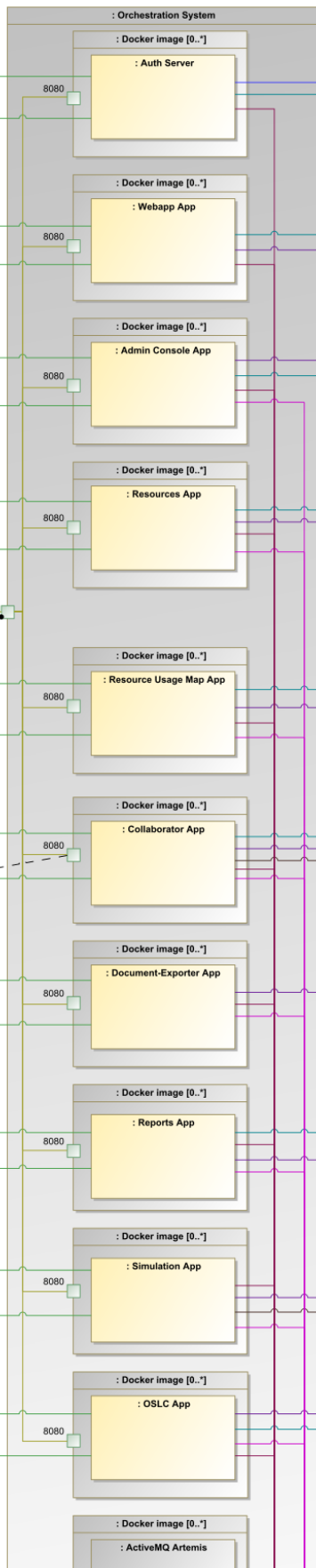# Enterprise deployment

Web Application Platform as well as all of its plugins are provided as separate web applications. Therefore, it is possible to containerize applications and this way achieve better performance and scalability.

To learn how to containerize Web Application Platform applications using Docker, see the following topics:
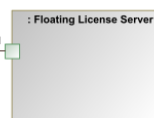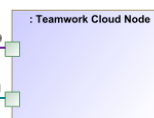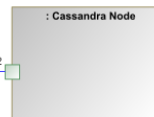
- Building Teamwork Cloud containers
- Building Web Application Platform containers
- Deployment example for Web Application Platform
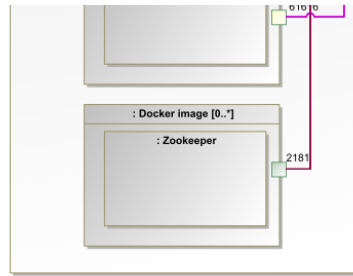
ibd [Block] WAP enterprise deployment[WAP enterprise deployment]

Legend
- Internal network - HTTP
- TWC Java API - AKKA
- TWC REST API - HTTP
- Flex protocol - TCP/IP
- Web browser communicaiton - HTTP
- Service discovery - TCP/IP
- Internal WEBAPP comunication - HTTP
- Cassandra protocol - TCP/IP
- ActiveMQ Artemis - TCP/IP

Internal communication of applications via HTTP requests. Communication port depends on Proxy server configuration.

Sticky sessions via cookie.

: Orchestration System

: Docker image [0..*]
: Auth Server
8080

: Docker image [0..*]
: Webapp App
8080

: Docker image [0..*]
: Admin Console App
8080

: Docker image [0..*]
: Resources App
8080

: Docker image [0..*]
: Resource Usage Map App
8080

: Docker image [0..*]
: Collaborator App
8080

: Docker image [0..*]
: Document-Exporter App
8080

: Docker image [0..*]
: Reports App
8080

: Docker image [0..*]
: Simulation App
8080

: Docker image [0..*]
: OSLC App
8080

: Docker image [0..*]
: ActiveMQ Artemis

: Web Browser

: Proxy
80
443
80

Can be configured to 443 and 8443 accordingly for any App depending on the need to secure internal communication.

: Cassandra Node
9042

: Teamwork Cloud Node
3579
8111

: Floating License Server
1101

: Docker image [0..*]

: Zookeeper

2181

The enterprise deployment option for Web Application Platform and its services.