

# Summary

I have been using the following standards and conventions with the Azure Cloud Platform for years. I've found it to be critial in working with teams to be opinionated in the conventions used as most teams need this level of structure but your milage may vary. This becomes particularly important when your team starts to implement Infrastructure-as-Code with a technology like Terraform, as all of the rules need to be well-understood.

I developed them before [Microsoft created their documentation](#). Surprisingly, we ended up with a very similar structure.

The main differences is that I put the resource type at the end of my convention and they put it at the beginning. My reasoning is that the resource type is typically less relevant than the ownership and purpose for the resource and that a convention should always go from general to specific.

I also like to develop patterns (codes) that I can use regular expressions and automation to access and ensure proper resource controls. For example, I know the number of characters that the `<DepartmentCode>` will be and where it will reside be with relation to the dashes. Even more important than the pattern is keeping the resource name within the associated length defined by Azure. This means that I can search for these patterns and enforce them.

Regarding the subscription naming, I'm a big believer in getting the subscription level down to a physical team level. The reason for this is billing. I want the teams to be responsible for their own subscriptions and to have Billing Consumption reports built into the subscription.

Additionally, I like to use a `-prod` and a `-nonprod` designation on subscriptions for regulated companies that need to keep different controls and budgeting on production vs non-production environments and different controls on the sensitive data stored in them. For example, developers can access `-nonprod` freely, but `-prod` is more controlled.

Below I have my recommendations. The items in the angular brackets means that it is defined in the [Glossary](#) below.

## General Entities

Entity	Scope	Length	Casing	Valid Characters	Required Pattern
Subscription	Subscription	1-64	Insensitive	All characters	<DepartmentCode>-<TeamCode>-<ServiceLevelCode>
Resource Group	Subscription	1-90	Insensitive	Alphanumeric, hyphens, underscores, periods (except at end), and parentheses	<DepartmentCode>-?<TeamCode>-<ServiceLevelCode>-<LocationCode>-<Purpos
Tag	Associated Entity	512 (named), 256 (value)	Insensitive	Alphanumeric including Unicode characters; special characters except <, >, %, &, , ?, /. See limitations here . Maximum of <a href="#">50</a> .	"key" : "value"
Key Vault	Global	3-24	Insensitive	Alphanumeric and hyphens	<DepartmentCode>-?<TeamCode>-<ServiceLevelCode>-vlt
Service Principal	Scope-dependent	1-120	Insensitive	Alphanumeric and hyphens (not < > ; & %)	<DepartmentCode>-?<TeamCode>-<ServiceLevelCode>-<Description>-sp

# Azure DevOps (ADO) Entities

Entity	Scope	Length	Casing	Valid Characters	Required Pattern	Ex
Variable Group in ADO	Global	1-128	Insensitive	Alphanumeric, underscores, hyphens, periods (except at end), parentheses	<DepartmentCode>-<TeamCode>-<ServiceLevelCode>-?<PurposeCode>-Values	IT-sap-
Variable Group linked with Azure Key Vault	Global	1-128	Insensitive	Alphanumeric, underscores, hyphens, periods (except at end), parentheses	<DepartmentCode>-<TeamCode>-<ServiceLevelCode>-?<PurposeCode>-vlt	IT-SAP-
Variable	Variable Group	1-128	Insensitive	Any URL characters	PascalCase	BuildCo

## Glossary

### DepartmentCode

2 characters to represent the associated Department. (see examples below)

Abbreviation	Description
AC	Accounting
IT	Information Technologies
EN	Engineering

### TeamCode

3 letter code for the team that the resource is related to. This could be a product that you're working on, or team within a department.

Code	Team Name
sap	The SAP component of the application, if one exists.
ops	The operational team within IT
dot	The DevOps team
spm	The Stay Puft Marshmallow development team

### LocationCode

3 characters to represent the location of the VM. This can be a cloud region or it can be a physical datacenter. I try to use airport appreviations for the city if one exists that defines the city.

Code	Description
CHI	Chicago Data Center

Code	Description
PGH	Pittsburgh Data Center
ZET	Microsoft Azure US East
ZWT	Microsoft Azure US West
ZWC	Microsoft Azure US West Central
WET	AWS US East
WWT	AWS US West

ServiceLevelCode

1 character for the lifecycle stage of development.

Code	Abbreviation	Description
A	train	Training
B	bld	Build
C	poc	Proof of Concept
D	dev	Development
F	perf	Performance Testing
H	hf	Hotfix
I	inf	Infrastructure (SMTP Servers, etc)
N	np	Non-Production
O	demo	Demo
P	prod	Production
R	dr	Disaster Recovery
S	sta	Staging/Stable
T	test	Test/QA
U	uat	User Acceptance Testing
X	sbox	Sandbox

PurposeCode

3 character code for the role/purpose of the VM.

Code	Description
APP	Application server (Windows Services, Console Applications)
BLD	On-Premise Build Server
EXC	Exchange Server
FIL	File Server
PRT	Print Server
SQL	SQL Server
WEB	Web App/API Server

**Description**

Free form field for any notes or descriptions about the resources.