

ThoughtWorks®

*DevOps Community*活动

WINDOWS下自动化配置管理实践

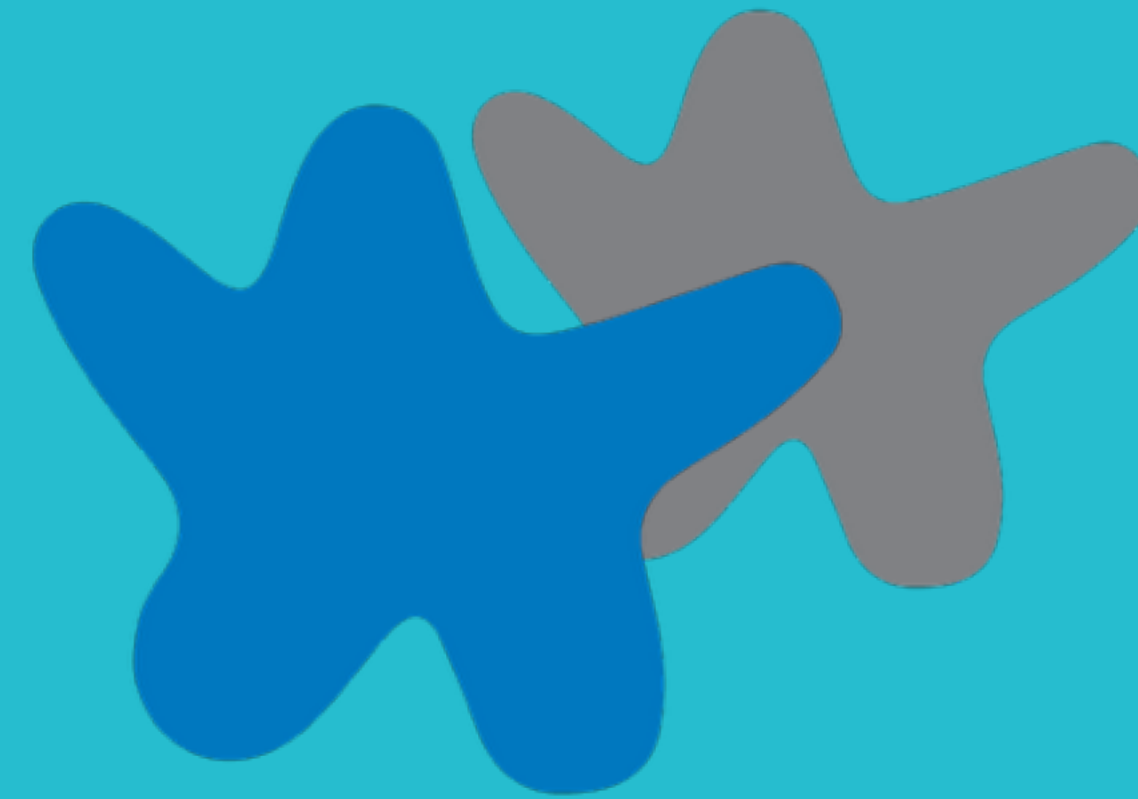
姚文杰 - *ThoughtWorks*研发工程师
wjyao@thoughtworks.com

日常环境的配置管理包括哪些

- 安装IDE、数据库等应用
- 管理文件、服务、网络等基础设施
- 部署网站等等
- 管理云平台基础设施



自动化 ≈ 代码/脚本化



WINDOW VS LINUX/UNIX

对比而言，微软在自动化声明配置及部署的领域确实有些迟钝

项目实际需求与困境

- 基于**.NET**开发
- 开发、测试、部署都是基于**Windows Server**机器
- 现存大量的Bat/Powershell脚本
- 几十台开发、测试机器
- 十几个云上产品环境



- 本地开发环境应用程序安装**不统一**
- 开发测试环境基本配置**容易变化**
- 搭建一台新的机器需要的**时间长**
- 新人对基本环境了解**不一致**

ThoughtWorks®

从安装应用自动化开始说起

最开始的方案



手动

+



BAT/PS脚本 (MSI)



- **Linux下包管理**
 - yum
 - apt-get

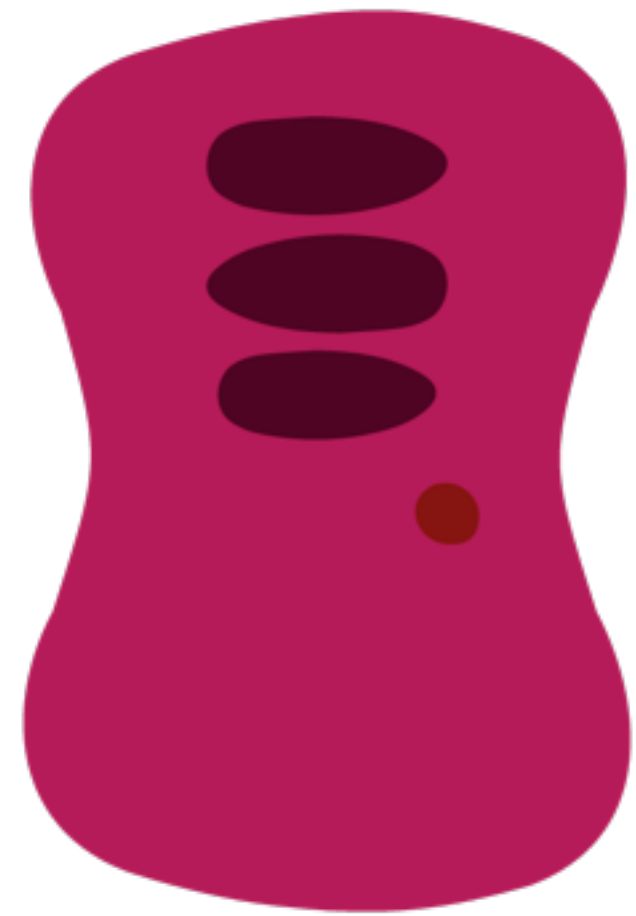


- 关注开发平台工具



- 对Nuget的一层封装，关注用户可用软件

- ▶ **chocolatey install git –version *.*.***
- ▶ **choco install git**
- ▶ **cinst git**
- ▶ **cinst git -source <http://192.168.56.11/nuget>**



Nuget Server
项目专用软件包仓库

- 实现了Windows下绝大多数软件包的安装和管理**自动化**
- 可以自己创建包源，搭建自己的软件包仓库，更加**安全**
- **扩展性高**，可以自己制作分享软件包

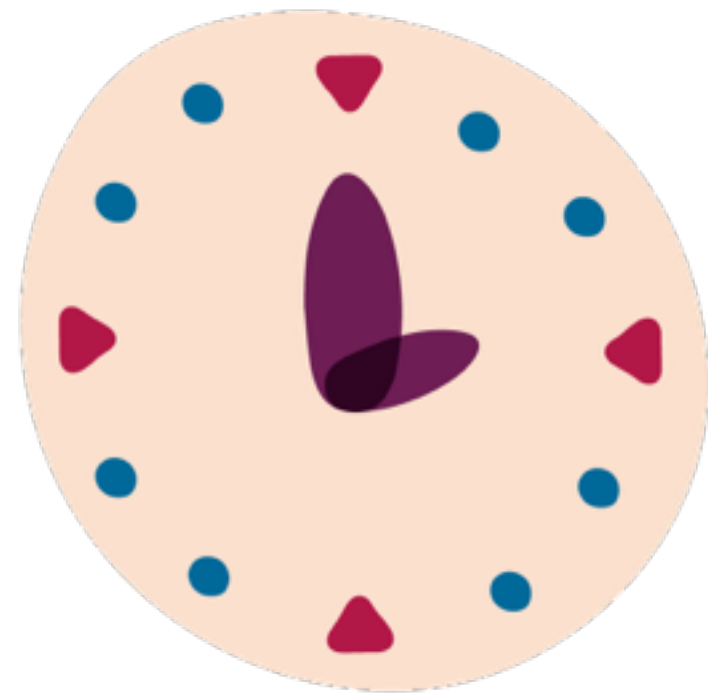
WINDOWS下其他配置管理/部署项

比如，管理文件、服务、进程，配置工具，部署网站等等

之前的做法



- **开发/产品环境**
采用手动和脚本并行的策略(半自动)



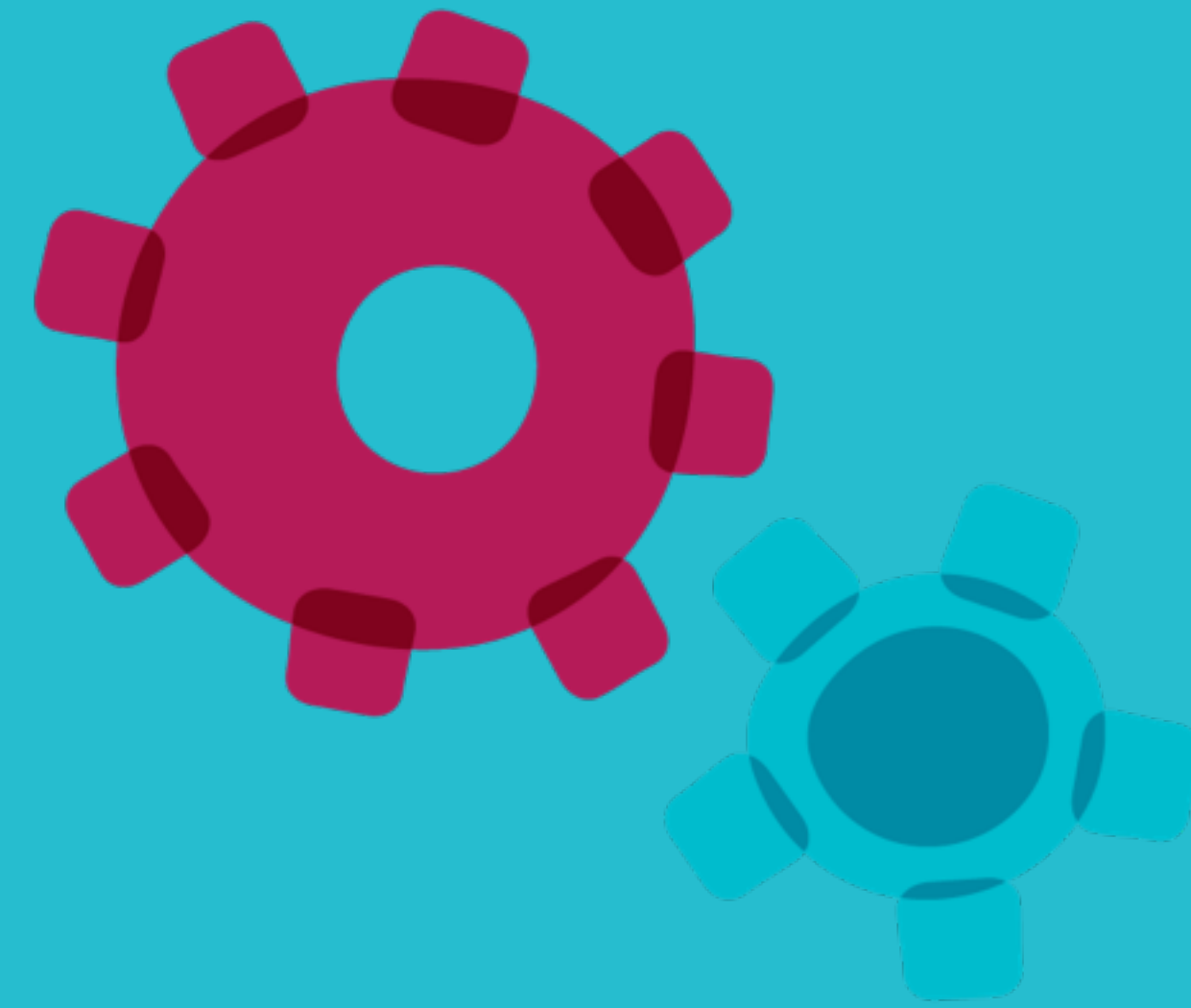
- **持续交付流水线中的测试环境（运行build、tests的机器）**
做好对应的镜像，每天凌晨定时从镜像中心拷贝
并重建对应的环境

几个缺点

- 新人对环境如何配置，具体配置成什么样**无从了解**
- 环境相对固定，**更改**会相对麻烦
- 维护的Windows镜像过于**庞大**
- 半自动化的管理和部署策略**风险较大**







ThoughtWorks®



使用配置管理工具?

常见的自动化配置管理工具

Tools	DSL	Support Windows?
 CHEF™	Ruby-based DSL	✓
 puppet labs	Ruby-based DSL	✓
 ANSIBLE	YAML	✓
 SALTSTACK	SLS格式文件（支持 YAML）	✓

*WINDOWS*自己原生的 自动化配置管理方案

DESIRED STATE CONFIGURATION



基于WMF4.0的

DESIRED STATE CONFIGURATION(DSC)
期望状态配置

几点原因 - DSC的特点

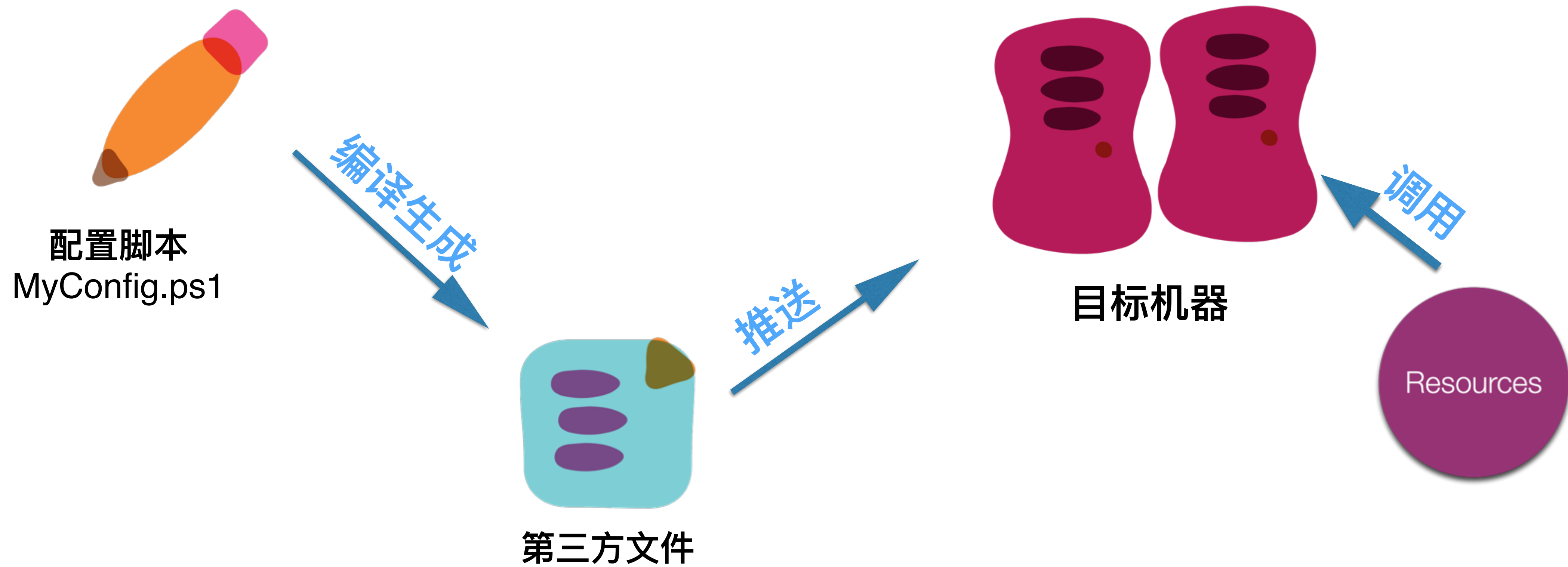


- 微软原生
- 远程通信方式多样、可靠
- 声明式的配置脚本
- 可扩展性高
- 支持多机器集群配置,部署
- 支持云平台及虚拟化技术



- **依赖PowerShell4.0(WMF4.0)**
 - 预装在Windows8.1及Windows Server 2012 R2的机器上
 - Windows 7, Windows Server 2008 R2, Windows Server 2012升级Powershell到4.0后也可以使用
- **对权限要求相对较高**
 - 用户操作权限
 - 网络访问权限等

DSC的实现方式



DSC配置脚本及运行

```
Configuration MyConfig
{
  Node "HostName_or_IP"
  {
    WindowsFeature IIS
    {
      Ensure = "Present"
      Name = "Web-Server"
    }

    File MyFileExample
    {
      Ensure = "Present"
      Type = "Directory"           # Default is "File"
      Recurse = $true
      SourcePath = $WebsiteFilePath
      DestinationPath = "C:\inetpub\wwwroot"
      DependsOn = "[WindowsFeature]IIS"
    }
  }
}
```

```
PS C: > MyConfig
#生成一个目录, 目录下MyConfig.mof
```

```
PS C: > Start-DscConfiguration -Path .\MyConfig
#执行一个DSC配置
```



DSC实现配置管理的核心?

Resource

WINDOWS自带的RESOURCE

Resource	描述
Archive	在目标机器上解压zip文件
Environment	管理目标机器的环境变量
File	管理目标机器的文件和目录
Group	管理目标机器上的本地用户组
Log	日志配置信息
Package	在目标机器上安装和管理应用程序包
WindowsProcess	管理目标机器上进程
Registry	管理目标机器上注册表key value
WindowsFeature	在目标机器上添加功能或者角色
Script	在目标机器上运行PowerShell脚本
Service	管理目标机器上的服务
User	管理目标机器上本地用户账号

扩展 - 微软实验性的RESOURCE

一些常用的扩展Resource		
xFileShare	xIPAddress	xSqlServer
xAzure	xActiveDirectory	xDatabase
xPhp	xSmbshare	xNetworking
xDisk	xDnsServer	xWordPress
xWordPress	xAdcDeployment	xWebAdministration

利用DSC自动化配置工具 - 以SQL SERVER为例



手动安装配置过程

A screenshot of a Google search for "sql server 安装教程". The search results show a video titled "SQL Server标准教程01 SQL Server安装" with a duration of 32:15. A red circle highlights the video thumbnail, and a red arrow points from the text "32分钟" to the video. The search results also include a link to a Microsoft MSDN page and a blog post from downcc.com.

Google sql server 安装教程

Web Videos Images News Maps More Search tools

About 1,220,000 results (0.35 seconds)

教程: SQL Server Management Studio - MSDN - Microsoft
<https://msdn.microsoft.com/zh-cn/library/bb934498.aspx> 轉為繁體網頁
请注意, 本教程适用于除SQL Server Express 外所有SQL Server 版本随附的 Management Studio 的完整安装。针对Management Studio 的基本安装和...

SQL Server标准教程01 SQL Server安装 - 在线播放 - 优酷网 ...
v.youku.com > 科技列表 > IT
SQL Server标准教程01 SQL Server安装本章从SQL Server的发展历史、安装要求及 ...

Win7 系统上安装SQL Server 2008一步一步图解教程_绿色 ...
www.downcc.com/tech/4135.html 轉為繁體網頁
Nov 11, 2014 - 这几天因为需要, 一直想安装SQL Server 2008来作为Web后台的数据库进行些实验, 但总是没有时间, 今天终于有时间了, 便安装了SQL Server ...

32分钟

安装配置SQL SERVER - 以前的实现 (纯POWERSHELL)

```
if(!(Test-Path 'C:\SQLServer2008.zip'))
{
    (New-Object Net.WebClient).DownloadFile('http://10.18.8.100/
sql_server.zip', 'C:\SQLServer2008.zip');
}
if(!(Test-Path 'C:\sql_server'))
{
    & 'C:\Program Files\7-Zip\7z.exe' x C:\SQLServer2008.zip -oC:\
C:\sql_server\setup.exe /ConfigurationFile=C:\ConfigurationFile.ini
}
```


安装配置SQL SERVER - DSC的实现

```
xSQLServerSetup MySQLServer
{
  SourcePath = ****
  SourceFolder = ****
  SetupCredential = ****
  Features = *****
  InstanceName = ****
  InstanceID = ****
  PID = ****
  UpdateEnabled = ****
  UpdateSource = ****
  ... ..
}
```

- 配置属性一目了然
- 依赖更少
- 更改方便



简单DEMO演示

利用*JS*部署一个简单的静态网站

把安装应用程序的过程也加入到DSC脚本中来



使用Package这个Resource



编写一个Resource, 名为Choco



自定义RESOURCE - CHOCO

- choco.psd1 - 基本数据（作者/版本等等）
- choco.schema.mof - 基本概要（属性及属性类型）
- choco.psm1 - 具体的执行脚本模块文件

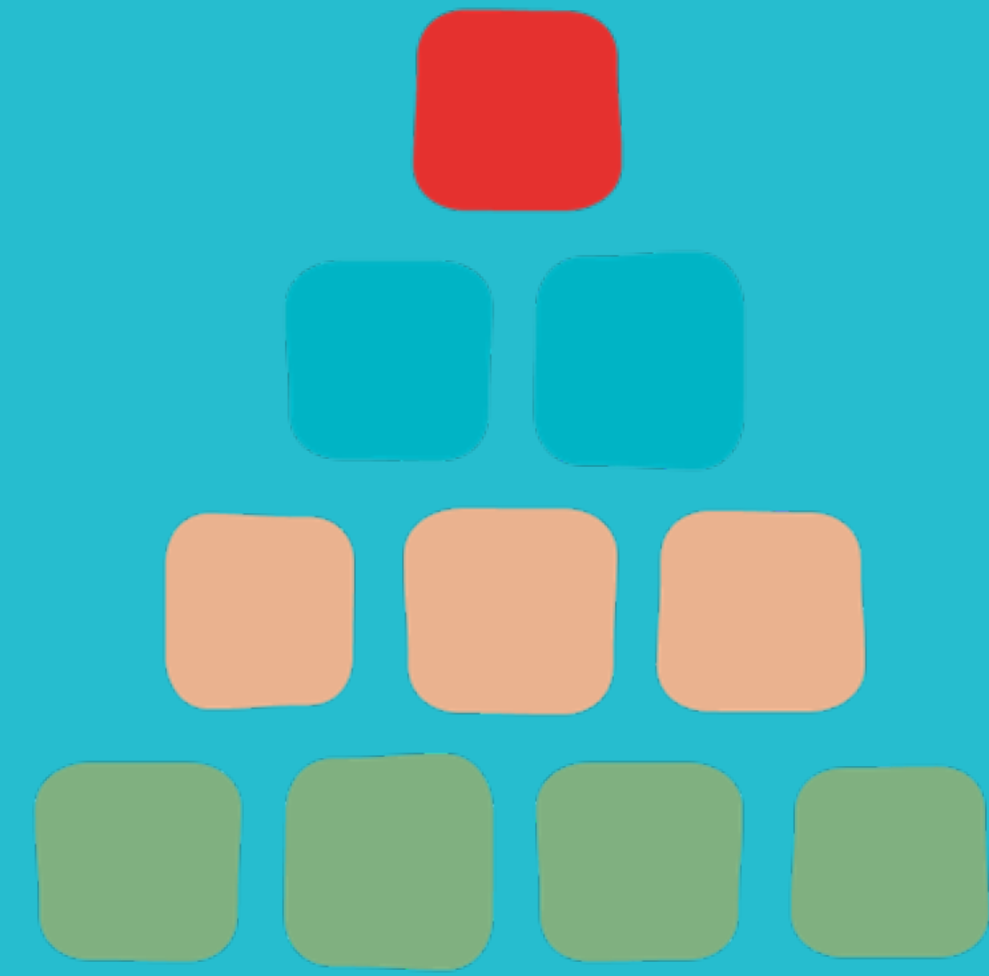
三个函数

- Get-TargetResource - 返回当前状态
- Set-TargetResource - 设置系统状态
- Test-TargetResource - 比较当前状态与理想状态

自定义RESOURCE - CHOCO

```
Configuration MyConfig
{
  Node "HostName_Or_IpAddress"
  {
    Choco Git
    {
      Name = "git"
      Ensure = "present"
      Version = ***
      Source = ***
    }
  }
}
```

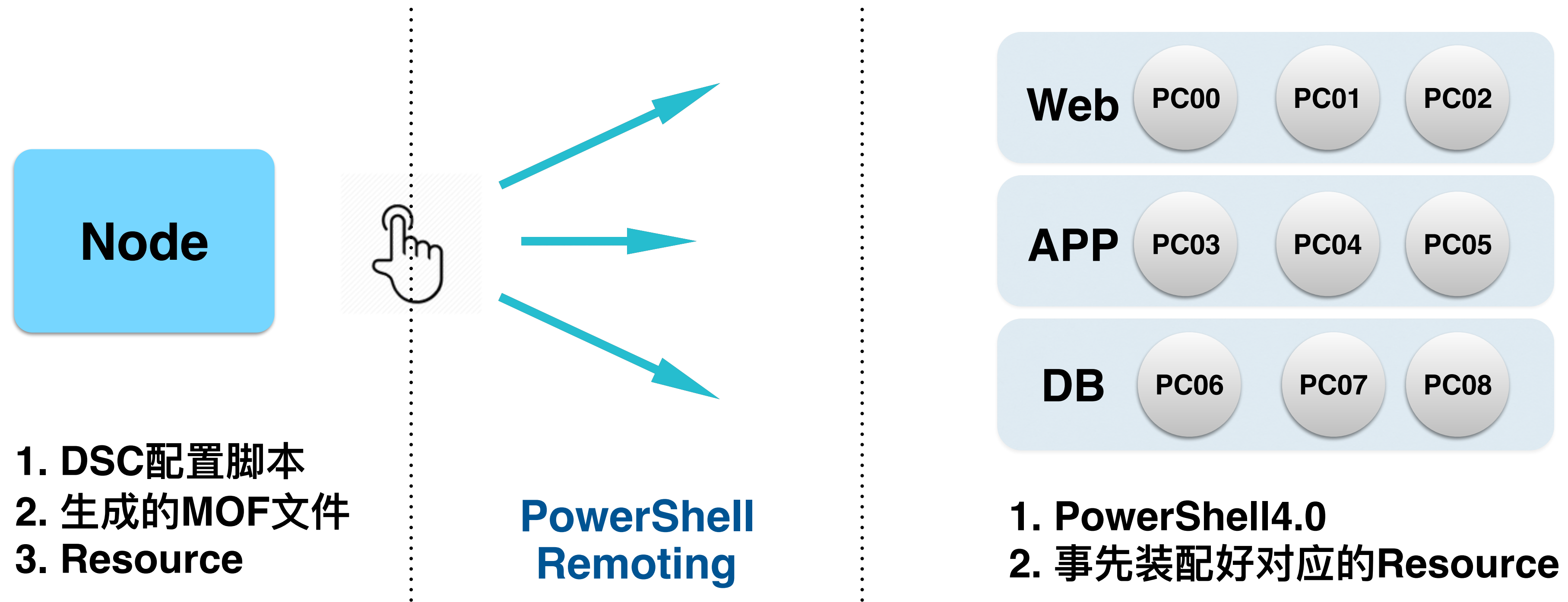
- 统一了配置脚本风格
- 状态管理
- 依赖 *Chocolatey*



集群部署方案

两种模式

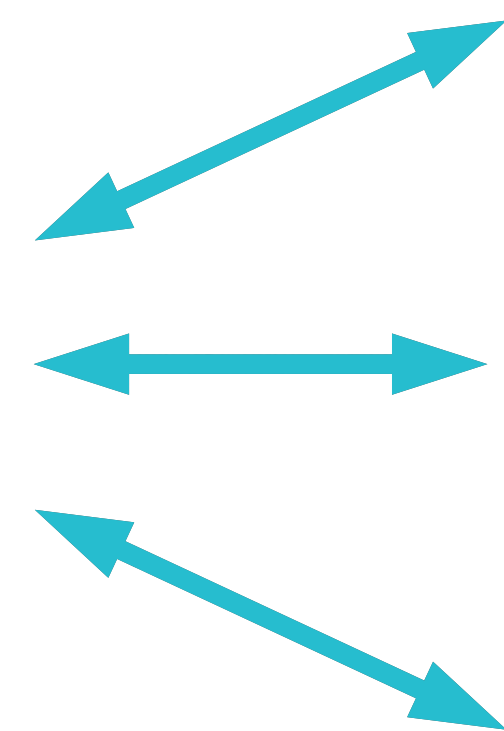
PUSH模式



PULL模式

Pull Server

1. DSC配置脚本
2. 生成的MOF文件
3. Resource



**HTTP(S)
或SMB协议**

Web

PC00

PC01

PC02

APP

PC03

PC04

PC05

DB

PC06

PC07

PC08

PowerShell4.0

两种模式对比

Push模式

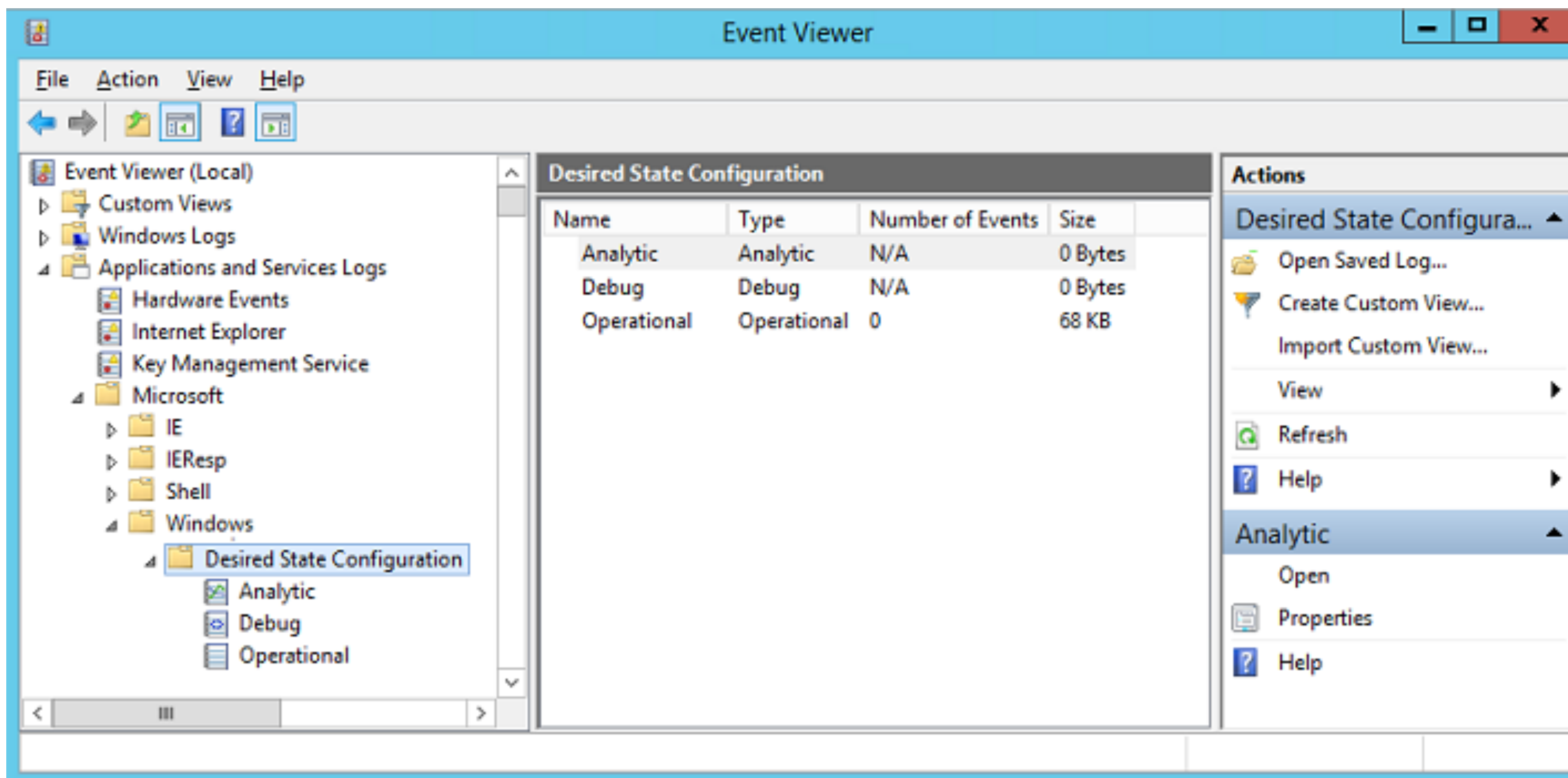
- 无实际服务器
- 部署前，Resource需要目标机器上
- 在执行时生效
- 适合少量机器管理及本地调试

Pull模式

- 需要Pull服务器
- 目标机器按需从服务器拉取Resource
- **定期监控**状态，确保机器处于期望状态
- 适合大量机器**集群**管理

日志及故障排除

- 在运行至加Verbose:
 如**Start-DSCConfiguration – Verbose**
- 使用Log、xDSCDiagnostic这样的内外部Resource
- 使用Windows自带的**Event Viewer**



总结 - 带来的好处

- 本地开发环境应用程序安装**不统一**
- 开发测试环境基本配置**容易变化**
- 搭建一台新的机器需要的**时间长**
- 新人对基本环境了解**不一致**
- 整体环境配置实现“**基础设施即代码**”
- Pull模式下**监控**的机器环境更加**可控**
- 搭建环境的过程更加**简单，规范**
- 环境配置更加**可视化**
- 易于配置**修改或变动**
- 后期**可扩展**



WMF 5.0, DSC功能增强, 添加Resource库, 愈发完善

该种实践的应用场景



适合的场景

- Windows服务器系统环境
- 环境配置复杂、易变



不适合的场景

- 非Windows平台或过老的Windows系统
- 太多的网络或者权限限制
- 环境配置简单，单纯

- ▶ 我对《The DSC Book》的中文翻译

<https://yaowenjie.gitbooks.io/the-dsc-book/content/>

- ▶ 本次Session的PPT

<https://yaowenjie.github.io/share/dsc-slide/>

- ▶ Demo样例

<https://github.com/Yaowenjie/PowerShell-DSC-Stuff.git>

THANK YOU

Q&A

姚文杰

yaowenjie.github.io

ThoughtWorks®