

# DevOps and Decoys- Be Part of the Solution with Automation

Kellyn Pot'Vin-Gorman,  
DevOps Engineer and now Data Platform Architect,  
Analytics and AI at Microsoft



# Kellyn Pot'Vin-Gorman

Data Platform Architect at Microsoft, EDU Team



/kellyngorman



@DBAKevlar

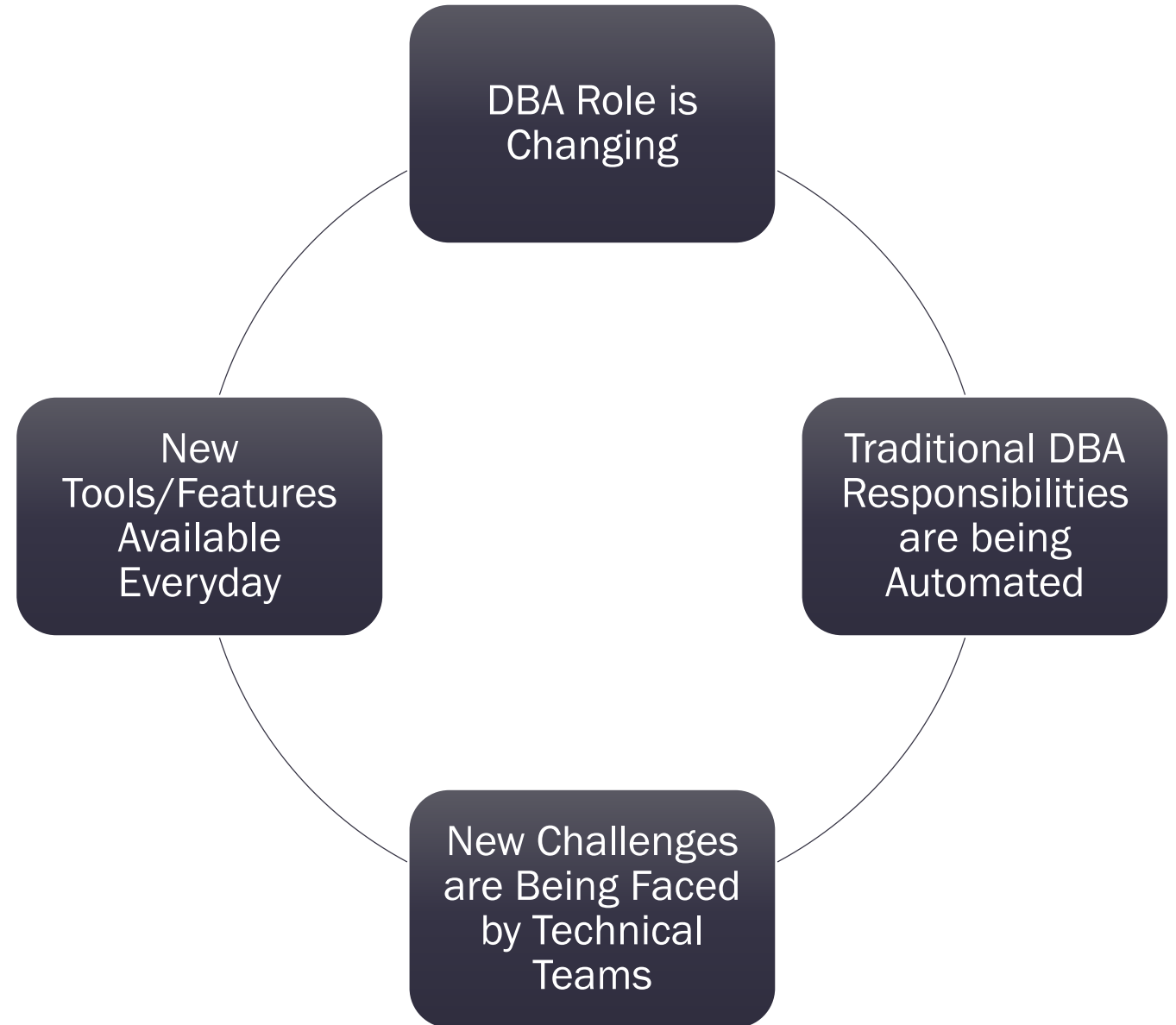


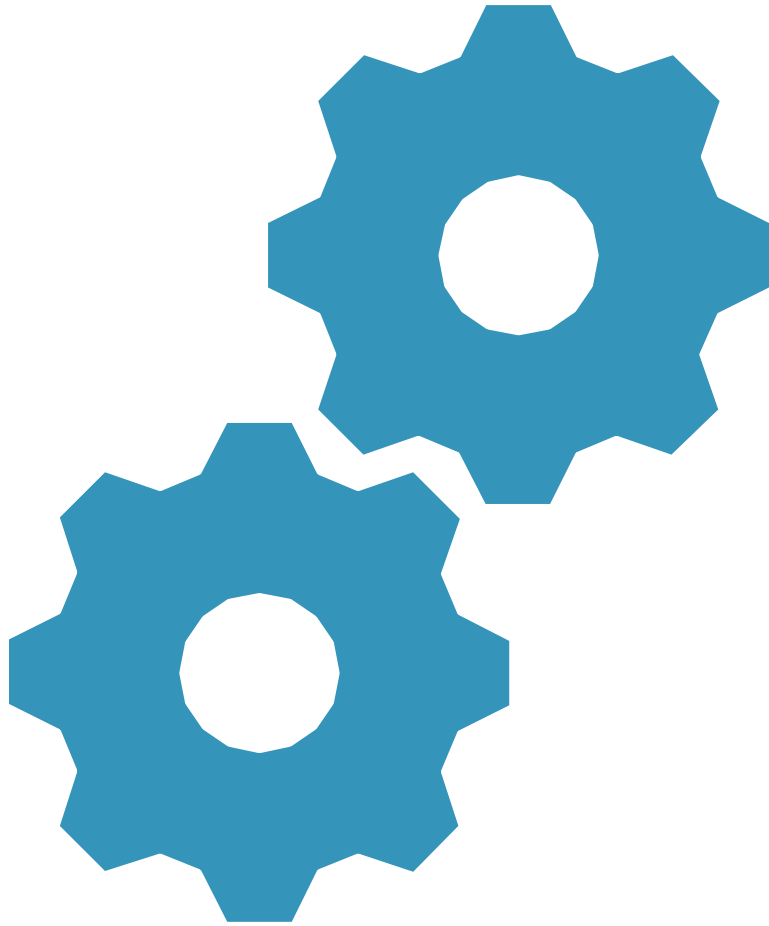
<http://dbakevlar.com>

We be bloggin'

- Multi-platform DBA, (Oracle, MSSQL, MySQL, Sybase, PostgreSQL, Informix...)
- DevOps Engineer in Azure and AWS
- Oracle ACE Director, (Alumni)
- Oak Table Network Member
- Idera ACE Alumni 2018
- STEM education with Raspberry Pi and Python, including DevOxx4Kids, Oracle Education Foundation and TechGirls
- Former President, Rocky Mtn Oracle User Group
- Current President, Denver SQL Server User Group
- Optimization, Linux and DevOps author, instructor and presenter.
- Blogger, (<http://dbakevlar.com>) Twitter: @DBAKevlar
- <https://linkedin.com/in/kellyngorman>

# A Few Facts





## Automation is a Key to Much of the Tech Acceleration

- Less intervention from resources to maintain, manage and operate database platforms.
- Advancement in tools to eliminate resource demands.
- More tool interconnectivity, allowing less steps to do more
- A resurgence of scripting to enhance automation and graphical interfaces to empower those who have wider demands.

# Workloads to the Cloud

- Simplify Resource Allocation and Management
- Initial ease on cost, but not an Apple to Apple comparison
- Can be migrated to any cloud, including- AWS, Oracle,Google Cloud, Azure or combination.
- Makes DevOps much easier.
  - *Automation*
  - *Tool availability*
  - *Cross platform consistency*
  - *The cloud LIVES on DevOps*

# What I'm Demonstrating Today...

I work with Microsoft customers, so yeah, it's going to be Azure...

This can easily be done with Oracle and I do it with Azure VMs in Azure all the time:

- *Create Azure VM, flashgrid, Create RAC environment with Oracle.*
- *Create Azure VM with Oracle databases, (PDBs)*
- *Create Oracle database and migrate over to Azure DB and use the Oracle Extension pack to simulate any features that aren't direct migrations.*
- *Use Azure Data Gateway to connect to Oracle DBs and use Microsoft products*

*We're all friends here.... 😊 The goal is to demonstrate how DevOps skills can be used for a number of situations...*

# First Things First-



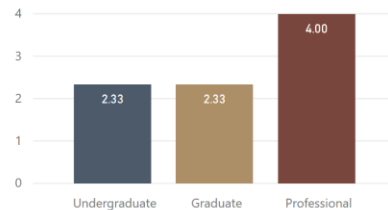
# Our Use Case

- Large Percentage of University Customers in MHE Searching for same solution
- EDU team created a Popular Solution that allowed for community involvement
- Source data is Oracle and SQL Databases, spreadsheets and Access.
- Multi-Tier Deployment- SQL Databases, ADF, Analysis Services, Data and Power BI
- Ever-evolving

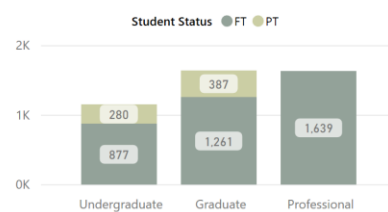


Undergraduate Graduate Professional

Median Time to Degree in Years - 2012-2016



Enrollment - Fall 2016

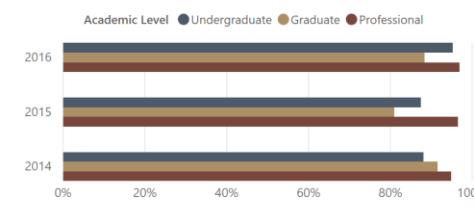


## Student Success

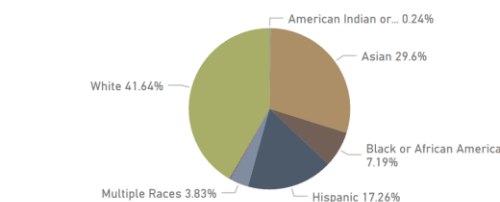
The success of Loma Linda University's vision of "Transforming lives through education, health care, and research" resonated when graduates from LLU ranked number one in a national survey on "How much satisfaction do you get from your job, and do you feel your job helps make the world a better place?"

Loma Linda University is a Seventh-day Adventist educational health-sciences institution with more than 4,000 students located in Southern California.

Retention Rate by Academic Level



Class of 2016 - Student Diversity



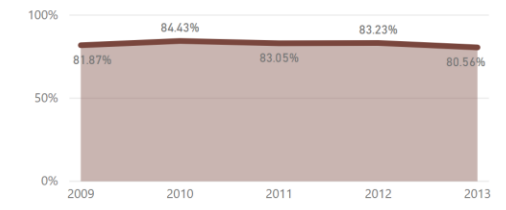
Recipients - All Time Awards - All Time

50,000+ 61,000+

Students from more than 70 countries around the world and every state in the nation are represented in Loma Linda University's diverse student body.

Eight schools comprise the University organization. More than 100 programs are offered by the schools of Allied Health Professions, Behavioral Health, Dentistry, Medicine, Nursing, Pharmacy, Public Health and Religion. Both traditional and online degrees are offered.

Completion Rate by Year of Matriculation



Loma Linda University  
Loma Linda, California 92350  
University: (909) 558-1000  
Public Relations: (909) 558-4526  
Admissions: (909) 651-5029

More information is available at:  
<http://nces.ed.gov/collegenavigator/?q=Loma+Linda+University&s=all&id=117636>

[Office of Educational Effectiveness](#)

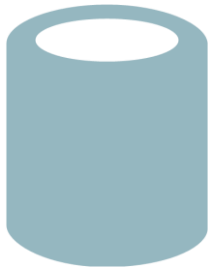


# Limited Time, Limited Resources

- Three TSP Data Platform Architects on the Team to Cover the US
- 100's of HigherEd Customers
- All of them interested in a solution vs. a product.
- All of them with different source databases or data sources.

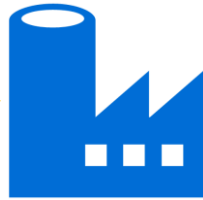
# Higher Ed Analytics Solution

Oracle/SQL  
Database, Access,  
Spreadsheets



Student Information System

VPN Gateway



Data Factory



Staging Database



Data Warehouse



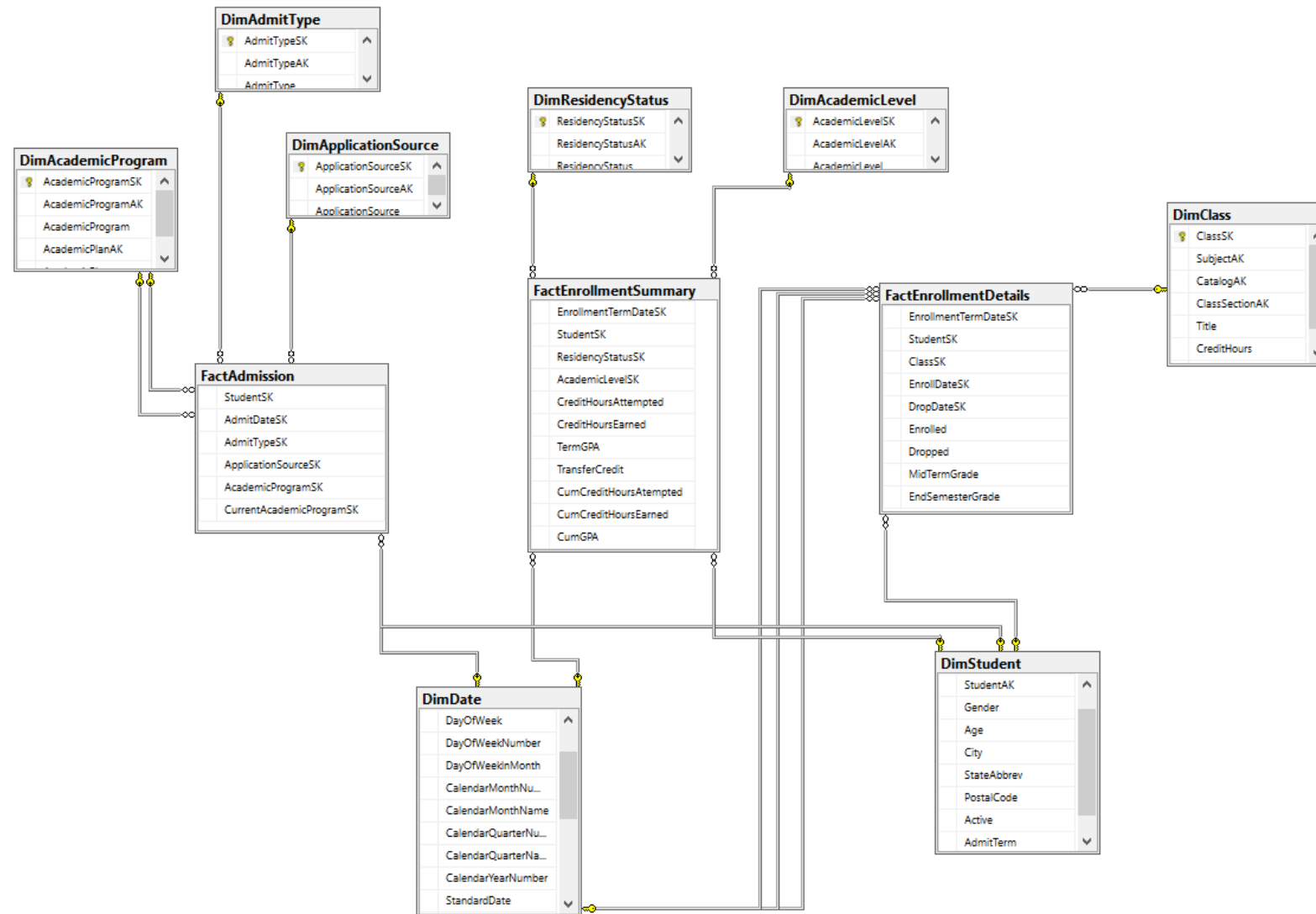
SSIS DB

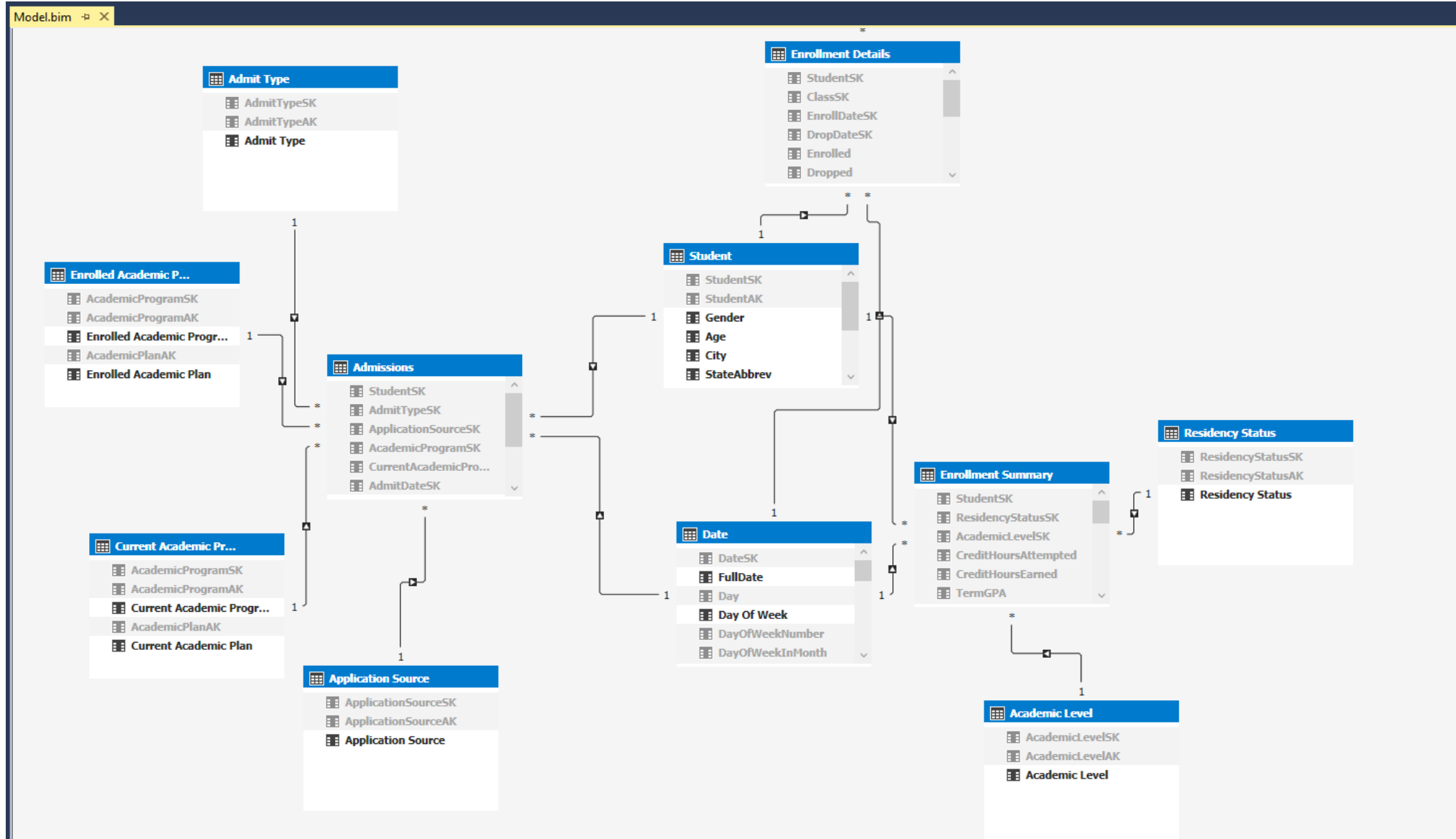


Analysis Services



Power BI





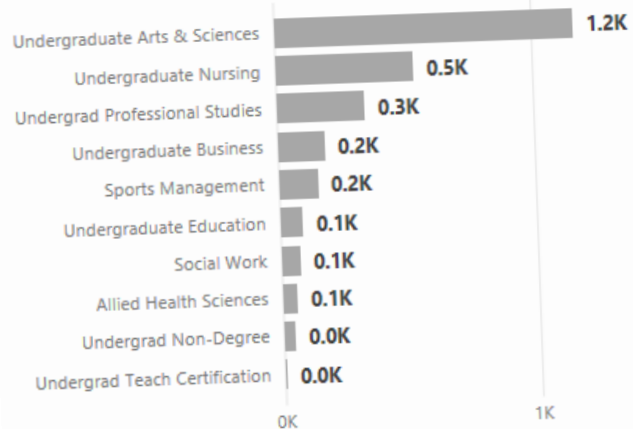
# Admissions Report



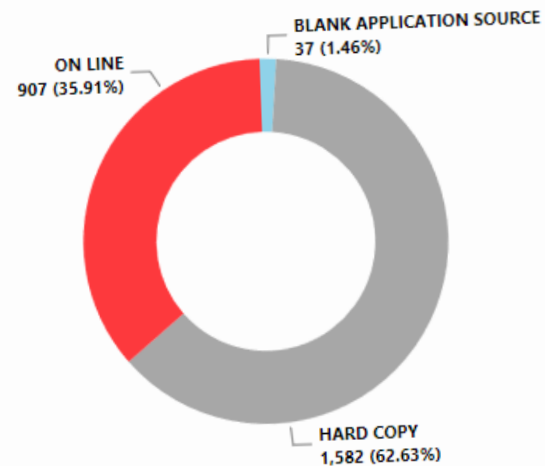
**Headcount**  
**2,453**

**Headcount Last Year**  
**2,535**

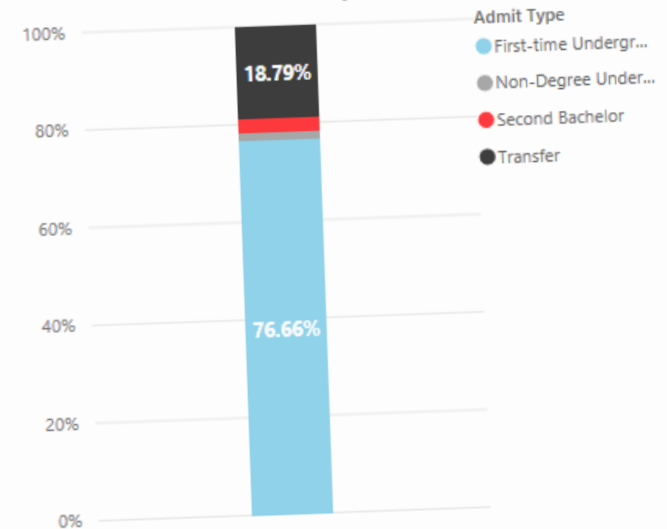
Headcount by Enrolled Academic Program



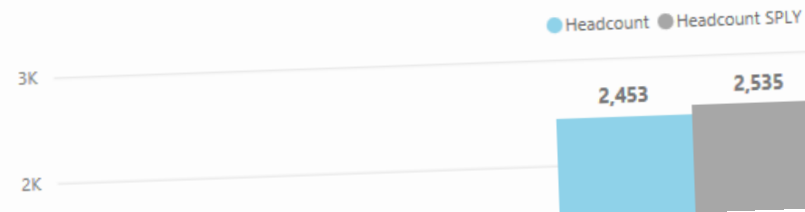
Headcount by Application Source



Headcount by Admit Type



Headcount and Headcount SPLY by School Year and Term



Year			Term	
2014	2011	2008		

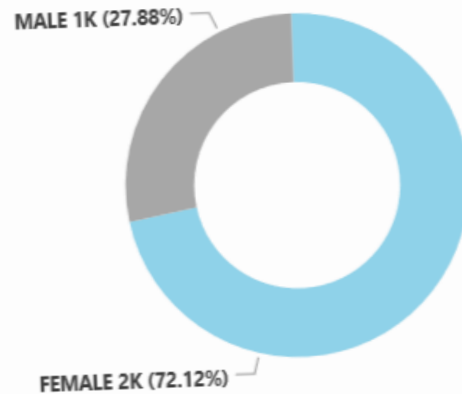
# Enrollment Summary Report



Students

2,514

Students by Gender



Year			Term	
2014	2011	2008		

School Year	Students	Returning Students	New Students	Retention	Retention Status
2013	1,521		1,521		
Spring	1,518	1,340	178	88.27 %	●
OutState	120	101	19	84.17 %	●
InState	1,399	1,231	168	87.99 %	●
Fall	98	97	1	98.98 %	●
OutState	9	9	0	100.00 %	●
InState	89	88	1	98.88 %	●
2012	2,457		2,457		
Summer I	128	12	116	9.38 %	◆
OutState	8		8		
InState	120	12	108	10.00 %	◆
Spring	2,245	1,636	609	72.87 %	▲
OutState	210	145	65	69.05 %	▲
Not Available	1	1	0	100.00 %	●
InState	2,039	1,451	588	71.16 %	▲
Fall	1,885	1,616	269	85.73 %	●
OutState	152	133	19	87.50 %	●
InState	1,733	1,477	256	85.12 %	●
Total	2,514		2,514		

Students and Returning Students by School Year and Term

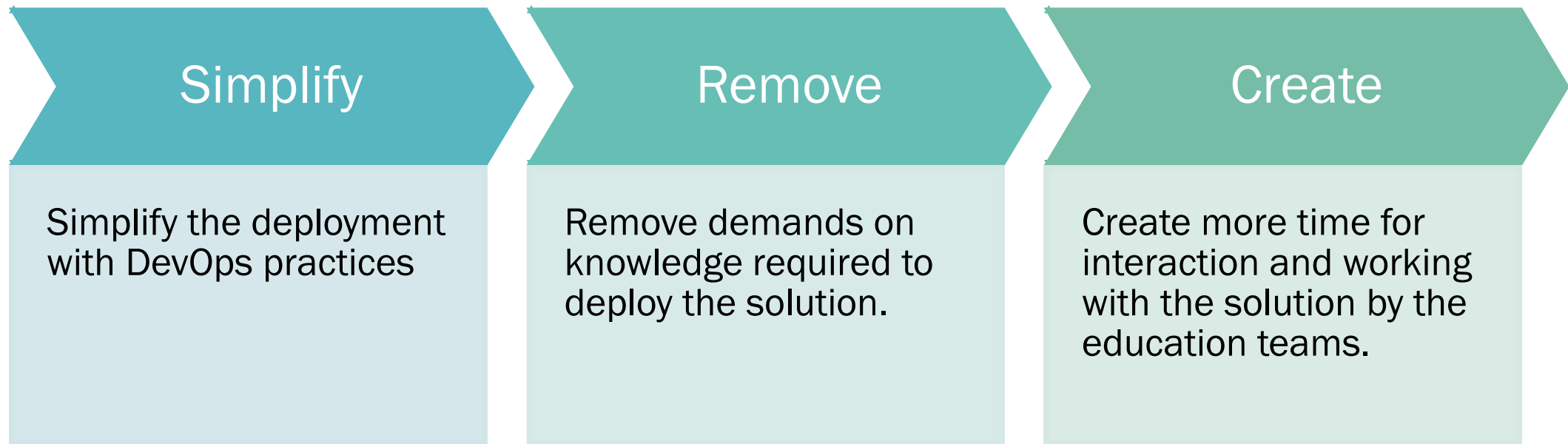
● Students ● Returning Students

# Deployment vs. Enjoyment

- Teams from universities are made up of varied technical backgrounds
  - *Data Scientists, DBAs, Data Analysts and Educators*
  - *Spend More time deploying than working with the solution*
  - *Slows down the time the EDU's limited resources get to work one-on-one with the customers*
  - *Discovered some customers lost interest during the deployment phase or didn't have the time to deploy*



# Goal





# Build a Roadmap

1

## Document All Pieces of Deployment

- Identify any interactive vs. default entries that will benefit the deployment.
- Update documentation as you go. Don't try to do it in the end.

2

## Categorize by Physical, Logical and Interactive

- Build out Physical Deployment first, as it is the foundation.
- Build in easy restart and clean up steps to physical deployments
- Test and update any documentation to reflect the automation

3

## Begin to automate logical slowly and in phases.

- Remove any manual steps or configurations.
- Take advantage of any plugins that ease work on end-users side
- Continue to accept feedback and enhance.

# What is Involved

- 1. Two SQL Databases- one staging and one data warehouse
- 2. Azure Data Factory with a SSIS Database
- 3. Azure Analysis Services
- 3. Three Power BI Reports
- 4. CSV files for ongoing data loads, which will need to be configured for ongoing workloads
- 5. Multiple Solution and project files, some deprecated in VS 2017
  
- \* Data loads and configuration via Visual Studio or SSDT solutions already built in.
- \* Sample data files in Excel could be replaced with the customers own data.

# The Solution Was Very Repeatable

- Most resources and databases could have the same name in every deployment.
- Outside of the CSV files containing example data, everything else could be deployed without any changes by the customer if a few dynamic parameters were pushed to the deployment.
- Although official documentation existed, there were numerous versions and the process had evolved with the introduction and ease of Azure deployment.

If you repeat, then automate

# Already Using Github

- Stable code, rarely changed

The screenshot shows the GitHub interface for a repository named 'HigherED BI Solution'. At the top, there are tabs for 'Code', 'Issues' (0), 'Pull requests' (1), 'Projects' (0), 'Wiki', and 'Insights'. Below the tabs, a message states 'No description, website, or topics provided.' A horizontal bar displays repository statistics: 41 commits, 2 branches, 0 releases, and 2 contributors. Below this bar are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and a green 'Clone or download' button. The commit history table shows a recent merge pull request by 'pleblanc72' and two previous commits: 'Updated README' and 'Initial commit', both dated 'a year ago'.

Commit	Author	Message	Time
15854f7	pleblanc72	Merge pull request #11 from spontello/master	Jun 8 2017
		Updated README	a year ago
		Initial commit	a year ago

<https://github.com/pleblanc72/msedubi>

# Visual Studio/SSMS Dev Tools

- My predecessor and team members built in some automation already using solution files!
  - *Awesome, these can be reused!*

Don't throw away value  
while evolving.

# How Do From Point A to Point Automate?

- Perform the task in the User Interface FIRST.
- Gather the information with the CLI to build out your scripts
- Test and retest comparing to the UI deployment
- Manage and maintain automation.
- Don't go back to manual processing or manual intervention.
- Build out in phases- physical to logical, enhancing and improving as we go along.

# So Many Choices...

- Multiple options to automate-
  - *Terraform*
  - *PowerShell with Azure Commands*
  - *Azure DevOps*
  - *Azure CLI with BASH*



# Terraform

- Freemium Product
- Came to Azure in 2016
- Another robust product that allows for automation of processing via script.
- Supports complex tasks and multi-platform deployments



```
#-----  
# Terraform Demo for MS Summit  
# Based off MS examples and updated by K. Gorman  
# Deploys Resources- group, vnet, VM and db  
#-----
```

```
# Configure the Microsoft Azure Provider
```

```
provider "azurerm" {  
  subscription_id = "  
  client_id       = "  
  client_secret   = "  
  tenant_id       = "
```

```
# Create a resource group if it doesn't exist
```

```
resource "azurerm_resource_group" "summittfgroup" {  
  name      = "Summit_TF_Group"  
  location  = "eastus"  
  
  tags {  
    environment = "Terraform Summit Demo"  
  }  
}
```

```
# Create virtual network
```

```
resource "azurerm_virtual_network" "summittfnetwork" {  
  name            = "summitVnet"  
  address_space   = ["10.0.0.0/16"]  
  # subnet         = "summitSubnet"
```

# Terraform CLI

# Azure CLI

- Allows a command line interface to the Azure cloud
- Can be installed locally on Windows, Mac and other OS platforms
- Can be run inside a Docker Container
- Can be used with the Azure Cloud Shell without installation.
- Flexible and robust, allows for a CLI solution and automation via scripting in PowerShell/BASH of Azure deployments.

# Once Installed

- Test:
- >az login
- >az account show

```
kellyn@Azure:~$ az account show
{
  "environmentName": "AzureCloud",
  "id": "          <Subscription ID>          ",
  "isDefault": true,
  "name": "Microsoft Azure Internal Consumption",
  "state": "Enabled".
  "tenantId": "          <tenant ID>          ",
  "user": {
    "cloudShellID": true,
    "name": "kegorman@microsoft.com",
    "type": "user"
  }
}
```

# AZ CLI is Simple to Use and Robust

- `>C:\EDU_Docker>az vm create -n LinuxTstVM -g dba_group --image UbuntuLTS --generate-ssh-keys`
- SSH key files 'C:\Users\kegorman.NORTHAMERICA\.ssh\id\_rsa' and 'C:\Users\kegorman.NORTHAMERICA\.ssh\id\_rsa.pub' have been generated under ~/.ssh to allow SSH access to the VM. If using machines without permanent storage, back up your keys to a safe location.
- - Running ..
- `C:\EDU_Docker>az vm list -g dba_group`
- `C:\EDU_Docker>az vm delete -n LinuxTstVM -g dba_group`
- Are you sure you want to perform this operation? (y/n): Y

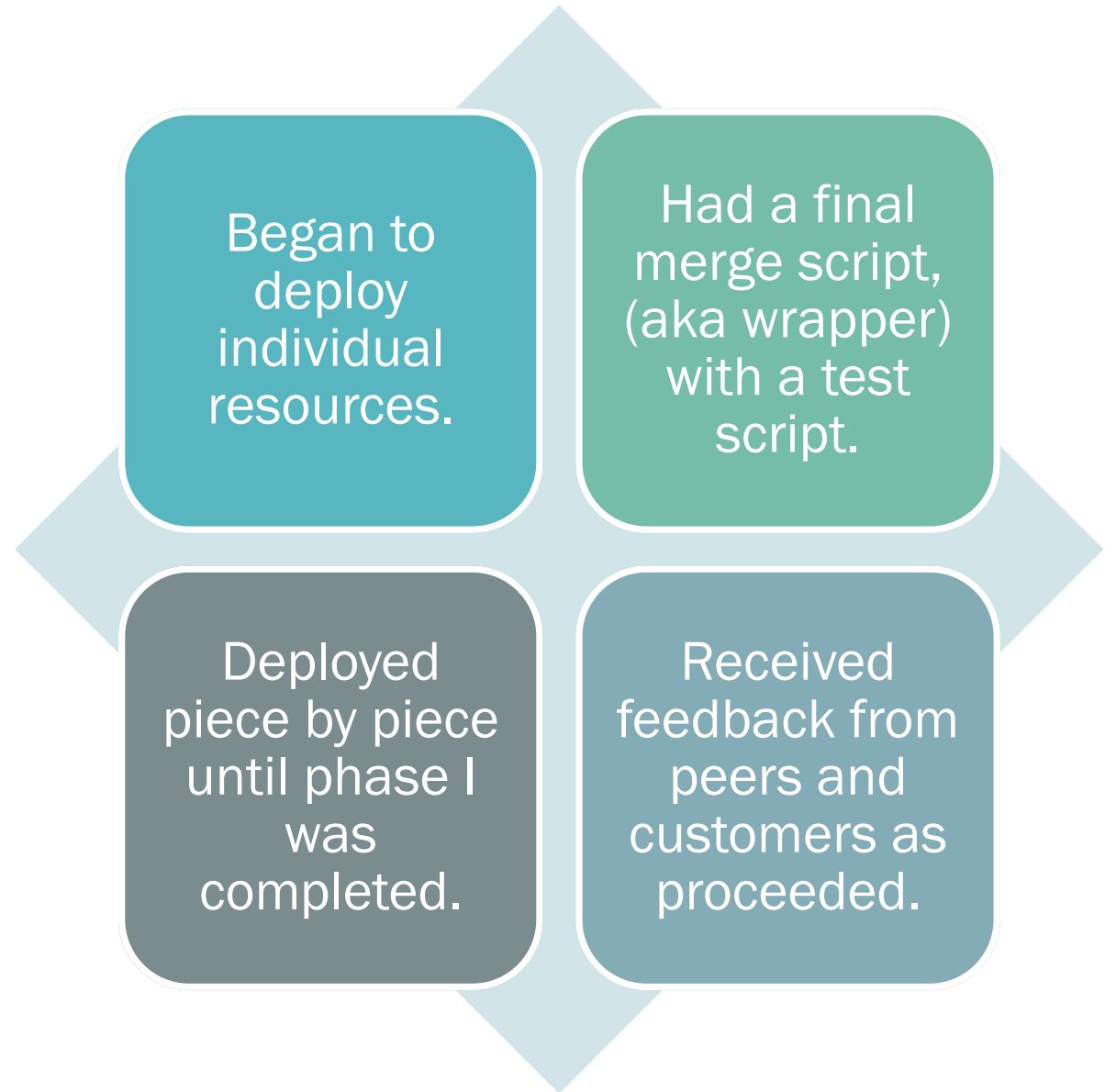
```
C:\Users\kegorman.NORTHAMERICA>az resource list --location eastus
[
  {
    "id": "/subscriptions/73aa270e-fffd-411a-b368-b44263f61deb/resourceGroups/higheredssas1",
    "identity": null,
    "kind": null,
    "location": "eastus",
    "managedBy": null,
    "name": "higheredssas1",
    "plan": null,
    "properties": null,
    "resourceGroup": "DBA_Group",
    "sku": {
      "capacity": 1,
      "family": null,
      "model": null,
      "name": "B1",
      "size": null,
      "tier": "Basic"
    },
    "tags": {},
    "type": "Microsoft.AnalysisServices/servers"
  },
]
```

## Locating Information on ADF

- 1. Create one in the GUI
- 2. Inspect the resource facts via the CLI:

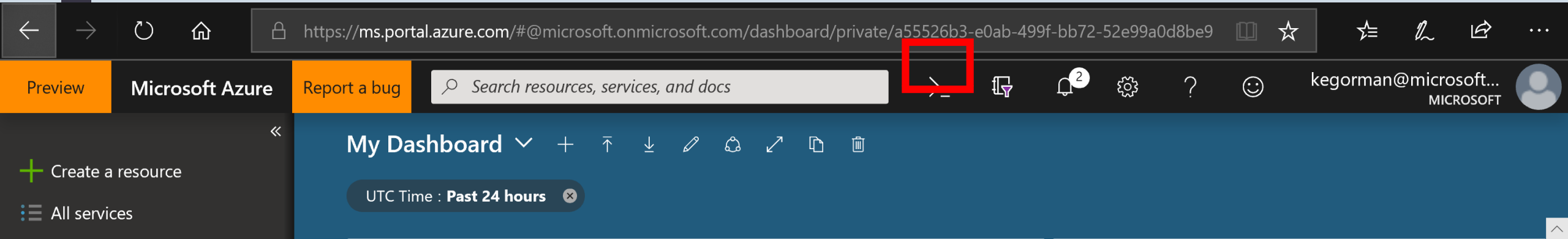
```
az resource list --
location eastus
```

## Walk Before You Run..



# Azure CLI Isn't Enough – Cloud Shell

- Enhanced Azure CLI commands into BASH script to deploy and automate.
- A script to enhance automation and set variables to ease customer skill requirements was required.
- From the Azure Portal:



Or Direct: <https://shell.azure.com/>

# Initial Build in BASH



- Well, I'm a Linux Person
- Script is interactive, accepting customer's requested naming conventions and requirements.
- Builds out the physical resources in Azure
  - *SQL Server with a data warehouse and staging database*
  - *Azure Data Factory*
  - *Azure Analysis Server*
- Creates firewall rules for Azure Cloud Shell
- Creates all user access
- Creates Database objects



# New GitHub Repository

EDUSolution / AutoEDU

Unwatch

1

Star

0

Fork

0

<> Code

Issues 0

Pull requests 0

Projects 0

Wiki

Insights

Settings

Repository for Automated EDU Solution Deployment for the MHE Community

Manage topics

16 commits

1 branch

0 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

Dbakevlar

Main Deployment Bash Script

...

Latest commit bdc0aaa 5 hours ago

AutoDeploy EDU PaaS.docx	Deployment Documentation	23 hours ago
README.md	Initial commit	9 days ago
ck_views.sql	Updated with Step II of logical deployment	a day ago
edu_deploy.sh	Main Deployment Bash Script	5 hours ago
edu_hied_DW.sql	Final checks on scripts and updates	a day ago
edu_hied_staging.sql	Support SQL and JSON	22 hours ago
parameters1.json	Support SQL and JSON	22 hours ago
parameters2.json	Support SQL and JSON	22 hours ago
template1.json	Support SQL and JSON	22 hours ago
template2.json	Support SQL and JSON	22 hours ago

<https://github.com/EDUSolution/AutoEDU>

Demo

# Two JSON Files, Dynamically Generated

```
cat >./$parametersFile1 <<EOF
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentParameters.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "name": {
      "value": "$adfname"
    },
    "location": {
      "value": "$zone"
    },
    "apiVersion": {
      "value": "2018-06-01"
    }
  }
}
EOF
```

# Living Documentation

Made easier transition as redesigned.

Kept track of all moving parts.









Offered insight to those who knew previous, manual process.

Allowed for roadmap to be included.

Allowed for troubleshooting section as other sections shrunk with automation.

# A Common Deployment Group

8 items ☐ Show hidden types ⓘ

<input type="checkbox"/>	NAME ↕	TYPE ↕	LOCATION ↕
<input type="checkbox"/>	 adftst9	Data factory (V2)	East US
<input type="checkbox"/>	 dbtst9sql1	SQL server	East US
<input type="checkbox"/>	 AzureDW (dbtst9sql1/AzureDW)	SQL data warehouse	East US
<input type="checkbox"/>	 HiEd_DW (dbtst9sql1/HiEd_DW)	SQL database	East US
<input type="checkbox"/>	 HiEd_Staging (dbtst9sql1/HiEd_Staging)	SQL database	East US
<input type="checkbox"/>	 dbtst9sql1_vnet1	Virtual network	East US
<input type="checkbox"/>	 MyRouteTable	Route table	East US
<input type="checkbox"/>	 tst9aas1	Analysis Services	East US

# Why Not PowerShell



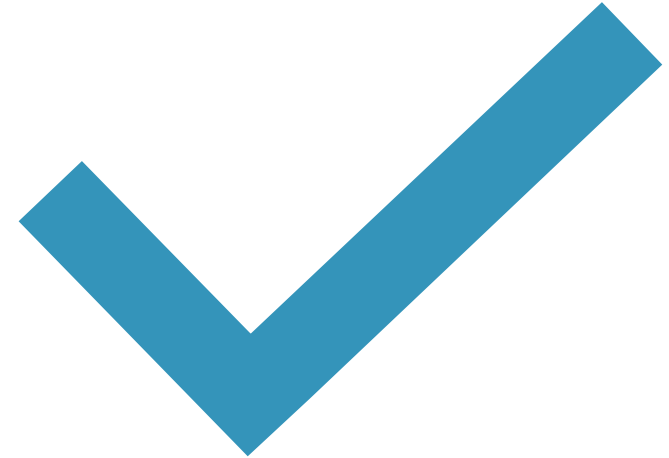
- Love of the SQL Community
- BASH is starting to become more prevalent for Azure.
- Numerous deployment areas still not robust enough to manage the requirements.

# Automate the Logical

- Migrate any onsite SSIS pkgs and workflows to Azure Data Factory
- ADF will build out an SSISDB in Azure SQL database
- Store projects in pipelines
- Schedule, report and check into a Github repository to automate development cycle.
- SLN and PROJ files are recycled.

# Next Steps

- PowerShell version on hold-
  - *More likely know PowerShell over BASH if Microsoft professional.*
  - *BASH only requires one main script to be updated with enhancements and additions to the repository.*
  - *Hardening script for security standards.*
  - *Migrate all SSIS, (Integration Services) to Data Factory steps, (Azure DevOps CLI already installed as part of the deployment).*
  - *Build out scripts to scale up the current “demo” version to an enterprise version.*
  - *Script to automatically pause integration and factory services to save on cost.*
  - *Script out data workload inputs for new features/data model and proprietary data loads.*
  - *Build out all steps using Azure DevOps to continue the growth of the automation.*





# Challenges?

No Reference Material on  
How to Create an Azure  
Data Factory from the  
Command Line

Azure CLI offered  
HOPE, but some of  
it had to be built out  
in the self-service  
command of  
“resource” and use  
json strings.

The motto when  
working with new  
tools- “Learn to  
Fish”

- Help menus and examples online are your best friend.
- Get a good text editor.
- You will be learning new scripting languages
- Find support groups, twitter handles and help channels.



# Success

- Manual Process takes between two full days of onsite meetings, to 8 weeks of remote meetings to deploy.
- New Automated process deploys in less than 15 minutes after customer answered questions in interactive script.
- Offers extensively more time for customer to work with solution and Microsoft architects to work with providing value to the customer on how to use Power BI with Azure.
- In first week, over two dozen customers requested POC deployment with little assistance from heavily limited resource team, allowing for more valuable allocation of resources.
- Now part of Partner development project to assist partners with Azure deployments.
- If the resources aren't needed any longer, **Dropping Resource Group removes it all.**

1

Data WILL come from more than just SQL Server and Azure SQL Databases.

2

You will be expected to do more with less.

3

Embrace automation tools inside the database platform, (backup, recovery, optimization, maintenance tasks.)

# You Must Evolve

# Thank You

Learn more from Kellyn Pot'Vin-Gorman



@DBAKevlar



kegorman@microsoft.com

