Administering AppFabric with Windows PowerShell

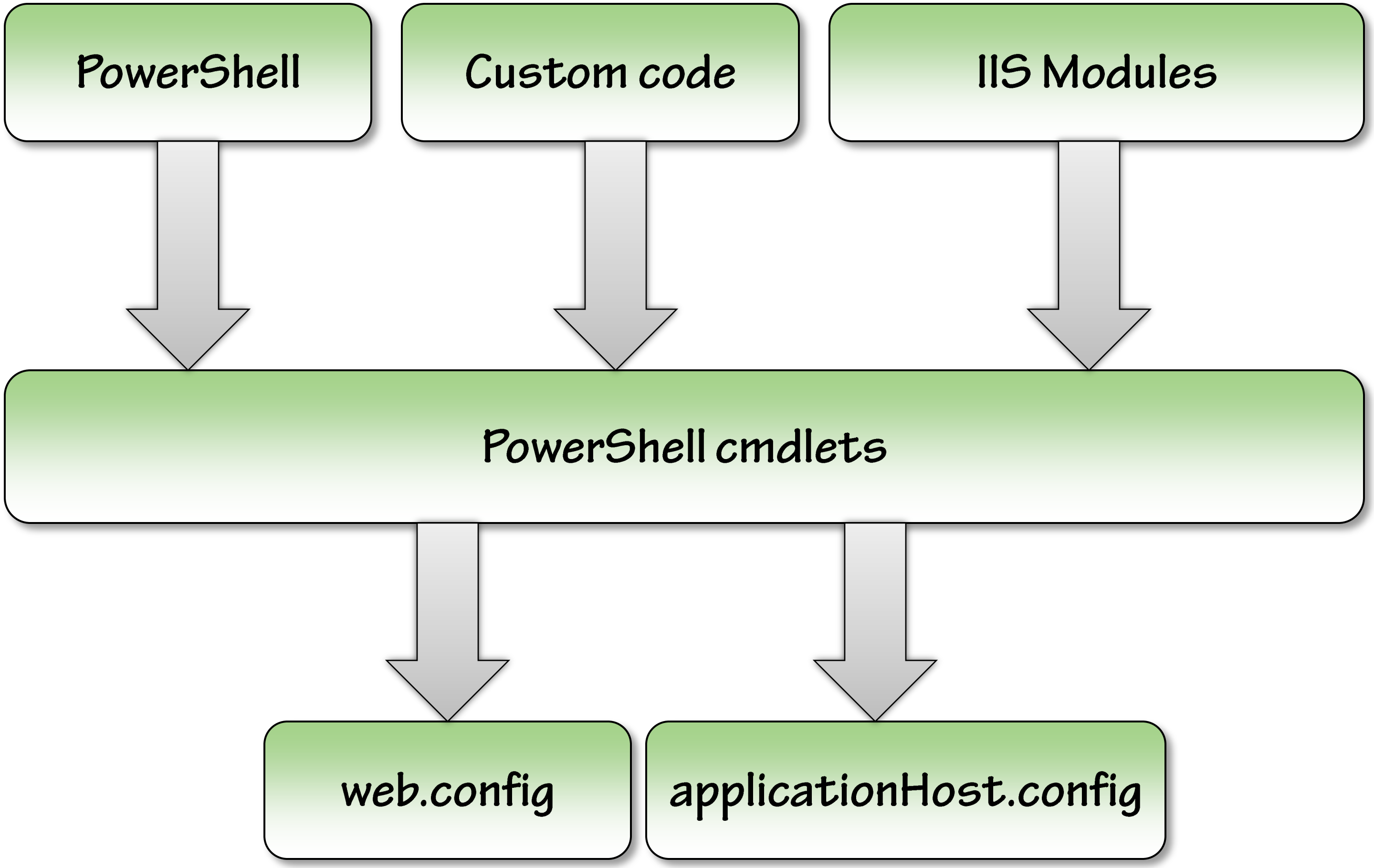
Using scripting to manage services

pluralsight
see what you can learn

Overview

- PowerShell basics
- The Application Server cmdlets
- Managing services and instances with cmdlets

Management tool architecture



Windows PowerShell

- **THE shell for Microsoft server administrators**
- **Built on the .NET framework**
- **Enables working with COM, WMI, ADSI, XML, etc.**
- **Access data stores with file system-like commands**
  - Registry
  - Certificate store
- **Being enhanced by server teams with custom commands**
  - SQL, IIS, Exchange, AppFabric
- **Version 2.0 enables remote command execution**

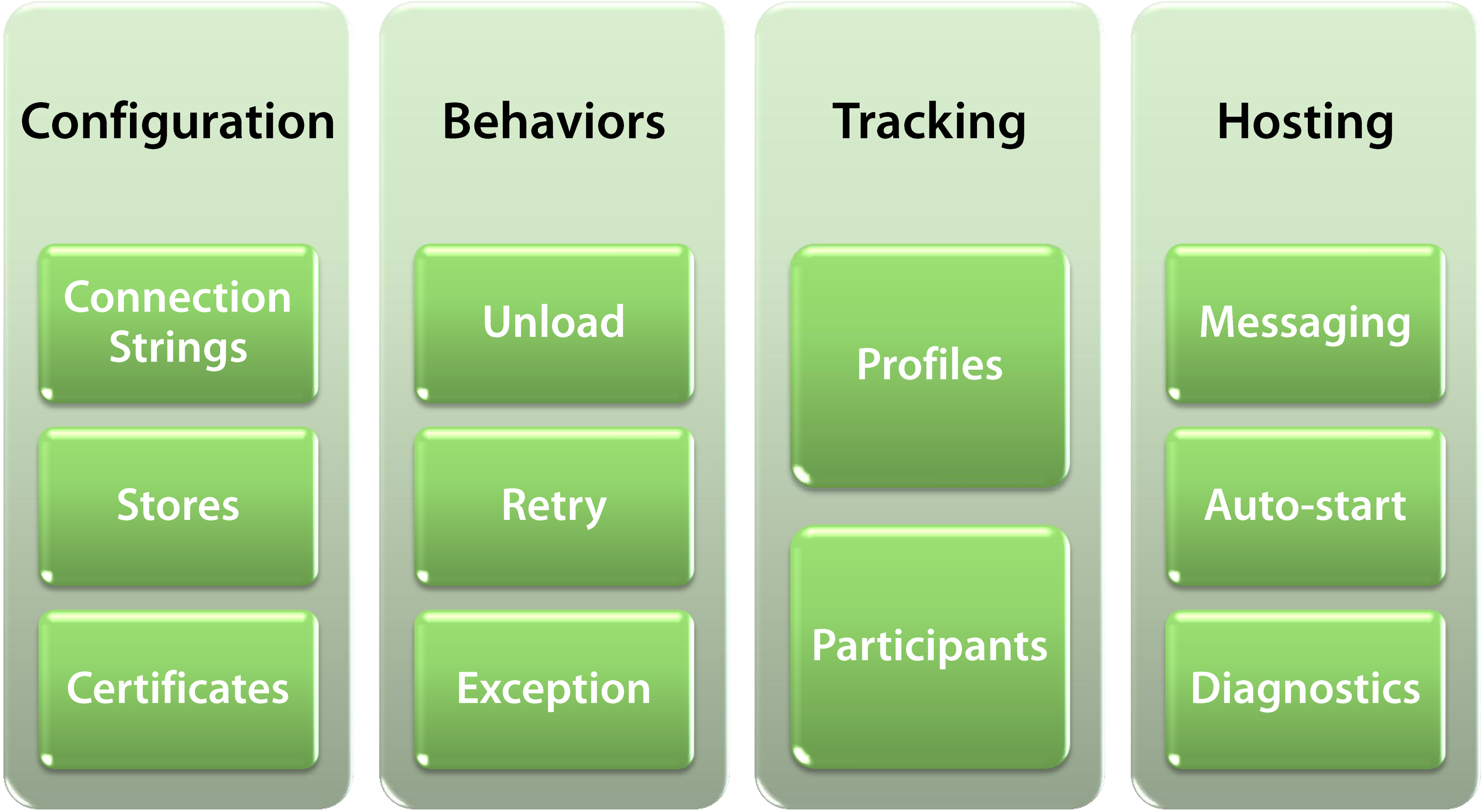
PowerShell terminology

- **Shell**
  - The host application for PowerShell
  - Default shell included upon installion
- **Provider**
  - Enables access to data store as hierarchy (certs, registry, SQL)
- **Cmdlet**
  - Encapsulated logic or command
- **SnapIn**
  - PowerShell v1 installation container for providers and cmdlets
  - Custom shells can pre-load snap ins
- **Module**
  - PowerShell v2 container for commands, aliases, etc.

PowerShell is objects

- **PowerShell is like other shells**
  - Interactive mode for quick work or testing
  - Scripted mode for repeatability
  - Output from one command can be “piped” to the next command
- **PowerShell is NOT like other shells**
  - Built on the .NET Framework and integrates with COM and WMI
  - Pipes objects not strings

AppServer cmdlets



Configuration

- Connection Strings
- Stores
- Certificates

Behaviors

- Unload
- Retry
- Exception

Tracking

- Profiles
- Participants

Hosting

- Messaging
- Auto-start
- Diagnostics

Summary

- PowerShell provides the interactive or scripted experience
- IIS Manager provides the GUI experience
- Use PowerShell as the base for your tooling experience