



DETAILED DESIGN SPECIFICATION (DDS)

SSC NAMING AND TAGGING STANDARD FOR AZURE AZURE VIRTUAL DATA CENTRE

Cloud Research and Development
Cloud Product Management and Services Directorate
Chief Technology Officer Branch
Shared Services Canada



Document Revision:	Release v2.0
Status:	Final
Publish Date:	Jan 6, 2021



Shared Services
Canada

Services partagés
Canada

Canada

DOCUMENT APPROVALS

The signing authorities below concur with the conditions and responsibilities specified within this document.

Jacqueline Morcos

Director,
Cloud Research and Development
Cloud Services Directorate
Chief Technology Office
Shared Services Canada

Date

Signature

Martin Vallières

Director,
SSC Cloud Oversight and
Application Portfolio Management
Chief Information Office
Shared Services Canada

Date

Signature

DOCUMENT HISTORY

Document Author(s) & Co-Author(s), Position – Title

Full Name	Role	Department - Position - Title
Simon Earle	Author	SSC, CSD Research and Development

History of Changes

Ver. #	Date	Consulted/Reviewers (name of individual & working group consulted)	Brief description of Change	Author of Change
v0.1	2019-06-10	Project Team Review	Initial Release for R1b VDC	SSC Cloud R&D
v0.2	2019-06-14		CSD Review	SSC Cloud R&D
v0.3	2019-06-17		Additional network objects	SSC Cloud R&D
v0.4	2019-06-17		Fixed inconsistencies and extended the field definition tables	SSC Cloud R&D
v0.5	2019-09-10		Standard for PBMM Release 2c – first draft	SSC Cloud R&D
v0.6	2019-09-16	SSC CIO	Align with Azure GC Accelerators	SSC Cloud R&D
v0.7	2019-09-24		Document renamed: SSC Naming and Tagging Standard for Azure v0.7	SSC Cloud R&D
V0.8	2019-10-12	SSC CSD R&D	Arch team review and final decisions for R2c naming standard Note: This version does not have section 4 updated	SSC Cloud R&D
V0.9	2019-10-26	Project Team Review	Tagging section completed for R2c – updated with input from TBS	SSC Cloud R&D
V1.0	2019-11-7	Cloud Technical Working Group	Release candidate for signature	SSC Cloud R&D
V1.1	2019-12-10	SSC CSD & CIO	Director Review	SSC Cloud R&D
V1.2	2019-12-14	TBS Feedback	Updated TBS Governance tags	SSC Cloud R&D
V1.3	2020-1-13	GC Working Group Feedback	Removed case dependency on tags	SSC Cloud R&D
V1.4	2020-3-11		Updated object tables	SSC Cloud R&D
V1.5	2020-3-20		Updated Tenant AD object naming for multi-region	SSC Cloud R&D
V1.6	2020-04-15		Subscription and resource group change, new definition for owner and contact tags	SSC Cloud R&D
V1.7	2020-05-22		Added PaaS device type and new suffixes	SSC Cloud R&D
V1.8	2020-08-24		Update for R2c	SSC Cloud R&D
V1.9	2020-09-16		Fix object tables and general cleanup	SSC Cloud R&D
V2.0	2020-12-29	SSC Cloud teams	Replace field value table tracking with coding strategy, remove “_” as reserved character, add automation section	SSC Cloud R&D

Security Classification **Unclassified**

Status: DRFT

Subject: SSC NAMING AND TAGGING STANDARD FOR AZURE

TABLE OF CONTENTS

1	Overview	6
1.1	<i>GC Cloud Governance for Object Naming</i>	<i>6</i>
1.1.1	Field Delimitation	8
2	Azure Naming and Tagging High Level Design	9
2.1	<i>Design Principles and Objectives</i>	<i>9</i>
2.1.1	Naming and Tagging Design Principles.....	9
2.2	<i>Infrastructure as Code (IaC)</i>	<i>10</i>
2.3	<i>Document Lifecycle and Intended Audience.....</i>	<i>10</i>
3	Azure Naming Standard Detailed Design	11
3.1	<i>SSC Global Naming Rules</i>	<i>11</i>
3.1.1	Optional Implementation Strategy	13
3.1.2	GC Governance Fields	13
3.1.3	SSC Device Type Field - Service, Asset, and Configuration Management (SACM)	14
3.1.4	Suffix – SSC Resource Type Table	16
3.1.5	Workload Migration Considerations (WLM)	17
3.1.6	Exception Management	17
3.2	<i>Resource Object Naming Tables.....</i>	<i>18</i>
3.2.1	General Object Rules	18
3.2.2	Compute Object Rules.....	20
3.2.3	Storage Object Rules	21
3.2.4	Networking Objects Rules	23
3.2.5	Azure Automation Accounts and Service Principals.....	28
3.2.6	Container Object Rules (placeholder)	29
4	Azure Tagging Standard Detailed Design	30
4.1	<i>Tagging Strategy</i>	<i>30</i>
4.2	<i>Tagging Convention Limitations.....</i>	<i>32</i>
4.2.1	Azure Tagging Limitations	33
4.3	<i>Global Object Tagging.....</i>	<i>33</i>
4.3.1	Global Tagging Rules	33
4.3.2	Tag Reservations by Scope	34
4.3.3	Governance Tags	34
4.3.4	Resource Tags (User Defined Tags)	39
5	Appendix A.....	42

LIST OF TABLES

Table 1: GC Naming Structure	7
Table 2: Generic SACM Device Types	14
Table 3: SACM Functional Device Types	15
Table 4: Suffix Lookup Table	16
Table 5: General Object Rules	19
Table 6: Compute Object Rules	21
Table 7: Storage Object Rules	23
Table 8: Network Object Rules	28
Table 9: Automation Accounts and SP Object Rules	28
Table 10: Container Object Rules	29
Table 11: Tagging Limitations	32
Table 12: Tagging Scope	34
Table 13: Governance Tags	38
Table 14: User Defined Tables	41
Table 15: Department Codes	43

LIST OF FIGURES

No table of figures entries found.

1 OVERVIEW

A consistent Cloud Resource Naming and Tagging Convention is required to help manage cost, improved resource tracking, standardize automation and reduce ambiguity in cloud resourcing. It is also required to implement a multi-Tenant GC wide Cloud initiatives. The primary objective of having a single, standardized, naming/tagging convention is to enable reliability on the name/tag to reflect the state of resources and simplify management, tracking, reporting and automation. The key outcomes of this standards are as follows:

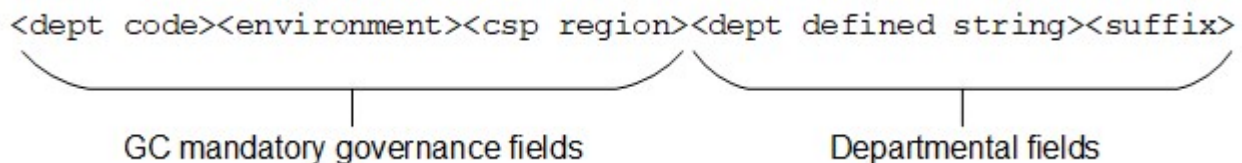
- **Application of Governance Frameworks:** The implementation and maintenance of the cloud governance framework is dependent on consistent naming standards to enable enforcement and compliance auditing.
- **Resource Management:** Quickly find resources associated with specific workloads, environments, ownership groups, or other important information. Resource identification is critical to assigning organizational roles and access permissions for resource management.
- **Security:** Naming and tagging plays a key role in security monitoring and incident management. A naming and tagging standard is required for ITSG-33 compliance and the implementation of GC guardrails.
- **Automation:** In addition to making resources easier for IT to manage, a proper organizational naming scheme enables automation as part of resource creation, operational monitoring, and the creation of DevOps processes.
- **Accounting:** Business owners need to be aware of cloud resource consumption to support chargeback/showback accounting, cloud resources need to be organized to reflect ownership and usage.

Defining a departmental Cloud Naming and Tagging Standard early on in the adoption process is critical to successful Cloud Governance. The convention must be well documented and agreed on by stakeholders to ensure consistent adoption and support enforcement through policy and automation. The standard will follow a periodic review cycle to ensure new requirements and inefficiencies are addressed, and be flexible enough to allow business owners to meet requirements without breaking mandatory field strings and tags. A single standard should be applied at the departmental level across all Cloud Service Providers.

1.1 GC CLOUD GOVERNANCE FOR OBJECT NAMING

A base structure is defined for cloud resource naming that allows for governance while supporting flexibility to meet departmental business and technical requirements. The following proposal has been developed in discussion with TBS, SSC, CSE, and Vendors:

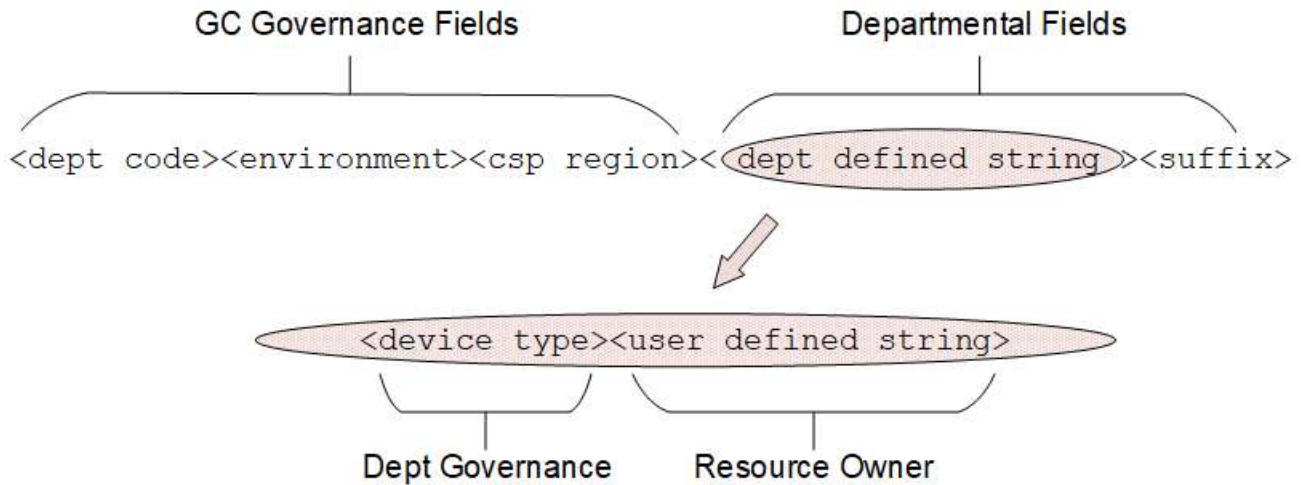
Cloud resource names use the following five fields:



Field	Description																												
<dept code>	Two character code that uniquely identifies departments across the GC. This will assist with GC wide initiatives such as the Canadian Centre for Cyber Security (CCCS) and the Secure Cloud Enablement and Defense (SCED) projects. System event logs do not contain cloud resource tags, including a departmental identifier in the resource name is considered a GC best practice.																												
<environment>	Single character code that identifies environment (P=Production, D=Development, S=Sandbox, Q=Quality Assurance, etc.) Should be defined at the Department level and consistent throughout all CSP deployments.																												
<csp region>	<p>Single character code that identifies Cloud Service Provider and Region. The following assignments are used by SSC and proposed for GC Adoption</p> <table border="1"> <thead> <tr> <th>Cloud Regional Code</th><th>CSP – Region</th></tr> </thead> <tbody> <tr> <td>C or (c)</td><td>Azure – Canada Central</td></tr> <tr> <td>D or (d)</td><td>Azure – Canada East</td></tr> <tr> <td>G or (g)</td><td>AWS – Ca-central-1</td></tr> <tr> <td>H or (h)</td><td>AWS – (reserved for future 2cd region)</td></tr> <tr> <td>I or (i)</td><td>Reserved for International CSP and Regions</td></tr> <tr> <td>K or (k)</td><td>GCP- Northamerica-northeast1</td></tr> <tr> <td>L or (l)</td><td>GCP- (reserved for future 2cd region)</td></tr> <tr> <td>M or (m)</td><td>Reserved for multi-region</td></tr> <tr> <td>O or (o)</td><td>IBM – Canada Montreal</td></tr> <tr> <td>P or (p)</td><td>IBM – Canada Toronto</td></tr> <tr> <td>...</td><td>Extensible</td></tr> <tr> <td>Y or (y)</td><td>Azure Stack SSC East (reserved for EDC Montreal)</td></tr> <tr> <td>Z or (z)</td><td>Azure Stack SSC Central (EDC Barrie)</td></tr> </tbody> </table>	Cloud Regional Code	CSP – Region	C or (c)	Azure – Canada Central	D or (d)	Azure – Canada East	G or (g)	AWS – Ca-central-1	H or (h)	AWS – (reserved for future 2cd region)	I or (i)	Reserved for International CSP and Regions	K or (k)	GCP- Northamerica-northeast1	L or (l)	GCP- (reserved for future 2cd region)	M or (m)	Reserved for multi-region	O or (o)	IBM – Canada Montreal	P or (p)	IBM – Canada Toronto	...	Extensible	Y or (y)	Azure Stack SSC East (reserved for EDC Montreal)	Z or (z)	Azure Stack SSC Central (EDC Barrie)
Cloud Regional Code	CSP – Region																												
C or (c)	Azure – Canada Central																												
D or (d)	Azure – Canada East																												
G or (g)	AWS – Ca-central-1																												
H or (h)	AWS – (reserved for future 2cd region)																												
I or (i)	Reserved for International CSP and Regions																												
K or (k)	GCP- Northamerica-northeast1																												
L or (l)	GCP- (reserved for future 2cd region)																												
M or (m)	Reserved for multi-region																												
O or (o)	IBM – Canada Montreal																												
P or (p)	IBM – Canada Toronto																												
...	Extensible																												
Y or (y)	Azure Stack SSC East (reserved for EDC Montreal)																												
Z or (z)	Azure Stack SSC Central (EDC Barrie)																												
<dept defined string>	Departmental standard is developed by Cloud Team, should align with existing dept. naming convention, DNS and asset management/CMDB datasets.																												
<suffix>	Placeholder for mandatory Departmental governance field. Can be implementation specific at the tenant, subscription or resource group level (SSC recommends using the cloud resource type as a tenant wide suffix based on Microsoft best practice documentation https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/ready/azure-best-practices/naming-and-tagging).																												

Table 1: GC Naming Structure

The `<dept defined string>` is further broken down by each department's cloud governance team. Shared Services has implemented a `<device type>` mandatory field as part of the departmental standard/governance that aligns with the current end-state data center (EDC) naming convention. This ensures alignment with the current asset management and trouble ticketing systems (CMDB). This document defines the SSC implementation of this standard in detail.



1.1.1 FIELD DELIMITATION

The Cloud naming convention uses specific field patterns and definitions. A strategy is applied at the department level to address field delimitation and string identification. The ability to extract specific fields from the resource name is critical to monitoring, reporting, automation and DevOps.

Optional strategies for field delimitation include:

- **Reserved Characters:** Using the lowest common denominator approach across CSPs, the hyphen "-" is reserved for field delimitation. The hyphen is used to separate mandatory fields applied at the department level. Resource owners may also leverage the "-" or use other CSP supported characters for field delimitation within the `<user defined string>` (i.e. underscore "_").
- **Use of Camel Case:** Each field must use uppercase for the first and lowercase for all remaining characters. Numbers are not allowed in the first character. Special characters are typically not supported (**note:** Camel Case is not used in this standard).
- **Fixed Length Fields:** Each field has a fixed length and all resource names must have the correct number of characters for the object type.

A consistent strategy across all cloud service providers at the Departmental level is recommended. All of the above approaches have pros and cons, care should be taken when defining a departmental standard for field delimitation. A mixed approach may be required to meet requirements. The SSC convention leverages a hybrid of the reserved character and fixed length strategies. Case is not used as a field delimiter in the SSC standard, case is difficult to maintain and enforce, and is not applied consistently across Cloud Service Provider tagging implementations. Parsing strings for case also adds unnecessary complexity to automation (code).

2 AZURE NAMING AND TAGGING HIGH LEVEL DESIGN

This document provides implementation specifics of the SSC naming and tagging convention for IaaS/PaaS objects within the Azure portal and Azure Resource Manager (ARM) API. It is *not* the intention of this standard to address naming and tagging requirements for SaaS solutions. The environment will follow an agile approach to development and undergo iterative changes through periodic review. The goal is to provide an adaptable naming standard that includes enough governance structure to meet high level GC requirements but keep it flexible enough that departments can customize the implementation to meet business and technical needs.

2.1 DESIGN PRINCIPLES AND OBJECTIVES

This section defines the high level design principles applied in the Azure naming and tagging convention. This document was developed from the proposed SSC Azure Governance Framework, the existing SSC naming convention used in the Enterprise Data Centers (EDC), and discussions with SSC CIO, TBS, and Vendors.

The standard follows a hierarchical approach that allows delegation of field definition based on resource type and scope (GC → Dept → Service). The objective is to produce an approved convention/implementation for Protected B consumption of Azure for Shared Services Canada. It is also the objective of this convention to share with the GC Cloud Technical Working Group to assist in the development of GC wide and departmental standards.

2.1.1 NAMING AND TAGGING DESIGN PRINCIPLES

The following design principles and considerations are applied.

- Support multi-cloud implementation (Azure, AWS, GCP, and IBM).
- Align with existing departmental naming conventions, Domain Name System (DNS) and Configuration Management Database (CMDB) where possible.
- Allow for flexible adoption, extensibility and exceptions within the standard.
- Enable easy identification of resource type and owner through the Cloud Service Provider (CSP) portal to simplify operational support.
- Support automation through templates and CI/CD pipelines.
- Tagging standard should provide a hierarchical approach to allow for GC wide requirements (CSE, TBS, and SSC) but also support delegation of authority for departments, business lines and application developers to define tags within their scope.
- Support enforcement of names and tags through CSP policy and other automation mechanisms.
- Allowed characters should support a lowest common denominator approach to ensure the standard applies across AWS, Azure, IBM, and GCP where possible (and other).
- Support vendor templates where possible (some fields are mandatory).
- Align with technical limitations (i.e. 15 character restriction on Windows NetBIOS names).
- Address legacy system requirements and workload migration (WLM).
- Define global date format at the departmental scope that aligns with CSP lowest common denominator approach. This implementation uses the GC standard (ISO8601) which is YYYY-MM-DD (note: the “/” is not supported in tags by all CSPs).

Note: Azure has character restrictions on some resource types (most notably storage accounts can only use lowercase alphanumeric characters). This naming standard allows for these exceptions, the rules and conventions for each cloud object type are clearly articulated in the resource object tables in section 3.2 of this document.

2.2 INFRASTRUCTURE AS CODE (IAC)

One of the objectives of this naming standard is to enable the deployment and management of cloud resources through code. Compliance enables the creation of reusable templates and modules that are centrally maintained and shared through SSC managed repositories. SSC has selected Terraform as the language of choice for automating deployments within Azure but this standard can easily be applied to other scripting languages (JSON, ARM templates, PowerShell, etc.).

This standard supports the delegation of ownership of resource names using a hierarchical approach starting at the GC level, then down through the department and subscription levels to the individual cloud resources. The use of global and local named variables enables individual application solutions to pass values to GC wide templates/modules for compliance and string concatenation. Naming modules exist (terraform) that can be leveraged to create compliant strings for Infrastructure as Code (IaC) deployments. Application developers *always* have the ability to assign `<user defined strings>` to meet specific coding requirements within this standard.

This naming standard is currently being used by two IaC Government of Canada initiatives supported by SSC:

- Base Cloud Architecture (BCA): Part of the GC Accelerator initiative that provides Terraform modules that deploy ITSG-22 compliant Azure Virtual Data Centers based on Microsoft's Cloud Adoption Framework (CAF).
[<https://github.com/canada-ca-terraform-modules>] and [<https://github.com/ssc-spc-cloud-nuage/>]
- Cloud Operating Model (COM): Advanced automation and orchestration solution that extend the Base Cloud Architecture to include CI/CD pipelines using yaml and Terragrunt. [*contact SSC CCOE for access to COM*]

Note: Specific coding examples and variable assignments are purposely left out of this document to avoid out-of-date references – refer to the links above or contact ssc.azurelzbca-azureaizdb.spc@canada.ca for more information.

2.3 DOCUMENT LIFECYCLE AND INTENDED AUDIENCE

This naming standard is maintained by SSC Cloud Research and Development and will undergo yearly review. Periodic updates are applied as required and tracked in the naming and tagging quick reference guide in the SSC code repositories.

The intended audience of this document is GC cloud governance, technical, and operational teams working in Azure.

3 AZURE NAMING STANDARD DETAILED DESIGN

This section provides the detailed naming convention for Azure adopted by Shared Service Canada. The intent is to extend this standard to other GC departments once alignment between stakeholders is achieved. It is important to note that tagging rules are separate and included in Section 4. There are subtle differences between the rules for naming vs. tagging (i.e. allowed case, field extensibility, string length, etc.).

Note: This convention uses the SSC, TBS and Microsoft recommendation for the first four characters to enforce GC wide unique names to assist with system identification at the Canadian Centre for Cyber Security (CCCS) and other SSC/TBS governance initiatives.

3.1 SSC GLOBAL NAMING RULES

The following object naming rules **must** be respected by all cloud deployments that use this convention. The examples in this section illustrate how to **build** a compliant resource name. The resource type object tables in section 3.2 provide the detailed format, restrictions and examples for most IaaS resource types, these will be extended as the Azure footprint expands to include more resource types.

- Infrastructure devices that require DNS entries (A records) **should** not exceed 15 characters. CNAME records are used to implement user friendly names as required. Windows virtual machine names **must** be a maximum of 15 characters to support synchronization with the NetBIOS host names.
- Allowed characters: Camel Case (mix upper and lower), alphanumeric, hyphen "-", and underscore "_".
- Reserved character: Hyphen "-" is reserved for field delimitation. Mandatory fields in an object name **must** be separated by a hyphen "-" (unless otherwise stated in the object definition). The hyphen "-" **may** be used as an **optional subfield delimiter** in the <userDefined-string>.
- Restricted characters: Spaces and other special characters are **not** allowed unless explicitly stated in object tables below (CSP restrictions apply).
- Camel Case: This standard does not use case for field delimitation, it is recommended that case follow the examples in the object tables for consistency but this is **not** mandatory for compliance. Case is determined by the resource owner and used to improve readability of object names. This standard encourages mixed upper and lower case of the mandatory prefix as illustrated in the next bullet.
- Prefix: All object names use mandatory prefix fields that align with GC Cloud Governance best practice. The compliance fields are fixed length (4 characters) and do **not** use a delimiter.

<dept code><environment><csp region> : <2 char><1><1>

Example: ScPc (SSC, Production, Azure Central)

- After the GC four character prefix, the <dept defined string> is further defined to align with departmental requirements such as existing conventions, DNS, and asset tracking/CMDB implementations. This document was developed for SSC, other departments using this standard may choose to apply all, some, or none of the departmental defined fields. Maintaining compliance with the GC wide governance framework outlined in section [1.1 GC Cloud Governance for Object Naming] is mandatory for departments using this convention.

- **Device Type:** The device type field has been implemented as part of the SSC departmental convention, it is fixed length (3 characters) and follows the 4 character GC mandatory governance prefix. A delimiter is **not** used. The device types align with the SSC End-State naming convention (CMDB) and are included in the *Azure Naming and Tagging Quick Reference v.21.docx* guide for easy reference. This field is extensible, new values can be added if they are not used in the current naming convention. The authority for the three character device type is the SSC Service, Asset, and Configuration Management (SACM) team. Version 3.6 of the SACM cheat sheet is included below. New device types should periodically be synchronized with the SACM team so they can be reserved and tracked.

`<dept code><environment><csp region><device type>`

Examples: ScPcSWA (Server, Windows, Domain Controller)
ScPcADC (Application Delivery Controller / Load Balancer)

- **User Defined String:** The mandatory field delimiter (" - ") proceeds the user defined string to support code parsing and improve readability. The string is variable length and **may** use the hyphen ("-") or underscore("_") as an *optional* subfield delimiter. Cloud objects **must** have at least one user defined string unless explicitly stated otherwise in the object tables. Subscription, resource group, and resource owners have the flexibility to implement their own standards within their scope, mandatory fields within the user defined string **should** be tracked by the subscription owner.

`<dept code><environment><csp region><Device Type>-<userDefined-String>`

Examples: ScPcSWA-MyApp01
ScPcSWA-App-01

- **Suffix:** A single mandatory suffix is defined within this naming standard and is used to identify object type on certain resources (defined in the object tables below). The rule of thumb is: *if an object requires a DNS entry the object type suffix is not required*. Typically an object will use a suffix if it is a child object of an infrastructure device (i.e. VM / VM-nic1). The "-" **must** be used as the field delimiter for object suffixes. Object suffixes align with the Microsoft best practice guide: <https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/ready/azure-best-practices/naming-and-tagging>.

`<dept code><environment><csp region><Device Type>-<UserDefined-string>-<suffix>`

Examples: ScPcSWA-MyApp01 (virtual machine)
ScPcSWA-MyApp01-nic1 (first network interface)
ScPcSWA-MyApp01-pip1 (first public network interface)
ScPcSWA-MyApp01-nsg (network security group)
ScPcSWA-MyApp01-datadisk1 (first data disk)

- **Suffix (cont.):** Resource type suffixes are also applied on some cloud constructs that do not have a parent infrastructure device (note: in the example below the `<device type>` "CNR" is used as a "catch all" for Cloud Network Resource object types – these are tracked in the Device Type table).

Examples: ScPcCSV-MyAppVault1-kv (Key Vault)
ScPcCNR-Core-vnet (vnet)

ScPcCNR-Core-Transit-rt (route table)

- Exceptions: CSP restrictions may require exceptions to the global naming rules, these are identified in the object tables below (i.e. Azure Storage accounts and objects).
- Management Groups, Subscriptions and Resource Groups are considered governance (or container) objects and have different filed restrictions and character exceptions. The <Device Type> field is **not** used in container object names. Instead, a 3 or 4 character <owner> field is defined and tracked at the subscription level. Governance container owners are responsible for naming compliance, security and cost management (the “how” is out of scope of this document and may vary by implementation – subscriptions and resource groups are used to manage these responsibilities).
- Naming Conflicts: As the Cloud footprint grows there will be inevitable naming conflicts. The use of CNAMEs will help with DNS isolation of conflicting A records but unique names will still be required for applications and systems management (i.e. directory services, monitoring, asset management). If possible, extend current processes for assigning device names to the cloud teams. A single business unit should be responsible for naming standards across both ground and cloud infrastructure deployments (this falls under cloud governance and is out of scope of this document).

The structure of the *Object Rule Tables* are department specific. As previously mentioned, this document uses the SSC adoption of this standard to provide examples and **may** be leveraged.

3.1.1 OPTIONAL IMPLEMENTATION STRATEGY

The above implementation strategy for SSC splits the <dept defined string> into two sub-strings <Device Type>-<userDefined-string>, the <Device Type> string (or field) is implemented as a department wide mandatory governance field for the purpose of aligning with SSC’s end-state data center naming standard and asset tracking / trouble ticket systems (CMDB). It is the responsibility of the subscription owner to ensure compliance with departmental standards and preform periodic audits of deployed resources.

Another implementation strategy that could be adopted by a department is to delegate the complete <dept defined string> to the subscription / resource owner, removing departmental governance from the cloud naming convention. The device type suffix **should** be maintained. This would result with:

<dept code><environment><csp region>-<userDefined-String>-<suffix>

Examples: ScPc-MyApp001 (virtual machine)
ScPd-MyApp001-nic1 (first network interface)
ScPd-MyApp001-pip1 (first public network interface)
ScPc-MyApp001-nsg (network security group)
ScPc-MyApp001-datadisk1 (first data disk)

Variants of this naming and tagging strategy are supported within this framework. The departmental cloud governance team should take responsibility for defining and maintaining the Naming and Tagging standard.

3.1.2 GC GOVERNANCE FIELDS

The first four characters of the naming convention follow a GC wide structure to align with the proposed governance framework as described in section 1. These values are repeated here for the SSC implementation:

Two character department code <dept code> – SSC will use <Sc>

Security Classification **Unclassified**

Status: DRFT

Subject: SSC NAMING AND TAGGING STANDARD FOR AZURE

Single character <environment> - additional codes can be added to the Value tables

P=Production, D=Development, Q=Quality Assurance, S=Sandbox

Single character Cloud Service Provider Region <csp region>

c=Azure Central, d=Azure East

3.1.3 SSC DEVICE TYPE FIELD - SERVICE, ASSET, AND CONFIGURATION MANAGEMENT (SACM)

The device type field falls within the <dept defined string> section of the naming convention. Device Type field definitions and values are assigned and maintained by the SSC Service, Asset and Configuration Management team (SACM). These were created in 2013 for use in the end-state data center and configuration management database (CMDB). Alignment will ensure cloud resources are consistent with existing SSC naming standards, simplify operational support, and provide for future integration with SSC's trouble ticketing system.

Generic Device Type	Short Name	New Cloud Generic Device Types	Short Name
Network Sniffer Device	NSD	Generic Cloud Entity / Object	CLD
Network Monitor Probe	NMP	Cloud Network Resource	CNR
Network TAP	TAP	Cloud Storage Account	CSA
Network TAP Aggregator	AGR	Cloud Secret Vault	CSV
Intrusion Detection System	IDS	Cloud Container Registry	CCR
Intrusion Prevention System	IPS	Cloud Platform Service	CPS
Router	RTR	...extensible...	
Access Point	APT		
Server (*see note below table for functional device types)	SRV		
Network Switch	XSW		
Storage SAN Switch	QFC		
Network Time Server	NTP		
Terminal Console Server	TCS		
Storage Disk Array	QSN		
Bastion Host	BST		
Firewall	FWL		
Application Delivery Controller (load-balancers, , etc.)	ADC		
Appliance (Reverse and Forward Proxy & URL filtering, etc.)	AUP		
Server based virtual desktop (i.e. Citrix or VMWare View)	VDI		
Alarm	ALM		
Intelligent Power Bar	IPB		
Camera	CAM		
Unified Communication System	UCS		
Power Distribution Unit	PDU		
Rack	RCK		
Chassis	CHS		
Wireless Management System	WMS		
Wireless Mobility Controller	WMC		
Wireless Captive Portal	WCP		

Table 2: Generic SACM Device Types

Within the existing end-state SSC naming standard is the concept of functional device types. Essentially this allows for resource owners to optionally include additional detail within the three-character device type string. For more detail on how this is implemented refer to the SSC end-state data center naming convention, a summary is provided here:

Functional Device Types: The *functional* device type is implemented as an optional extension to the SSC end-state naming convention to allow for additional information about the resource to be included in the name. The <generic device type> is split into 3 separate fields that do **not** use a delimiter. Resource owners can implement functional device types to remain compliant with legacy device naming standards or choose to use the generic types (i.e. SRV for server). Tracking functional device types is managed by the SACM group within SSC, the character “C” has been reserved for cloud native resources (approval pending).

Example - Server Naming: Assuming that the first character of a hostname is ‘S’ for server (reserved) in the SACM standard, the second and third characters of the functional device type portion of the hostname should be specified using the tables below.

Second Character	
W	Windows OS
L	RedHat Enterprise Linux or SUSE Linux Enterprise Server
A	Appliance based server (not OS specific)
X	Hypervisor
Third Character	Function
A	Domain Services (Active Directory, DNS, DHCP, etc.)
B	Data bases (Oracle, MS-SQL, etc.)
C	Web Servers [http/https only] (Apache, IIS, etc.)
D	Application Servers (Web Trends, PeopleSoft, SAP, Web Application Servers [WebLogic, WebSphere, Tomcat], etc.)
E	Management Servers (venter, SCOM, SCCM, Puppet, etc.)
F	File and Print Servers
G	Clusters
H	Messaging
V	Microsoft Hyper-V
X	VMWare ESX
J	Jump Server
...	Extensible

Table 3: SACM Functional Device Types

3.1.4 SUFFIX – SSC RESOURCE TYPE TABLE

The SSC implementation of this standard defines a mandatory cloud resource type suffix on **some** objects. These are defined in detailed in the object tables in section 3.2 of this document. The following suffix table provides sample values for approved mandatory suffix fields, the list will be extended as more Azure resource types are implemented. New suffix short codes will follow Microsoft best practice guide by default, exceptions are allowed.

<https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/ready/azure-best-practices/naming-and-tagging>

Object Type (Resource/Service):	Short Code:	Object Type (Resource/Service):	Short Code:
Subscription	"CBS billing #"	Application Gateway	agw
Resource Group	rg	App Service	asv
Virtual Machine	vm	Key Vault	kv
Availability Set	as	App Service Plan	asp
Virtual Network	vnet	Sql Database	sdb
Subnet	snet	Sql Server	sql
Public IP Address	pip#	Disk	dsk
Firewall	fw	Azure Bastion end-point	bstn
Network Security Group	nsg	Log Analytics Workspace	law
Storage Account	stg	Function	fnc
Traffic Manager	tm	Logic App	log
Load Balancer	lb	Network Interface	nic#
Azure Load Balancer	lbi / lbe	Connection	con
Load Balancer Rule	lbr	Route Table	rt
Load Balancer Health Probe	lbhp	Route	route
Load Balancer Backend Pool	lbbp	VM OS Disk	osdisk#
Virtual Network Gateway	vgw	VM Data Disk	datadisk#
Local Network Gateway	lgw	Management Group	mg
Vnet peering	gwp	Azure Firewall	azfw
Network Watcher	nw	App Service Plan	svcplan
Azure Purview	pview	Application Service	svc
Application Insights	appi	Search Service	ss
Web Application	wapp	SQL Assessment	sgla
Databricks Service	dbw	Synapse Workspace	syn
Azure Data Factory	adf	Express Route circuit	erc

Table 4: Suffix Lookup Table

3.1.5 *WORKLOAD MIGRATION CONSIDERATIONS (WLM)*

Moving existing infrastructure (virtual) to the cloud introduces additional challenges and complexity. The naming and tagging standard allows “as is” lift and shift of systems provided a mandatory tag is attached containing the compliant cloud object name. SSC Cloud team is investigating options and will update this strategy as requirements are identified in future releases of this document. Departments should develop a strategy to manage legacy names in the cloud.

The actual host name used by the Operating System/DNS does not have to match the resource name in the cloud fabric/portal for the virtual machine. An option is to correctly name the cloud resource but leave the legacy name for the operating system instance. Tags are used to map names between the cloud and legacy naming standards. These may also be used to bridge names between systems using automation.

WLM tag *key:value* pair examples:

```
<azname>:<dept code><environment><csp region><Device Type>-<UserDefined-string>-<suffix>  
<legacyname>:<pre-migration name>
```

3.1.6 *EXCEPTION MANAGEMENT*

Microsoft Marketplace, GitHub quick starts, and third party vendor templates may include hardcoded device naming parameters and reserved script delimiters that may not work within this standard. In particular, VM disk naming, NIC numbering and object type suffix fields typically do not align when deploying resources from the Azure Marketplace through the portal. These are allowed as exceptions within the convention but **should** be tracked. Alignment with the first four GC governance characters as the prefix however is absolutely required, templates that do not support this are **not** allowed. Automation **may** be used to periodically remove/suspend cloud resources that violate this mandatory naming standard within the Azure Virtual Data Center for PBMM approval.

3.2 RESOURCE OBJECT NAMING TABLES

The object naming tables provide the naming structure, <> brackets are used to denote mandatory fields, [] denote optional fields, use of delimiters are described above. Object specific rules and examples are included, subscription and resource group owners can extend the standard to meet specific requirements within their scope. Mandatory governance fields cannot be modified.

Important: A *child object* will always use the parent device name with the suffix replaced to reflect object type. A child object is defined as a resource that exists only as a sub component of its parent. Examples are: <vm>/<vm>-nic1/<vm>-osdisk1 or <vnet>/<vnet>-snet.

3.2.1 GENERAL OBJECT RULES

General Object Rules			
Object Type	Scope	Character Set	Naming Pattern / Example
Management Group	Tenant	Length: 1-90 Casing: Case insensitive (Camel Case) Valid Characters: Alphanumeric, underscore, hyphen, spaces Spaces are allowed in subscription name to assist in readability	<dept code><environment><CSP Region>-<owner>-<userDefined-string> Management Groups do not require object type suffix. Azure AD objects are not included in this version of the convention. Cloud team should standardize on an Azure AD object naming strategy. Note: Departmental Governance Team should define Management Group and Subscription structure prior to releasing the naming convention. Example: ScPc-CTO-VDC
Subscription	Tenant	Length: 1-90 Casing: Case insensitive (Camel Case) Valid Characters: Alphanumeric, underscore, hyphen, spaces Spaces are allowed in subscription name to assist in readability	<dept code><environment><CSP Region>-<owner>-userDefined-string> Subscriptions do not require object type suffix Notes: Departmental Governance Team should define Management Group and Subscription structure prior to releasing the naming convention. If <CSP region> span Azure regions use the multi-region code "m". Spaces are allowed in subscription names, at least one <userDefined-string> is mandatory in the subscription name (subfields are optional). Subscriptions are the primary containers for managing application life cycle, cost recovery, role-based access control, and development pipelines. Examples: ScPc-CTO-VDC-CORE ScDc-CTO-VDC-PROD

Security Classification **Unclassified**

Status: DRFT

Subject: SSC NAMING AND TAGGING STANDARD FOR AZURE

General Object Rules			
Object Type	Scope	Character Set	Naming Pattern / Example
Resource Group	Subscription	Length: 1-90 Casing: Case insensitive (Camel Case) Valid Characters: Alphanumeric, underscore, and hyphen	<p><dept code><environment><CSP Region>-<owner>-<userDefined-string>-rg</p> <p>Notes: Resource Group naming standard should be defined by the departmental cloud governance team and applied tenant wide. Resource Group tagging is also key to effective cloud governance.</p> <p>Resource Groups can't be renamed, changing or extending the convention after deployment may result in complete rebuilds of existing infrastructure.</p> <p>At least one <userDefined-string> is mandatory in the resource group name, subfields are optional and defined by the subscription owner or delegated to the resource group owner.</p> <p>Examples: ScPc-CTO-VDC-CORE-rg ScDc-CTO-ICAS--rg</p>

Table 5: General Object Rules

3.2.2 COMPUTE OBJECT RULES

Compute			
Object Type	Scope	Description	Naming Pattern / Example
Virtual Machine	Resource Group	Length: 1-15 (Windows), 1-64 (Linux) Casing: Case insensitive Valid Characters: Alphanumeric and hyphen (underscore not supported in Azure)	<dept code><environment><CSP Region><device type>-<UserDefined-stringUserDefined-string> Note: Suffix is not used on Virtual Machines. Example: ScPcSWA-MyApp01
Managed Disk Virtual Machine – OS Disk	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<vm name>-osdisk<##> Notes: This is a child object of the VM, use VM name and apply the resource type suffix. The VM name includes the mandatory governance fields. Example: ScPcSWA-MyApp01-osdisk1
Managed Disk Virtual Machine – Data Disk	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<vm name>-datadisk<##> Notes: This is a child object of the VM, use VM name and apply the resource type suffix. The VM name includes the mandatory governance fields. Example: ScPcSWA-My_Ap01-datadisk1
Availability Set	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, underscore, and hyphen	<VM name>-<userDefined-stringUserDefined-stringUserDefined-string>-as Note: An availability set is considered a child object of its virtual machine (instance number is dropped on clusters) Suffix applies

Compute			
Object Type	Scope	Description	Naming Pattern / Example
			Example: ScDcFWL-FGPROD-as
Function App	Global	Length: 1-60 Casing: Case insensitive Valid Characters: Alphanumeric and hyphen	<code><dept code><environment><CSP Region><device type>-<userDefined-stringUserDefined-string>-fn</code> Note: In the example below the default cloud device type is used since there is no entry in the Field Value tables for Function Apps – the subscription owner should update the table. Example: ScSdCLD-AvSet001-fn
Key Vault	Global	Length: 3-24 Casing: Case insensitive Valid Characters: Alphanumeric and hyphen	<code><dept code><environment><CSP Region><device type>-<userDefined-stringUserDefined-stringUserDefined-string>-kv</code> Example: ScSdCSV-MyVault01-kv

Table 6: Compute Object Rules

3.2.3 STORAGE OBJECT RULES

Note: Storage object names require globally unique lowercase alphanumeric strings, at this time the only mandatory string is the GC Governance field prefix. In this release, most storage objects follow the `(tolower)<dept code><environment><CSP Region><userdefinedstring><random>` standard discussed in section 2.2 of this document.

Storage			
Object Type	Scope	Description	Naming Pattern / Example
Storage account (data)	Global	Length: 3-24 Casing: Lowercase Valid Characters: Alphanumeric	<code><dept code><environment><CSP Region><userdefinedstring></code> Note: must be globally unique, can use code to calculate a unique string for naming storage accounts, number is optional. "-" not supported as a field delimiter. See

Storage			
Object Type	Scope	Description	Naming Pattern / Example
			<p>section 2.2 for automation examples</p> <p>Example : scpcmynuniqueguid</p>
Storage account (disks)	Global	<p>Length: 3-24</p> <p>Casing: Lowercase</p> <p>Valid Characters: Alphanumeric</p>	<p><dept code><environment><CSP Region><userdefinedstring></p> <p>Note: must be globally unique, can use a function to calculate a unique guid for naming storage accounts, number is optional. "-" not supported as a field delimiter.</p> <p>Example : scpcmynuniqueguidfromfunction</p>
Container name	Storage account	<p>Length: 3-63</p> <p>Casing: Lowercase</p> <p>Valid Characters: Alphanumeric and hyphen</p>	<p><storage account name>-<userdefinedstring></p> <p>Notes: This is a child object of the Storage Account, use Storage Account name followed by – and a userdefinedstring. Fabric/system generated strings are allowed</p> <p>Example: scpcmynuniqueguid-mycontainername</p>
Queue name	Storage account	<p>Length: 3-63</p> <p>Casing: Lowercase</p> <p>Valid Characters: Alphanumeric and hyphen</p>	<p><storage account name>-<userdefinedstring></p> <p>Notes: This is a child object of the Storage Account, use Storage Account name followed by – and a userdefinedstring. Fabric/system generated strings are allowed</p> <p>Example: scpcmynuniqueguid-myqueue</p>

Storage			
Object Type	Scope	Description	Naming Pattern / Example
Table name	Storage account	Length: 3-63 Casing: Case insensitive Valid Characters: Alphanumeric	<p><storage account name>-<userdefinedstring></p> <p>Notes: This is a child object of the Storage Account, use Storage Account name followed by – and a userdefinedstring. Fabric/system generated strings are allowed</p> <p>Example: scpcmynuniqueguidmytable</p>
File name	Storage account	Length: 3-63 Casing: Lowercase Valid Characters: Alphanumeric and hyphen	<p><storage account name>-<userdefinedstring></p> <p>Notes: This is a child object of the Storage Account, use Storage Account name followed by – and a userdefinedstring. Fabric/system generated strings are allowed</p> <p>Example: scpcmynuniqueguid-myshare</p>
Data Lake Store	Global	Length: 3-24 Casing: Lowercase Valid Characters: Alphanumeric	<p><dept code><environment><CSP Region><userdefinedstring></p> <p>Note: must be globally unique, can use a function to calculate a unique guid for naming storage accounts, number is optional. “-” not supported as a field delimiter.</p> <p>Example : scpcmynuniqueguid</p>

Table 7: Storage Object Rules

3.2.4 NETWORKING OBJECTS RULES

Typically, networking object names do not require DNS entries, resource suffixes are mandatory. Current SSC standard uses a “catch all” <device type> of CNR (Cloud Network Resource) to identify network infrastructure components and align with SACM functional device type reservation (“C”). Network Interface Cards (NIC), Public IPs, etc. use their parent object name to track ownership (VM / VM-nic1 or

VNET / VNET-snet). The <userDefined-string> is leveraged to identify resource specifics. When using the parent object name as part of the child object name the suffix is truncated (see examples).

Networking			
Object Type	Scope	Description	Naming Pattern / Example
Virtual Network (VNet)	Resource Group	Length: 2-64 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<dept code><environment><CSP Region><device type>-<userDefined-string>-vnet Note: IP addresses are not allowed in object names, zone acronym aligns with ITSG-22 zoning convention. Example: ScPcCNR-VDC-MRZ-vnet
Subnet	Virtual Network	Length: 2-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<Vnet name>-<userDefined-string >-snet Note: Do not include the IP address or mask in the subnet names, it is easily queried and appears in the portal anyway. Parent vnet name is used with vnet suffix truncated. Note: “GatewaySubnet” and “AzureFirewallSubnet” are reserved name in Azure SDN fabric Examples: ScPcCNR-VDC-MRZ-Logging-snet ScPcCNR-VDC-MRZ-Security-snet ScPcCNR-VDC-MRZ-AccessZone-snet
Route Table	Resource Group	Length: 2-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<VNET name>-rt or <SNET name>-rt depending on scope of the route table. Note: A single route table can be assigned to multiple subnets and is used to override <i>some</i> default system routes. Examples: (vnet scope): ScPcCNR-VDC-Core-rt (snet scope): ScPcCNR-VDC-Core_Mgmt-rt
Route	Resource Group	Length: 2-80 Casing: Case insensitive Valid Characters:	<UserDefined-string>-route Note: Routes can be used in multiple route tables. The naming convention is defined by the cloud network team or resource

Security Classification **Unclassified**

Status: DRFT

Subject: SSC NAMING AND TAGGING STANDARD FOR AZURE

Networking			
Object Type	Scope	Description	Naming Pattern / Example
		Alphanumeric, hyphen and underscore	owner of the route. The SSC BCA implementation uses the following syntax: to<device>_<source>_<destination>-route Examples: toCoreLB_Transit_Internet-route toCoreFW_Transit_”spoke”-route toCoreFW_Mgmt_Core-route
Azure Firewall End-point	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<dept code><environment><CSP Region><device type>-<userDefined-string>-afw Note: Suffix applies Example: ScPcSFWL-Core-Internet-azfw
Azure Bastion End-point	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<dept code><environment><CSP Region><device type>-<userDefined-string>-bstn Note: Suffix applies Example: ScPcSBST-Core-Internet-bstn
Virtual Network Gateway	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<vnet name>-[<userDefined-string>-]vng Note: User defined string is optional Example: ScPcCNR-Core-MRZ-vng
Local Network Gateway	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<vnet name>-[<userDefined-string>-]lng Note: User defined string is optional Example: ScPcCNR-Core-MRZ-lng
Connection	Virtual Network	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<vnet name>-[<userDefined-string>-]con Note: User defined string is optional Example: ScPcCNR-Core-MRZ-con

Security Classification **Unclassified**

Status: DRFT

Subject: SSC NAMING AND TAGGING STANDARD FOR AZURE

Networking			
Object Type	Scope	Description	Naming Pattern / Example
Network Interface	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<p><parent object name>-[<userDefined-string>-]nic<#></p> <p>Note: The network interface is an exception to the suffix rule (parent object is typically a VM but may also be a load balancer, firewall, etc.). User defined field is not required, instance number is assigned to the suffix</p> <p>Example: ScPcSWA-MyApp01-nic1</p>
Network Security Group	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<p><parent object name>-[<userDefined-string>-]nsg</p> <p>Note: Typically <userDefined-string> is not assigned if the parent object is a single VM or subnet.</p> <p>Examples: ScPcSWA-MyApp0001-nsg (VM) ScPcCNR-Core-MRZ-Logging-nsg (snet)</p>
Network Security Group Rule	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<p><userDefined-string></p> <p>Note: The naming convention is defined by the cloud security team or resource owner. The SSC BCA implementation uses the following syntax:</p> <p><action>-<description></p> <p>Examples: Allow-RDP Allow-22 Deny-InboundMRZ</p>
Public IP Address	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<p><parent object name>-[<userDefined-string>-]pip<#></p> <p>Note: The public interface is an exception to the suffix rule (parent object is typically a VM but may also be a load balancer, firewall, etc.). User defined field is not required, instance number is assigned to the suffix. Public IPs are attached to Network Interfaces (option to use nic# in the [<userDefined-string>-]-pip#</p> <p>Example: ScPcADC-LB01-pip1</p>

Security Classification **Unclassified**

Status: DRFT

Subject: SSC NAMING AND TAGGING STANDARD FOR AZURE

Networking			
Object Type	Scope	Description	Naming Pattern / Example
Vnet Peering	Vnet	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<source vnet>-<dest vnet>-gwp Note: parent object type truncated, user defined field not allowed. Example: ScDcCNR-VDC-Dev-ScDcCNR-VDC-Prod-gwp
Load Balancer	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<dept code><environment><CSP Region><device type>-<userDefined-string>-lb Note: Load balancers typically serve a parent object. In this case use the parent object name (truncate instance ## for cluster) and add suffix Example: ScPcADC-F5PROD-lb
Load Balancer Front End Interface	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<load balancer>-lbr# Note: Parent object name suffix truncated. User defined field not supported Example: ScPcADC-F5PROD-lbr3
Load Balanced Rules	Load Balancer	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<load balancer>-lbbp Note: Parent object name suffix truncated. User defined field not supported Example: ScPcADC-F5PROD-lbbp
Load Balancer Backend Pool	Load Balancer	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<load balancer>-lbhp Note: Parent object name suffix truncated. User defined field not supported Example: ScPcADC-F5PROD-lbhp

Networking			
Object Type	Scope	Description	Naming Pattern / Example
Load Balancer Health Probe	Load Balancer	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<load balancer>-<userDefined-string>-lbhp Note: <userDefined-string> required, Parent object name as suffix truncated. Example: ScPcADC-AZLB01-RDSFarm-RDP-TCP3389-lbhp
Azure Application Gateway	Resource Group	Length: 1-80 Casing: Case insensitive Valid Characters: Alphanumeric, hyphen and underscore	<dept code><environment><CSP Region><device type>-<userDefined-string>-agw
Traffic Manager Profile	Resource Group	Length: 1-63 Casing: Case insensitive Valid Characters: Alphanumeric and hyphen	<dept code><environment><CSP Region>-<userDefined-string>-tm

Table 8: Network Object Rules

3.2.5 AZURE AUTOMATION ACCOUNTS AND SERVICE PRINCIPALS

Automation Accounts and Service Principals			
Object Type	Scope	Description	Naming Pattern / Example
Azure Run As Account Azure Service Principle	Subscription	Length: 6-50 Casing: Case insensitive (Camel Case) Valid Characters: Alphanumeric and hyphen	<dept code><environment><CSP Region>-<owner>-<userDefined-string>-spn Note: Standard for automation accounts and Service Principals is flexible and should be based on Azure Active Directory naming standard – User Defined String is mandatory. Spaces are not allowed (AAD objects are out of scope of this document). Example: ScPd-AutomationAccount-spn

Table 9: Automation Accounts and SP Object Rules

3.2.6 CONTAINER OBJECT RULES (PLACEHOLDER)

Containers			
Object Type	Scope	Description	Naming Pattern / Example
Container Registry	Global	Length: 5-50 Casing: Case insensitive Valid Characters: Alphanumeric	<dept code><environment><CSP Region><device type>-<userDefined-string>-registry (under review)

Table 10: Container Object Rules

Excerpt: "Blob and container names are passed to the Blob service within a URL. Certain characters must be percent-encoded to appear in a URL, using UTF-8 (preferred) or MBCS. This encoding occurs automatically when you use the Azure Storage client libraries. However, there are certain characters that are not valid in URL paths even when encoded. These characters cannot appear in blob or container names. Code points like \uE000, while valid in NTFS filenames, are not valid Unicode characters, so they cannot be used. In addition, some ASCII or Unicode characters, like control characters (0x00 to 0x1F, \u0081, etc.), are also not allowed.

For rules governing Unicode strings in HTTP/1.1 see:"

- [**RFC 2616, Section 2.2: Basic Rules and RFC 3987**](#)

4 AZURE TAGGING STANDARD DETAILED DESIGN

Tags are metadata labels that are assigned to cloud resources. Each tag consists of a `key:value(s)` pair in which the key represents a property (i.e. date of birth) and the value represents the associated data (i.e. 1985-04-21). To the Cloud Service Provider, tags are simply strings of characters with no semantic meaning, they are only relevant to the cloud consumer. An effective tagging strategy is built around the organization's business and technical requirements. Tags can be easily changed which supports an agile, trial and error approach to implementation.

The extension of the tagging limit within Azure from 15 to 50 tags allows for increased flexibility in how the GC assigns tags (some resources have different restrictions). The SSC tagging convention follows a hierarchical approach, each level has the authority to reserve and assign tag `key:value` pairs appropriate to their scope and to set mandatory tagging rules on child objects (i.e. a subscription owner may assign mandatory tags on resource groups, resource group owner may set mandatory tags on resources, etc.). Tag enforcement is applied based on scope and is a shared responsibility. Auditing and automation is applied through policy/code. A complete list of tag support and restrictions for Azure resources can be found at this link:

<https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/tag-support>

4.1 TAGGING STRATEGY

The departmental cloud governance team takes responsibility for defining the tagging strategy. The tagging policies must be documented and accessible to all teams working in the cloud. Tags are used for a wide variety of purposes from implementing cost allocation and reporting, to supporting authorization and automation. Implementation can be challenging due to the number of stakeholders and business lines involved in consuming the standard. The key is to start with a small number of mandatory departmental tags and delegate authority to resource owners to enable them to design their own solutions to meet their unique requirements.

This release focuses on "governance tags" (defined below). Resource owner tags (or `<user defined>` to stay consistent) are not managed or enforced by the departmental cloud governance team. Instead guidance is provided through design principles and guidelines to ensure compliance but allow for flexibility and extensibility. This convention identifies five reserved GC wide tags for use by TBS, CSE, and SSC. Common tagging strategies include:

- Tag for resource organization: When working with the Azure portal or ARM resource provider tags provide a great way to query and organize resources. They can be used to customize console views, preform resource searches, or enumerate objects for reporting.
- Tag for cost management: Creating effective cost management requires the ability to easily identify resource ownership and usage for integration into billing systems.
- Tag for automation: Resource-type and service-specific tags are used to filter resources during infrastructure deployment and modification. They can be used as opt-in or opt-out flags during automated processes in code pipelines to differentiate between dev, test, prod environments. Resource tagging is heavily leveraged in the DevOps lifecycle.
- Tag for security and access control: Implement tags to support compliance auditing, identification of sensitive data, and access controls. Tags can be used to identify cloud resources such as flow logs, network security groups or edge appliances that require additional scrutiny.

Tags are assigned at the scope level (subscription, resource group, resource) and may or may not be replicated to child objects. For example, the `<owner>` tag is defined as a mandatory subscription level tag that **must** be applied on all child objects. This means that all resource groups and resources will have the `<owner>` key but the value fields of the tag pair will change based on the actual object owner. The `<owner>` tag is, at a minimum assigned for each subscription and resource group (a single owner may have multiple resource groups). The resource group owner may choose to extend this tag using the “-” field delimiter in the value string creating a `<user-defined>` subfield. User defined subfields are not tracked centrally by the cloud team. A subscription owner **may** choose to track resource owner tags.

Another example of this implementation strategy; the `<enrollment>` tag is defined as a mandatory departmental governance tag at the subscription scope but is not replicated to child objects. This means that each subscription **must** have the `<enrollment>:<enrollment ID>-<SSC broker request ID>-<expiry>` key:value pair but this is not replicated to child resource groups or resources. Within the Microsoft billing system, financial credits are applied to enrollment tiles. These credits have expiry dates based on contractual agreements and are not carried forward beyond the contract expiry date (this is under review with Microsoft). Committed credits are lost after the expiry date unless applied to *reserved instances* (details on the billing system are out of scope of this document). It is critical for cost management that financial commitments are tracked and not allowed to expire (lost) without cost center awareness. Departments using cost center identifiers may choose to include these in the value string of this tag.

Note: The tagging convention should follow a periodic review cycle to ensure new requirements and inefficiencies are addressed.

4.2 TAGGING CONVENTION LIMITATIONS

Each Cloud Service Provider has different limits and restrictions on tags (table may change over time). This convention implements the least common denominator approach based on the table below.

	AWS	Azure	Google (GCP)	IBM
Tags per resource	50	50 (most objects)	64	Treated as string – limit not tested. Key:value , comma delimited
Length of key	127	512	63	128 shared with value
Length of value	256	256	63	128 shared with key
Case sensitive	Yes (keys and values)	No	Lowercase only	No
Allowed characters	Letters, spaces, numbers, and + - = . _ : / @	Alphanumeric and - _ @ : = + (etc.) Not Allowed: <, >, %, &, \, ?, /	Lowercase letters, numeric characters, underscores, and dashes. International characters are allowed.	A-Z, 0-9, white space, underscore, hyphen, period, and colon, and tags are case-insensitive.
Notes	Don't use aws: prefix as that is reserved for AWS. You must " activate " particular tags for cost allocation so that they show up in billing reports. Maximum active tag keys for Billing and Cost Management Reports: 500.	Can tag on Azure Resource Manager (ARM) resources only (not classic Azure). Tag at Resource Group or Resource level. Suggest resource levels for better cost allocation Combine tags or use JSON string if exceeding the 15 tag limit.	Labels are a Beta service. Keys must start with a lowercase letter. Tags are called "Labels" in GCP. There are "network tags" in GCP used to apply firewall rules. These are separate from labels.	Colons turn the tag into a string where you can isolate two logical parts, like a key:value pair. You can't use a colon in a tag without creating this pairing. A comma separates tags and can't be used within the tag name itself.
Taggable resources	EC2 Resources Other Services	All ARM resources can be tagged. List of ARM services	List	
Documentation	Tag Docs User-Defined Tag Restrictions	Tag Docs Best Practices	Label Docs	

Table 11: Tagging Limitations

4.2.1 AZURE TAGGING LIMITATIONS

- Not all resource types support tags. To determine if you can apply a tag to a resource type, see [Tag support for Azure resources](#).
- Each resource or resource group can have a maximum of 50 tag name/value pairs. This limitation applies only to tags directly applied to the resource group or resource. A resource group can contain many resources that each have 50 tag name/value pairs. If you have more than 50 values that you need to associate with a resource, use a JSON string for the tag value. The JSON string can contain many values that are applied to a single tag name.
- The tag name is limited to 512 characters, and the tag value is limited to 256 characters. For storage accounts, the tag name is limited to 128 characters, and the tag value is limited to 256 characters.
- Virtual Machines and Virtual Machine Scale Sets are limited to a total of 2048 characters for all tag names and values.
- Tags applied to the resource group are not inherited by the resources in that resource group.
- Tags can't be applied to classic resources such as Cloud Services.
- Tag names can't contain these characters: <, >, %, &, \, ?, /

4.3 GLOBAL OBJECT TAGGING

This convention is **defined** at the Tenant level and **implemented** at the subscription level in Azure. This design supports flexible implementation while ensuring compliance with key governance requirements. User defined tags are maintained by the resource owner and do not need to be tracked unless mandated by the subscription or resource group owner. This standard does **not** mandate tracking user defined tags. Governance tags are enforced.

4.3.1 GLOBAL TAGGING RULES

The following object tagging rules **must** be respected by all resource deployments that use this convention.

- Tags are assigned at the resource level and are **not** inherited automatically.
- Tags **should** be implemented through code/automation, not manually.
- A governance tag could be global or defined for a specific scope (i.e. a cloud broker tag <cbs> could be implemented at the subscription and resource group level only to track client and billing information but not pushed down to resources).
- In this convention, lowest common denominator is applied, after discussion with SSC partners it was agreed that camel case is allowed in tags. This convention **does not** enforce case within tags but case **must** not be used as a field delimiter within the convention.

Note: Tags in AWS, Azure and Google treat case differently; AWS is case sensitive, Azure is not, and Google only allow lowercase. It is considered best practice to use lowercase tags and implement a “to lower” function in tag automation code. This decisions is made by the departmental cloud governance team.

- The hyphen (“-”) is a reserved character used for field delamination in governance tags.
- Tag format is written <key>:<value> and support extensible fields in either position, the hyphen is a reserved character within this convention for field delamination within tags.

Example 1: SSC is assigned a GC Wide governance key <ssc>, the Cloud Broker implements a mandatory billing code and expiry date tag at the subscription and resource group level <ssc>:<cbs>:<billing code>:<expiry>.

Example 2: The Cloud Operations team implements a mandatory departmental network zone key for all IaaS resources <cnz>:<paz>.

- Minimum of thirty tags **must** be reserved for the resource owner.
- Resource owner <user defined> tags can use any character supported by the cloud provider.
- Spaces are **not** allowed in either the key or value fields of governance tags.
- Date format within this convention uses the ISO8601 GC standard <YYYY-MM-DD>. The slash “/” is not compliant with the least common denominator approach and **must not** be used in governance tags.

4.3.2 TAG RESERVATIONS BY SCOPE

This standard reserves specific tags for future governance, security and reporting purposes. The tables below illustrate these reservations, this is under development and will be updated in the future releases. At each level, the scope **owner** has the flexibility to implement mandatory tags within their area of responsibility. These **should** be tracked and can be queried by resource owners to assist with deployment and compliance. Tags are extensible, which means that a tag owner can include multiple fields in a single tag <value> by using the global tagging delimiter (“-”). In this release of the convention, the hyphen is a reserved character in governance tags only, resource owners are free to use any CSP supported character in their <user defined> tags. These are not tracked.

Scope	Responsibility
GC Wide – 5 Tags	TBS/SSC/CSE
Global (Dept) – 5 Tags	Azure Tenant / Department
Subscription – 5 Tags	Subscription owner
Resource Group – 5 Tags	Resource group owner
Resource – 30 Tags	Resource owner

Table 12: Tagging Scope

4.3.3 GOVERNANCE TAGS

Tags in this section are mandatory and can be implemented through code (IaC) and/or enforced through PowerShell scripts and Azure policies at the appropriate scope.

- GC Global Tags:
 - Managed and enforced at the Tenant level by the Departmental Cloud Governance team.
 - Allowed characters: Camel case, alphanumeric, hyphen “-” in both <key> and <value> fields.
 - Reserved characters: Hyphen “-” is used as a field delimiter.
 - Restricted characters: Lowest common denominator approach **should** be followed.
 - Maximum of 5 GC Wide governance tags are reserved on all resources throughout the hierarchy.
 - Governance tags **must** be tracked.

- Departmental Tags (SSC):
 - Managed and enforced at the Tenant level by the Departmental Cloud Governance team.
 - Allowed characters: Camel case, alphanumeric, hyphen "-" in both <key> and <value> fields.
 - Reserved characters: Hyphen "-" is used as a field delimiter.
 - Restricted characters: Lowest common denominator approach **should** be followed.
 - Maximum of 5 tags are assigned at the Tenant scope.
 - Departmental tags **must** be tracked.

- Subscription Owner Tags:
 - Managed and enforced at the Subscription level by the Subscription owner.
 - Allowed characters: Camel case, alphanumeric, hyphen "-" in both <key> and <value> fields.
 - Reserved characters: Hyphen "-" is used as a field delimiter.
 - Restricted characters: Lowest common denominator approach **should** be followed.
 - Subscription Owner has the flexibility to implement tagging guidelines within their scope.
 - Maximum of 5 tags are assigned at the Subscription scope.
 - Subscription tags **must** be tracked.

- Resource Group Owner Tags:
 - Managed and enforced at the Resource Group level by the Resource Group owner.
 - No character restriction other than those imposed by the cloud service provider.
 - Resource Group owner has the flexibility to implement tagging guidelines within their scope (for example):
 - Allowed characters: Camel case, alphanumeric, hyphen "-" in both <key> and <value> fields.
 - Reserved characters: Hyphen "-" is used as a field delimiter.
 - Restricted characters: Lowest common denominator approach **must** be followed.
 - Maximum of 5 tags are allowed at the resource group scope.
 - Resource Group tags **should** be tracked.

In the Azure Landing Zone Base Cloud Architecture (BCA) implementation the Resource Owner has the flexibility to implement tags using all CSP supported characters. Governance tag <key> strings defined in the table below are reserved and **must** not be used.

Extensibility refers to whether or not [-<user defined string>] may be added to the Tag <value> field. This is experimental and will be leveraged to test tag query integration into external applications where specific values are required.

GOVERNANCE TAGS – Max 20 (on supporting objects)			
Scope: GC Wide – Max 5 / Dept – Max 5 / Subscription – Max 5 / Resource Group – Max 5			
TAG Authority	Scope	Description	Tag fields / Example
CSE	GC global - reserved	Description: placeholder Supports extensibility: YES Length: 1 - 99 Casing: Camel case Valid Characters: Camel case alphanumeric and hyphen	<key>:<value> pair reserved for CSE <cse>:<future value>
SSC	GC global - reserved	Description: placeholder Supports extensibility: YES Length: 1 - 99 Casing: Camel case Valid Characters: Camel case alphanumeric and hyphen	<key>:<value> pairs reserved for SSC <ssc>:<future value>
SSC Cloud Service Broker	GC Global - Subscription only	Description: Billing tag to track Cost Center and other service broker fields Supports extensibility: NO Length: 1 - 99 Casing: Camel Valid Characters: Alphanumeric, hyphen, underscore	Single <key> assigned to SSC CBS, <value> strings are separated by "-", expiry date is optional <ssc-cbs>:<enrollment#>-<cloud broker request ID>[-<enrollment expiry>] Examples: ssc-cbs:4997130-24049 ssc-cbs:4997130-24049-2020-12-31 Note: ISO8601 GC date standard <YYYY-MM-DD>
TBS	GC Global - reserved	Description: Application Portfolio Management Supports extensibility: NO Length: 1 - 99 Casing: Camel case Valid Characters: Camel case alphanumeric, hyphen and underscore	<key>:<value> pair under design to align with TBS CIOB initiative. Link Note: Not implemented at this time Example: apm:appid-00154

GOVERNANCE TAGS – Max 20 (on supporting objects)			
Scope: GC Wide – Max 5 / Dept – Max 5 / Subscription – Max 5 / Resource Group – Max 5			
TAG Authority	Scope	Description	Tag fields / Example
TBS	GC Global – reserved	Description: placeholder Supports extensibility: NO Length: 1 - 99 Casing: Camel case Valid Characters: Camel case alphanumeric and hyphen	<key>:<value> pair to align with TBS cloud profiles. See page 12: GC Cloud Guardrails - DRAFT for Consultation Example: profile:[1-6]
Departmental	Subscription, resource group and all resources	Description: Cost Center Supports extensibility: Yes Length: 1 - 99 Casing: Camel case Valid Characters: Camel case alphanumeric, hyphen and underscore	<key>:<value> pair is defined by the department <costcenter>:<#####>[-<user defined string>] Example: costcenter:22578-proj_123
Subscription	Subscription, resource group and all resources	Description: All resources are assigned the environment tag Supports extensibility: No (reserved for future release) Length: 1 - 99 Casing: Camel Valid Characters: Alphanumeric, hyphen, underscore	Single <key>:<value> <env>:<environment> Example: env:dev
Subscription	Subscription, resource group and all resources	Description: classification – confidentiality, integrity, availability (CIA) Supports extensibility: No Length: 1 - 99 Casing: Camel Valid Characters: alphanumeric, hyphen, underscore	Single <key>:<value> Note: The subscription owner is responsible for systems and data classification <classification>:<pbmm>, <ull>, etc.

GOVERNANCE TAGS – Max 20 (on supporting objects)			
Scope: GC Wide – Max 5 / Dept – Max 5 / Subscription – Max 5 / Resource Group – Max 5			
TAG Authority	Scope	Description	Tag fields / Example
Resource Group and Resources	Resource group and all resources	Description: object owner - can be delegated to the resource group level Supports extensibility: No Length: 1 - 99 Casing: Camel Valid Characters: alphanumeric, hyphen and @	Single <key>:<value> Note: The owner is a valid email address and represents the person for non-technical responsibility of the object – specifically security compliance, risk acceptance and financial. <owner>:<email> Example: owner:jane.doe@canada.ca
Resource Group and Resource	Resource group and all resources	Description: object contact – can be delegated to resource owner Supports extensibility: No Length: 1 - 99 Casing: Camel Valid Characters: alphanumeric and @	Single <key>:<value> Note: The contact is a valid email address and represents the person or functional mailbox for technical responsibility of the object – specifically the level 2 operational support team. <contact>:<email> Example: contact:john.doe@canada.ca

Table 13: Governance Tags

4.3.4 RESOURCE TAGS (USER DEFINED TAGS)

Resource tags (user defined tags) are managed by the resource owner. `Key:value` pairs may be assigned as mandatory or optional by the subscription or resource group owner but are implemented at the resource scope. Each resource object can have up to 30 tags within the standard. The resource owner may choose to implement mandatory tags for some object types and not for others. For example, network objects such as vnets may require an ITSG22 zone tag which is not applied to other objects.

- Resource Owner Tags (User Defined):
 - No character restriction other than those imposed by the cloud service provider.
 - 30 tags are allowed per resource (Microsoft tagging restrictions apply).
 - Tracking `key:value` pairs is out of scope in the current standard.

Note: The Resource tags listed below are included as **examples only**, other departmental tags could include `<resource center>`, `<application>`, `<project>`, etc. Tags are defined and maintained by the **scope** owner.

RESOURCE OWNER (USER DEFINED TAGS)			
Tag	Scope	Description	Tagging Pattern / Example
Code Release (version)	Resource owner	Description: used to track deployment templates Supports extensibility: YES Length: 1 - 99 Casing: Camel case Valid Characters: Alphanumeric, hyphen, slash, period and "@" (any supported by CSP)	Single <code><key></code> assigned to scripting authority, <code><value></code> strings could be separated by "-". Any CSP supported character is allowed. <code><version>:<build>[-<version>]</code> Example: <code>version:LZBCA-r1a-v0.1</code> Note: Tag is user defined and managed by the resource owner. Implementation can include code repository reference, sub versions, etc. Not tracked in the field definition tables.

RESOURCE OWNER (USER DEFINED TAGS)			
Tag	Scope	Description	Tagging Pattern / Example
Legacy data source	Resource owner	Description: Tag used to map data to legacy data set to support api/bridge Supports extensibility: YES Length: 1 - 99 Casing: Camel case Valid Characters: Alphanumeric, hyphen and hyphen (compliant across all CSP)	See Legacy Data Set test case in Appendix A <i>key:value</i> pair - <cmdb-ecd>:<data centre code>-<device type>-<serviceline resolver group>-<... extensible ...> Note: Best practice for data mapping tags is to comply with lowest common denominator strategy.
Resource expiration date	Resource owner	Description: End date for the project Supports extensibility: YES Length: 1 - 99 Casing: Camel case Valid Characters: Alphanumeric, hyphen, slash and ':' (etc.)	<i>key:value</i> pair <expiry>:<date> Example: expiry:2022-02-27
Maintenance dependencies	Resource owner	Description: Used to define system dependencies for maintenance (i.e. windows, pre/post execution scripts, startup order, etc.). Supports extensibility: Yes Length: 1 - 99 Casing: Camel case Valid Characters: Alphanumeric and hyphen.	<i>key:value</i> pair <maintenance>:<user defined strings>[-<string>[-<string>]] Example: maintenance:tue0400-tue0430-edt-executemyscript

RESOURCE OWNER (USER DEFINED TAGS)			
Tag	Scope	Description	Tagging Pattern / Example
Legacy Configuration Management Database	Departmental scope	Description: Used to link to trouble ticketing / asset management systems Supports extensibility: Yes Length: 1 - 99 Casing: Camel case Valid Characters: Alphanumeric and hyphen.	<i>key:value</i> pair <code><cmdb>:<user defined strings>[-<string>[-< string >]]</code> Note: SSC uses Enterprise Control Desk (ECD) resolver groups to assign tickets Example: cmdb:ecd-dc00243

Table 14: User Defined Tables

5 APPENDIX A

Department Codes – Pending GC EARB Approval – Only includes SSC partners

No.	Dept Abbr.	2 Character Code for Cloud	Department name
1	AAFC	Aa	Agriculture and Agri-Food (Department of)
2	ACOA	Ac	Atlantic Canada Opportunities Agency
3	CBSA	Cb	Canada Border Services Agency
4		Ed	Canada Economic Development for Quebec Regions
5		Hr	Canadian Human Rights Commission
6	CRA	Ra	Canada Revenue Agency - (Administered Activities)
7		Sp	Canada School of Public Service
8	CFIA	Cf	Canadian Food Inspection Agency
9	PCH	Ch	Canadian Heritage (Department of)
10	CanNor	Ne	Canadian Northern Economic Development Agency
11	CNSC	Ns	Canadian Nuclear Safety Commission
12	CSA	Sa	Canadian Space Agency
13	IRCC	Ci	Citizenship and Immigration (Department of)
14	CSC	Cs	Correctional Service of Canada
15	FIN	Df	Finance (Department of)
16	JUS	Dj	Justice (Department of)
17	ESDC	Es	Employment and Social Development (Department of)
18		Ec	Environment Canada (Department of the)
19	FedDev Ontario	Fe	Federal Economic Development Agency for Southern Ontario
20	FINTRAC	Ft	Financial Transactions and Reports Analysis Centre of Canada
21	DFO	Fo	Fisheries and Oceans Canada (Department of)
22		Fa	Foreign Affairs, Trade and Development
23	IRB	Ir	Immigration and Refugee Board of Canada
24	INAC	In	Indigenous and Northern Affairs Canada (Department of)
25	ISED	Is	Innovation, Science and Economic Development Canada
26	INFC	Ic	Infrastructure Canada
27	ICH	Iv	Invest in Canada Investir au Canada
28	LAC	La	Library and Archives of Canada

Security Classification **Unclassified**

Status: DRFT

Subject: SSC NAMING AND TAGGING STANDARD FOR AZURE

No.	Dept Abbr.	2 Character Code for Cloud	Department name
29	DND	Nd	National Defence (Department of)
30	NRC	Nr	National Research Council of Canada
31	NRCan	Na	Natural Resources Canada (Department of)
32	PC	Pc	Parks Canada Agency
33	PCO	Pr	Privy Council Office
34	PHAC	Ph	Public Health Agency of Canada
35	PS	Ps	Public Safety and Emergency Preparedness (Department of)
36	PSC	Ps	Public Service Commission of Canada
37	PSPC	Pw	Public Works and Government Services (Department of)
38	RCMP	Mp	Royal Canadian Mounted Police
39	SSC	Sc	Shared Services Canada
40	StatCan	St	Statistics Canada
41	TC	Tc	Transport Canada (Department of)
42	TBS	Tb	Treasury Board Secretariat
43	VAC	Va	Veterans Affairs (Department of)
44	WD	We	Western Economic Diversification (Department of)
45	HC	Hc	Health Canada (Department of)

Table 15: Department Codes