

Bitbucket Data Center on the AWS Cloud

Quick Start Reference Deployment

October 2016
([last update](#): April 2020)

Chris Szmajda, Felix Haehnel – Atlassian
Shiva Narayanaswamy, Tony Vattathil – Amazon Web Services

Contents

Overview.....	2
Costs and licenses.....	3
Architecture.....	4
Amazon Aurora PostgreSQL.....	5
Auto Scaling groups in this Quick Start.....	6
Prerequisites.....	6
Specialized knowledge.....	6
Technical requirements.....	7
Deployment options.....	7
Deployment steps.....	7
Step 1. Prepare an AWS account.....	7
Step 2. Launch the Quick Start.....	8
Step 3. Review the parameters and create your stack.....	10
Step 4. Configure Bitbucket Data Center.....	21
Step 5: Add nodes to the cluster.....	23
FAQ.....	24

Additional resources	25
Git repository	26
Document revisions	26

This Quick Start deployment guide was created by Atlassian in collaboration with Amazon Web Services (AWS).

[Quick Starts](#) are automated reference deployments that use AWS CloudFormation templates to deploy key technologies on AWS, following AWS best practices.

Overview

This Quick Start reference deployment guide provides step-by-step instructions for deploying Atlassian Bitbucket Data Center on the AWS Cloud.

Bitbucket Data Center is an on-premises Git repository management solution from Atlassian. It provides source code collaboration for enterprises that require high availability and performance at scale. Bitbucket Data Center provides the following features:

- **Clustering:** Enables you to run your Bitbucket instance on multiple nodes in the same data center. The cluster of nodes share the workload and provide scalable capacity, performance, and high availability. The cluster improves performance by distributing the workload across multiple dedicated machines. You can add nodes to your cluster to boost your capacity instantly, with no downtime. In the event of a system outage in one node, the remaining nodes can continue to handle incoming requests with little or no loss of availability.
- **Smart mirroring:** Lets you set up synchronized repository copies (*mirrors*) in geographically distributed locations, so that users can clone and fetch content faster. For more information about smart mirroring, see the [Atlassian documentation](#).
- **Disaster recovery:** Helps you deploy an offsite disaster recovery system for business continuity even in the event of a complete system outage.

For more information about Bitbucket Data Center, see the [Atlassian Bitbucket documentation](#).

This Quick Start is for users who want to deploy Bitbucket Data Center in a supported configuration in the AWS Cloud, following AWS best practices.

Please know that we may share who uses AWS Quick Starts with the AWS partner that collaborated with AWS on the content of the Quick Start.

Costs and licenses

You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using the Quick Start.

The AWS CloudFormation templates for this Quick Start includes configuration parameters that you can customize. Some of these settings, such as instance type, will affect the cost of deployment. For cost estimates, see the pricing pages for each AWS service you will be using. Prices are subject to change.

Tip: After you deploy the Quick Start, we recommend that you enable the [AWS Cost and Usage Report](#) to track costs associated with the Quick Start. This report delivers billing metrics to an S3 bucket in your account. It provides cost estimates based on usage throughout each month, and finalizes the data at the end of the month. For more information about the report, see the [AWS documentation](#).

A Bitbucket Data Center license is required to deploy this Quick Start. For information on obtaining a Bitbucket Data Center license, see the [Atlassian pricing page](#).

Note: This Quick Start can't be used with Bitbucket Server licenses.

If you want to evaluate Bitbucket Data Center, you can also request an evaluation license after you deploy the Quick Start, when you configure Bitbucket Data Center.

Architecture

Deploying this Quick Start with the **default parameters** builds the following Bitbucket Data Center environment in the AWS Cloud.

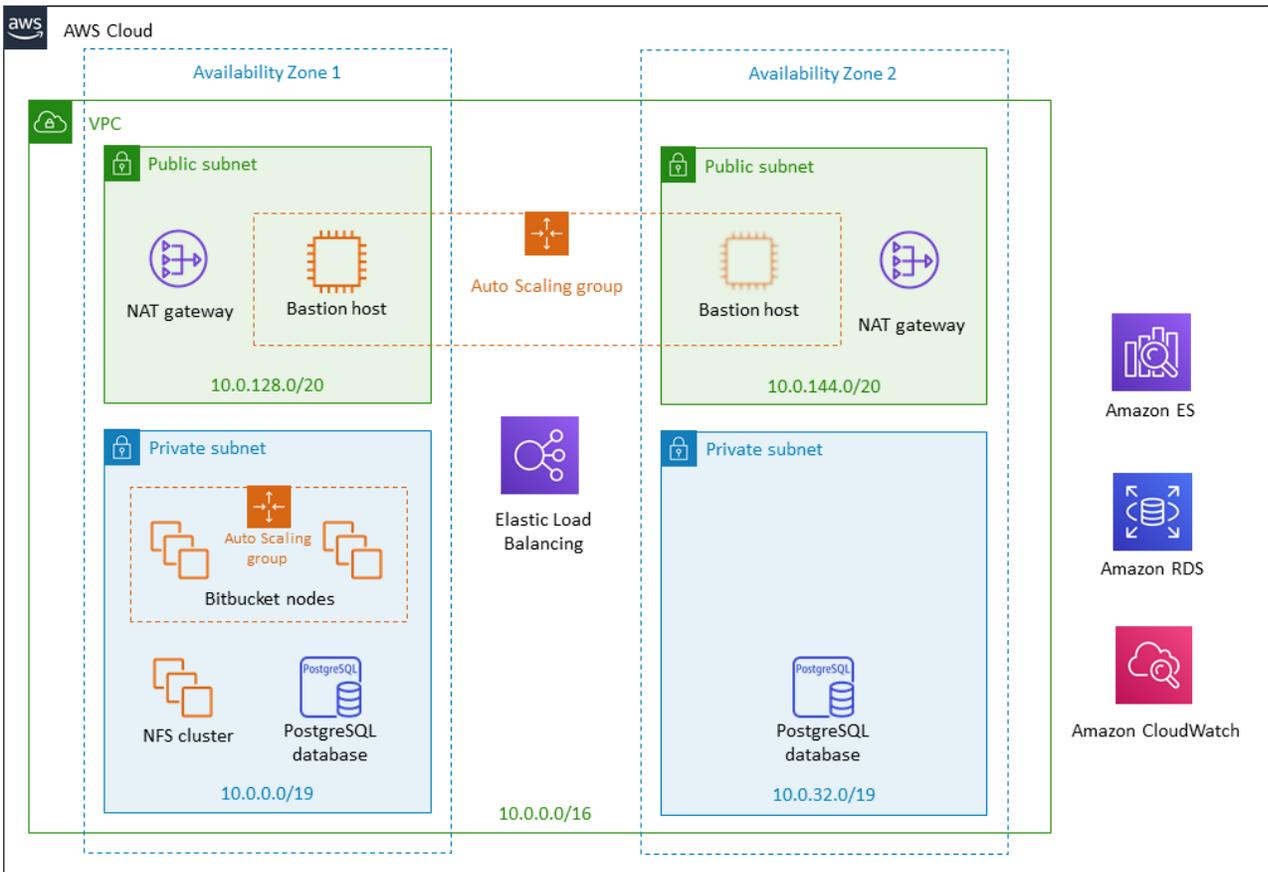


Figure 1: Bitbucket Data Center architecture on AWS

This Quick Start includes two AWS CloudFormation templates: the first one builds out the Atlassian Standard Infrastructure (ASI), which is a VPC that contains the components required by all Atlassian Data Center applications, and then provisions Bitbucket Data Center into this ASI. The second template provisions Bitbucket Data Center in an existing ASI.

The resulting infrastructure from this Quick Start contains the following:

- **Amazon RDS PostgreSQL:** Bitbucket Data Center requires a supported external database. Amazon Relational Database Service (Amazon RDS) PostgreSQL in a Multi-AZ configuration allows failover in the event the master node fails. Alternatively, you can use a more resilient [Amazon Aurora PostgreSQL](#) clustered database.

- **NFS server:** Bitbucket Data Center uses a shared network file system (NFS) to store the repositories in a common location that is accessible to multiple Bitbucket nodes. The Quick Start architecture implements the shared file system in an Amazon Elastic Compute Cloud (Amazon EC2) instance with an attached Amazon Elastic Block Store (Amazon EBS) volume. We recommend that you create regular snapshots of the EBS volume at a frequency that meets the recovery point objective (RPO) of your organization. If using a single EBS volume doesn't meet your availability and disaster recovery requirements, you should consider a highly available NFS implementation using AWS partner products.
- **Bitbucket Auto Scaling group:** The Bitbucket Data Center product is installed on EC2 instances in an Auto Scaling group. The instances are based on Amazon Linux and use an Atlassian-provided Amazon Machine Image (AMI). The scaling metric is CPU utilization.
- **Amazon Elastic Load Balancing:** This works both as a load balancer and a Secure Sockets Layer (SSL) termination reverse proxy.
- **Amazon ES:** Bitbucket Data Center uses Elasticsearch 2.3 for indexing and searching. The Quick Start architecture uses the Amazon Elasticsearch Service (Amazon ES), which is a managed service that makes it easy to deploy, operate, and scale Elasticsearch in the AWS Cloud.
- **Bastion host:** This host enables secure access to your Bitbucket application without exposing it to the internet. (For more information, see [Bastion Hosts](#).)
- **Amazon CloudWatch:** This provides basic monitoring for all application and database nodes in your deployment. By default, CloudWatch will also collect and store logs from each monitored node. This Quick Start configures a CloudWatch dashboard to help you read and analyze collected data. You can configure this dashboard with additional metrics and monitoring after deployment.

Amazon Aurora PostgreSQL

You can also use this Quick Start to deploy Bitbucket Data Center with an Aurora clustered database that's compatible with PostgreSQL. The cluster configuration is illustrated in Figure 2. It features a primary database writer that replicates to two database readers in a different Availability Zone. If the writer fails, Aurora promotes one of the readers to take its place with no downtime. For more information, see [Amazon Aurora Features: PostgreSQL-Compatible Edition](#).

The Aurora configuration with a single database writer and two readers is designed to provide high availability without degrading performance.

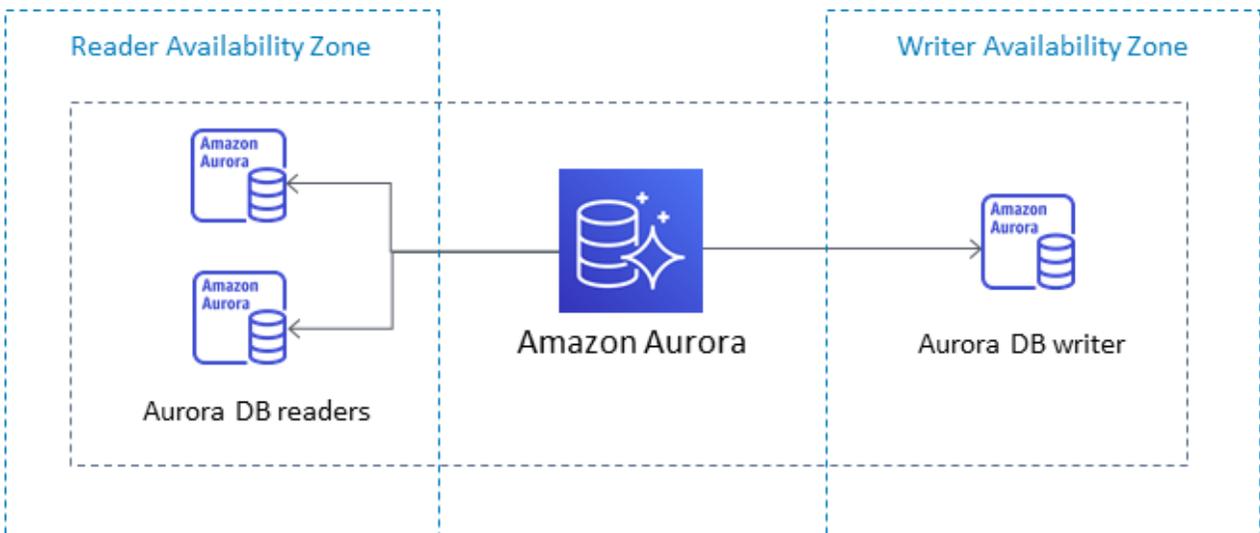


Figure 2: Aurora reader/writer configuration

Amazon Aurora is supported on Bitbucket Data Center 6.7 and later.

Auto Scaling groups in this Quick Start

This Quick Start uses [Auto Scaling groups](#), but only to statically control the number of its nodes. We don't recommend that you use Auto Scaling to dynamically scale the size of your cluster. Adding an application node to the cluster usually takes more than 20 minutes, which isn't fast enough to address sudden load spikes.

If you can identify any periods of high and low load, you can schedule the application node cluster to scale accordingly. See [Scheduled Scaling for Amazon EC2 Auto Scaling](#) for more information.

To study trends in your organization's load, you'll need to monitor the performance of your deployment. Refer to Bitbucket Data Center sample deployment and monitoring strategy for tips on how to do so.

Prerequisites

Specialized knowledge

Before you deploy this Quick Start, we recommend that you become familiar with the following AWS services. (If you are new to AWS, see the [Getting Started Resource Center](#).)

- [Amazon Virtual Private Cloud \(Amazon VPC\)](#)
- [Amazon EC2](#)
- [Amazon EBS](#)
- [Amazon ES](#)
- [PostgreSQL on Amazon RDS](#)

Technical requirements

This Quick Start requires an Atlassian account and a license to use Bitbucket Data Center, as discussed earlier in the [Costs and Licenses](#) section.

Deployment options

This Quick Start provides two deployment options:

- **Deploy Bitbucket into a new ASI (end-to-end deployment).** Choose this option if you're a new user. This option builds the Atlassian Standard Infrastructure (ASI), which is a VPC that consists of the subnets, NAT gateways, security groups, bastion hosts, and other infrastructure components required by all Atlassian applications, and then deploys Bitbucket into this new VPC.
- **Deploy Bitbucket into an existing ASI.** Choose this option if you've already deployed the ASI separately by using the [ASI Quick Start](#), or by deploying another Atlassian product ([Jira](#) or [Confluence Data Center](#)). This option provisions Bitbucket in your existing ASI.

Important: You can provision Bitbucket Data Center into an ASI that has an existing Atlassian application (for example, Jira Software Data Center). If you are deploying and integrating multiple Atlassian applications, we recommend that you provision them all in the same ASI.

The Quick Start provides separate templates for these options. It also lets you configure CIDR blocks, instance types, and Bitbucket settings, as discussed later in this guide.

Deployment steps

Step 1. Prepare an AWS account

1. If you don't already have an AWS account, create one at <https://aws.amazon.com> by following the on-screen instructions.

2. Use the region selector in the navigation bar to choose the AWS Region where you want to deploy Bitbucket Data Center on AWS.
3. Create a [key pair](#) in your preferred region.
4. If necessary, [request a service quota increase](#) for the Amazon EC2 **c4.xlarge** instance type. You might need to do this if you already have an existing deployment that uses this instance type, and you think you might exceed the [default quota](#) with this reference deployment.

Step 2. Launch the Quick Start

Note: You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using this Quick Start. For full details, see the pricing pages for each AWS service you will be using in this Quick Start. Prices are subject to change.

There are two methods of deploying Bitbucket Data Center into your AWS account. The fastest way is by [launching the Quick Start directly from our Amazon S3 bucket](#). However, when you do, any updates we make to the Quick Start templates will propagate directly to your deployment. These updates sometimes involve adding or removing parameters from the templates. This could introduce unexpected (and possibly breaking) changes to your deployment.

As such, for production environments, we recommend a second method, where you [copy the Quick Start templates into your own S3 bucket](#). Then, launch them directly from there. Launching from your own templates lets you control when to apply our latest changes to your environment.

Method 1: Launch the Quick Start from its original Amazon S3 bucket

1. Use one of the following options to deploy Bitbucket Data Center into your AWS account. For help choosing an option, see [deployment options](#) earlier in this guide.

Launch Quick Start
(with new ASI)

Launch Quick Start
(for existing ASI)

Deploying Bitbucket into a new ASI takes about 50 minutes. Deploying Bitbucket into an existing ASI takes about 30 minutes.

2. Check the AWS Region that's displayed in the upper-right corner of the navigation bar and change it if necessary. This is where the network infrastructure for your AWS

environment will be built. The template is launched in the US West (Oregon) Region by default.

3. On the **Select Template** page, keep the default setting for the template URL, and then choose **Next**.

At this point, you can go directly to [Step 3. Review the parameters and create your stack](#) to continue.

Method 2 (recommended): Launch the Quick Start from your own S3 bucket

1. Clone a local copy of the [Quick Start templates](#) (including all of its submodules). From the command line, run:

```
git clone --recurse-submodules https://github.com/aws-quickstart/quickstart-atlassian-bitbucket.git
```

2. **(Optional)** The Quick Start templates repository uses the directory structure required by the Quick Start interface. If needed (for example, to minimize storage costs), you can remove all other files except the following:

```
quickstart-atlassian-bitbucket
├── submodules
│   ├── quickstart-atlassian-services
│   │   └── templates
│   │       └── quickstart-vpc-for-atlassian-services.yaml
│   └── templates
│       ├── quickstart-bitbucket-dc-with-vpc.template.yaml
│       └── quickstart-bitbucket-dc.template.yaml
```

3. Install and set up the AWS Command Line Interface (CLI). This tool enables you to create an S3 bucket and upload content to it.
4. Create an S3 bucket in your region:

```
aws s3 mb s3://<bucket-name> --region <AWS_REGION>
```

5. Choose which Quick Start template you'll be using and open it:
 - a. **quickstart-bitbucket-dc-with-vpc.template.yaml**: use this for deploying Bitbucket into a new ASI (end-to-end deployment).
 - b. **quickstart-bitbucket-dc.template.yaml**: use this for deploying Bitbucket into an existing ASI.

- In the template you've chosen, the `QSS3BucketName` default value is set to `aws-quickstart`. Replace this default with the name of the bucket you created earlier.
- Go into the parent directory of your local clone of the Quick Start templates. From there, upload all the files in your local clone to your S3 bucket:

```
aws s3 cp quickstart-atlassian-bitbucket s3://<bucket-name> --recursive --acl public-read
```

Once you've uploaded everything, you're ready to deploy your production stack from your S3 bucket. Go to **CloudFormation** → **Create Stack**. When specifying a template, paste in the Object URL of the Quick Start template you'll be using.

You can now continue to [Step 3. Review the parameters and create your stack](#).

Step 3. Review the parameters and create your stack

At this point, you should have launched the Quick Start. Whether you launched it from its [original S3 bucket](#) or [from your own](#), the next steps are the same:

- On the **Specify Details** page, review the parameters for the template. Enter values for the parameters that require your input. For all other parameters, you can customize the default settings provided by the template.

In the following tables, parameters are listed by category and described separately for the two deployment options:

- [Parameters for deploying Bitbucket with a new ASI](#)
- [Parameters for deploying Bitbucket into an existing ASI](#)

When you finish reviewing and customizing the parameters, choose **Next**.

- Parameters for deploying Bitbucket with a new ASI**

[View template](#)

Bitbucket setup:

Parameter label (name)	Default	Description
Version * (BitbucketVersion)	6.3.1	The version of Bitbucket to install. Bitbucket version 5.0.0 and later are supported. For a list of versions, see the Atlassian documentation .

Cluster nodes:

Parameter label (name)	Default	Description
Enable CloudWatch Integration (CloudWatchIntegration)	Metrics and Logs	Controls the extent to which Amazon CloudWatch should monitor your deployment. Metrics and Logs monitors node CPU, disk, and network activity, while also performing centralized logging. If the costs incurred by Amazon CloudWatch are an issue, use Metrics Only (to disable logging) or Off (to disable it completely).
Bitbucket cluster node instance type (ClusterNodeInstanceType)	C5.xlarge	The EC2 instance type for the Bitbucket Data Center nodes. See the Atlassian Bitbucket documentation for guidance.
Maximum number of cluster nodes (ClusterNodeMax)	2	The maximum number of Bitbucket Data Center nodes in the cluster. When you first launch the Quick Start, leave the default value of 2 nodes unchanged. In step 4 , after configuring Bitbucket Data Center for multinode clustering, you can change this parameter setting to the desired number of cluster nodes.
Minimum number of cluster nodes (ClusterNodeMin)	1	The minimum number of Bitbucket Data Center nodes in the cluster. Leave this parameter at its default setting for a new deployment. You can update it after deployment.
Deployment Automation Git Repository URL (DeploymentAutomationRepository)	https://bitbucket.org/atlassian/dc-deployments-automation.git	The deployment automation repository to use for per-node initialization. Leave this as default unless you have customizations.
Deployment Automation Branch (DeploymentAutomationBranch)	master	The deployment automation repository branch to pull from.
Ansible playbook to invoke during instance initialization (DeploymentAutomationPlaybook)	aws_bitbucket_dc_node.yml	The Ansible playbook to invoke to initialize the application node on first start.
Custom command-line parameters for Ansible (DeploymentAutomationCustomParams)	<i>Optional</i>	Additional command-line options for the `ansible-playbook` command. See https://bitbucket.org/atlassian/dc-deployments-automation/src/master/README.md for more information about overriding parameters.
SSH key name to use with the repository (DeploymentAutomationKeyName)	<i>Optional</i>	Named KeyPair name to use with this repository. The key should be imported into the SSM parameter store.

File server:

Parameter label (name)	Default	Description
File server instance type (FileServerInstanceType)	m4.xlarge	The EC2 instance type for the file server that is hosting the Bitbucket shared home directory. See the Atlassian Bitbucket documentation for guidance.
Home directory size (HomeSize)	100	The storage size for the home directory, in GB. Allowed range is 100-16,384.
Home directory volume type (HomeVolumeType)	Provisioned IOPS	The volume type for the home directory. You can choose General Purpose SSD or Provisioned IOPS SSD. For more information, see the AWS documentation .
Home directory IOPS (HomeIops)	1000	The IOPS for the home directory. This value is used only when the Home volume type parameter is set to Provisioned IOPS. Allowed range is 100-20,000. The ratio of IOPS to volume size must be 50 or less. For example, if you set this parameter to 5000 IOPS, the home directory size must be at least 100 GB.

Database:

Parameter label (name)	Default	Description
The database engine to deploy with (DBEngine)	PostgreSQL	The database engine to use for the application. Choose either PostgreSQL (for Amazon RDS) or Amazon Aurora PostgreSQL.
Database instance class (DBInstanceClass)	db.m4.large	The EC2 instance type for the Amazon RDS database.
RDS Provisioned IOPS (DBIops)	1000	Must be in the range of 1000 - 30000 and a multiple of 1000. This value is only used with Provisioned IOPS. Note: The ratio of IOPS per allocated-storage must be between 3.00 and 10.00. Not valid for Amazon Aurora.
Master password * (DBMasterUserPassword)	<i>Requires input</i>	The password for the master (postgres) account. Must be at least 8 characters and include 1 uppercase, 1 lowercase, 1 number, and 1 of the following symbols: ! # \$ { * : [= ,] - _ @ + % &.
Enable RDS Multi-AZ deployment (DBMultiAZ)	True	If True , your database deploys with high availability. When DBEngine equals PostgreSQL , a multi-AZ RDS instance is provisioned. When DBEngine equals Amazon Aurora PostgreSQL , a multi-node Amazon Aurora cluster is provisioned. If high availability isn't a concern, set this parameter to False .

Parameter label (name)	Default	Description
Bitbucket database password * (DBPassword)	<i>Requires input</i>	Database password used by BitBucket. Must be at least 8 characters and include 1 uppercase, 1 lowercase, 1 number, and 1 of the following symbols: ! # \$ { * : [= ,] - _ @ + % &.
Database storage (DBStorage)	10	The storage size, in GB, to allocate to the database. This value should be 100-6144, if you've selected Provisioned IOPS for the database storage type. This setting isn't used for Amazon Aurora.
Database storage type (DBStorageType)	General Purpose (SSD)	The database storage type. You can choose General Purpose SSD or Provisioned IOPS SSD. For more information, see the AWS documentation .

Elasticsearch:

Parameter label (name)	Default	Description
Elasticsearch instance type (ElasticsearchInstanceType)	m4.xlarge.elasticsearch	The EC2 instance type for the Amazon Elasticsearch service to run on.
Elasticsearch disk space per node (ElasticsearchNodeVolumeSize)	100	EBS volume size (in gigabytes) per Elasticsearch node.

Networking configuration:

Parameter label (name)	Default	Description
Trusted IP range (AccessCIDR)	<i>Requires input</i>	The CIDR IP range that is permitted to access Atlassian services. We recommend that you set this value to a trusted IP range. For example, you might want to grant only your corporate network access to the software.
Make instance internet facing (InternetFacingLoadBalancer)	True	Controls whether the load balancer should be visible to the internet (True) or only within the VPC (False).
SSH Key Pair Name (KeyPairName)	—	The EC2 key pair used by SSH to access the instances. If you don't provide one, the Atlassian Standard Infrastructure stack's key pair is used.
Existing DNS name (CustomDnsName)	—	Use a custom existing DNS name for your Data Center instance. You must own this domain and configure it to point to the load balancer.

Parameter label (name)	Default	Description
SSL certificate ARN (SSLCertificateARN)	—	The Amazon Resource Name (ARN) of your SSL certificate. The AWS Certificate Manager (ACM) will display the ARN upon certificate creation. If you want to use your own certificate that you generated outside AWS, you must first import it to ACM by following the instructions in the AWS documentation . The import command outputs the ARN.
IP address block for the VPC (VPCCIDR)	10.0.0.0/16	The CIDR block for the VPC.
Availability Zones (AvailabilityZones)	<i>Requires input</i>	The list of Availability Zones to use for the subnets in the VPC. The Quick Start uses two Availability Zones from your list and preserves the logical order you specify.
AZ1 private IP address block (PrivateSubnet1CIDR)	10.0.0.0/19	The CIDR block for the private subnet located in Availability Zone 1.
AZ2 private IP address block (PrivateSubnet2CIDR)	10.0.32.0/19	The CIDR block for the private subnet located in Availability Zone 2.
AZ1 public IP address block (PublicSubnet1CIDR)	10.0.128.0/20	The CIDR block for the public (DMZ) subnet located in Availability Zone 1.
AZ2 public IP address block (PublicSubnet2CIDR)	10.0.144.0/20	The CIDR block for the public (DMZ) subnet located in Availability Zone 2.

Advanced (Optional):

Parameter label (name)	Default	Description
Bitbucket properties (BitbucketProperties)	—	A comma-separated list of Bitbucket properties in the format key1=value1, key2=value2, etc. For details, see the Atlassian documentation .
JVM Heap Size Override (JvmHeapOverride)	—	Override the default amount of memory to allocate to the JVM for your instance type - set size in mg or gb (e.g., 1024 mg or 1 gb).
Additional JVM options (JvmSupportOpts)	—	Pass in any additional JVM options to tune the Bitbucket instance.
Create S3 bucket for Elasticsearch snapshots (CreateBucket)	True	Set this parameter to True to create the S3 bucket for Elasticsearch snapshots. Use this parameter in conjunction with the Elasticsearch snapshot S3 bucket name parameter.

Parameter label (name)	Default	Description
Bitbucket primary database (DBMaster)	—	The database ARN of the Amazon RDS instance to replicate. Setting this parameter will bring up Bitbucket as a disaster recovery standby with an Amazon RDS read replica database. This parameter is not supported when using Amazon Aurora.
Database snapshot ID to restore (DBSnapshotId)	—	The Amazon RDS snapshot ID of an existing database backup to restore. This parameter must be used with the Home volume snapshot ID to restore (HomeVolumeSnapshotId) parameter. Leave this value blank for a new instance. This parameter is not supported when using Amazon Aurora.
Elasticsearch snapshot S3 bucket name (ESBucketName)	—	The name of a new or existing S3 bucket configured for Elasticsearch snapshots.
Elasticsearch snapshot ID to restore (ESSnapshotId)	—	The ID of a snapshot in the configured Amazon S3 snapshot repository. This parameter must be used with the Elasticsearch snapshot S3 bucket name parameter.
Home volume snapshot ID to restore (HomeVolumeSnapshotId)	—	The Amazon EBS snapshot ID of an existing database backup to restore as the home directory. This parameter must be used with the Database snapshot ID to restore parameter. Leave this value blank for a new instance.
Delete Home on termination (HomeDeleteOnTermination)	True	Keep the default setting of True to delete the home directory when the instance is terminated. If True , you must back up your data before terminating your instance. Set to False to keep the home directory volume upon termination.

AWS Quick Start configuration:

Parameter label (name)	Default	Description
Quick Start S3 Bucket Name (QSS3BucketName)	atlassian-aws	The S3 bucket you have created for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. The bucket name can include numbers, lowercase letters, uppercase letters, and hyphens, but should not start or end with a hyphen.
Quick Start S3 Key Prefix (QSS3KeyPrefix)	quickstart-atlassian-bitbucket/	The S3 key name prefix used to simulate a folder for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. This prefix can include numbers, lowercase letters, uppercase letters, hyphens, and forward slashes.
ASI Exported Prefix (ExportPrefix)	ATL-	Identifier used in all variables (VPCID, SubnetIDs, KeyName) exported from this deployment's Atlassian Standard Infrastructure. Use different identifiers if you're deploying multiple Atlassian Standard Infrastructures in the same AWS Region.

- **Parameters for deploying Bitbucket Data Center into an existing ASI**

[View template](#)

Bitbucket setup:

Parameter label (name)	Default	Description
Version * (BitbucketVersion)	6.3.1	The version of Bitbucket to install. Bitbucket version 5.0.0 and later are supported. For a list of versions, see the Atlassian documentation .

Cluster nodes:

Parameter label (name)	Default	Description
Enable CloudWatch Integration (CloudWatchIntegration)	Metrics and Logs	Controls the extent to which Amazon CloudWatch should monitor your deployment. Metrics and Logs monitors node CPU, disk, and network activity, while also performing centralized logging. If the costs incurred by Amazon CloudWatch are an issue, use Metrics Only (to disable logging) or Off (to disable it completely).
Bitbucket cluster node instance type (ClusterNodeInstance Type)	c5.xlarge	The EC2 instance type for the Bitbucket Data Center nodes. See the Atlassian Bitbucket documentation for guidance.
Maximum number of cluster nodes (ClusterNodeMax)	2	The maximum number of Bitbucket Data Center nodes in the cluster. When you first launch the Quick Start, leave the default value of 2 nodes unchanged. In step 4 , after configuring Bitbucket Data Center for multinode clustering, you can change this parameter setting to the desired number of cluster nodes.
Minimum number of cluster nodes (ClusterNodeMin)	1	The minimum number of Bitbucket Data Center nodes in the cluster. Leave this parameter at its default setting for a new deployment. You can update it after deployment.
Deployment Automation Git Repository URL (DeploymentAutomation Repository)	https://bitbucket.org/atlassian/dc-deployments-automation.git	The deployment automation repository to use for per-node initialization. Leave this as default unless you have customizations.
Deployment Automation Branch (DeploymentAutomation Branch)	master	The deployment automation repository branch to pull from.
Ansible playbook to invoke during instance initialization	aws_bitbucket_dc_node.yml	The Ansible playbook to invoke to initialize the application node on first start.

Parameter label (name)	Default	Description
(DeploymentAutomationPlaybook)		
SSH key name to use with the repository (DeploymentAutomationKeyName)	–	Named KeyPair name to use with this repository. The key should be imported into the SSM parameter store.
Custom command-line parameters for Ansible (DeploymentAutomationCustomParams)	–	Additional command-line options for the `ansible-playbook` command. See https://bitbucket.org/atlassian/dc-deployments-automation/src/master/README.md for more information about overriding parameters.

File server:

Parameter label (name)	Default	Description
File server instance type (FileServerInstanceType)	m4.xlarge	The EC2 instance type for the file server that is hosting the Bitbucket shared home directory. See the Atlassian Bitbucket documentation for guidance.
Home directory size (HomeSize)	100	The storage size for the home directory, in GB. Allowed range is 100-16384.
Home directory volume type (HomeVolumeType)	Provisioned IOPS	The volume type for the home directory. You can choose General Purpose SSD or Provisioned IOPS SSD. For more information, see the AWS documentation .
Home directory IOPS (HomeIops)	1000	The IOPS for the home directory. This value is used only when the Home volume type parameter is set to Provisioned IOPS. Allowed range is 100-20,000. The ratio of IOPS to volume size must be 50 or less. For example, if you set this parameter to 5000 IOPS, the home directory size must be at least 100 GB.

Database:

Parameter label (name)	Default	Description
The database engine to deploy with (DBEngine)	PostgreSQL	The database engine to use for the application. Choose either PostgreSQL (for Amazon RDS) or Amazon Aurora PostgreSQL.
Database instance class (DBInstanceClass)	db.m4.large	The EC2 instance type for the Amazon RDS database. Only r4 and r5 families are supported for Amazon Aurora.
RDS Provisioned IOPS (DBIops)	1000	The IOPS for database storage. This value is used only when the Database storage type parameter is set to Provisioned

Parameter label (name)	Default	Description
		IOPS. Allowed range is 1,000-30,000. The ratio of IOPS to allocated storage must be between 3 and 10. Not valid for Amazon Aurora.
Master password * (DBMasterUserPassword)	<i>Requires input</i>	The password for the master (postgres) account. Must be at least 8 characters and include 1 uppercase, 1 lowercase, 1 number, and 1 of the following symbols: ! # \$ % { * : [= ,] - _ @ + % &.
Enable RDS Multi-AZ deployment (DBMultiAZ)	True	If True , your database deploys with high availability. When DBEngine equals PostgreSQL , a multi-AZ RDS instance is provisioned. When DBEngine equals Amazon Aurora PostgreSQL , a multi-node Amazon Aurora cluster is provisioned. If high availability isn't a concern, set this parameter to False .
Bitbucket database password * (DBPassword)	<i>Requires input</i>	Database password used by BitBucket. Must be at least 8 characters and include 1 uppercase, 1 lowercase, 1 number, and 1 of the following symbols: ! # \$ % { * : [= ,] - _ @ + % &.
Database storage (DBStorage)	10	The storage size, in GB, to allocate to the database. This value should be 100-6144, if you've selected Provisioned IOPS for the database storage type.
Database storage type (DBStorageType)	General Purpose (SSD)	The database storage type. You can choose General Purpose SSD or Provisioned IOPS SSD. For more information, see the AWS documentation .

Elasticsearch:

Parameter label (name)	Default	Description
Elasticsearch instance type (ElasticsearchInstanceType)	m4.xlarge. elasticsearch	The EC2 instance type for the Amazon Elasticsearch service to run on.
Elasticsearch disk space per node (ElasticsearchNodeVolumeSize)	100	EBS volume size (in gigabytes) per Elasticsearch node.

Networking:

Parameter label (name)	Default	Description
Permitted IP range * (CidrBlock)	<i>Requires input</i>	The CIDR IP range that is permitted to access Bitbucket. We recommend that you set this value to a trusted IP range (for example, to restrict access to your corporate network). If you

Parameter label (name)	Default	Description
		use 0.0.0.0/0, your Bitbucket instances will be open to public internet access.
Make instance internet facing (InternetFacingLoadBalancer)	True	Controls whether the load balancer should be visible to the internet (True) or only within the VPC (False).
SSH Key Pair Name (KeyPairName)	—	The EC2 key pair used by SSH to access the instances.
Existing DNS name (CustomDnsName)	—	Use a custom existing DNS name for your Data Center instance. You must own this domain and configure it to point to the load balancer.
SSL certificate ARN (SSLCertificateARN)	—	The Amazon Resource Name (ARN) of your SSL certificate. The AWS Certificate Manager (ACM) will display the ARN upon certificate creation. If you want to use your own certificate that you generated outside AWS, you must first import it to ACM by following the instructions in the AWS documentation . The import command outputs the ARN.

Advanced (Optional):

Parameter label (name)	Default	Description
Bitbucket properties (BitbucketProperties)	—	A comma-separated list of Bitbucket properties in the format key1=value1, key2=value2, etc. For details, see the Atlassian documentation .
JVM Heap Size Override (JvmHeapOverride)	—	Override the default amount of memory to allocate to the JVM for your instance type - set size in mg or gb (e.g., 1024 mg or 1 gb).
Additional JVM options (JvmSupportOpts)	—	Pass in any additional JVM options to tune the Bitbucket instance.
Create S3 bucket for Elasticsearch snapshots (CreateBucket)	True	Set this parameter to True to create the S3 bucket for Elasticsearch snapshots. Use this parameter in conjunction with the Elasticsearch snapshot S3 bucket name parameter.
Bitbucket primary database (DBMaster)	—	The database ARN of the Amazon RDS instance to replicate. Setting this parameter will bring up Bitbucket as a disaster recovery standby with an Amazon RDS read replica database. This parameter is not supported when using Amazon Aurora.
Database snapshot ID to restore (DBSnapshotId)	—	The Amazon RDS snapshot ID of an existing database backup to restore. This parameter must be used with the Home volume snapshot ID to restore (HomeVolumeSnapshotId) parameter. Leave this value

Parameter label (name)	Default	Description
		blank for a new instance. This parameter is not supported when using Amazon Aurora.
Elasticsearch snapshot S3 bucket name (ESBucketName)	—	The name of a new or existing S3 bucket configured for Elasticsearch snapshots.
Elasticsearch snapshot ID to restore (ESSnapshotId)	—	The ID of a snapshot in the configured Amazon S3 snapshot repository. This parameter must be used with the Elasticsearch snapshot S3 bucket name parameter.
Home volume snapshot ID to restore (HomeVolumeSnapshotId)	—	The Amazon EBS snapshot ID of an existing database backup to restore as the home directory. This parameter must be used with the Database snapshot ID to restore parameter. Leave this value blank for a new instance.
Delete Home on termination (HomeDeleteOnTermination)	True	Keep the default setting of True to delete the home directory when the instance is terminated. If True , you must back up your data before terminating your instance. Set to False to keep the home directory volume upon termination.

AWS Quick Start configuration:

Parameter label (name)	Default	Description
Quick Start S3 Bucket Name (QSS3BucketName)	atlassian-aws	The S3 bucket you have created for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. The bucket name can include numbers, lowercase letters, uppercase letters, and hyphens, but should not start or end with a hyphen.
Quick Start S3 Key Prefix (QSS3KeyPrefix)	quickstart-atlassian-bitbucket/	The S3 key name prefix used to simulate a folder for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. This prefix can include numbers, lowercase letters, uppercase letters, hyphens, and forward slashes.
ASI identifier (ExportPrefix)	ATL-	Each Atlassian Standard Infrastructure (ASI) uses a unique identifier. If you have multiple ASIs within the same AWS Region, use this field to specify where to deploy Bitbucket.

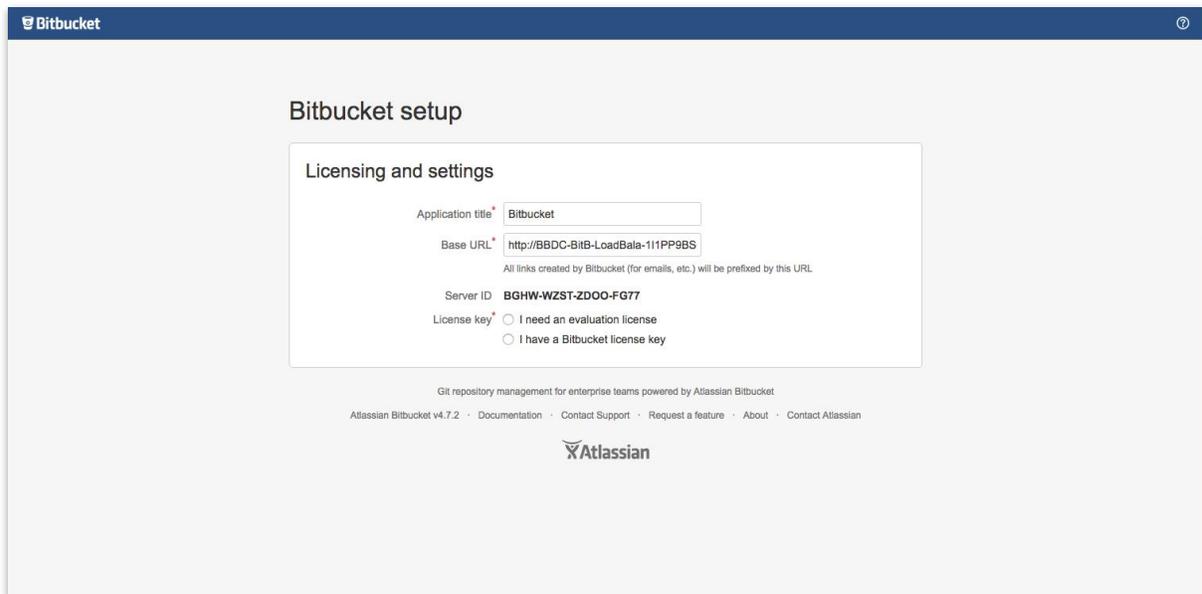
2. On the **Options** page, you can [specify tags](#) (key-value pairs) for resources in your stack and [set advanced options](#). When you're done, choose **Next**.
3. On the **Review** page, review and confirm the template settings. Under **Capabilities**, select the check box to acknowledge that the template will create IAM resources.
4. Choose **Create** to deploy the stack.

5. Monitor the status of the stack. When the status is **CREATE_COMPLETE**, the deployment is complete.
6. You can use the URL displayed in the **Outputs** tab for the stack to view the resources that were created.

Step 4. Configure Bitbucket Data Center

When you launch the Quick Start, it deploys a single Bitbucket node (Auto Scaling group of min=1 and max=2 by default).

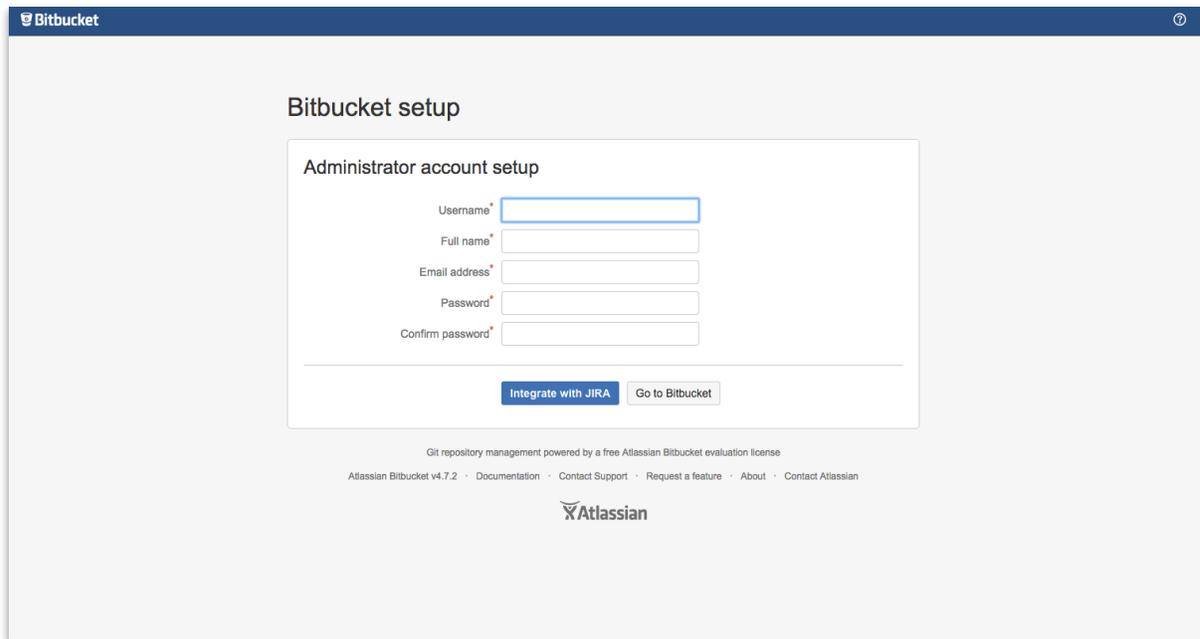
1. Choose the URL displayed in the **Outputs** tab to go to the Bitbucket setup screen.
2. On the **Licensing and Settings** page, enter a title for your Bitbucket deployment. Leave the base URL unchanged, and choose the appropriate licensing option. If you don't have a valid license for Bitbucket Data Center, sign up for an evaluation license.



The screenshot shows the Bitbucket setup interface. At the top, there is a dark blue header with the Bitbucket logo and a help icon. Below the header, the main content area is titled "Bitbucket setup". Inside this area, there is a white box titled "Licensing and settings". This box contains several input fields and radio buttons: "Application title" with the value "Bitbucket", "Base URL" with the value "http://BBDC-BitB-LoadBala-111PP9BS", and "Server ID" with the value "BGHW-WZST-ZDOO-FG77". Below these fields, there are two radio buttons for "License key": "I need an evaluation license" (which is selected) and "I have a Bitbucket license key". At the bottom of the white box, there is a small note: "All links created by Bitbucket (for emails, etc.) will be prefixed by this URL". Below the white box, there is a footer with the text "Git repository management for enterprise teams powered by Atlassian Bitbucket" and a list of links: "Atlassian Bitbucket v4.7.2", "Documentation", "Contact Support", "Request a feature", "About", and "Contact Atlassian". The Atlassian logo is centered at the bottom of the page.

Figure 2: Bitbucket licensing and settings

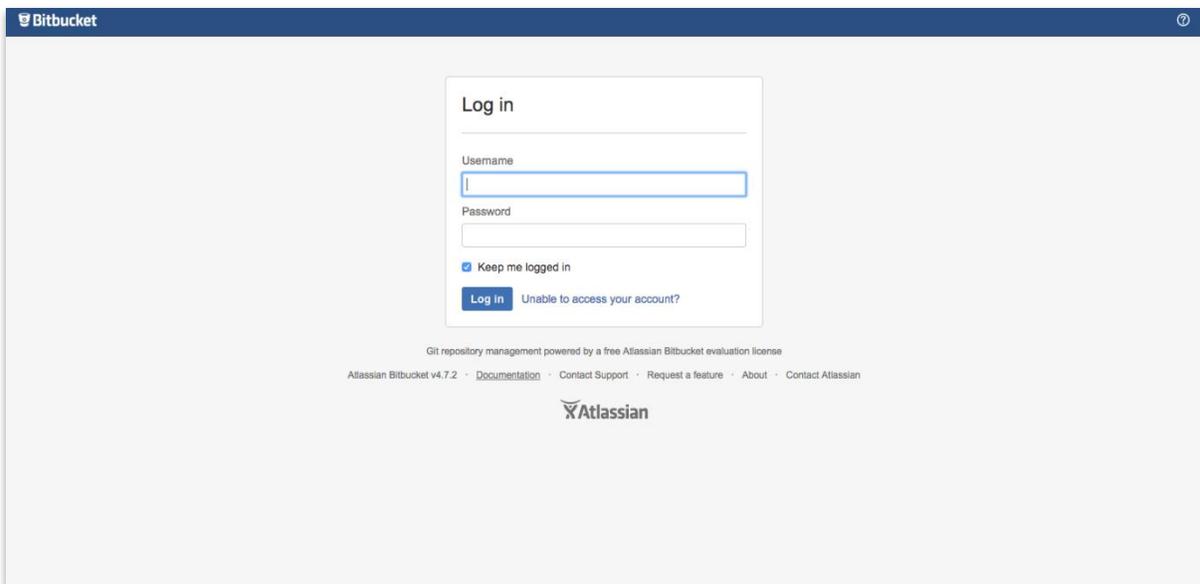
3. To set up Bitbucket Data Center, you need to create an Administrator account and password. The Administrator account has full access to all data in Bitbucket, so we highly recommend that you choose a strong password for this account. Enter the Administrator's user details in the setup screen illustrated in Figure 3, and choose **Go to Bitbucket**.



The screenshot shows the Bitbucket setup interface. At the top, there is a blue header with the Bitbucket logo and a help icon. Below the header, the main content area is titled "Bitbucket setup". Underneath, there is a section titled "Administrator account setup" which contains a form with five input fields: "Username", "Full name", "Email address", "Password", and "Confirm password". Below the form are two buttons: "Integrate with JIRA" (in blue) and "Go to Bitbucket" (in grey). At the bottom of the page, there is a small line of text: "Git repository management powered by a free Atlassian Bitbucket evaluation license", followed by a footer with links for "Atlassian Bitbucket v4.7.2", "Documentation", "Contact Support", "Request a feature", "About", and "Contact Atlassian". The Atlassian logo is centered at the bottom.

Figure 3: Bitbucket setup screen

4. Log in with the user name and credentials you created in the previous step.



The screenshot shows the Bitbucket login interface. At the top, there is a blue header with the Bitbucket logo and a help icon. Below the header, the main content area is titled "Log in". Underneath, there is a form with three input fields: "Username", "Password", and a checkbox labeled "Keep me logged in". Below the form are two buttons: "Log in" (in blue) and "Unable to access your account?" (in grey). At the bottom of the page, there is a small line of text: "Git repository management powered by a free Atlassian Bitbucket evaluation license", followed by a footer with links for "Atlassian Bitbucket v4.7.2", "Documentation", "Contact Support", "Request a feature", "About", and "Contact Atlassian". The Atlassian logo is centered at the bottom.

Figure 4: Logging in to Bitbucket Data Center

5. Choose **Settings** (the gear icon in the upper right), and then choose **Clustering**. You should see a page similar to Figure 5, which shows that the node is ready for clustering.

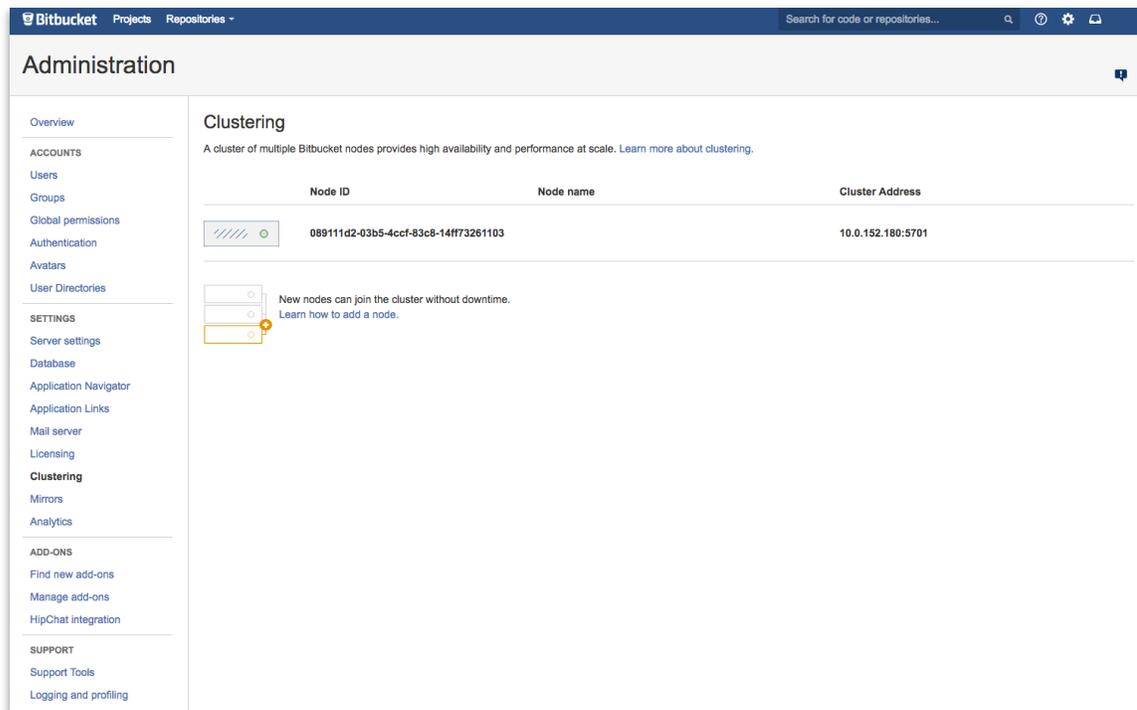


Figure 5: Standalone node is ready for clustering

Your Bitbucket Data Center deployment is now in a state where you can add nodes that will automatically cluster with your existing node.

Step 5: Add nodes to the cluster

1. Sign in to the AWS Management Console, use the region selector in the navigation bar to choose the AWS Region for your deployment, and open the AWS CloudFormation console at <https://console.aws.amazon.com/cloudformation/>.
2. Choose the Bitbucket Data Center template. From the **Actions** list, choose **Update Stack**.
3. On the **Select Template** page, leave **Use current template** selected, and then choose **Next**.
4. On the **Specify Details** page, in the **Cluster nodes** section of **Parameters**, enter your desired number of cluster nodes in **Minimum number of cluster nodes** and **Maximum number of cluster nodes**. Then, click through to update the stack. This will set a static number of nodes in your cluster.
5. After the stack has finished updating, confirm that the additional nodes have formed a cluster by viewing the **Clustering** page in Bitbucket Data Center (Figure 6).

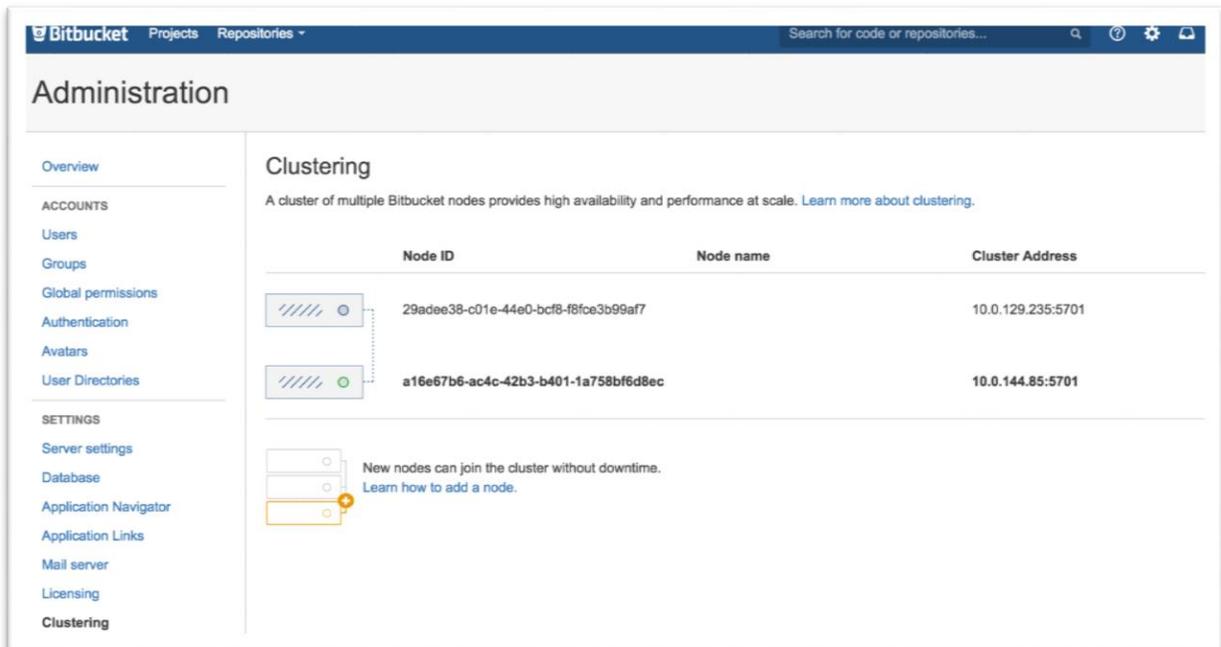


Figure 6: Two nodes in a clustered configuration

Your Bitbucket Data Center deployment is now ready to use.

Note: You can also migrate Bitbucket data from an existing deployment to this one. For more information, see the [migration guides](#) on the Atlassian website.

FAQ

Q. I encountered a CREATE_FAILED error when I launched the Quick Start.

A. If AWS CloudFormation fails to create the stack, we recommend that you relaunch the template with **Rollback on failure** set to **No**. (This setting is under **Advanced** in the AWS CloudFormation console, **Options** page.) With this setting, the stack's state will be retained and the instance will be left running, so you can troubleshoot the issue. (You'll want to look at the log files in `/var/log/at1.log` and `/var/log/cfn-init.log`.)

Important: When you set **Rollback on failure** to **No**, you'll continue to incur AWS charges for this stack. Please make sure to delete the stack when you've finished troubleshooting.

For additional information, see [Troubleshooting AWS CloudFormation](#) on the AWS website.

Q. I encountered a size limitation error when I deployed the AWS CloudFormation templates.

A. We recommend that you launch the Quick Start templates from the links in this guide or from another S3 bucket. If you deploy the templates from a local copy on your computer or from a non-S3 location, you might encounter template size limitations when you create the stack. For more information about AWS CloudFormation quotas, see the [AWS documentation](#).

Additional resources

AWS services

- AWS CloudFormation
<http://aws.amazon.com/documentation/cloudformation/>
- Amazon EBS
<http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonEBS.html>
- Amazon EC2
<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html>
- Amazon VPC
<http://aws.amazon.com/documentation/vpc/>

Atlassian documentation

- Bitbucket Data Center
<https://confluence.atlassian.com/display/BitbucketServer/Clustering+with+Bitbucket+Data+Center>

Quick Start reference deployments

- Atlassian Standard Infrastructure (ASI) on the AWS Cloud
<https://fwd.aws/xYyYy>
- Jira on the AWS Cloud: Quick Start Reference Deployment
<https://fwd.aws/Wz3Qb>
- Confluence Data Center on the AWS Cloud: Quick Start Reference Deployment
<https://fwd.aws/kBpWN>
- AWS Quick Start home page
<https://aws.amazon.com/quickstart/>

Git repository

You can visit our [GitHub repository](#) to download the templates and scripts for this Quick Start, to post your comments, and to share your customizations with others.

Document revisions

Date	Change	In sections
April 2020	Updated the descriptions for parameters DBMultiAZ and DBPassword	Step 3. Review the parameters and create your stack
March 2020	Added Elasticsearch ElasticsearchNodeVolumeSize parameter	Architecture ; Step 3. Review the parameters and create your stack
February 2020	Added Amazon CloudWatch information, added ExportPrefix parameter	Architecture ; Step 3. Review the parameters and create your stack
October 2019	Added basic Amazon CloudWatch monitoring and logging; added advanced deployment customizations via Ansible	Architecture ; Step 3. Review the parameters and create your stack
September 2019	Updated architecture diagram to replace Elastic Load Balancing with Application Load Balancer	Architecture
August 2019	Added Auto Scaling Groups section	Architecture
February 2019	Rearchitected to deploy into an ASI	Changes in templates and throughout guide
August 2017	Added bastion host functionality and associated parameters	Architecture diagram ; Step 3. Review the parameters and create your stack
October 2016	Initial publication	–

© 2020, Amazon Web Services, Inc. or its affiliates, and Atlassian. All rights reserved.

Notices

This document is provided for informational purposes only. It represents AWS's current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

The software included with this paper is licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with the License. A copy of the License is located at <http://aws.amazon.com/apache2.0/> or in the "license" file accompanying this file. This code is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.