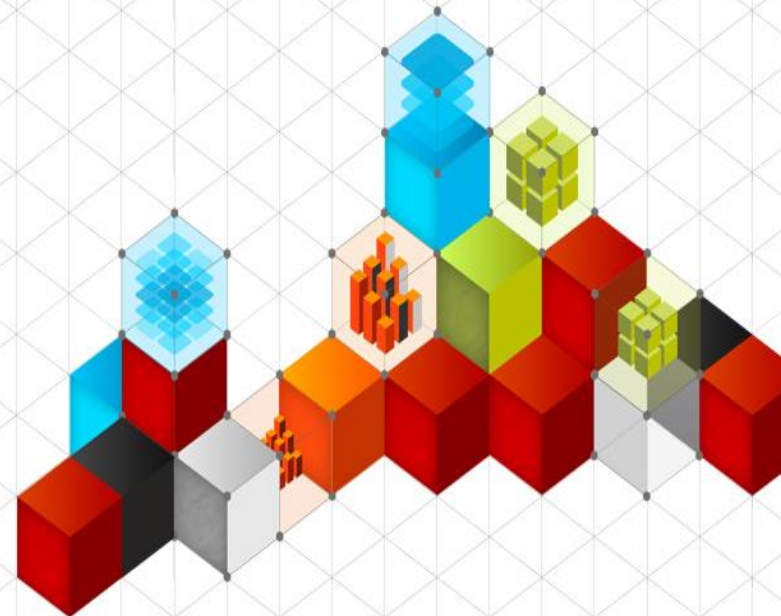


#544 | SPARTANBURG 2016

# POLICY BASED MANAGEMENT BEYOND THE BASICS

**Patrick Keisler**

Senior Premier Field Engineer



# ABOUT THE SPEAKER

## Patrick Keisler

- Senior Premier Field Engineer
- Over 15 years of database experience
- Specialize in administration of high availability and disaster recovery environments

Email: [pkeisler@microsoft.com](mailto:pkeisler@microsoft.com)

Blog: [www.patrickkeisler.com](http://www.patrickkeisler.com)

Twitter: [@patrickkeisler](https://twitter.com/patrickkeisler)

**Microsoft**  
**CERTIFIED**  
Solutions Expert  
Data Platform

**Microsoft**  
Specialist  
Implementing Microsoft  
Azure Infrastructure  
Solutions

**Microsoft**  
**CERTIFIED**  
IT Professional  
Database Administrator  
2008

**Microsoft**  
**CERTIFIED**  
IT Professional  
Database Developer  
2008

CompTIA  
**Security+**  
**CERTIFIED**



# SESSION AGENDA

- What is PBM?
- Basic Features
- Create a Policy
- Evaluating Policies
- PBM & Central Management Server
- Customizing PBM
- PBM & PowerShell



# WHAT IS A POLICY BASED MANAGEMENT?

- Policy-based system for managing one or more SQL Servers
- A policy uses a condition to check a target object
  - Returns PASS or FAIL



# BASIC FEATURES – TARGET, FACET, CONDITION

- Target – object you are evaluating
  - Server
  - Database
  - Table
- Facet – type of object & its properties
  - Database property examples
    - Recovery model setting
    - Page verify setting
- Condition – state or value of a property
  - Page Verify property of the Database facet
    - None
    - TornPageDetection
    - Checksum



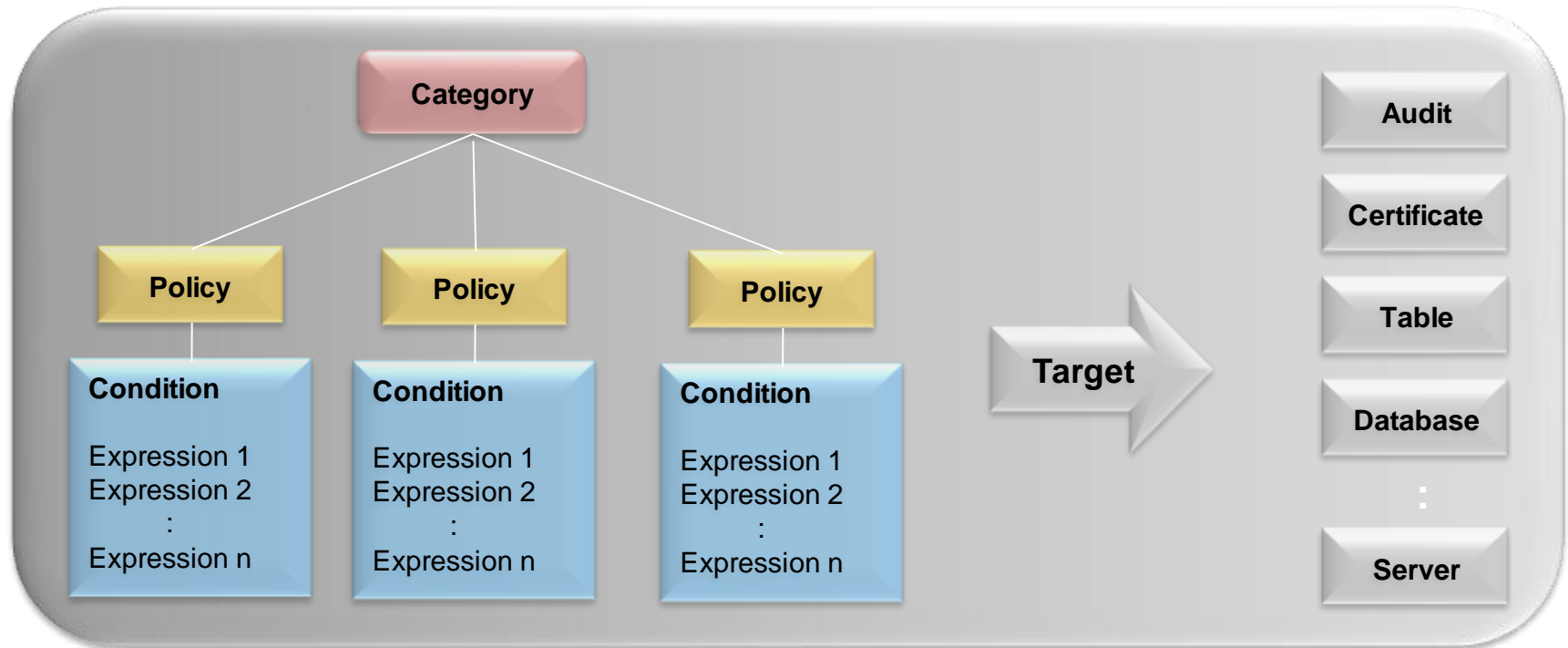
# BASIC FEATURES – CATEGORY

- Group of policies
- Example
  - Production Server Standards
  - Test Server Standards
  - Database standards



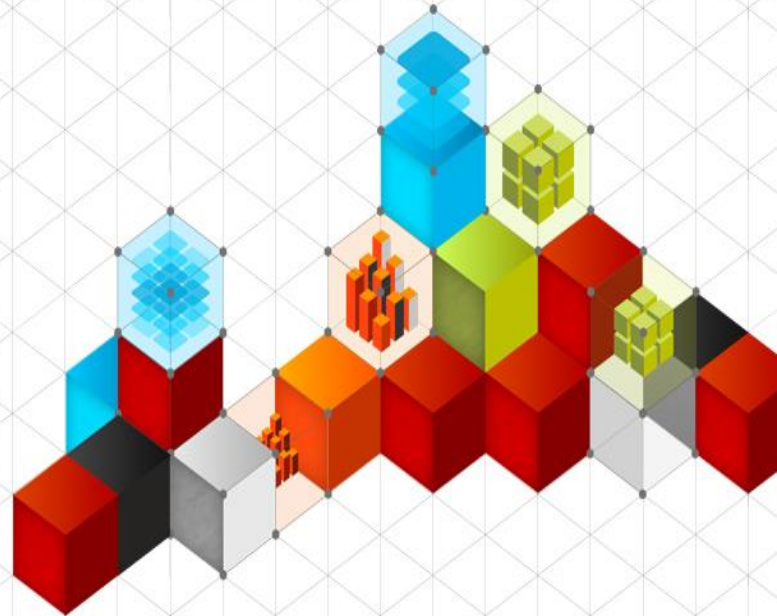
# BASIC FEATURES – POLICY

- Relationships between the objects
- Order of creation:
  - Choose a Facet → Create a Condition → Choose a Target → Save it as a Policy





# DEMO 1: CREATE A POLICY





# BASIC FEATURES – EVALUATION MODES

- **Manual Evaluation Modes:**
  - On Demand
- **Automatic Evaluation Modes:**
  - On Schedule
  - On Change: Log Only
  - On Change: Prevent

# BASIC FEATURES – EVALUATION MODES

- Evaluation mode is determined by the facet

<http://blogs.msdn.com/b/sqlpbm/archive/2008/05/24/facets.aspx>

| Facet Name                   | CoC: Prevent | CoC: Log | CoS |
|------------------------------|--------------|----------|-----|
| Application Role             | X            | X        | X   |
| Asymmetric Key               | X            | X        | X   |
| Audit                        |              |          | X   |
| Backup Device                |              |          | X   |
| Broker Priority              |              |          | X   |
| Broker Service               |              |          | X   |
| Certificate                  |              |          | X   |
| Credential                   |              |          | X   |
| Cryptographic Provider       |              |          | X   |
| Data File                    |              |          | X   |
| Database                     |              |          | X   |
| Database Audit Specification |              |          | X   |
| Database Ddl Trigger         |              |          | X   |
| Database Maintenance         |              |          | X   |
| Database Option              |              | X        | X   |
| Database Performance         |              |          | X   |
| Database Role                | X            | X        | X   |
| Database Security            |              |          | X   |
| Default                      |              |          | X   |
| Endpoint                     | X            | X        | X   |
| File Group                   |              |          | X   |

# EVALUATING POLICES

- **For a single policy**
  - From SSMS, right click a policy
  - Choose Evaluate
- **For multiple polices**
  - From SSMS, right click the Policies folder
  - Choose Evaluate
  - Select your policies
- **If the policies are not stored locally**
  - From SSMS, right click the Policies Folder
  - Choose Evaluate
  - Click the ellipse button (...)
  - Choose an XML or a SQL Server as the source

# BEST PRACTICE POLICIES

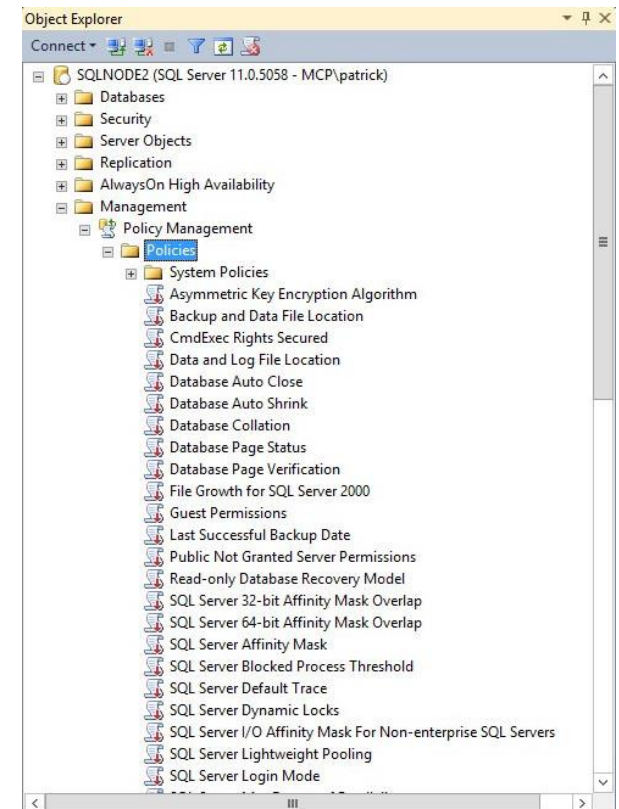
- XML files are installed along with the client tools

**SQL 2012:** C:\Program Files (x86)\Microsoft SQL Server\110\Tools\Policies\DatabaseEngine\1033

**SQL 2014:** C:\Program Files (x86)\Microsoft SQL Server\120\Tools\Policies\DatabaseEngine\1033

**SQL 2016:** C:\Program Files (x86)\Microsoft SQL Server\130\Tools\Policies\DatabaseEngine\1033

- 50 policies for all versions
- Import these to a server

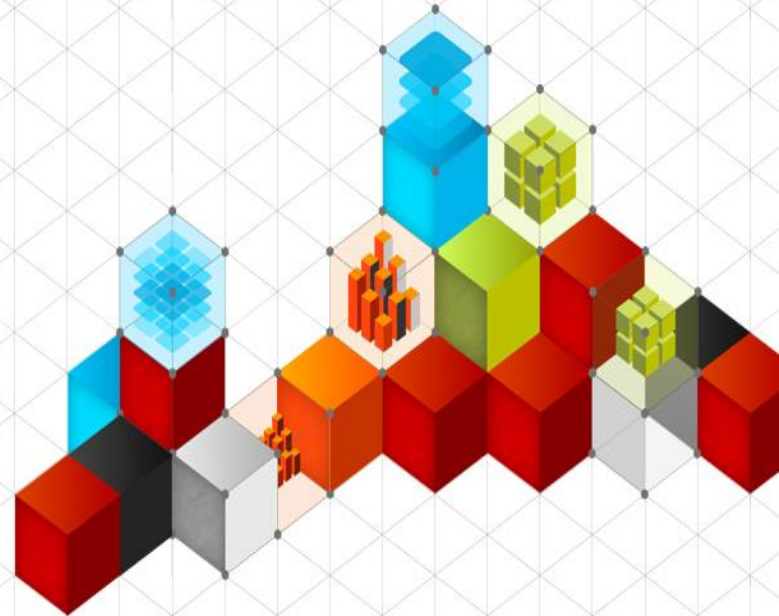


# PBM & CENTRAL MANAGEMENT SERVER

- CMS is a central location for registering SQL Servers
- That list can be shared with anyone
  - As long as they have access
- Using CMS, policies can be evaluated against all or a group of servers
  - Results are not stored locally
  - Can only be exported as XML

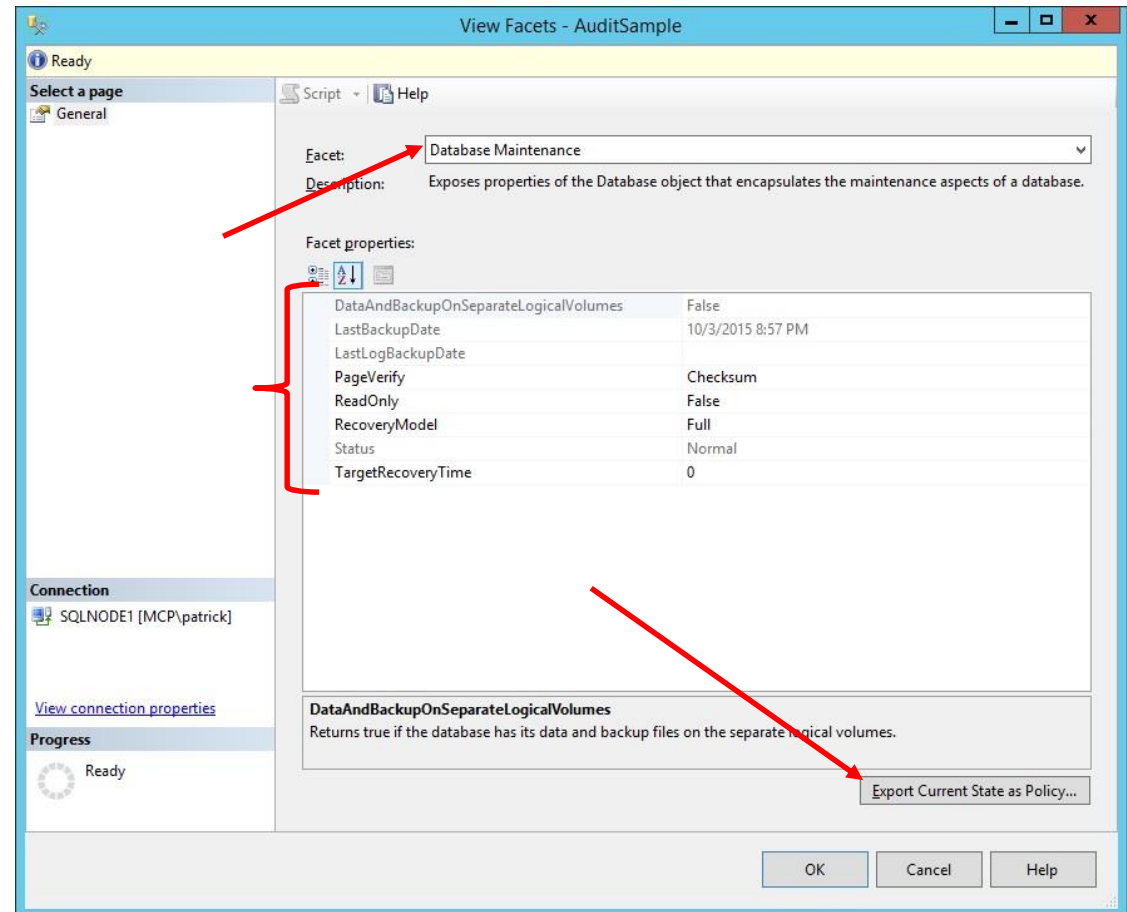


# DEMO 2: DECODING THE XML



# CUSTOMIZING PBM

- Setup a server or database with all the correct settings
- Export those settings as a policy





## CUSTOMIZING PBM

- Policy Based Management
  - aka - Declarative Management Framework

- "Framework" definition –

"A structure around or over which something is built"

Source: <http://dictionary.cambridge.org/us/dictionary/english/framework>

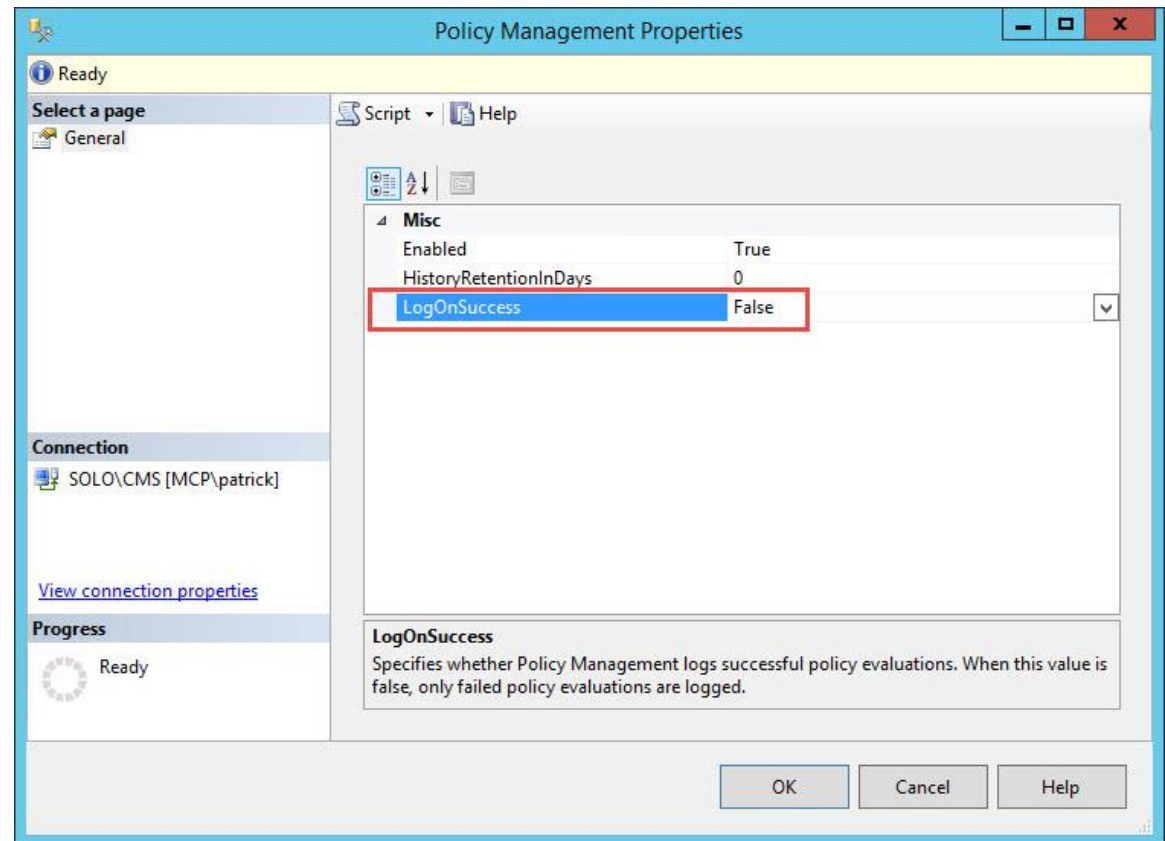
- PBM provides you with the tools that you will need to build a group of policies specific to your business.

- You must be creative when using PBM



# CUSTOMIZING PBM

- By default, PBM only stores details for failed policies
  - Enable this to keep more historical detail



# CUSTOMIZING PBM

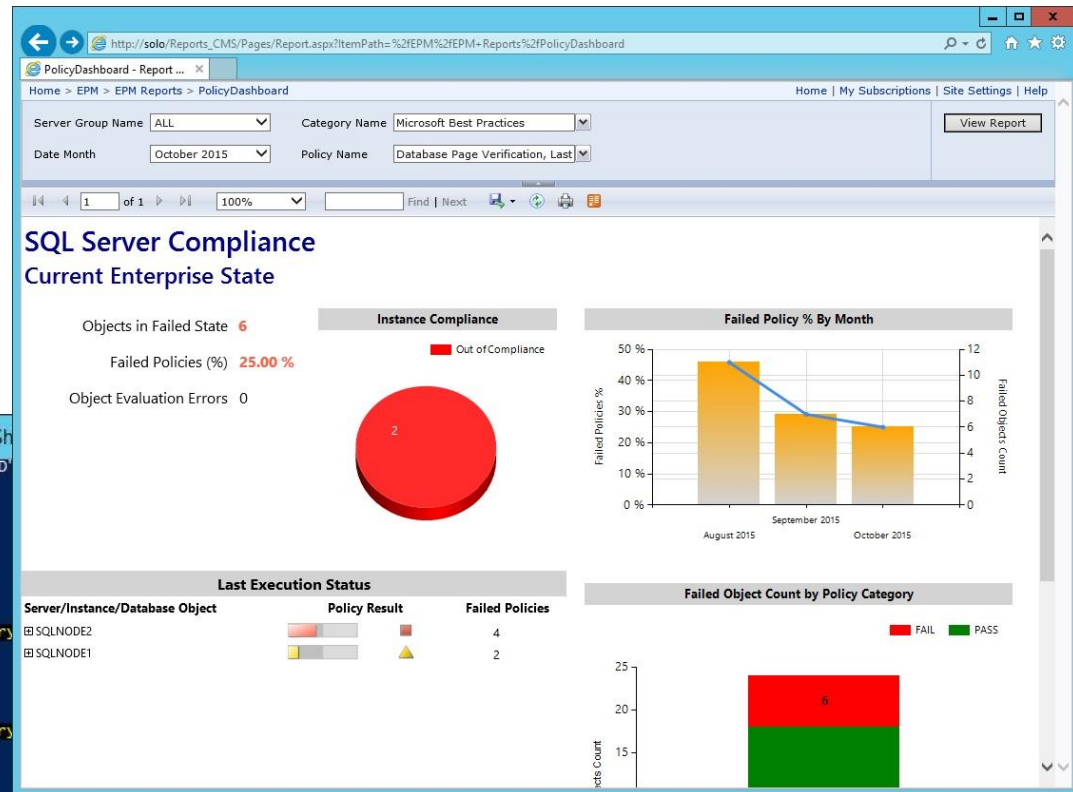
- Enterprise Policy Management Framework

<https://epmframework.codeplex.com/>

- Best when used with a CMS server

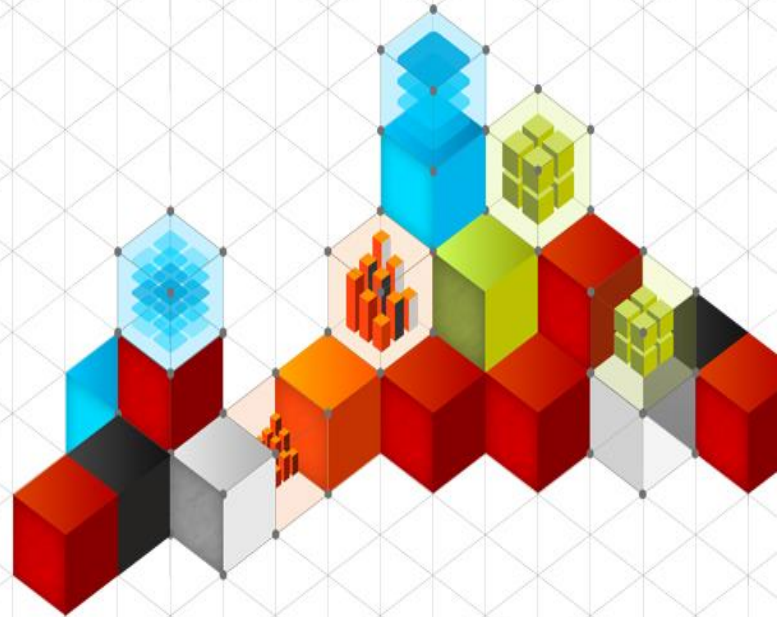
- Includes:

- PowerShell scripts to collect and import data
- SSRS reports



```
Administrator: Windows PowerShell
PS D:\results> .\EPM_EnterpriseEvaluation_412.ps1 -ConfigurationGroup "PROD"
ices" -EvalMode "Check"
EPM Framework v4.1.2
Starting policy category evaluation - 10/3/2015 9:03:49 PM
Loading Assemblies
SQLPS module already loaded
Connecting to the policy store
Get list of servers to evaluate
Starting server loop
Finished server loop
VERBOSE: Oct 3 2015 9:03:51:523PM - Starting data integration for Category
VERBOSE: Oct 3 2015 9:03:51:533PM - 24 rows inserted...
VERBOSE: Oct 3 2015 9:03:51:533PM - Starting no target data integration
VERBOSE: Oct 3 2015 9:03:51:533PM - 0 rows inserted...
VERBOSE: Oct 3 2015 9:03:51:533PM - Starting errors data integration
VERBOSE: Oct 3 2015 9:03:51:537PM - 0 rows inserted...
VERBOSE: Oct 3 2015 9:03:51:537PM - Finished data integration for Category
Finished policy category evaluation - 10/3/2015 9:03:51 PM
PS SQLSERVER:\SQLPolicy\SOLO\CMS\Policies>
```

# DEMO 3: EPM FRAMEWORK



# CUSTOMIZING PBM

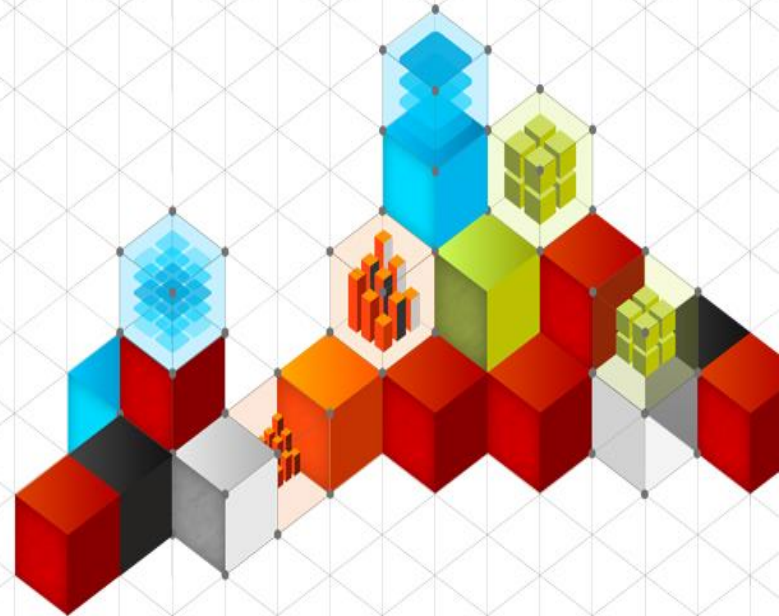
- Write your own policies using
  - ExecuteSQL()
  - ExecuteWQL()
- Write TSQL or WMI queries to return any data
- Valid return types:
  - Numeric
  - String
  - Bool
  - DateTime
  - Array
  - GUID
- Queries should only return 1 row



<https://msdn.microsoft.com/en-us/library/bb895209.aspx>



# DEMO 4: SCRIPTED POLICES



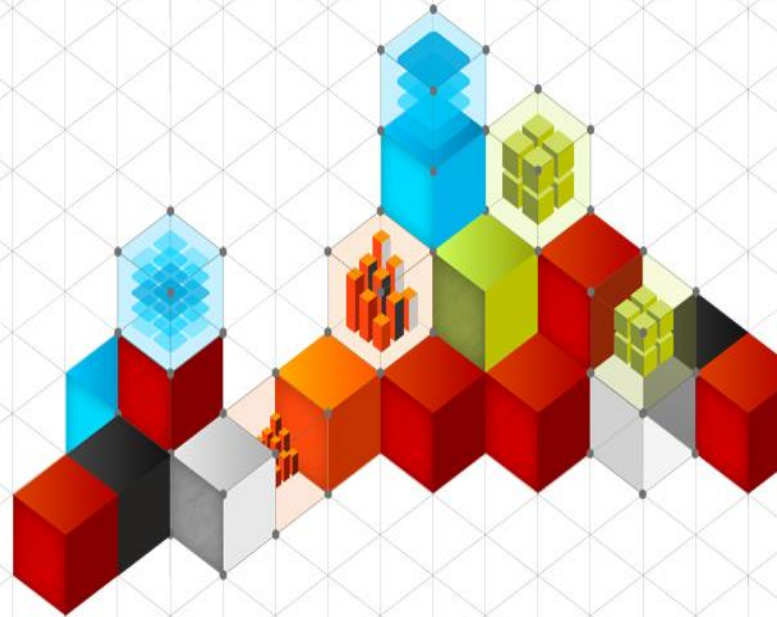
# PBM & POWERSHELL



- PBM is most flexible when used with PowerShell
  - Import-Module SQLPS
  - Invoke-PolicyEvaluation
- Policies can be evaluated against remote servers
- Detailed results can be saved as XML then imported into a database



# DEMO 5: POWERSHELL



## OTHER RESOURCES

- White Paper: SQL Server 2008 Compliance Guide  
<https://www.microsoft.com/en-us/download/details.aspx?id=6808>
- SQL PBM Blog  
<http://blogs.msdn.com/b/sqlpbm/>
- Pro SQL Server 2008 Policy-Based Management  
<http://www.apress.com/9781430229100>
- SQLPSX  
<https://github.com/MikeShepard/SQLPSX>

# Questions?