

# AWS Tagging/Naming Conventions

## Naming of Objects

Nearly all AWS resources can be named. These should follow a consistent pattern to be able to more clearly identify them and their context. The standard naming convention described below should be used in all AWS accounts. If the name tag is available, then the object must be given one. Below are examples of resources and the associated naming conventions as they would be following this pattern. NOTE: If a resource type is not listed below, it should still conform to this naming convention.

<b>AWS Resource</b>	<b>Name recipe (Camel Case + ALL CAP appended to the end (e.g., SG, VPC, S3, KEY))</b>	<b>Comment</b>	<b>Example</b>
VPC	ITS-Region-<name>		ITS-USEast1-Prod ITS-USEast2-DevQA
Subnet	ITS-Region-<vpc name>-<scope>-<AZ>	Scope: Public or Private	ITS-USEast1-Prod-Public-2a ITS-USEast2-DevQA-Private-2c
Elastic Load Balancer	<project>-whatever your team needs-<(dev/stage/prod/whatever)>-ELB		
EC2	<project>-whatever your team needs-<(dev/stage/prod/whatever)>-EC2		AURA-WebSvr-prod-EC2
Security Group	<project>-whatever your team needs-<(dev/stage/prod/whatever)>-SG		
S3	<project>-whatever your team needs-<(dev/stage/prod/whatever)>-s3 (note: s3 bucket names are lowercase)		

Key	<project>-whatever your team needs-<(dev/stage/prod/whatever)>-KEY		
RDS	<project>-whatever your team needs-<(dev/stage/prod/whatever)>-RDS		
SQS	<project>-whatever your team needs-<(dev/stage/prod/whatever)>-SQS		
SNS	<project>-whatever your team needs-<(dev/stage/prod/whatever)>-SNS		
USER	<team>-brief description of need-<(dev/stage/prod/whatever)>API-USER	these users are ONLY programmatic users. no console access. console access is ONLY through grouper.	linuxteam-ec2-autoscale-prod-API-USER
POLICY	if tied to an API-USER, use <same name as API-USER>-POLICY		linuxteam-ec2-autoscale-prod-API-USER-POLICY
RDS Parameters	<project>-whatever your team needs-<(dev/stage/prod/whatever)>-RDS-PARAM		

## TAGS

Based on: <https://aws.amazon.com/answers/account-management/aws-tagging-strategies/>

as well as convention for DLT: <https://opscenter.dlt.com/hc/en-us/articles/235230267-AWS-Tagging-with-a-DLT-AWS-Account>

<https://confluence.cornell.edu/display/CLOUD/Standard+Tagging>

## RULES

1. use on every type of AWS object
2. you can use whatever other tags you like
3. but these tags must be used according to the environment

	<b>TAG</b>	<b>PROD - Required</b>	<b>SANDBOX - Required</b>	<b>standardized/open</b>	<b>details/example</b>
1	CostCode	yes, get it from <a href="mailto:its-aws-requests@lists.uchicago.edu">its-aws-requests@lists.uchicago.edu</a> , or make one up and we'll correct it later	yes, your cnetid	(Standardized: What has been provided by Cornelia).	If you don't know your CostCode, or need one, email <a href="mailto:its-aws-requests@lists.uchicago.edu">its-aws-requests@lists.uchicago.edu</a>
2	Purpose	yes	yes	(Open: What is the resource used for? Use your nomenclature.)	
3	Role	yes	yes	(Open-ish: What architectural purpose does this serve?)	e.g., database, web server
4	Environment	yes	optional	(Standardized:)	dev, test, stage, prod
5	ResponsibleParty	yes	yes	(Standardized: Service owner, Who is the first responder when it breaks? for the value, use name + contact, e.g.,	unix-team@ <a href="mailto:lists.uchicago.edu">lists.uchicago.edu</a>
6	Owner	yes	optional	(Standardized: Who is the business owner? We know it'll change, but we can batch that later.)	"First Name / Last Name <email>"

7	Project	yes	optional	(Open: what's the project?) identifies the project(s) the resource supports. This should be consistent for resources supporting the same project for ease or programming, querying, reporting, etc.	griffin, aura,
---	---------	-----	----------	---	----------------