



Jonathan
Flightbase

조나단 플라이트베이스™ v2.0

멀티 GPU 기반 머신러닝 플랫폼

(Jonathan Flightbase™ v2.0 – Machine Learning Platform with Multiple GPUs)

사용 매뉴얼

Jonathan Flightbase™ v2.0 사용 매뉴얼

발행	행	주식회사 아크릴
주소	소	서울시 강남구 선릉로 704, 청담벤처프라자 11층
전화	화	02) 557-4958
팩스	스	02) 558-4958
고객지원		support@iacryl.com



© 2020. ACRYL Inc. All rights reserved.

The contents of this document are all rights reserved by ACRYL Inc.

A license is hereby granted to download and print a copy of this document for personal use only.

No other license to any other intellectual property rights is granted herein.

Unless expressly permitted herein, reproduction, transfer, distribution of storage of part or all of the contents in any form without the prior written permission of ACRYL Inc. is prohibited.

The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without notice.

No liability will be accepted by the publisher for any consequence of its use.

Publication thereof does not convey nor imply any license under patent or other industrial or intellectual property rights..

목차

1. Jonathan Flightbase™ 설치 요구 사항	- 4 -
2. Jonathan Flightbase™ 설치 가이드	- 6 -
3. 단계별 이용 가이드 – 관리자 : 사용자 작업 공간 생성	- 8 -
4. 단계별 이용 가이드 – 사용자 : 학습 실행	- 17 -
5. 알려진 문제에 대한 해결 방안	- 28 -
6. Jonathan Flightbase™ 유지 보수	- 32 -

1. Jonathan Flightbase™ 설치 요구 사항

1-A. 하드웨어 (HW) 요구 사항

Jonathan Flightbase™(이하 Flightbase)는 1개 이상의 서버에 설치될 수 있으며, 분산 학습을 위해서는 2개 이상의 NVIDIA GPU가 장착되어야 합니다. 아래는 Flightbase를 설치하기 위한 서버의 최소 요구 사항입니다:

no.	HW 명칭	요구 사양	역할
1	NVIDIA GPU	CUDA 지원	- GPU의 다중 코어를 이용한 프로그래밍을 지원하는 GPU 모델 (GPGPU)
2	10Gbps Ethernet Network Card (NIC)	10Gbps Dual-Ports	- 분산 학습 시 발생하는 서버 간의 bottleneck으로 인한 학습 성능 저하 해결
3	데이터 저장용 디스크 (HDD or SSD)	500GB 이상	- 학습 데이터를 저장하기 위한 저장 장치
4	Ethernet Switch (3대 이상의 서버 연결 시 필수)	1 Gbps 지원	- Master Server와 Worker 서버 사이의 명령어 전달 - Client에서 NAT 없이 IP만으로 Master Server 와 Worker Server에 접근 가능한 환경 필요
		10 Gbps 지원	- Worker Server와 다른 Worker Server 사이의 분산 학습 bottleneck을 줄이기 위한 통신 장비

Flightbase는 추가적으로 GPU 간의 통신 가속을 위한 장비가 장착되어 있을 경우, 이를 최대한 활용하여 더욱 효율적인 학습을 가능하도록 해줍니다. 아래는 이를 지원하는 장비 요구 사항입니다:

no.	HW 명칭	요구 사양	역할
1	NVIDIA NVLink	NVLink Bridge / Switch (DGX)	- NVLink는 단일 서버 내에서 GPU 간의 Peer-to-Peer (P2P) 가속 통신 지원
2	Mellanox InfiniBand Connect-X VPI	40 GbE 이상	- 서버 간 InfiniBand로 연결하고, GPUDirect RDMA 기능을 활성화하여 서버 간의 GPU 가속 통신을 가능하게 함
3	Mellanox InfiniBand Switch	Connect-X VPI 지원	- 3대 이상의 서버 간의 InfiniBand 연결을 위한 장비

1-B. 소프트웨어 (SW) 요구 사항

Flightbase의 설치를 위해서는 다음과 같은 환경이 필요합니다:

no.	SW 명칭	요구 사양	역할
1	OS	Ubuntu 16.04	<ul style="list-style-type: none"> -클러스터를 구성하고 있는 개별 서버의 운영체제 -Docker 설치 시 시스템 root 권한이 필요합니다.
2	NVIDIA GPU Driver	430.40 이상	-CUDA를 실행하기 위한 NVIDIA GPU Driver 버전
3	Firewall	해제	<ul style="list-style-type: none"> -설치 시 필요 패키지 다운로드를 위한 방화벽 해제 -분산 학습을 위한 서버간 통신을 위해 필요

2. Jonathan Flightbase™ 설치 가이드

2-A. Master, Worker 설치

Step 1) 설치 파일을 이용하여 각 서버에 설치 실행

→ Master

```
Shell
[COMMAND]
python3 jf_installer.py --master_ip [MASTER IP] --flag MASTER
Option: --ib_ip [InfiniBand IP] --front_port [Front Port] --api_port [API Port]
```

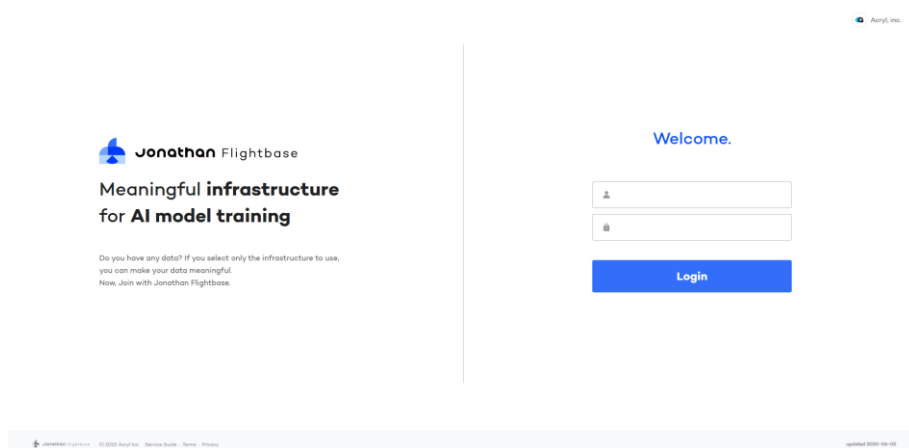
→ Worker

```
Shell
[COMMAND]
python3 jf_installer.py --master_ip [MASTER IP] --flag WORKER
Option: --ib_ip [InfiniBand IP]
```

→ InfiniBand (IB) IP 미입력시 IB 사용이 불가능하며, Front Port는 443, API Port는 56789가 기본값입니다.

Step 2) 설치 완료 후, 클라이언트 웹페이지 접속하여 관리자 계정으로 로그인

→ 주소 : [설치 Master IP]:[Front Port] 에 접속 (예: https://192.168.1.18:443)



→ 아래의 관리자 계정으로 로그인 (로그인 후 Users 메뉴에서 비밀번호 변경 가능)

ID	Password
root	Master 서버의 root 계정 비밀번호

2-B. Master, Worker 설치 제거

Step 1) Uninstall 파일을 이용하여 각 서버에서 실행

→ Master

```
Shell  
[COMMAND]  
python3 jf_uninstaller.py --flag MASTER
```

→ Worker

```
Shell  
[COMMAND]  
python3 jf_uninstaller.py --flag WORKER
```

2-C. 재부팅 시 Flightbase 재시동

Step 1) 재부팅된 서버에서 API를 재시작 합니다.

→ Master

```
Shell  
[COMMAND]  
python3 start_api.py --master_ip [MASTER IP] --flag MASTER  
Option: --ib_ip [InfiniBand IP]
```

→ Worker

```
Shell  
[COMMAND]  
python3 start_api.py --master_ip [MASTER IP] --flag WORKER  
Option: --ib_ip [InfiniBand IP]
```

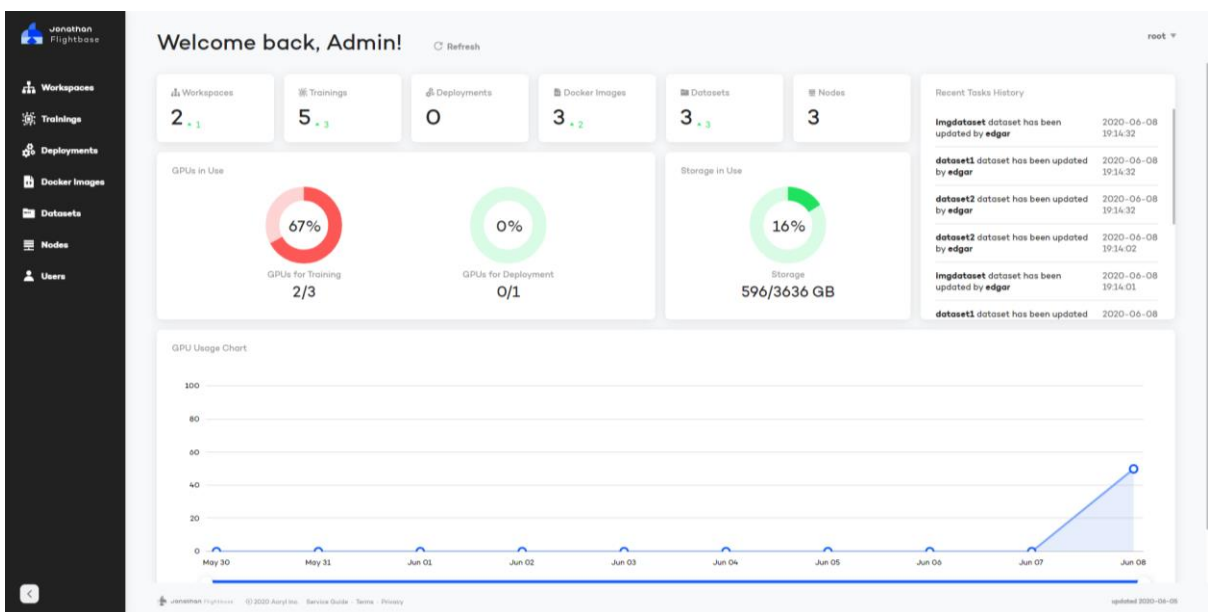
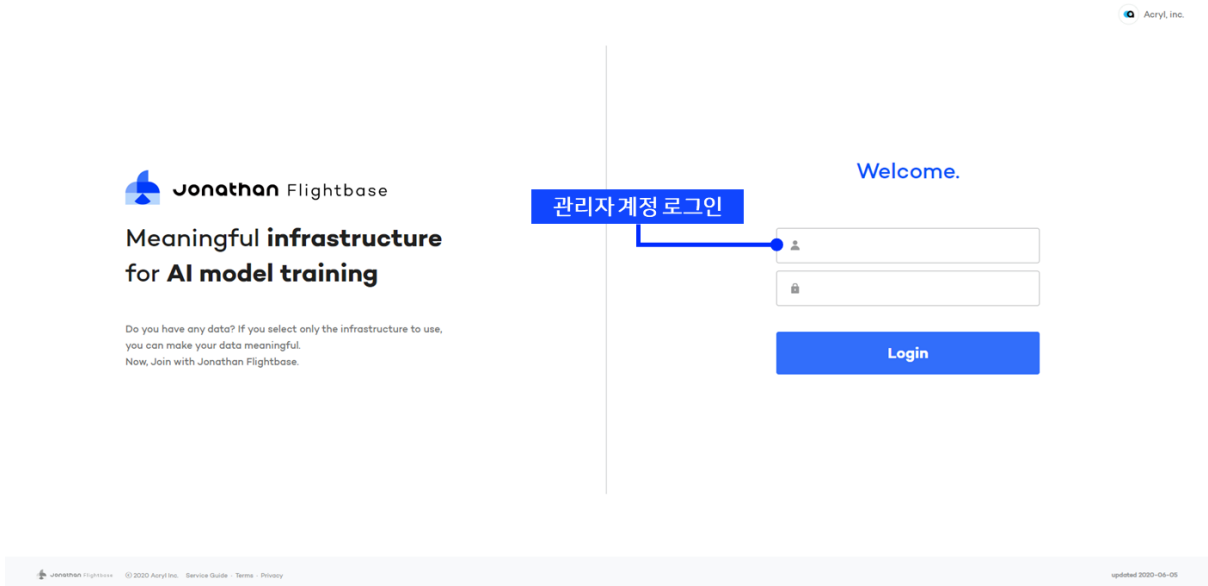
* 설치 시 유의 사항

- 설치 시간은 약 40 분 정도 소요됩니다. 단, 인터넷 환경 및 서버 사양에 따라 달라질 수 있습니다.

* 이외의 문제 발생 시 support@iacryl.com 으로 문의주시기 바랍니다.

3. 단계별 이용 가이드 – 관리자 : 사용자 작업 공간 생성

Step 1) 관리자 계정 로그인 후, Dashboard에서 플랫폼 현황 확인



Step 2) 네트워크 구성 및 Worker Server 추가를 위해 Nodes 메뉴 선택

The screenshot shows the 'Nodes' management page in JupyterLab. The table contains the following data:

Status	Node Name	IP Address	Network Interface	GPU Configuration(s)	GPU in Use	Connection
Attached	jf-node-04	192.168.114	10Gbps Ethernet (enp4s0f0)	GeForce GTX 1080 Ti x 2ea	0/2 GPUs	[Toggle]
Attached	jf-node-03	192.168.113	10Gbps Ethernet (enp4s0f0)	GeForce GTX 1080 Ti x 2ea	0/2 GPUs	[Toggle]
Attached	jf-node-02	192.168.112	10Gbps Ethernet (enp3s0f1)	GeForce GTX 1080 Ti x 2ea	1/2 GPUs	[Toggle]

→ Master 서버의 네트워크 정보 입력을 위해, Master 서버의 정보 수정을 선택합니다.

The 'Edit Node' dialog box displays the following configuration:

- IP: 192.168.113
- Network Interface Settings:
 - Ethernet - 1Gbps: enp4s0f1 (selected)
 - Ethernet - 10Gbps: enp4s0f0
 - InfiniBand - Optional: None

→ Network Interface Settings 목록에서 해당 Interface 항목을 선택하여 설정합니다.

Step 3) "Add Node"를 선택하여 Worker Node의 정보 입력 후 Add 선택

Add Node

IP
You can enter the IP address of the node you want to add. Get Info

Network Interface Settings

Ethernet - 1Gbps
You can choose the Interface name

Ethernet - 10Gbps
You can choose the Interface name

InfiniBand - Optional
You can choose the Interface name

Cancel Add

연결할 서버의 IP 입력

서버의 네트워크 정보 불러오기

서버의 Network 정보 선택

→ JF-Worker가 설치된 서버의 IP를 입력하고, Network 정보를 선택합니다.

Step 4) 생성된 Node를 연결하기 위해 Connection 실행

Nodes
Dashboard > Nodes

3 nodes

Status	Node Name	IP Address	Network Interface	GPU Configuration(s)	GPU in Use	Connection
Attached	jf-node-04	192.168.114	10Gbps Ethernet (enp4s0F0)	GeForce GTX 1080 Ti x 2ea	0/2 GPUs	<input type="checkbox"/>
Attached	jf-node-03	192.168.113	10Gbps Ethernet (enp4s0F0)	GeForce GTX 1080 Ti x 2ea	0/2 GPUs	<input type="checkbox"/>
Attached	jf-node-02	192.168.112	10Gbps Ethernet (enp3s0F1)	GeForce GTX 1080 Ti x 2ea	1/2 GPUs	<input type="checkbox"/>

Add Node Delete

Rows per page: 10 | 1-3 of | < >

Node 연결/연결 해제

→ Connection 실행 시 Node를 연결하는 작업을 하며, 완료 후 클러스터링 작업 (Attaching)을 실행합니다. 이 작업에는 최대 5분정도의 시간이 소요될 수 있습니다.

Step 5) 사용자 계정 생성을 위해 Users 메뉴에서 Add User를 선택

The screenshot shows the 'Users' management page in Flightbase. A table lists existing users with columns for User Type, User ID, Updated, and Registered. Below the table, the 'Add User' button is highlighted with a blue callout box containing the text '사용자 계정 추가'.

User Type	User ID	Updated	Registered
Workspace Manager	edgar	2020-06-08 11:32:56	2020-06-08 11:32:56
Training Owner	ttotest	2020-06-08 11:32:20	2020-06-08 11:32:20
User	tempo	2020-06-08 10:55:47	2020-06-08 10:55:47
Workspace Manager	hi	2020-06-05 18:25:49	2020-06-05 18:25:49
User	demo	2020-06-05 18:24:23	2020-06-05 18:24:23
Admin	root	2020-06-05 18:24:21	2020-06-05 18:24:21

Step 6) 사용자 ID와 Password를 입력 후 Create 선택하여 계정 생성

The screenshot shows the 'Add User' form with three input fields. Blue callout boxes point to each field with the following text:

- 생성할 User ID 입력** (Input field: "You can enter the user ID you want to add.")
- Password 입력** (Input field: "You can enter password of the user.")
- Password 재입력** (Input field: "Please enter the same password again.")

Buttons for 'Cancel' and 'Add' are visible at the bottom right of the form.

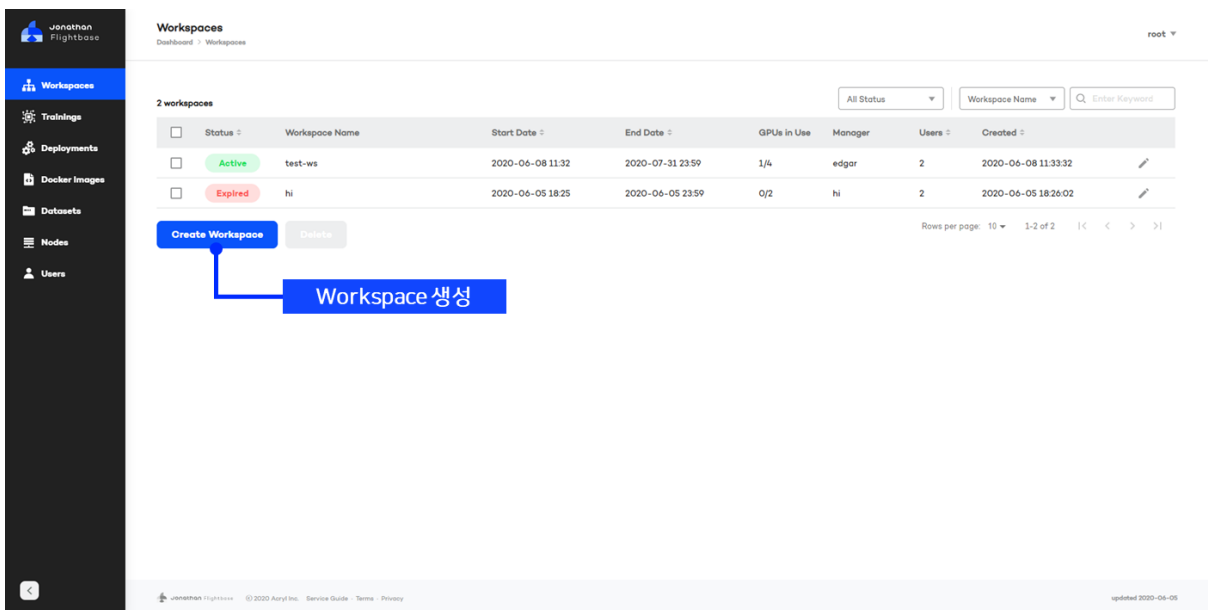
→ 관리자는 사용자 계정을 생성하여 부여함으로써 사용자가 Flightbase를 사용하도록 할 수 있습니다.

→ 사용자의 비밀번호 관리는 인프라 및 플랫폼 전체를 관리하는 관리자만 가능하며, 비밀번호 변경 시 관리자에게 요청해야 합니다.

→ 사용자 계정 ID와 비밀번호는 다음의 규칙에 따라 생성할 수 있습니다.

항목	조건
사용자 계정 ID	- 3자 이상 20자 이하 영어 소문자, 숫자, 하이픈 (-) 가능 단, 하이픈은 연속될 수 없으며, 이름의 처음과 끝에 올 수 없습니다.
암호	- 알파벳 대소문자, 숫자, 특수문자 (~!@#%\$%^&* <>?) 조합으로 8자리 이상 20자리 이하

Step 7) Workspace를 생성하기 위해, 왼쪽 메뉴에서 “Workspaces” 선택



→ 사용자에게 GPU 인프라 제공을 위한 작업 공간인 Workspace를 생성합니다.

Step 8) "Create Workspace"를 선택하여 새 Workspace를 생성

생성할 Workspace의 이름 입력

Workspace 설명 입력

사용기간 설정

학습 GPU 수 입력

관리자 배정

사용자 배정

→ Workspace의 이름과 설명을 작성하고, 사용 기간을 입력합니다.

→ 학습을 위한 GPU의 수를 입력하고, Workspace의 운영자와 사용자를 배정합니다.

→ GPU 수는 Workspace 사용 기간 동안 점유되며, 기간이 만료되는 경우 Workspace의 정보 확인은 가능하나 GPU가 점유되지 않으며 관련 기능이 동작하지 않습니다.

Step 9) 생성된 Workspace의 상세 정보 확인

Workspace 세부 정보

GPU 사용 현황

Workspace 관련 정보

Status	Workspace Name	Start Date	End Date	GPUs in Use	Manager	Users	Created
Active	test-ws	2020-06-08 11:32	2020-07-31 23:59	0/4	edgar	2	2020-06-08 11:33:32

test-ws's details

Workspace for Test

GPUs for Training: 0/3 (0%)

GPUs for Deployment: 0/1 (0%)

Users: ttatst, edgar

Trainings: 3 trainings

Deployments: 3 deployments

Docker Images: 3 docker images

Datasets: 3 datasets

Step 10) 플랫폼을 통해 사용자가 업로드한 Datasets 관리

The screenshot displays the 'Datasets' management page in the Jonathon Flightbase dashboard. The left sidebar contains navigation options: Workspaces, Trainings, Deployments, Docker Images, Datasets (selected), Nodes, and Users. The main content area shows a table of datasets with columns for Dataset Name, Files, Workspace, Creator, Updated, and Created. Three datasets are listed: 'imgdataset' (4 files), 'dataset2' (2 files), and 'dataset1' (1 files). Below the table, there are buttons for 'Create Dataset' and 'Delete'. Annotations with blue boxes and arrows point to these elements: 'Dataset 업로드' points to the 'Create Dataset' button, '업로드한 Datasets 목록' points to the dataset table, and 'Dataset 수정' points to the edit icon in the first row of the table.

Dataset Name	Files	Workspace	Creator	Updated	Created
imgdataset	4 files	test-ws	edgar	2020-06-08 21:31:05	2020-06-08 16:53:50
dataset2	2 files	test-ws	edgar	2020-06-08 21:31:05	2020-06-08 16:51:38
dataset1	1 files	test-ws	edgar	2020-06-08 21:31:05	2020-06-08 16:51:16

Step 11) 사용자에게 개발 환경을 위한 Docker Image 목록 및 생성

The screenshot displays the 'Docker Images' management page in the Jonathon Flightbase dashboard. The left sidebar contains navigation options: Workspaces, Trainings, Deployments, Docker Images (selected), Datasets, Nodes, and Users. The main content area shows a table of Docker images with columns for Status, Upload Type, Release Type, Docker Image Name, Size, Workspace, Uploader, and Uploaded. Three images are listed: 'dockertest' (0.07GB, Workspace), 'test' (0.07GB, All Workspaces), and 'jf-default' (984GB, All Workspaces). Below the table, there are buttons for 'Upload Docker Image' and 'Delete'. An annotation with a blue box and arrow points to the 'Upload Docker Image' button, labeled 'Docker Image 생성/업로드'.

Status	Upload Type	Release Type	Docker Image Name	Size	Workspace	Uploader	Uploaded
Ready	Pull	Workspace	dockertest	0.07GB	robert-test-ws	robert	2020-06-16 15:25:15
Ready	Pull	Global	test	0.07GB	All Workspaces	root	2020-06-15 19:12:43
Ready	Built-in	Global	jf-default	984GB	All Workspaces	root	2020-06-15 15:00:45

Upload Docker Image

Docker Image Name
You can enter Docker Image Name within 20 characters.

Docker Image Description - Optional 0/200
You can enter the Docker Image Description.

Docker Image Upload Type
 Pull Dockerfile build Tar

Please enter only URL of Docker image.

Release Type
 Workspace Global

Workspaces

Available Workspaces	Chosen Workspaces
robert-test-ws	
lico-test-ws	

Cancel Upload

→ "Upload Docker Image"를 선택하여 Docker Image를 생성합니다. Docker Image 이름과 설명을 작성하고, Docker Image Pull을 위한 URL을 입력합니다.

(예: nvcr.io/nvidia/tensorflow:19.02-py3)

→ 접근 범위를 선택합니다.

(Workspace: 선택된 Workspace에서만 접근 가능 / Global: 플랫폼 전체 접근 가능)

Step 12) 사용자 계정 전달 및 신규 사용자 생성

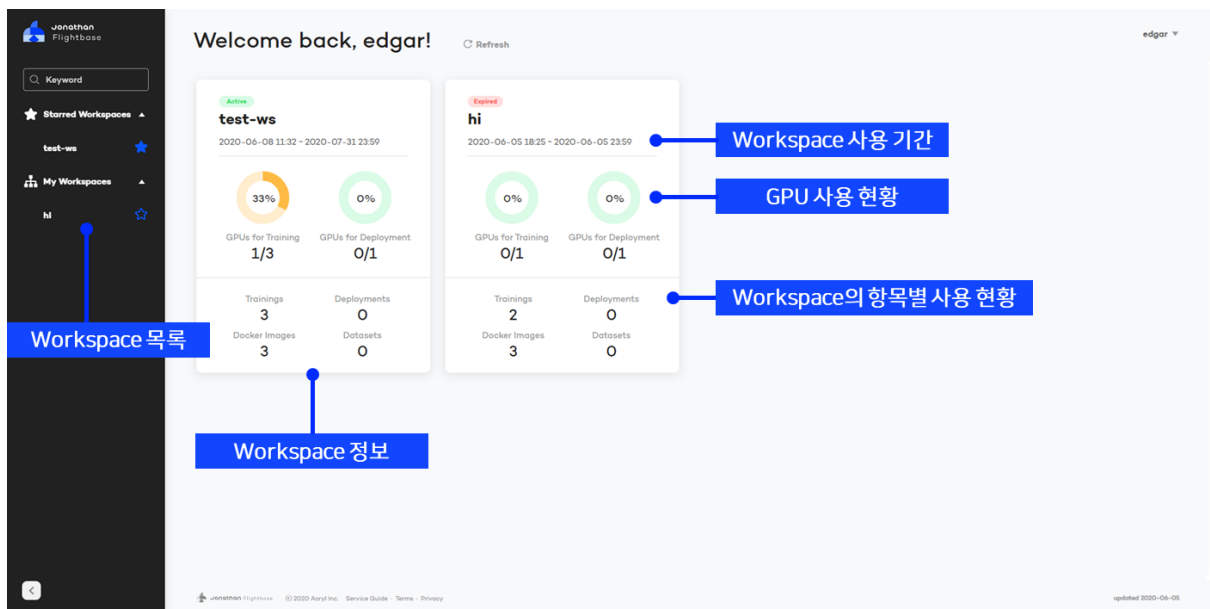
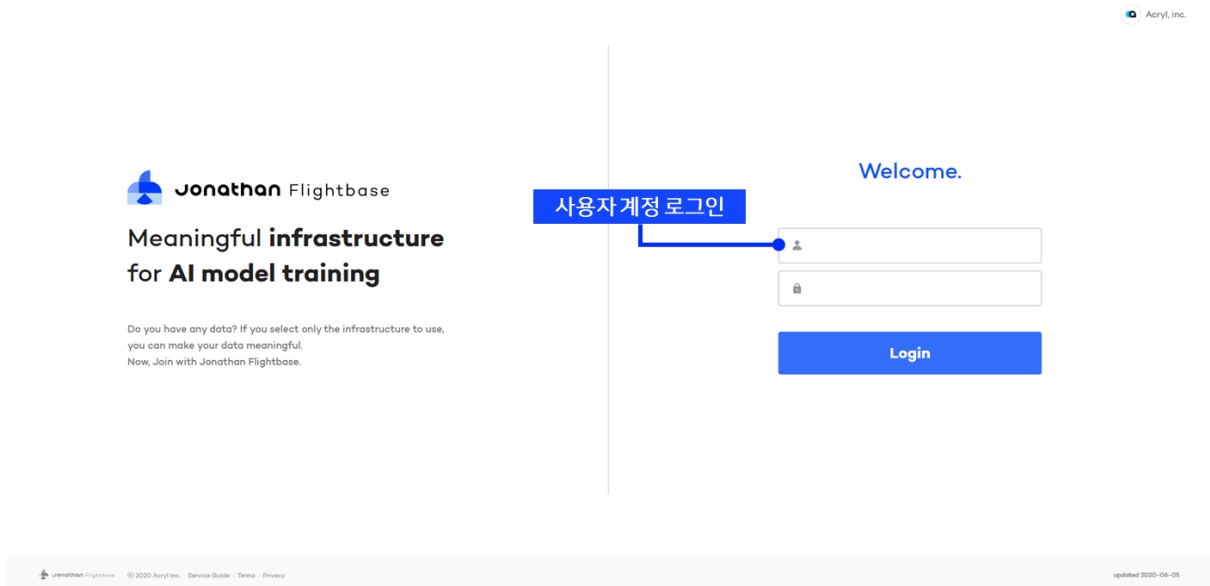
- Workspace에 접속이 가능하도록 생성된 계정 정보를 사용자에게 전달합니다.
- 추가 사용자 생성시 Step 5-6을 수행하며, 추가 사용자에게 Workspace 할당을 위해 다음과 같이 Workspace 수정을 통해 사용자를 추가 배정합니다.

The screenshot shows the 'Edit Workspace' dialog box with the following fields and annotations:

- GPUs for Training:** A spinner box containing the number '1'. A blue callout box labeled '학습 GPU 수 수정' (Modify Learning GPU Count) points to this field.
- GPUs for Deployment:** A spinner box containing the number '0'.
- Manager:** A dropdown menu showing 'robert'. A blue callout box labeled '관리자 수정' (Modify Administrator) points to this field.
- Users - Optional:** A section with two lists: 'Available Users' (containing root, demo, lico, test-account, yb) and 'Chosen Users' (currently empty). A blue callout box labeled '추가 사용자 배정' (Assign Additional Users) points to the 'Chosen Users' list.
- At the bottom right, there are 'Cancel' and 'Edit' buttons.

4. 단계별 이용 가이드 – 사용자 : 학습 실행

Step 1) 관리자에게 할당 받은 사용자 계정으로 로그인



→ 사용자에게 배정된 Workspace의 정보를 확인할 수 있습니다.

Step 2) Workspace 선택 후 Workspace Home으로 이동, Workspace 현황 확인

The screenshot shows the workspace home page for 'test-ws'. The interface includes a sidebar with navigation options like 'Workspace 정보', '항목별 수 확인', and 'GPU 최대 사용률 Chart'. The main content area displays 'test-ws's Info', 'GPU 사용 현황' (GPU Usage Status) with a 33% progress indicator, and a 'Recent Tasks History' table. Blue callout boxes highlight key features: 'Workspace 메뉴 목록' (Workspace menu list), 'Workspace 정보' (Workspace info), '항목별 수 확인' (Check item counts), 'GPU 사용 현황' (GPU usage status), '최근 작업 내역 확인' (Check recent task history), and 'GPU 최대 사용률 Chart' (GPU maximum usage chart).

Step 3) 사용할 Docker Image 확인

The screenshot shows the 'Docker Images' page for the 'test-ws' workspace. It features a table with columns for Status, Upload Type, Release Type, Docker Image Name, Size, Uploader, and Uploaded. Three images are listed: 'dockertest' (0.07GB, Pull, Workspace), 'test' (0.07GB, Pull, Global), and 'jf-default' (984GB, Built-in, Global). The page also includes an 'Upload Docker Image' button and a 'Deletes' button.

Status	Upload Type	Release Type	Docker Image Name	Size	Uploader	Uploaded
Ready	Pull	Workspace	dockertest	0.07GB	robert	2020-06-16 15:25:15
Ready	Pull	Global	test	0.07GB	root	2020-06-15 19:12:43
Ready	Built-in	Global	jf-default	984GB	root	2020-06-15 15:00:45

→ 관리자로부터 생성된 Docker Image 목록을 확인합니다.

→ "Upload Docker Image"를 선택하여 사용자가 직접 Pull할 수도 있습니다.

Step 4) 학습을 위한 Datasets 확인 및 업로드

The screenshot shows the Flightbase interface for a workspace named 'test-ws'. The 'Datasets' tab is active, displaying a table with 3 datasets:

Dataset Name	Files	Creator	Updated	Created
imgdataset	4	edgar	2020-06-08 16:53:58	2020-06-08 16:53:50
dataset2	2	edgar	2020-06-08 16:53:58	2020-06-08 16:51:38
dataset1	1	edgar	2020-06-08 16:53:58	2020-06-08 16:51:16

A blue callout box labeled 'Dataset 업로드' points to the 'Create Dataset' button located below the table.

→ Workspace에 업로드되어 있는 Datasets의 목록입니다.

→ “Create Dataset”을 통해 다수의 파일을 업로드할 수 있습니다.

The screenshot shows the details page for the 'imgdataset'. It displays the following information:

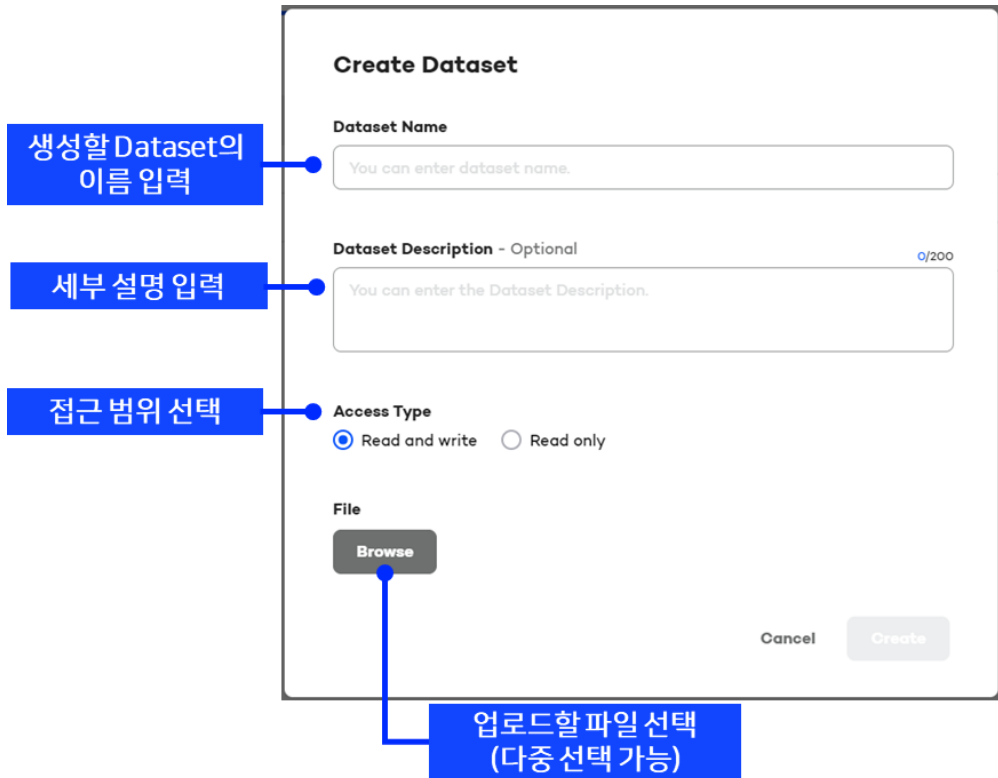
- Access Type: Read and Write
- Total number of Files: 4
- Total File Capacity: 0.36MB

Below this, a table lists the 4 files:

File Type	File Name	Size	Modifier	Modified
file	img2.PNG	0.04MB	edgar	2020-06-08 16:53:50
file	img1.PNG	0.10MB	edgar	2020-06-08 16:53:50
file	img3.PNG	0.05MB	edgar	2020-06-08 16:53:50
file	img4.PNG	0.16MB	edgar	2020-06-08 16:53:50

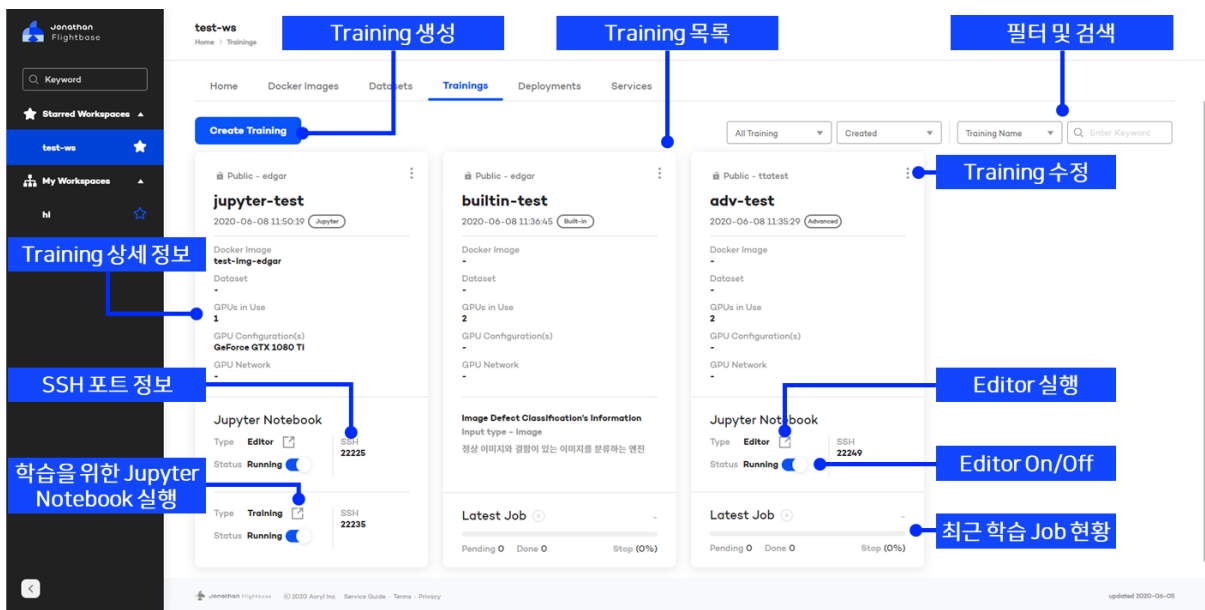
Two blue callout boxes are present: 'Dataset 내 File 추가' points to the 'File Upload' button, and 'Dataset 내 File 삭제' points to the 'Delete' button.

→ 목록의 Dataset을 선택하여, 내부 파일들을 확인할 수 있습니다.



→ "Create Dataset"을 통해 파일 업로드 시 다음과 같이 파일명이 제한됩니다.
 (한글, 일부 특수문자 (*, ?, :, <, >, ", |, ₩) 및 공백 (space)이 있는 파일명 업로드 불가)

Step 5) Trainings 현황 확인 및 생성



- 현재 생성되어 있는 Trainings의 목록입니다. Trainings의 정보 및 학습 상태, Editor 실행 등의 기능을 사용할 수 있습니다.
- Editor (Jupyter Notebook)를 실행하여 학습 코드를 작성하거나 업로드 할 수 있습니다.
- SSH 포트 정보를 이용하여 Terminal을 통해 직접 가상화 공간에 접속할 수 있습니다. 관련 사항은 "Step 5-D"를 참고해주시기 바랍니다.

- "Create Training"을 선택하여 Training을 생성합니다.
- Training 생성 시 입력 가능한 GPU의 최대 수는 Workspace에 할당된 수이며, Training 실행 시 Workspace의 잔여 GPU 수가 부족한 경우 학습은 Pending 됩니다.
- Flightbase에서는 3가지 Type의 Training을 지원하며, Type별 특징은 다음과 같습니다:

no	Training Type	특징
1	Advanced Training	<ul style="list-style-type: none"> - 사용자 학습 코드를 이용한 학습 실행 - 멀티 GPU를 이용한 분산 학습을 지원하는 Training Type - Flightbase내 탑재된 기본 이미지를 사용
2	Built-in Model Training	<ul style="list-style-type: none"> - Flightbase 내 탑재된 학습 모델을 이용하여 학습 실행 - 멀티 GPU를 이용한 분산 학습을 지원하는 Training Type - Flightbase내 탑재된 기본 이미지를 사용
3	Jupyter Notebook Training	<ul style="list-style-type: none"> - Jupyter Notebook을 이용하여 코드 작성 및 학습 가능 - 단일 GPU를 지원하는 Training Type - 사용자 이미지를 이용하여 학습 가능

Step 5-A) Advanced Training 생성

→ Training 이름과 설명, 학습에 사용할 GPU 수를 입력하고, 접근 범위 및 Training Owner를 선택합니다.

Step 5-B) Built-in Model Training 생성

→ Training 이름과 설명, 학습에 사용할 GPU 수를 입력하고, 접근 범위 및 Training Owner를 선택합니다.

→ Flightbase에 탑재된 Built-in Model을 선택하여 Training을 생성합니다.

Step 5-C) Jupyter Notebook Training 생성

→ Training 이름과 설명을 입력하고, Docker Image, 접근 범위 및 Training Owner를 선택합니다. Jupyter Notebook Training의 경우, 사용자 Docker Image를 선택하여 작업 공간을 구성할 수 있습니다.

Step 5-D) SSH 접속 방법 (Editor 및 Jupyter Notebook Training)

→ SSH 포트 정보를 이용하여 다음과 같이 가상화된 Editor 혹은 단일 GPU 를 이용한 학습 환경에 직접 접속할 수 있습니다.

```
Shell
[COMMAND]
ssh [사용자 ID]@[Mater Server IP] -p [SSH Port]
```

Step 6) 학습 Job 목록 및 Log 확인

The screenshot shows the 'adv-test's Jobs' page in the JupyterLab interface. The table lists 8 jobs with various statuses: Running, Pending, and Done. The 'View' column for each job is highlighted with a blue box and labeled '학습 로그 확인'. A 'Create Job' button is located at the bottom left of the table, with a blue box and label 'Job 생성' pointing to it. The sidebar on the left has a 'Job 상태' label pointing to the job list area.

- Advanced Training 및 Built-in Model Training은 학습 Job 기능을 제공합니다.
- Job 목록에서 현재 Job의 상태 및 완료된 Job의 Log를 확인할 수 있습니다.

Step 7) 학습 Job 생성

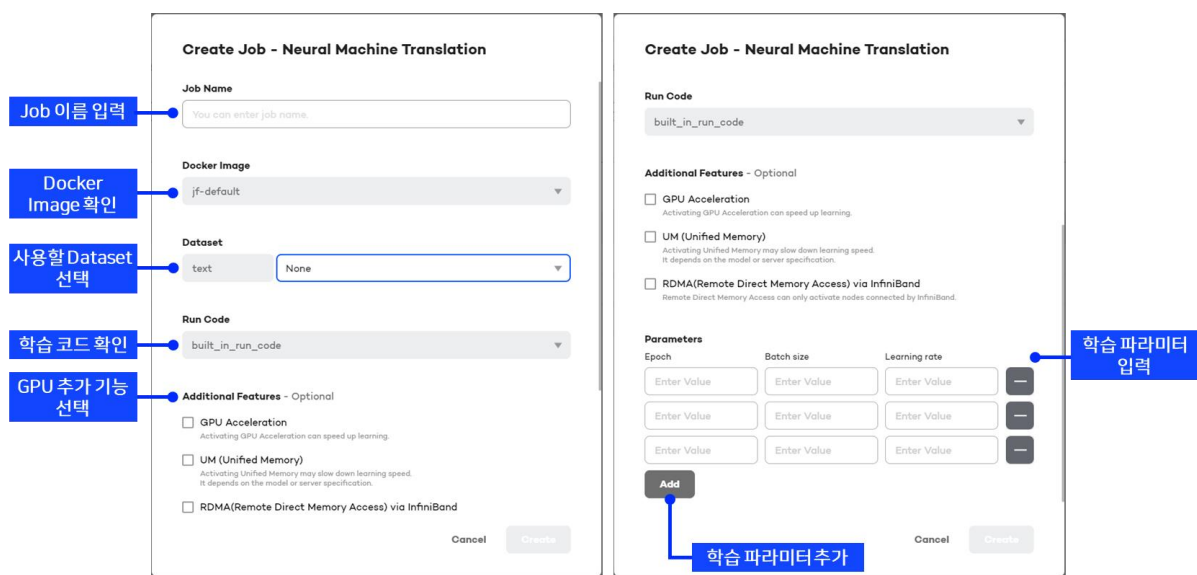
- "Create Job"을 선택하여 Job을 생성할 수 있습니다.

Step 7-A) Advanced Training 학습 Job 생성

The image shows two screenshots of the 'Create Job' dialog box. The left screenshot shows the 'Job Name', 'Docker Image', 'Dataset', and 'Run Code' fields, with labels 'Job 이름 입력', 'Docker Image 확인', '사용할 Dataset 선택', and '학습 코드 선택'. The right screenshot shows the 'Additional Features' and 'Parameters' sections, with labels 'GPU 추가 기능 선택', '학습 파라미터 입력', and '학습 파라미터 추가'. The 'Additional Features' section includes checkboxes for 'GPU Acceleration', 'UM (Unified Memory)', and 'RDMA(Remote Direct Memory Access) via InfiniBand'. The 'Parameters' section includes input fields for batch size and number of epochs, with an 'Add' button to add more parameters.

- Advanced Training의 학습 Job을 생성합니다. Editor를 통해 업로드한 학습 코드를 선택하여 학습 할 수 있습니다.
- Advanced Training의 학습은 Flightbase에 기본 탑재된 Docker Image를 이용합니다.
- 일부 요구 HW 조건에 따라 GPU 추가 기능을 선택하여 가속 학습이 가능하며, 학습 파라미터 추가를 통해 다수의 Job을 순차적으로 자동 실행할 수 있습니다.

Step 7-B) Built-in Model Training 학습 Job 생성



- Built-in Model Training의 학습 Job을 생성합니다. Flightbase에 사전 탑재된 학습 모델 코드를 이용하며, 변수를 입력하여 다수의 Job을 순차적으로 자동 실행 할 수 있습니다.
- Built-in Model Training은 Flightbase에 기본 탑재된 Docker Image를 이용합니다.
- 일부 요구 HW 조건에 따라 GPU 추가 기능을 선택하여 가속 학습이 가능합니다.

GPU Options	특징
GPU Acceleration	- 서버내 GPU 간 가속 통신 기술을 활성화합니다.
Unified Memory	- GPU 메모리가 부족한 경우, RAM을 이용하여 학습합니다. - Built-in Model에서만 사용 가능하며, 사용시 CPU와 통신으로 인해 속도 저하가 발생합니다.
RDMA via InfiniBand	- InfiniBand를 통해 GPU 서버간 RDMA 가속 통신을 활성화합니다.

Step 8) 학습 Job Log 확인 및 Log를 통해 GPU 가속화 기능 적용 여부 확인

Log 확인

View Job Log

Download

Log 다운로드

```

/user_dataset
Epoch 1 Batch 0 Loss 1.6592
Epoch 1 Batch 100 Loss 0.8853
Epoch 1 Batch 200 Loss 0.9081
Epoch 1 Batch 300 Loss 0.7301
Epoch 1 Batch 400 Loss 0.7278
Epoch 1 Batch 500 Loss 0.6855
Epoch 1 Batch 600 Loss 0.7424
Epoch 1 Batch 700 Loss 0.7011
Epoch 1 Batch 800 Loss 0.7989

.
.

2020-06-16 11:03:44.018981: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1159] Device
interconnect StreamExecutor with strength 1 edge matrix:
2020-06-16 11:03:44.018986: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1165] O
2020-06-16 11:03:44.018990: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1178] O: N
2020-06-16 11:03:44.019038: I
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:1006] successful
NUMA node read from SysFS had negative value (-1), but there must be at
least one NUMA node, so returning NUMA node zero
2020-06-16 11:03:44.019404: I
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:1006] successful
NUMA node read from SysFS had negative value (-1), but there must be at
least one NUMA node, so returning NUMA node zero
2020-06-16 11:03:44.019745: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1304] Created
TensorFlow device (/job:localhost/replica:0/task:0/device:GPU:0 with 22353 MB
memory) -> physical GPU (device: 0, name: GeForce GTX 1080 Ti, pci bus id:

```

Confirm

```

Log
NCCL INFO Ring 00 : 0[0] -> 1[1] via P2P/IPC
NCCL INFO Ring 00 : 2[0] -> 3[1] via P2P/IPC

```

→ Log 상에서 상기 부분이 포함되는 경우, "GPU Acceleration" 기능이 적용

```

Log
NCCL INFO NET : Using interface ib0:10.0.20.20<0>
NCCL INFO NET/IB : Using interface ib0 for sideband communication
NCCL INFO NET/IB: [1] mlx5_0:1/IB
NCCL INFO Using internal Network IB
NCCL INFO NET : Using interface ib0:10.0.20.20<0>
:
NCCL INFO Ring 00 : 0 -> 1 via NET/IB/0/GDRDMA

```

→ Log 상에서 상기 부분이 포함되는 경우, "RDMA via InfiniBand" 기능이 적용

*** 이름 및 계정 생성 조건**

항목	조건
사용자 계정 ID	- 3 자 이상 20 자 이하 영어 소문자, 숫자, 하이픈 (-) 가능 단, 하이픈은 연속될 수 없으며, 이름의 처음과 끝에 올 수 없습니다.
암호	- 알파벳 대소문자, 숫자, 특수문자 (~!@#\$\$%^&* <>?) 조합으로 8 자리 이상 20 자리 이하
이름	- 20 자리 이하의 알파벳 소문자, 숫자, 하이픈 (-) 가능 단, 하이픈은 연속될 수 없으며, 이름의 처음과 끝에 올 수 없습니다.

*** Training Access Type**

no	Access Type	접근 허용 범위
1	Public	- Workspace에 속한 사용자가 모두 접근 가능
2	Private	- Training 생성/수정 시 생성자가 배정한 사용자만 접근 가능

*** 중요 데이터 파손 방지를 위한 DB 백업**

→ 24 시간 주기로 Master 서버에서 DB 를 백업하며, 다음과 같은 경로에서 확인 가능합니다.

```
Shell
/jfbc/core/jf-data/backup/jfb-YYYYMMDD-hhmmss.sql
```

*** 사용자 접근 이력 기록 및 로그 확인**

→ Master 서버에서 사용자 접속 기록을 로깅하며, 다음과 같은 경로에서 확인할 수 있습니다.

```
Shell
/jfbc/core/jf-data/log/YYYYMMDD.log
```

→ 해당 로그는 1일 단위로 저장되며, 관리자가 직접 삭제 가능합니다.

5. 알려진 문제에 대한 해결 방안

5-A. Master 및 Worker 설치 후 웹 서비스 상에 화면이 나오지 않는 경우

→ Flightbase – Front 를 다시 시작합니다.

```
Shell  
[COMMAND]  
python3 start_front.py  
python3 installer_front.py --flag MASTER --master_ip [MASTER IP]
```

5-B. Master 및 Worker 설치 후 웹서비스 상에 "Error" 가 표시되는 경우

→ Master 서버에서 Flightbase – Master 를 다시 시작합니다.

```
Shell  
[COMMAND]  
/jfbcore/installer/master_run.sh [MASTER IP]
```

5-C. [5-B] 이후에도 동작하지 않는 경우

→ Master 서버에서 Flightbase – API 를 다시 시작합니다.

```
Shell  
[COMMAND]  
python3 python start_api.py --flag MASTER --master_ip [MASTER IP]  
Option: --ib_ip [InfiniBand IP]
```

5-D. 관리자 Nodes에서 Node 추가-Get Info 시 다음과 같은 Error가 발생하는 경우 :

- Get node network interface error
- Get node Network interfaces error (check worker status)

→ 해당 서버에서 Worker 를 다시 실행합니다.

```
Shell
[COMMAND]
/jfbcore/installer/worker_run.sh
```

5-E. [5-D] 이후에도 동작하지 않는 경우

→ Flightbase – Worker API 를 다시 실행합니다.

```
Shell
[COMMAND]
python3 start_api.py --flag WORKER --master_ip [MASTER IP]
Option: --ib_ip [InfiniBand IP]
```

5-F. “Failed to get node info” Error가 발생하는 경우

→ 다음과 같은 순서로 Kubernetes 초기화를 실행합니다.

```
Shell
[COMMAND]
/jfbcore/installer/kuber_master_init.sh
/jfbcore/installer/kuber_plug_in_init.sh
```

5-G. 관리자 Nodes에서 Add Node 후 Connection 시, "Attach Error"가 발생하는 경우

→ Master 서버에서 Kubernetes Worker 초기화를 실행하고 다시 Connection 실행합니다.

```
Shell
```

```
[COMMAND]
```

```
/jfbcore/installer/kuber_worker_init.sh
```

→ 문제가 해결되지 않는 경우, 재부팅 후 Worker 를 재시작합니다.

```
Shell
```

```
[COMMAND]
```

```
/jfbcore/installer/worker_run.sh
```

5-H. 웹서비스 – 관리자 – Nodes List에서 Node의 Status가 "Error" 인 경우, 커서를 이동해서 메시지를 확인하고 다음 케이스에 대한 해결 방안을 시도합니다.

[Error : Worker connection error]

→ [5-D]와 [5-E]의 방법으로 Worker 를 재시작합니다.

[Error : NodeStatusUnknown]

→ 해당 서버에 Kubernetes 가 종료되었을 수 있습니다. 관리자-Nodes 페이지에서 해당 Node 의 Connection 을 해제하고 다시 Connection 을 실행합니다.

→ 위의 방법으로 해결되지 않는 경우 Worker 를 초기화합니다.

```
Shell
```

```
[COMMAND]
```

```
/jfbcore/installer/kuber_worker_init.sh
```

5-1. 그 외의 상황

[기본 탑재 Docker Image 를 실수로 삭제한 경우]

→ API 를 재시작합니다.

```
Shell  
[COMMAND]  
/jfbcore/installer/jf_default_image_init.sh  
python3 start_api.py --flag MASTER or WORKER --master_ip [MASTER IP]  
Option: --ib_ip [InfiniBand IP]
```

* 이외의 문제 발생 시 support@iacryl.com 으로 문의주시기 바랍니다.

6. Jonathan Flightbase™ 유지 보수

6-A. 무상 유지 보수 기간

구매 제품이 설치된 날로부터 1년간은 무상으로 하자보수를 진행합니다.

6-B. 유상 유지 보수 기간

무상 하자보수 기간이 종료된 이후, 고객의 요청이 있는 경우에는 유지 보수 조건을 별도로 협의하여 진행합니다.

Jonathan Flightbase™ v2.0 사용 매뉴얼

초 판 2020년 6월 17일 발행

발행원 주식회사 아크릴

Copyright 2020. ACRYL Inc. All rights reserved.
