizing virtual application patterns.

If you need some basic hands-on practice to familiarize yourself with the toolkit, Using the IBM Workload Plug-in Development Kit shows you how to:

- Set up the samples environment in Eclipse.
- Create new projects from the command line.
- Build a single plug-in and pattern type with command-line tools.
- Develop a plug-in with the Eclipse framework.

If you want a look at a real scenario for developing a plug-in with the toolkit, you can relive the experiences of the IBM Cloud lab team’s efforts to develop the first plug-in for the PureSystems environment in Part 1 of the article, Deploy new applications on IBM PureSystems with plug-ins. They’ll show you how to craft a plug-in, a bridge between the application package and the system, in this case for SugarCRM, an enterprise application package for customer relationship management. The PHP application requires a software stack that is not supported by PureApplication System out of the box, so the team had to develop a new pattern type and a set of plug-ins that support the modeling, deployment, and operation of the application atop a base image of Linux® and IBM AIX®. If you want to go straight to the general lessons the team learned in the process, Part 2 dives into those lessons which include knowledge on supporting multiple platforms, managing the databases, developing scripts, debugging your plug-in, and more.

Virtual appliances

Virtual cloud appliances — an application along with the appropriately configured operating system and middleware — provide a single unit for managing your application. The IBM Virtual Appliance Factory makes it even easier by automating the deployment of your cloud appliance onto KVM and PowerVM environments. The article, Automate your virtual cloud appliance takes you on a technical tour of the toolkit.

Get ready for PureApplication System

So how do you prepare for PureApplication System? These companion resources can help.

- Preparing for IBM PureApplication System, Part 1: Onboarding overview
- Preparing for IBM PureApplication System, Part 2: Is your application ready to be virtual?
- Preparing for IBM PureApplication System, Part 3: Choosing a database option
- Preparing for IBM PureApplication System, Part 4: Onboarding your applications to the cloud
- Preparing for IBM PureApplication System, Part 5: Developing virtual application patterns