# Product Installation

## Installation Instructions

Installation instructions are found in the Aerie repository deployment documentation. If you have any questions or issues, don't hesitate to ask on #mpsa-aerie-users.

## Docker Containers

Goto the Artifactory Aerie Docker repository and log in with your JPL credentials. The latest released containers are:

```
docker-release-local/gov/nasa/jpl/aerie/adaptation/release-0.4.0
docker-release-local/gov/nasa/jpl/aerie/plan/release-0.4.0
docker-release-local/gov/nasa/jpl/aerie-apollo/release-0.4.0
docker-release-local/gov/nasa/jpl/aerie-ui/release-0.4.0
```

## Example Docker-Compose

An example Docker Compose file is available for deployment. You can use these instructions to help you deploy.

```
## TARs

If you just want the Aerie JAR files you can find them at:
```

general/gov/nasa/jpl/aerie/aerie-release-0.4.0.tar.gz

```
The Aerie Editor (Falcon) can be found at:
```

general/gov/nasa/jpl/aerie/aerie-editor-release-0.4.0.tar.gz "`

## Known Issues

1. When using the IntelliJ IDE, upon a source file change, only the affected source files will be recompiled. This causes conflicts with the annotations processing being used for Activity Mapping. For now manually rebuilding every time is the solution.

# System Requirements

## Software Requirements

| Name | Version |
| --- | --- |
| DOCKER | 19.X |
| NODEJS | 12.X LTS |
| NPM | 6.X |
| OPEN JDK | 11.X |

*For build purposes only. Not needed for installing the application.

## Supported Browsers

| Name | Version |
| --- | --- |
| CHROME | LATEST |
| FIREFOX | LATEST |

## Hardware Requirements

| Hardware | Details |
| --- | --- |
| CPU | 2 GIGAHERTZ (GHZ) FREQUENCY OR ABOVE |
| RAM | 4 GB AT MINIMUM |
| DISPLAY RESOLUTION | 2560-BY-1600, RECOMMENDED |
| INTERNET CONNECTION | HIGH-SPEED CONNECTION, AT LEAST 10MBPS |

## TCP Port Requirements

| Service | Port |
| --- | --- |
| Aerie UI | 8080 |
| Adapataion | 27182 |
| Plan | 27183 |
| RabittMQ | 15762 |
| Aerie Apollo | 27184 |

# Administration

This product is using Docker containers to run the application. There are total of five Docker containers that are internally bridged (connected) to run the application. Containers can be restarted in case of any issues using Docker CLI. Only port 8080 from the UI container is exposed to outside.

## Environment Variables

Aerie software does not have any environment variables at this point in time.

## Network Communications

The Aerie deployment configures the port numbers for each container via docker-compose. The port numbers must match those declared within the services' config.json. In a large majority of Aerie deployments no change to these port numbers will be needed, nor should one be made. The only port number that might be desired to change is the Aerie-UI port (8080). in this case the number to change is the first port number of the pair [XXXX:XXXX]. The second number represents the port number within the container itself. An example of this would be ports: ["8080:80"]. The number that needs to be changed is the first port which is 8080.

## Environment Variables

| Port Type | Default Port Number | Description |
|-----------|---------------------|-------------|
| TCP       | 8080                | UI Port     |

## Administration Procedures

Aerie is orchestrated as a set of Docker containers. Each of the software components are packaged and run in an isolated docker container independently from one another. There exists seven docker containers: - Aerie-UI: Hosts the web application and communicates with Aerie via the GraphQL Apollo Server. - Aerie-Apollo: The GraphQL Apollo Server which functions as the Aerie API Gateway against which clients can submit GraphQL queries and mutations. - Adaptation: Handles all the logic and functionality for the model Adaptation for activity planning. - Adaptation-mongo: Holds the data for Adaptation container. - Plan: Handles all the logic and functionality for activity planning. - Plan-mongo: Holds the data for Plan container. - RabbitMQ: The Aerie services' message bus message exchange instance.

The Adaptation, Plan, and Apollo servers communicate to each other via a REST API (internal to Aerie) with the ports specified in the docker-compose file. The database containers, Adaptation-mongo and Plan-mongo are isolated to connect only with their respective service container (Adaptation or Plan).

# Product Support

## Defect Reporting Procedure

All defect reports should go to aerie_support@jpl.nasa.gov.

## Points of Contact

- Adaptation: Kenneally, Patrick W, Development Lead
- Administration: Kenneally, Patrick W, Development Lead
- General Help: Alper Ramaswamy, Emine Basak, Product Lead
- #mpsa-aerie-users: User help Slack channel