

General Web App Pattern

Most web applications can follow the same basic pattern, with the only difference being the middleware on the web/app instances. A combination of subnets and security groups are used to provide security and separation. The pattern can be scaled down to one availability zone.

Elastic Load Balancers

The Elastic Load Balancers take in traffic from the outside, either the public internet or through direct connect/VPN. The ELBs are the only devices routable from the internet, to provide the separation as required by the Harvard security policies.

Application Instances

The application instances combine web and application functions, allowing for a number of different supported middleware platforms:

- Java/Tomcat
- Java/WebLogic
- Python/Gunicorn
- Apache/PHP
- Windows/IIS

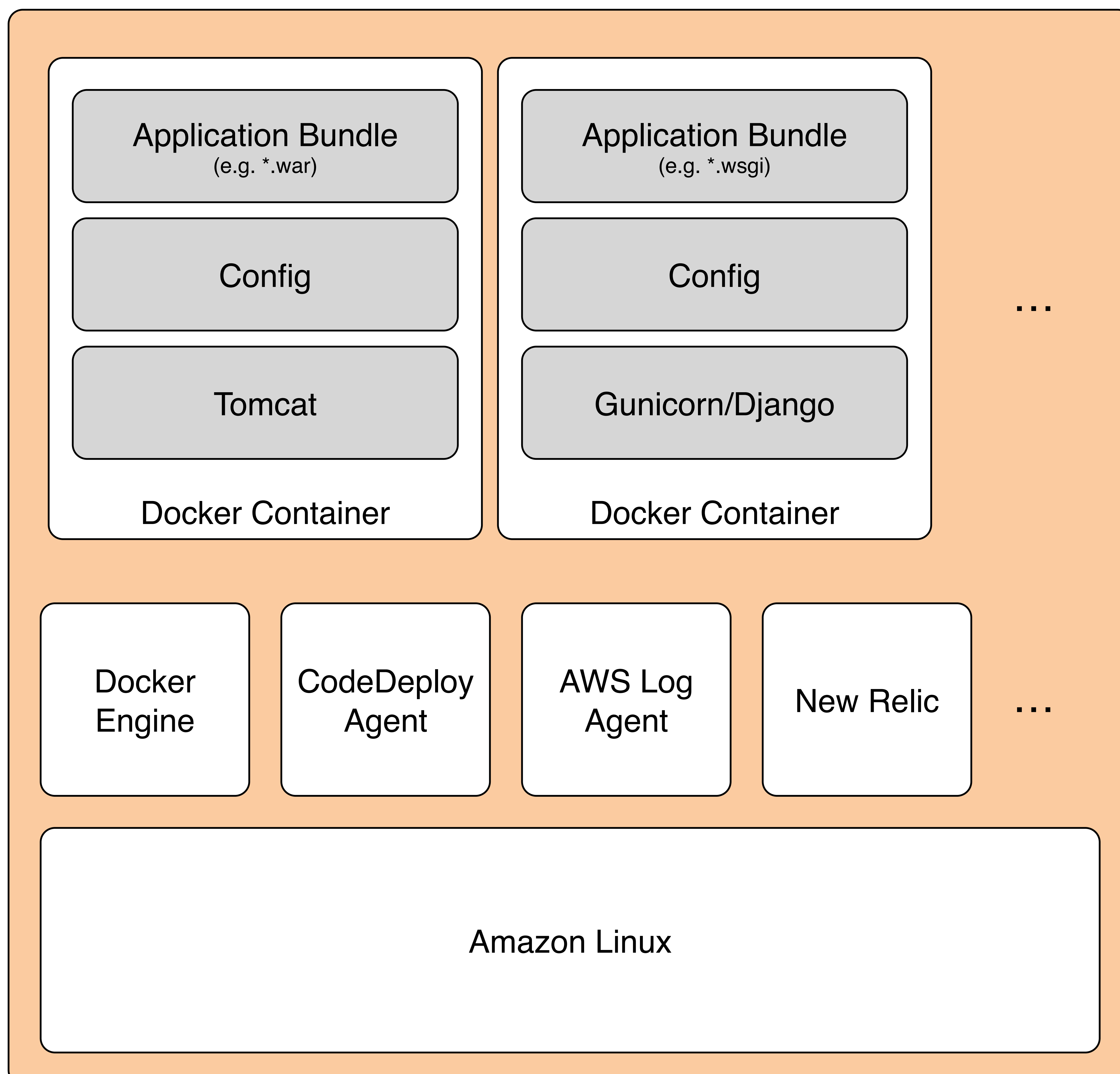
The standard patterns will provide a mechanism to isolate the middleware specific configurations on the instances, to allow for a single and standardized base pattern.

The instances are always placed into Auto Scaling Groups, even if there is only one instance, to provide ease of management and recovery.

RDS Instances

All applications do not need databases (or uses a shared one), but for those that do, the preferred option is to use RDS, allowing for a number of RDBMS products:

- Oracle
- MySQL
- Postgres
- SQL Server



General Web App Instance

The web app instances will have the same basic configuration to allow for uniform management and to reduce the complexity for Cloud Ops. The first instance is Linux based but there will be a corresponding instance type for Windows based systems.

Applications in Docker Containers

All the application specific artifacts and supporting middleware is in docker containers, allowing for easy separation and configuration. There will be specific containers for:

- Java/Tomcat
- Java/WebLogic
- Python/Uunicorn
- Apache/PHP
- Windows/IIS

Instance Tools

There will be a standard set of tools installed on every instance to allow for uniform management and a common set of functionality for running the instances. This set of tools might change over time, and by using standard patterns we can easily do full replacements of toolsets. Some currently relevant tools:

- *Docker Engine*. The base software for running docker containers.
- *CodeDeploy Agent*. Allows for automated blue-green deployments.
- *AWS Log Agent*. Allows for real time monitoring of logs in AWS CloudTrail, reducing the need for people to login to instances.
- *New Relic*. Performance monitoring and analytics tool.

Operating System

The Linux based systems will use Amazon Linux as the standard operating systems.

The Windows based systems will use a Windows version that supports Docker (will be released this year).