

Setting Up intdash Server Using AMI (For intdash All-in-One Users)

intdash All-in-One Ver. 202206

6th edition (August 2022)

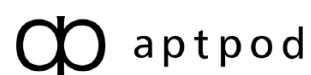


Table of contents

01 Introduction	3
1.1 Target audience	3
1.2 Prerequisites	3
1.3 Structure of this document	3
02 Create and launch an intdash instance	4
2.1 Preparation	4
2.2 Create an intdash instance	4
2.3 Confirm that the intdash instance has been created	8
2.4 Make sure you can SSH into your intdash instance	8
03 Set up the intdash server	10
3.1 Set domain name and server certificate	10
3.2 Set up the administrator account	16
3.3 Set the API key for Google Maps (only if you use Google Maps)	20
04 Check the operation	22
4.1 Create user accounts	22
4.2 Create an edge account	22
4.3 Prepare the dashboard	22
4.4 Send real-time video from iPhone	25
4.5 Visualize the data with Data Visualizer	25
05 Appendix: Services and configuration files in intdash	27
5.1 Original applications	27
5.2 Open source applications	29
06 Appendix: Software update procedure if you are using an older version of AMI	30
6.1 Prepare for an update	30
6.2 Update AMI version 202009 software to the version 202103 equivalent	31
6.3 Update AMI version 202103 software to the version 202106 equivalent	33
6.4 Update AMI version 202106 software to the version 202112 equivalent	34
6.5 Update AMI version 202112 software to the version 202203 equivalent	40
6.6 Update AMI version 202203 software to the version 202206 equivalent	43

01 Introduction

This document describes the setup procedure of an intdash server (intdash instance) using the Amazon Machine Image (AMI) provided on the AWS Marketplace.

Attention: This document has been translated using machine translation services and may contain inaccuracies and translation errors. Please also refer to the official version in Japanese.

1.1 Target audience

This document is written for those who set up and manages an intdash instance.

Those who set up an intdash environment are assumed to have basic knowledge of network management, server management, and instance setup with IaaS (AWS).

1.2 Prerequisites

The following software is required to set up and configure an intdash instance using the AMI.

- Terminal software that allows SSH connection
- Google Chrome web browser

1.3 Structure of this document

The following chapters are organized as follows.

Create and launch an intdash instance (p. 4)

Select the AMI on the AWS Marketplace and launch it. The operation is mainly in the AWS console.

Set up the intdash server (p. 10)

Make authentication and account settings on the launched intdash instance. The configuration is made via a terminal with SSH connection and a dedicated web application.

Check the operation (p. 22)

To confirm that the intdash instance started normally, perform a simple operation check by sending and displaying real-time data.

Appendix: Services and configuration files in intdash (p. 27)

Technical supplements for intdash applications.

Appendix: Software update procedure if you are using an older version of AMI (p. 30)

This section contains instructions for updating the software in your instance if you are using an earlier version of the AMI.

02 Create and launch an intdash instance

Create and launch an EC2 instance for intdash using the AMI provided on AWS Marketplace.

2.1 Preparation

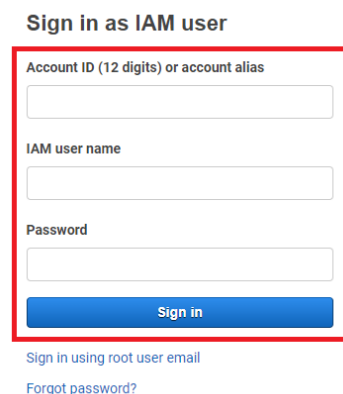
Prepare the following information in advance.

- AWS account
- Network settings for the intdash instance
 - VPC settings
 - Subnet settings
 - Routing table
- Elastic IP (required only when making the server available to the public)

2.2 Create an intdash instance

Open the AWS console and create an intdash instance.

1. Sign in to AWS.



The image shows the 'Sign in as IAM user' form in the AWS console. It is titled 'Sign in as IAM user' and contains three input fields: 'Account ID (12 digits) or account alias', 'IAM user name', and 'Password'. Below these fields is a blue 'Sign in' button. The entire form is enclosed in a red rectangular border. Below the form, there are two links: 'Sign in using root user email' and 'Forgot password?'.

Fig. 1 Signing in to AWS

2. Select [Compute] > [EC2].

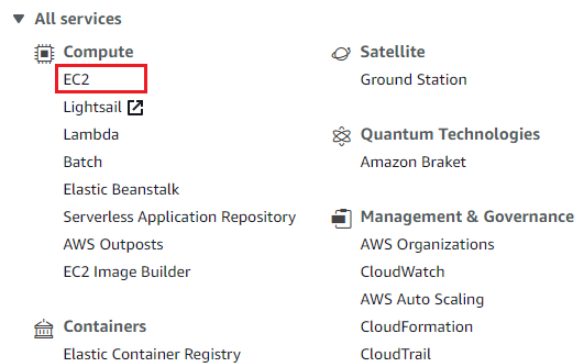


Fig. 2 Select [EC2]

3. From the dashboard, click [Launch instance].

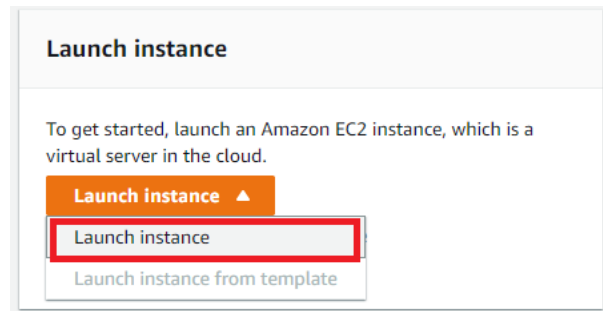


Fig. 3 Launching an instance

4. Search for "intdash" and select "intdash All-in-One" from the AWS Marketplace.

- Search word: intdash
- Tab: AWS Marketplace

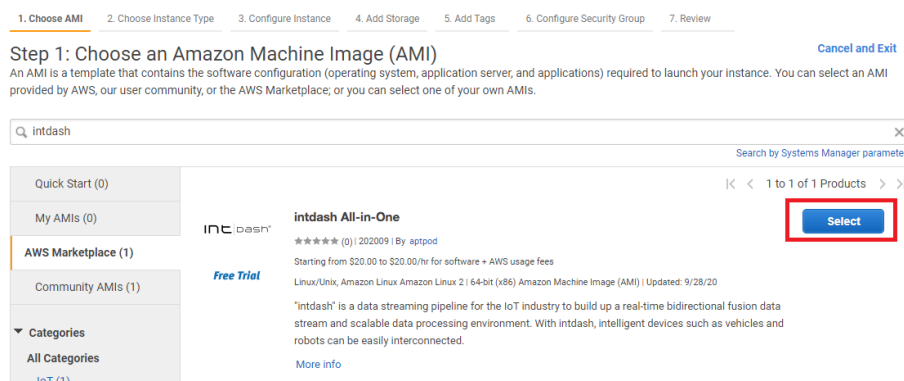


Fig. 4 Search and select "intdash"

Note: The number after the hyphen in the AMI version number ("2" in the case of version "202203-2") is incremented when minor modifications are made to the AMI. If multiple versions of the AMI are displayed, select the most recent one.

5. Read the content and click [Continue].

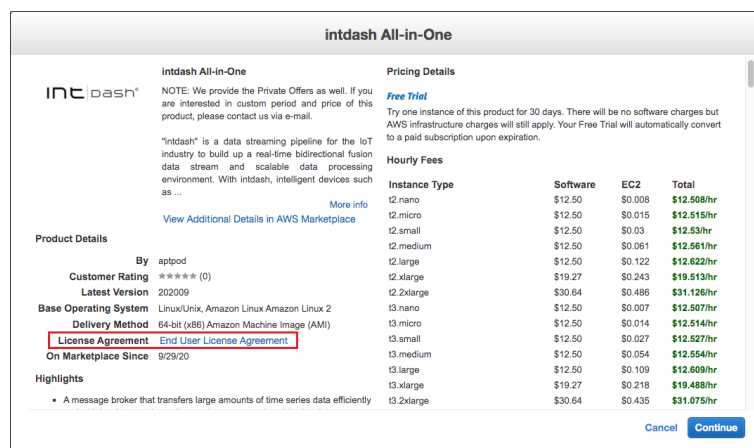


Fig. 5 Read the content and click [Continue]

6. Select the type of intdash instance. Instance types are a combination of different CPU, memory, storage, and network capacities. In this example, select [m5.large] and click [Next].

Step 2: Choose an Instance Type

<input checked="" type="checkbox"/>	General purpose	m5.large	2	8
<input type="checkbox"/>	General purpose	m5.xlarge	4	16
<input type="checkbox"/>	General purpose	m5.2xlarge	8	32
<input type="checkbox"/>	General purpose	m5.4xlarge	16	64

Fig. 6 Instance type selection

7. Set the details of the intdash instance. Select the network to which the instance belongs, assign an IP address, and make other detailed settings.

In this example, [Network], [Subnet], and [Auto-assign Public IP] are set. Set other items as needed. Click [Next].

Number of instances [Launch into Auto Scaling Group](#)

Purchasing option ☐ Request Spot Instances

Network [Create new VPC](#)

Subnet [Create new subnet](#)
4091 IP Addresses available

Auto-assign Public IP

Auto-assign IPv6 IP

Placement group ☐ Add instance to placement group

Capacity Reservation

Fig. 7 Instance details settings

8. Select the size of storage to attach to your intdash instance. In this example, the minimum required size is 8GB. The size of the storage can also be increased using the common AWS configuration methods after creating an instance. Click [Next].

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination
Root	/dev/xvda	snap-	<input type="text" value="8"/>	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>

[Add New Volume](#)

Fig. 8 Adding storage

9. Add a tag that applies to your intdash instance. Adding tags is optional. In this example, the "Name" tag with the value "intdash" is added. When you're done, click [Next].

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key (128 characters maximum) Value (256 characters maximum) Instances ⓘ

Name intdash

Add another tag (Up to 50 tags maximum)

Fig. 9 Adding tags

10. Set a security group to apply to the intdash instance. Allow the following in the security group. After completing the settings, click [Review and Launch].

- SSH (TCP, port 22)
- HTTP (TCP, port 80)
- HTTPS (TCP, port 443)

Assign a security group: ☒ Create a new security group
☐ Select an existing security group

Security group name: intdash-security-group

Description: intdash-security-group

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
SSH	TCP	22	Custom 0.0.0.0/0	for SSH
HTTP	TCP	80	Custom 0.0.0.0/0, ::/0	for HTTP
HTTPS	TCP	443	Custom 0.0.0.0/0, ::/0	for HTTPS

Fig. 10 Security group settings

Note: Allow SSH (TCP port 22) access to perform configuration of the server. It is recommended to restrict the connection source if necessary.

11. A confirmation screen for the settings is displayed. Confirm the settings, and click [Launch].

12. Select the key pair you want to use.

A key pair is a pair of private and public keys that is required to connect to your intdash instance. You can use an existing key pair or create a new one. If you are creating a new key pair, be sure to download the private key to your computer.

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Create a new key pair

Key pair name
intdash-keypair

Download Key Pair

You have to download the **private key file** (*.pem file) before you can continue. Store it in a secure and accessible location. You will not be able to download the file again after it's created.

Cancel Launch Instances

Fig. 11 Example creating a new key pair

13. Click [Launch Instances].

2.3 Confirm that the intdash instance has been created

1. In the AWS Console, select [Instances] > [Instances] from the left menu bar to see the list of instances.

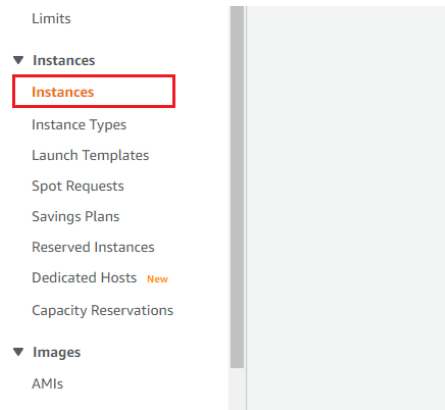


Fig. 12 View instance

2. Make sure you can see the intdash instance. Also, check that the instance state and the status checks are as follows.
 - Instance state: "running"
 - Status checks: "2/2 checks passed"

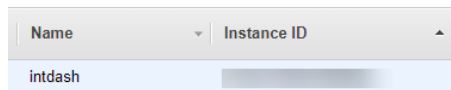


Fig. 13 Confirming that the created instance is displayed

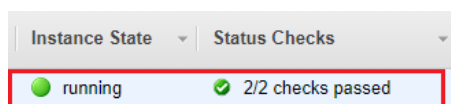


Fig. 14 [Instance State] and [Status Checks]

Note: When using Elastic IP, assign an Elastic IP address.

2.4 Make sure you can SSH into your intdash instance

Make sure that you can connect to the intdash instance via SSH from the terminal on your PC. Execute the command as follows in the terminal.

```
$ ssh -i PATH-TO-PRIVATE-KEY ec2-user@INSTANCE-IP-ADDRESS
```

PATH-TO-PRIVATE-KEY

File path of the private key

INSTANCE-IP-ADDRESS

IP address of the intdash instance

For example, if the file path of your private key is `./.ssh/intdash-keypair.pem` and the IP address of the intdash instance is `203.0.113.123`, the command would be `ssh -i ~/.ssh/intdash-keypair.pem ec2-user@203.0.113.123`.

If the connection is successful, the following screen will be displayed.

```
$ ssh -i ~/.ssh/intdash-keypair.pem ec2-user@203.0.113.123
Last login: .....

  __/  __/_  )
 _/  (    /  Amazon Linux 2 AMI
---/\---/---/

https://aws.amazon.com/amazon-linux-2/
NNN package(s) needed for security, out of NNN available
Run "sudo yum update" to apply all updates.

  _      _      /      _
(-) _ _  / /_  /  _ \  _ _  _ _  / /_
/ / / ' \ / _/  /  / / / / _ \ / _/ / ' \
/ / / / / / _/  /  / / / / (-) \_ \ / / /
/_/_/_ /_/_ \_/_ /  /_/_/_ \_ \_/_/_/_ /_/_ /
      /

https://www.aptpod.co.jp/products/intdash/

[ec2-user@ip-10-0-0-123 ~]$
```

The instance has been created. Proceed to [Set up the intdash server](#) (p. 10).

03 Set up the intdash server

After you have launched your intdash instance by following [Create and launch an intdash instance](#) (p. 4), configure intdash for your environment.

3.1 Set domain name and server certificate

In this document, the following domain name and certificate files are used as examples.

Item	Examples used in this document
Domain name for intdash	intdash.example.com
SSL server certificate issued by a third party certification authority (server certificate and intermediate certificate concatenated together)	/etc/pki/intdash/certs/intdash.pem
SSL server private key	/etc/pki/intdash/private/intdash.pem

Note: In most cases, a sub-domain of a domain name owned by your organization is used for intdash. For example, if you own and manage example.com, the domain name for intdash should be intdash.example.com.

Talk to your organization's domain name administrator for more information.

Note:

- The server certificate, also known as the SSL/TLS certificate, protects the connection between the client and the server and prevents it from being read or modified by a third party.
- For security reasons, use a certificate issued by a third party certification authority instead of a self-signed certificate. If you use a self-signed certificate, the client will fail to validate the server certificate. If you skip the server certificate verification, you will not verify the communication partner, allowing eavesdropping and tampering by a man-in-the-middle attack or other means.
- Some examples of certificate authorities are Cybertrust, DigiCert, and Let's Encrypt. [Let's Encrypt](#) allows you to get a certificate for free.

3.1.1 Set up the web server

In the nginx configuration file, set the domain name and the server certificate to be used.

1. Connect to your intdash instance with SSH and go to the configuration file directory.

```
# cd /etc/nginx/conf.d
```

2. Back up the default configuration file.

```
# cp -p intdash.conf intdash.conf.org
```

3. Open the configuration file in a text editor.

```
# vi intdash.conf
```

4. Set the domain name in [server_name] of port 80 (HTTP) and port 443 (HTTPS) settings.

```
...

server {
    listen      80;
    listen      [::]:80;
    server_name  intdash.example.com;  <--
    access_log  off;

    location / {
        return 301 https://$host$request_uri;
    }
}

server {
    listen      443 ssl http2;
    listen      [::]:443 ssl http2;
    server_name  intdash.example.com;  <--
    root        /usr/share/nginx/html;

    ssl_certificate      /etc/pki/intdash/certs/intdash.pem;
    ssl_certificate_key  /etc/pki/intdash/private/intdash.pem;

    ...
}
```

5. In the settings for port 443 setting in the same configuration file, specify the server certificate in [ssl_certificate] and the private key of the server certificate in [ssl_certificate_key]. Then close the configuration file.

```
...

server {
    listen      80;
    listen      [::]:80;
    server_name  intdash.example.com;
    access_log  off;

    location / {
        return 301 https://$host$request_uri;
    }
}
```

(continues on next page)

(continued from previous page)

```
server {
    listen      443 ssl http2;
    listen      [::]:443 ssl http2;
    server_name  intdash.example.com;
    root        /usr/share/nginx/html;

    ssl_certificate      /etc/pki/intdash/certs/intdash.pem;        <--
    ssl_certificate_key  /etc/pki/intdash/private/intdash.pem;      <--

    ...
}
```

- Restart nginx with the following command for the settings to take effect.

```
# systemctl restart nginx
```

- After restarting, make sure nginx is running properly.

```
# systemctl status nginx
```

If it is working properly, the status Active: active (running) is displayed.

3.1.2 Configure the edge router Traefik

Configure Traefik, which is responsible for intdash API services and proxying to microservices.

- Continuing from the previous step, SSH into your intdash instance and go to the configuration file directory.

```
# cd /etc/intdash/traefik.d
```

- Back up the default configuration file.

```
# cp -p middleware.toml middleware.toml.org
```

- Open the configuration file in a text editor.

```
# vi middleware.toml
```

- Enter the domain name in the following settings. Then close the configuration file.

```
[http]
[http.middlewares]

...

[http.middlewares.cors]
[http.middlewares.cors.headers]

...

accessControlAllowOriginList = ["https://intdash.example.com", "http://intdash.example.com",
→ "wss://intdash.example.com:443/api/v1/ws/measurements"] <--

...
```

- Restart the intdash-api-gateway service with the following command to apply the settings.

```
# systemctl restart intdash-api-gateway
```

6. After restarting, verify that the intdash-api-gateway has started successfully.

```
# systemctl status intdash-api-gateway
```

If it is working properly, the status `Active: active (running)` is displayed.

3.1.3 Configure intdash-micro-auth, which is responsible for authentication processes

Set the microservice (intdash-micro-auth) that performs authentication processes of intdash.

1. Continuing from the previous step, SSH into your intdash instance and go to the configuration file directory.

```
# cd /etc/intdash
```

2. Back up the default configuration file.

```
# cp -p authd.conf authd.conf.org
```

3. Open the configuration file in a text editor.

```
# vi authd.conf
```

4. Enter the domain name in the following settings. Then close the configuration file.

- [email] section
 - verification-page-uri

```
[email]
from = "noreply@example.com"
from-display-name = "VM2M"
reply-to = ""
limit-per-user = 1
verification-expiration-period = "3h"
verification-page-uri = "https://intdash.example.com/email/activate" <--
...
```

- [oauth2] section
 - issuer
 - sign-in-page-uri
 - change-password-page-uri

```
[oauth2]
issuer = "https://intdash.example.com" <--
access-token-expiration-period = "1h"
refresh-token-expiration-days = 30
sign-in-page-uri = "https://intdash.example.com/signin/" <--
change-password-page-uri = "https://intdash.example.com/oauth2/authorization/api/password-change" <-
...
↩
```

- [oauth2.default-client] subsection
 - password-recovery-redirect-base-url
 - web-redirect-base-urls

```
[oauth2.default-client]
password-recovery-redirect-base-url = "https://intdash.example.com" <--
web-redirect-base-urls = [
    "http://localhost:8080",
    "https://localhost:8080",
    "https://intdash.example.com", <--
]
...
```

5. To update the database with oauth2 related parameters, execute the following commands.

```
# sudo -u intdash authd db migrate -c /etc/intdash/authd.conf
```

6. For the configuration to take effect, restart the intdash-micro-auth service with the following command.

```
# systemctl restart intdash-micro-auth
```

7. After restarting, make sure that intdash-micro-auth is running properly.

```
# systemctl status intdash-micro-auth
```

If it is working properly, the status `Active: active (running)` is displayed.

3.1.4 Make the private keys of intdash-micro-auth JWT (JSON Web Token) constant

To keep the JWT and RSA private keys constant, set them in the intdash-micro-auth service configuration file `/etc/intdash/authd.conf`.

Note: If you do not set fixed values here, JWT and RSA private keys will be automatically generated each time the intdash-micro-auth service is restarted. In that case, the following will occur each time the service is restarted:

- API tokens become invalid
- User login statuses will be reset
- URLs for confirming email addresses become invalid

Therefore, it is recommended to set fixed values.

- `[keys.oauth2-rsa.private-key]` subsection
 - `key`: Set an RSA private key in PEM format.

```
[keys]
[keys.oauth2-rsa]
[keys.oauth2-rsa.private-key]
driver = "static"
[keys.oauth2-rsa.private-key.static]
key = "-----BEGIN RSA PRIVATE KEY-----\nXXXXXX...XXXXXX\n-----END RSA PRIVATE KEY-----" <--
```

- `[keys.oauth2-hmac]`, `[keys.api-token]`, `[keys.password-recovery]`, `[keys.email-verification]` subsections
 - `hash-secret`: Set randomly generated strings.

```
[keys.oauth2-hmac]
hash-secret = "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX" <--
rotated-hash-secrets = [""]
[keys.api-token]
```

(continues on next page)

(continued from previous page)

```

hash-secret = "XXXXXXXXXXXXXXXXXXXXXXXXXXXX" <--
rotated-hash-secrets = [""]
[keys.password-recovery]
hash-secret = "XXXXXXXXXXXXXXXXXXXXXXXXXXXX" <--
rotated-hash-secrets = [""]
[keys.email-verification]
hash-secret = "XXXXXXXXXXXXXXXXXXXXXXXXXXXX" <--
rotated-hash-secrets = [""]

```

Restart the intdash-micro-auth service after this configuration. It is not necessary to apply this change in the database.

Note: In the [keys.oauth2-rsa.private-key] subsection, you can specify `driver = "file"` to specify the file path of the RSA private key instead of the PEM format string. The private key file is read by the intdash-micro-auth service run by the intdash user; set the file permissions, owner, and group to allow reading by the intdash user.

```

[keys]
[keys.oauth2-rsa]
[keys.oauth2-rsa.private-key]
driver = "file"
[keys.oauth2-rsa.private-key.file]
path = "/etc/intdash/authd.keys/privkey.pem" <--

```

3.1.5 Configure intdash-web-oauth2

Configure the UI service (intdash-web-oauth2) that authenticates the web UI client in intdash OAuth2.

1. Continuing from the previous step, SSH into your intdash instance and go to the configuration file directory.

```
# cd /etc/sysconfig
```

2. Back up the default configuration file.

```
# cp -p intdash-web-oauth2 intdash-web-oauth2.org
```

3. Open the configuration file in a text editor.

```
# vi intdash-web-oauth2
```

4. Enter the domain name in `AUTHORIZATION_HOST` and `CLIENT_HOST`. Then close the configuration file.

```

HOST="127.0.0.1"
PORT="13003"

API_HTTP_URL="http://127.0.0.1:8080"
AUTHORIZATION_HOST="https://intdash.example.com" <--
CLIENT_HOST="https://intdash.example.com" <--
CLIENT_ID="533bc9ea_authorization_code.aptpod.co.jp"
RETURN_TO_URL="/"
ALLOW_LOGGING="false"

```

5. For the configuration to take effect, restart the intdash-web-oauth2 service with the following

command.

```
# systemctl restart intdash-web-oauth2
```

6. After restarting, make sure that intdash-web-oauth2 is running properly.

```
# systemctl status intdash-web-oauth2
```

If it is working properly, the status Active: active (running) is displayed.

3.2 Set up the administrator account

By default, there is an administrator user account named intdash. In this section, we will set the password and email address for this account. We will also configure the mail server.

Note: The initial password for the user intdash is the instance ID of the intdash instance. You can check the instance ID in the EC2 console. (Example: i-1234567890abcdef0)

3.2.1 Set the password for the administrator user

Set the password for the administrator user.

1. Open Visual M2M Data Visualizer (Data Visualizer) <https://intdash.example.com/> (use your actual domain name) with a web browser.
2. Enter the username intdash and the initial password, agree to the User Guideline, and then click [Sign In].

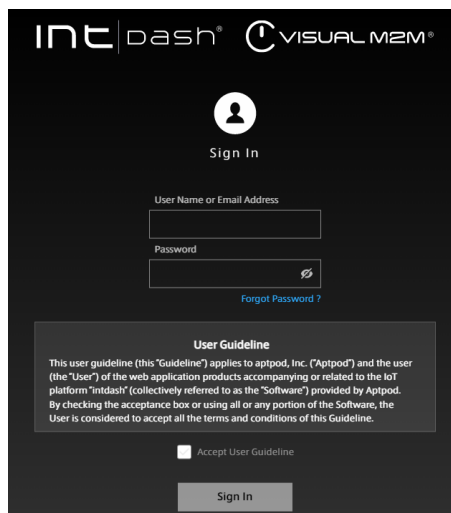


Fig. 15 Sign-in screen

3. When you sign in, the password setting screen will be displayed as shown below. Enter your new password and click [Change Password].

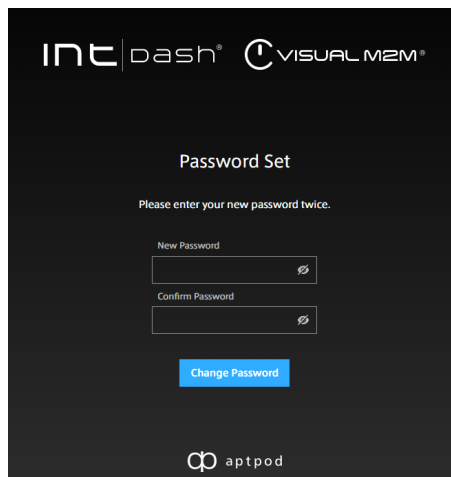


Fig. 16 Setting a password

Attention: Authentication information should be managed properly and carefully.

3.2.2 Email server and notification mail sender configurations

If you forget your password and attempt to authenticate, the intdash server will send an email to you. In this section, set the SMTP server and the email address used as the sender's address.

1. Connect to your intdash instance with SSH and go to the configuration file directory.

```
# cd /etc/intdash
```

2. Open the configuration file in a text editor.

```
# vi authd.conf
```

4. Enter the SMTP server information and outgoing mail information in the [email] section. Set the SMTP server information according to your environment. Then close the configuration file.

- [email] section
 - from: Email address used as the sender
 - reply-to: Email address used for Reply-to header (optional)
- [email.smtp] subsection
 - SMTP server information for your environment

```
[email]
from = "noreply@example.com"
from-display-name = "VM2M"
reply-to = ""

...

[email.smtp]
address = "127.0.0.1:25"
hostname = ""
enable-tls = false
enable-starttls-auto = false
insecure-tls = false
```

(continues on next page)

(continued from previous page)

```
authentication = ""  
username = ""  
password = ""
```

5. For the configuration to take effect, restart the intdash-micro-auth service with the following command.

```
# systemctl restart intdash-micro-auth
```

6. After restarting, make sure that intdash-micro-auth is running properly.

```
# systemctl status intdash-micro-auth
```

If it is working properly, the status Active: active (running) is displayed.

3.2.3 Set the email address of the administrator user

1. Open <https://intdash.example.com/users/me/> in your web browser.
2. When the sign-in screen appears, sign in with the administrator user name intdash and the password you set earlier.

My Page for the user intdash is displayed.

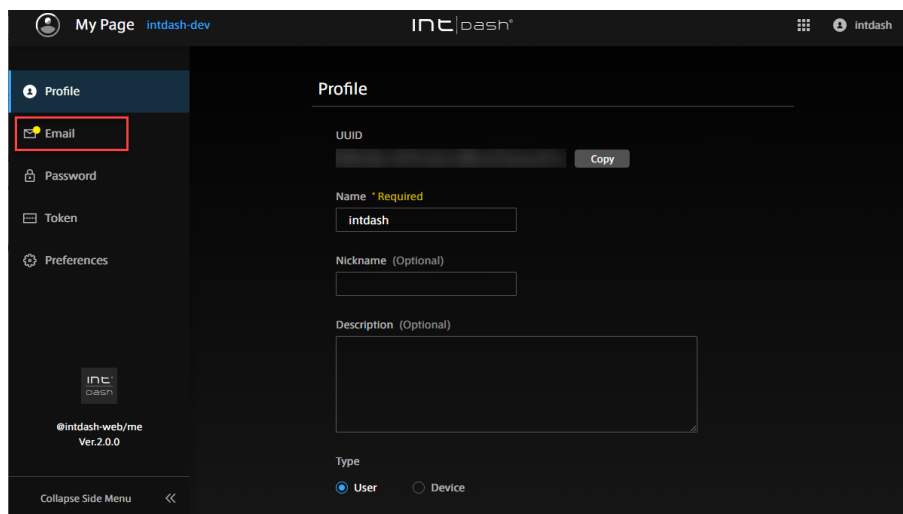


Fig. 17 My Page

3. Click [Email], enter the email address of the administrator user, and click [Save Changes].

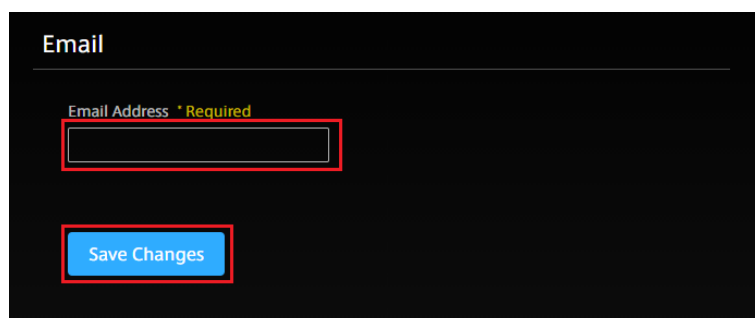


Fig. 18 Setting the email address

4. The following message will be displayed. Make sure your email address is correct and click [Close].

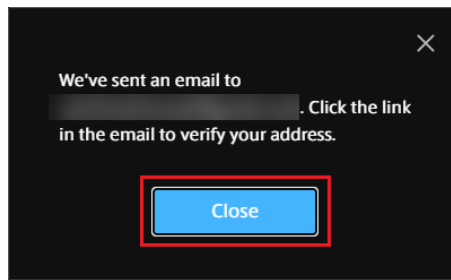
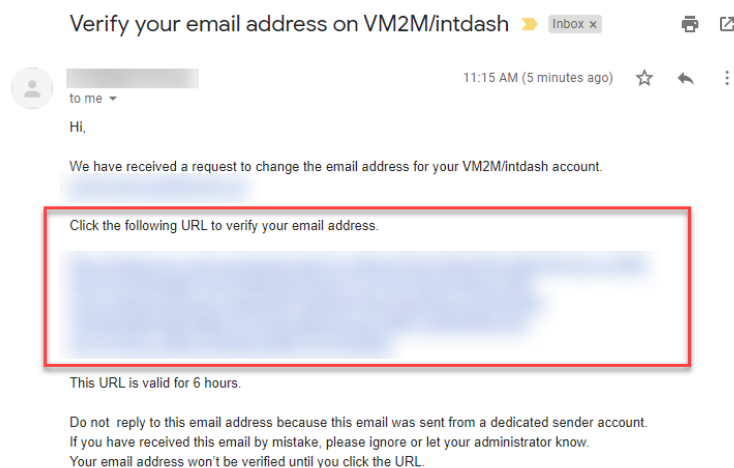


Fig. 19 Confirming your email address

A confirmation email will be sent to your email address.

5. Open the confirmation email in your email client.
Access the activation URL in the email.



6. Open [Email Address] on My Page again, and if it is [Verified] as shown below, the setting is complete.

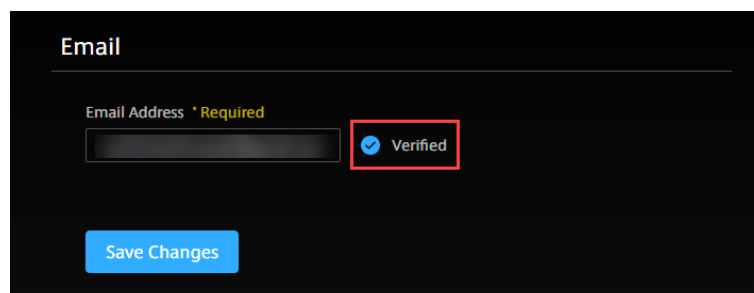


Fig. 20 Email address verified

3.3 Set the API key for Google Maps (only if you use Google Maps)

To use Google Maps with Data Visualizer, you need to set up a Google Maps API key on your intdash server.

Note:

- In Data Visualizer, you can also use Open Street Map; if you use Open Street Map, you do not need to set an API key here.
- For more information on Google Maps, see [Google Maps Official Website](#).

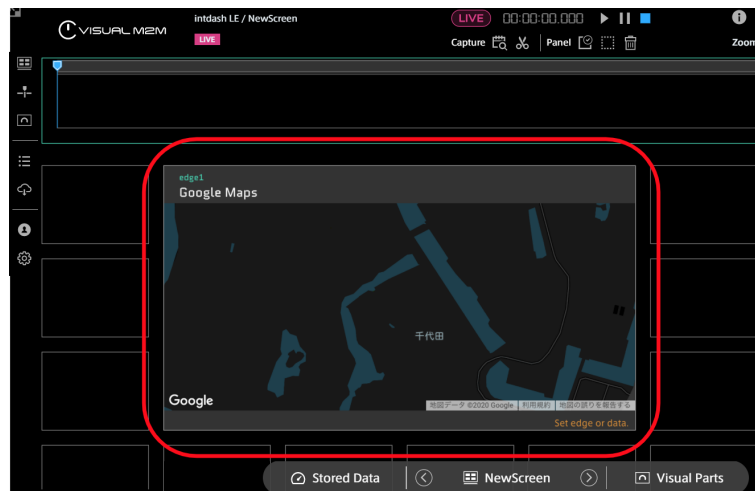


Fig. 21 Google Maps visual part

To set the API key for Google Maps, access the configuration API on the intdash server as an administrator user.

1. Connect to the intdash instance via SSH.
2. Execute the following command to set the password of the administrator user "intdash" to a shell variable.

```
# read -s PASSWORD
<-- Enter the password and press <Enter>.
```

3. Sign in to intdash with the following command.

```
# curl -X POST http://127.0.0.1:8080/api/auth/oauth2/token \
-H "Content-Type: application/x-www-form-urlencoded" \
-d "username"="intdash" \
-d "password"="${PASSWORD}" \
-d "grant_type"="password" \
-d "client_id"="99dcf67c_default.aptpod.co.jp"
```

4. Find access_token in the JSON response. Set its value to a shell variable.

```
# read -s ACCESS_TOKEN
<-- Enter the access token and press <Enter>.
```

5. Execute the following command to set the Google Maps API key.

Enter your Google Maps API key as the value of `googleMapsApiKey`.

In the following, -d '{"content" ... line is shown wrapped, but no line breaks are needed within this line.

```
# curl -X PUT http://127.0.0.1:8080/api/v1/configs/vm2m-2nd-variables \
-H "Content-Type: application/json" \
-H "Authorization: Bearer ${ACCESS_TOKEN}" \
-d '{"content":{"googleMapsApiKey":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX","\
↪"googleMapsApiClientID":"\""}'}
```

Note: If you have a Client ID instead of an API key, enter it as the value of googleMapsApiClientID as shown below.

```
-d '{"content":{"googleMapsApiKey":"","googleMapsApiClientID":"\
↪XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX\""}'}
```

This completes the settings. Go to [Check the operation](#) (p. 22).

04 Check the operation

In this chapter, we will perform a simple operation check as follows.

- Create a new user (user1) and an edge (edge1).
- Run the intdash Motion application on your iPhone and send the video to the intdash server as edge1.
- Display the video in Data Visualizer on your PC.
- Replay stored data with Data Visualizer.


4.1 Create user accounts

Sign in to the Admin Console as an administrator and create a user account (user1).

1. Access the Admin Console <https://intdash.example.com/admin/users> with a web browser.
2. When the sign-in screen appears, sign in with your administrator user name `intdash` and the password you just set.
3. Click [Create User], fill in the required information, and click [Create].
 - Name: `user1`
 - Role: `member` (normal user)
4. A user is created and a temporary password is displayed. Copy the temporary password.

4.2 Create an edge account

Continue to create an edge account in the Admin Console.

1. On the "Edges" page, click [Create Edge], fill in the required fields, and click [Create].
 - Name: `edge1`The UUID and client secret are displayed, but these are not required for operation checks.
2. Click  in the upper right corner to sign out. We will use the new user `user1` in the subsequent steps.

4.3 Prepare the dashboard

1. Access the Visual M2M Data Visualizer <https://intdash.example.com/vm2m/> in your Chrome browser and sign in as user `user1`.

If you are signing in for the first time, you will need to reset your password. Follow the on-screen instructions.
2. Download the data settings file (DAT file) for video from [Links] > [Download DAT File].

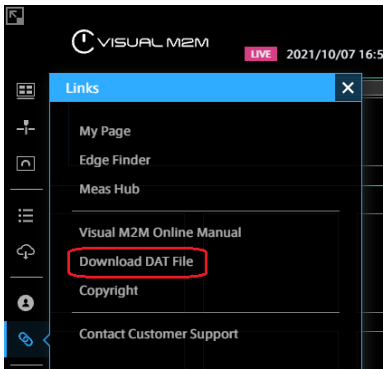


Fig. 22 Opening the DAT file download page

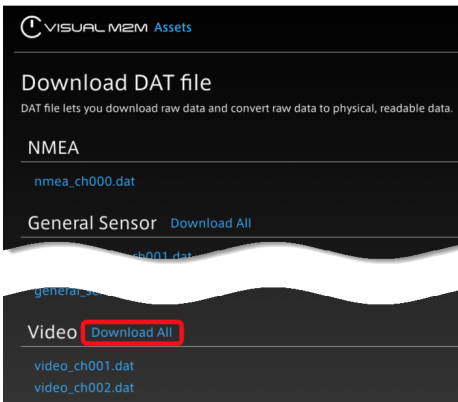


Fig. 23 Downloading DAT file for video

3. Click [Data Settings] > [Import] to import the downloaded data settings file.

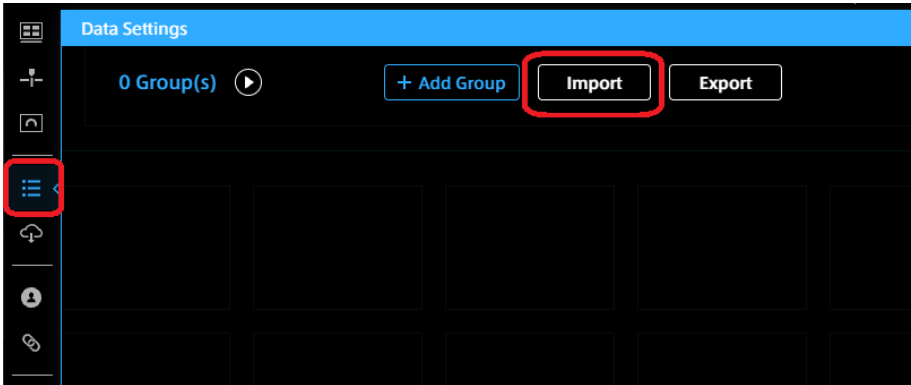


Fig. 24 Import DAT file

If the import is successful, a group called "Video" will be created as shown below.

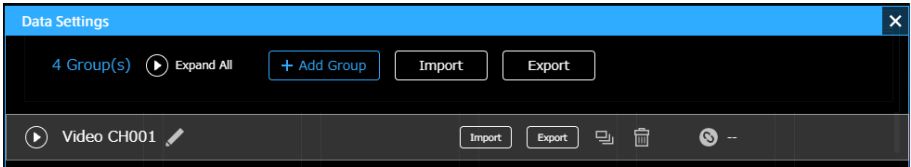


Fig. 25 Video group

4. Create a panel on the dashboard.

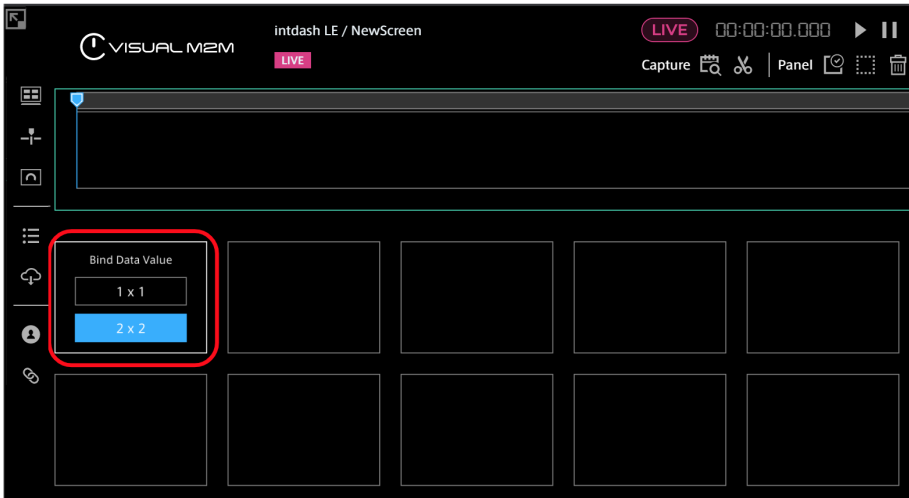


Fig. 26 Creating a panel

5. Click the panel to set the data source `edge1` and bind `Video:ch_001`.

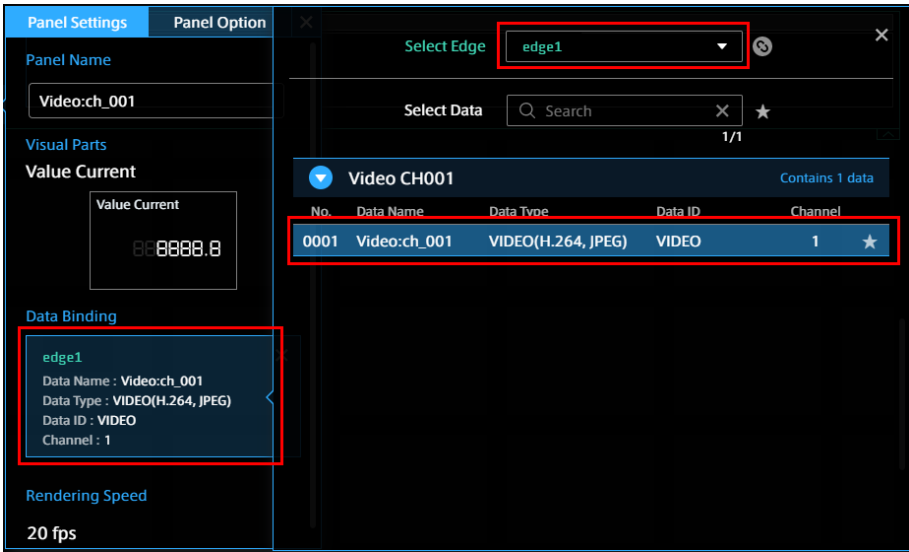


Fig. 27 Select the data source edge and bind the video data

6. Select Video Player as a visual part.

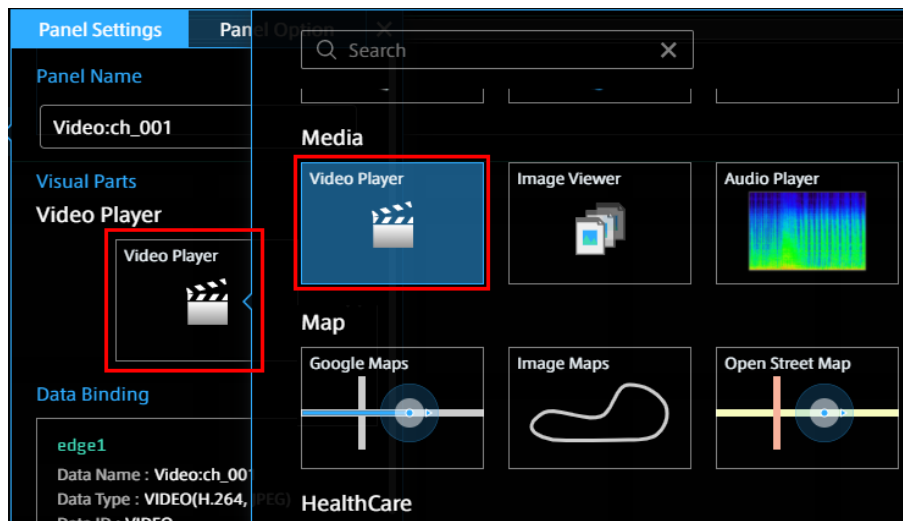


Fig. 28 Selecting Video Player

4.4 Send real-time video from iPhone

Launch the intdash Motion application on your iPhone and send the video.

1. Launch intdash Motion on your iPhone and sign in.

Use the following information to sign in:

- URL: URL of the intdash instance <https://intdash.example.com>
- User name: user1
- Password: the password set for user1

Note: Sign in as a user (user1), not as an edge.

2. Open [Settings] and configure as follows.

- [Send Data As]: edge1
- [Video]: On
 - [Stream to Server]: On
 - [Save to Server]: On
 - [Channel]: 1
 - [Codec and Options]: H.264

3. Return to the Main screen and tap ► to start shooting and sending.

4.5 Visualize the data with Data Visualizer

1. Set Data Visualizer to LIVE mode and click [Play].

The video being shot with intdash Motion is displayed in real time.



2. After confirming the operation, tap ■ in intdash Motion to end the measurement.

3. In Data Visualizer, click [Stored Data] > [Measurements] to see the previously acquired data.
The video you took in the above procedure should be displayed. Click the data to play.

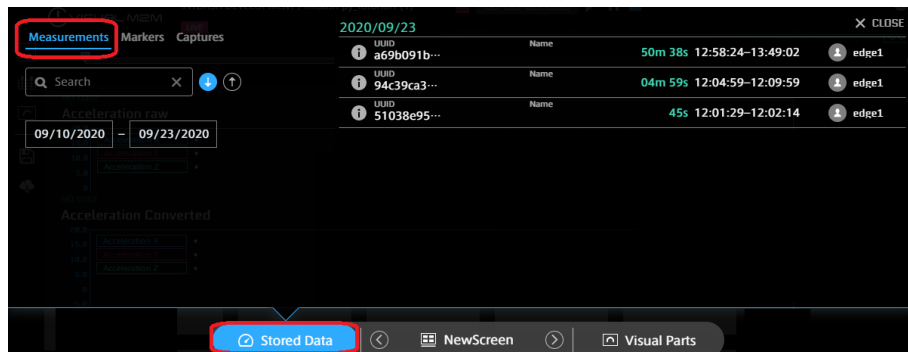


Fig. 29 Playing from a list of previously acquired data

This is the end of the operation check.

05 Appendix: Services and configuration files in intdash

The following is a list of services that make up intdash, their configuration files, the ports they use, and their purposes.

5.1 Original applications

The original applications that make up intdash are as follows.

- **intdash-api**
 - Configuration file: `/etc/intdash/intdashd.conf`
 - Service port: 8097/tcp, 8179/tcp
 - Purpose: API service for intdash. Also works as a proxy to various microservices.
- **intdash-micro-auth**
 - Configuration file: `/etc/intdash/authd.conf`
 - Service port: 8094/tcp
 - Purpose: Microservice for authentication
- **intdash-micro-broker**
 - Configuration file: `/etc/intdash/brokerd.conf`
 - Service port: 8180/tcp, 8178/tcp
 - Purpose: Microservice for real-time processing of measurement data
- **intdash-micro-measurement**
 - Configuration file: `/etc/intdash/measurementd.conf`
 - Service port: 8095/tcp
 - Purpose: Microservice for storing measurement data and loading stored measurements
- **intdash-micro-config**
 - Configuration file: `/etc/intdash/configd.conf`
 - Service port: 8096/tcp
 - Purpose: Microservice for managing smartphone app settings
- **intdash-micro-media**
 - Configuration file: `/etc/intdash/mediad.toml`
 - Service port: 8085/tcp
 - Purpose: Service for H.264 video data
- **intdash-web-oauth2-redirector**
 - Configuration file: `/etc/sysconfig/intdash-web-oauth2-redirector`
 - Service port: UNIX domain socket
 - `/var/run/intdash/intdash-web-oauth2-redirector.sock`
 - Purpose: Generates redirect URLs for OAuth2 authentication client applications.

- **intdash-web-signin-redirector**
 - Configuration file: `/etc/sysconfig/intdash-web-signin-redirector`
 - Service port: UNIX domain socket
 - `/var/run/intdash/intdash-web-signin-redirector.sock`
 - Purpose: Generates redirect URLs for the sign-in function in the authentication system.
- **intdash-web-me**
 - Configuration file: `/etc/sysconfig/intdash-web-me`
 - Service port: 13000/tcp
 - Purpose: "My Page", a web application that displays information about the user you are logged in as.
- **intdash-web-edges**
 - Configuration file: `/etc/sysconfig/intdash-web-edges`
 - Service port: 13001/tcp
 - Purpose: "Edge Finder", a web application for displaying information about the edges.
- **intdash-web-oauth2**
 - Configuration file: `/etc/sysconfig/intdash-web-oauth2`
 - Service port: 13003/tcp
 - Purpose: Service for client-side processing in OAuth2 authentication
- **intdash-web-measurements**
 - Configuration file: `/etc/sysconfig/intdash-web-measurements`
 - Service port: 13004/tcp
 - Purpose: "Meas Hub", a web application to manage measurements
- **intdash-web-widget-app-links**
 - Configuration file: `/etc/sysconfig/intdash-web-widget-app-links`
 - Service port: 13005/tcp
 - Purpose: Service for generating links to other web applications (links via application icons)
- **intdash-web-admin**
 - Configuration file: `/etc/sysconfig/intdash-web-admin`
 - Service port: 13006/tcp
 - Purpose: "Admin Console", a web application for user and edge management
- **vm2m-dataviz-backend**
 - Configuration file: `/etc/sysconfig/vm2m-dataviz-backend`
 - Service port: UNIX domain socket
 - Purpose: Service to store the settings of the measurement visualization application

5.2 Open source applications

intdash uses the following OSS (Open Source Software).

- **Nginx - Web server** (<https://nginx.org/en/>)
 - Configuration file: Under `/etc/nginx` (intdash web server settings: `conf.d/intdash.conf`, application settings: under `intdash.d`)
 - Service port: 80/tcp, 443/tcp
- **Edge router Traefik** (<https://doc.traefik.io/traefik/>)
 - Configuration file: Under `/etc/intdash` (main configuration: `traefik.toml`, configuration for each application: under `traefik.d`)
 - Service port: 8080/tcp, 18080/tcp
- **PostgreSQL - Relational database** (<https://www.postgresql.org/>)
 - Configuration file: Under `/var/lib/pgsql/data`
 - Service port: UNIX domain socket: (Peer authentication: 5432)
- **InfluxDB - Time series database** (<https://www.influxdata.com/>)
 - Configuration file: `/etc/influxdb/influxdb.conf`
 - Service port: 8086/tcp

06 Appendix: Software update procedure if you are using an older version of AMI

If you are using an earlier version of AMI, you can obtain and install a newer version of the software from the aptpod package repository.

To use the aptpod package repository, you need to apply in advance.

6.1 Prepare for an update

In order to update the software, you need to get the credentials of the repository and configure the repository definition file in your intdash instance.

6.1.1 Apply to use the package repository

Apply to use the aptpod package repository and get the credentials.

1. Using a web browser, go to <https://www.aptpod.co.jp/request-auth-info/>, where you can apply for the credentials.
2. Fill in the required information and submit it.
3. An email will be sent to you to inform you that your request has been accepted. Please wait until you receive the authentication information from aptpod.

6.1.2 Set up authentication information

Once you have received a user name and password to access the package repository, set the credentials in the repository definition file in your intdash instance.

1. Connect to the intdash instance via SSH and navigate to the repository definition file directory.

```
# cd /etc/yum.repos.d
```

2. Open the repository definition file in a text editor.

```
# vi intdash.repo
```

3. Edit the following. Then close the configuration file.

- [intdash] section
 - username
 - password
 - enabled

```
[intdash]
...
username=<user_name>
password=<password>
enabled=1
```

4. Verify that you can access the repository successfully.

```
# yum search intdash
```

If the repository is successfully accessed, the package list will be displayed.

6.2 Update AMI version 202009 software to the version 202103 equivalent

If you are using an instance created using AMI version 202009, you can update the software to the version 202103 equivalent by following the steps below.

Note: The following procedure will update the software in the instance, but will not change the version name of the AMI displayed in the EC2 console.

1. Connect to the intdash instance via SSH.
2. Install the new version of the software and new software to be added.

```
# yum install \  
intdash-api \  
intdash-client \  
intdash-media-h264 \  
intdash-micro-auth \  
intdash-micro-broker \  
intdash-micro-config \  
intdash-micro-measurement \  
intdash-nginx \  
intdash-web-edges \  
intdash-web-edges-admin \  
intdash-web-email \  
intdash-web-me \  
intdash-web-measurements \  
intdash-web-oauth2 \  
intdash-web-password \  
intdash-web-shared-assets \  
intdash-web-signin \  
intdash-web-utility-tools \  
intdash-web-widget-app-links \  
vm2m-assets \  
vm2m-dataviz \  
vm2m-dataviz-backend \  
vm2m-dataviz-plugins-core
```

If the configuration file needs to be updated, a warning will be displayed.

3. Update the configuration file.

If you need to update the configuration file, a new default configuration file with the extension `.rpmnew` will be created at the same level as the existing configuration file. Compare the existing configuration file with the `.rpmnew` file and apply the new settings to the configuration file. Refer to [Set up the intdash server](#) (p. 10) for more information about the configuration.

Note: For example, if a new configuration file named `intdash.conf.rpmnew` is created in the same directory as the configuration file `intdash.conf`, you can do the following

1. Check the differences between the current configuration file `intdash.conf` and the new configuration file `intdash.conf.rpmnew`.
2. Check the changes made in `intdash.conf.rpmnew` and reflect them in `intdash.conf`.
3. Remove `intdash.conf.rpmnew`.

4. Execute the following commands to update the database schema.

```
# sudo -u intdash intdashd db migrate -c /etc/intdash/intdashd.conf
# sudo -u intdash authd db migrate -c /etc/intdash/authd.conf
# sudo -u intdash measurementd db migrate -c /etc/intdash/measurementd.conf
# sudo -u intdash configd db migrate -c /etc/intdash/configd.conf
# sudo -u postgres psql -d template1 -c 'CREATE DATABASE "intdash-micro-broker" OWNER "intdash";'
# sudo -u intdash brokerd db migrate -c /etc/intdash/brokerd.conf
# sudo -u intdash mediad-h264 db migrate -c /etc/intdash/mediad-h264.toml
```

5. Execute the following queries to update the contents of the database. (This update is necessary because some of the functional limitations have been removed in version 202103.)

```
# sudo -u intdash psql -d intdash-micro-auth
intdash-micro-auth=> UPDATE "setting_master" SET "edge_limit" = 50;
intdash-micro-auth=> UPDATE "edge_authority" SET "authority" = '{"allows":["**:*"],"denys":[]}' WHERE_
↪edge_id = (SELECT id FROM "edge" WHERE uuid = 'a95d8502-7814-487c-8f8c-f774b55b20d2');

# sudo -u intdash psql -d intdash-api
intdash-api=> DELETE FROM "config" WHERE name IN ('vm2m-2nd-config', 'vm2m-2nd-visual-parts');
```

6. Execute the following commands to configure the added services to start automatically. (Newly added services will not be started automatically by default.)

```
# systemctl enable intdash-micro-broker.service
# systemctl enable intdash-web-edges.service
# systemctl enable intdash-web-edges-admin.service
# systemctl enable intdash-web-measurements.service
# systemctl enable vm2m-dataviz-backend.service
```

7. Execute the following commands to reload and restart the services.

```
# systemctl reload nginx.service
# systemctl restart intdash-api.service
# systemctl restart intdash-micro-auth.service
# systemctl restart intdash-micro-measurement.service
# systemctl restart intdash-micro-config.service
# systemctl restart intdash-micro-broker.service
# systemctl restart intdash-media-h264.service
# systemctl restart intdash-web-me.service
# systemctl restart intdash-web-edges.service
# systemctl restart intdash-web-edges-admin.service
# systemctl restart intdash-web-measurements.service
# systemctl restart intdash-web-widget-app-links.service
# systemctl restart intdash-web-oauth2.service
# systemctl restart intdash-web-oauth2-redirector.service
# systemctl restart intdash-web-signin-redirector.service
# systemctl restart vm2m-dataviz-backend.service
```

This completes the procedure to update the software of AMI version 202009 to the version 202103 equivalent.

6.3 Update AMI version 202103 software to the version 202106 equivalent

If you are using an instance created using AMI version 202103, you can update the software to the version 202106 equivalent by following the steps below.

Note: The following procedure will update the software in the instance, but will not change the version name of the AMI displayed in the EC2 console.

1. Connect to the intdash instance via SSH.
2. Install the new version of the software.

```
# yum install \  
intdash-api \  
intdash-micro-broker \  
intdash-micro-measurement \  
intdash-web-edges \  
intdash-web-edges-admin \  
intdash-web-email \  
intdash-web-me \  
intdash-web-measurements \  
intdash-web-oauth2 \  
intdash-web-password \  
intdash-web-shared-assets \  
intdash-web-signin \  
intdash-web-widget-app-links \  
vm2m-assets \  
vm2m-dataviz \  
vm2m-dataviz-backend \  
vm2m-dataviz-plugins-core
```

If the configuration file needs to be updated, a warning will be displayed.

3. Update the configuration file.

If you need to update the configuration file, a new default configuration file with the extension `.rpmnew` will be created at the same level as the existing configuration file. Compare the existing configuration file with the `.rpmnew` file and apply the new settings to the configuration file. Refer to [Set up the intdash server](#) (p. 10) for more information about the configuration.

Note: For example, if a new configuration file named `intdash.conf.rpmnew` is created in the same directory as the configuration file `intdash.conf`, you can do the following

1. Check the differences between the current configuration file `intdash.conf` and the new configuration file `intdash.conf.rpmnew`.
2. Check the changes made in `intdash.conf.rpmnew` and reflect them in `intdash.conf`.
3. Remove `intdash.conf.rpmnew`.

4. Execute the following commands to update the database schema.

```
# sudo -u intdash intdashd db migrate -c /etc/intdash/intdashd.conf  
# sudo -u intdash measurementd db migrate -c /etc/intdash/measurementd.conf  
# sudo -u intdash brokerd db migrate -c /etc/intdash/brokerd.conf
```

5. Execute the following commands to reload and restart the services.

```
# systemctl restart intdash-api.service  
# systemctl restart intdash-micro-measurement.service
```

(continues on next page)

(continued from previous page)

```
# systemctl restart intdash-micro-broker.service
# systemctl restart intdash-web-me.service
# systemctl restart intdash-web-edges.service
# systemctl restart intdash-web-edges-admin.service
# systemctl restart intdash-web-measurements.service
# systemctl restart intdash-web-widget-app-links.service
# systemctl restart intdash-web-oauth2.service
# systemctl restart intdash-web-oauth2-redirector.service
# systemctl restart intdash-web-signin-redirector.service
# systemctl restart vm2m-dataviz-backend.service
```

This completes the procedure to update the software of AMI version 202103 to the version 202106 equivalent.

6.4 Update AMI version 202106 software to the version 202112 equivalent

If you are using an instance created using AMI version 202106, you can update the software to the version 202112 equivalent by following the steps below. The following steps should be performed on the intdash instance connected via SSH.

Note: The following procedure will update the software in the instance, but will not change the version name of the AMI displayed in the EC2 console.

6.4.1 Update the repository definition file

1. Open the repository definition file in a text editor.

```
# vi /etc/yum.repos.d/intdash.repo
```

2. Edit the following. Then close the configuration file.

- [intdash] and [intdash-testing] sections
- baseurl

```
[intdash]
...
baseurl=https://private-repository.aptpod.jp/intdash/202112/linux/rpm
...

[intdash-testing]
...
baseurl=https://private-repository.aptpod.jp/intdash/202112/testing/linux/rpm
...
```

3. Remove package management metadata.

```
# yum clean metadata
```

6.4.2 Update and delete software

1. Install the new version of the software and new software to be added.

```
# yum install \  
intdash-api \  
intdash-micro-auth \  
intdash-micro-broker \  
intdash-micro-config \  
intdash-micro-measurement \  
intdash-micro-media \  
intdash-nginx \  
intdash-web-admin \  
intdash-web-edges \  
intdash-web-email \  
intdash-web-me \  
intdash-web-measurements \  
intdash-web-oauth2 \  
intdash-web-password \  
intdash-web-shared-assets \  
intdash-web-signin \  
intdash-web-utility-tools \  
intdash-web-widget-app-links \  
vm2m-assets \  
vm2m-dataviz \  
vm2m-dataviz-backend \  
vm2m-dataviz-plugins-core
```

If the configuration file needs to be updated, a warning will be displayed.

2. Delete software that is no longer needed.

```
# yum remove intdash-client
```

3. Update the configuration file.

If you need to update the configuration file, a new default configuration file with the extension `.rpmnew` will be created at the same level as the existing configuration file. Compare the existing configuration file with the `.rpmnew` file and apply the new settings to the configuration file. Refer to [Set up the intdash server](#) (p. 10) for more information about the configuration.

Note: For example, if a new configuration file named `intdash.conf.rpmnew` is created in the same directory as the configuration file `intdash.conf`, you can do the following

1. Check the differences between the current configuration file `intdash.conf` and the new configuration file `intdash.conf.rpmnew`.
2. Check the changes made in `intdash.conf.rpmnew` and reflect them in `intdash.conf`.
3. Remove `intdash.conf.rpmnew`.

Some software names have been changed. Running the update will remove the software with the old name. In some cases, however, the configuration files of the old software may remain with the `.rpmsave` extension added. You can refer to these old configuration files when configuring the new software. After that, you may delete the old configuration files.

Note: The following software has been renamed.

- `intdash-micro-media` (formerly: `intdash-media-h264`)
- `intdash-web-admin` (formerly: `intdash-web-edges-admin`)

6.4.3 Update the database schema of the Auth Service

In the update from AMI version 202106 to 202112, the major version of Auth Service has changed. You will need to run the migration tool to migrate the database schema.

1. Execute the following command to stop the Auth Service.

```
# systemctl stop intdash-micro-auth.service
```

2. Navigate to the directory where the Auth Service auxiliary commands are located.

```
# cd /usr/libexec/intdash-micro-auth
```

3. Use authd-v1.9.0 to update the database schema for version 1.9.0.

```
# sudo -u intdash ./authd-v1.9.0 db migrate -c /etc/intdash/authd.conf
```

4. Execute the following command to rename the existing database. Also, create a new database with the same name as the existing one.

```
# sudo -u postgres psql -d template1 -c 'ALTER DATABASE "intdash-micro-auth" RENAME TO "intdash-micro-  
↪auth-v1"'  
# sudo -u postgres psql -d template1 -c 'CREATE DATABASE "intdash-micro-auth" OWNER "intdash"'
```

5. Use authd-v2.0.0 to make the database schema of the new database for version 2.0.0.

```
# sudo -u intdash ./authd-v2.0.0 db migrate -c /etc/intdash/authd.conf
```

6. Create a configuration file for the migration tool with a text editor.

```
# vi /tmp/auth-mig.conf
```

7. Create a configuration file with the following contents. Then, close the configuration file.

```
dry-run = false  
  
[src]  
  driver = "postgres"  
  address = "/var/run/postgresql"  
  dbname = "intdash-micro-auth-v1"  
  username = "intdash"  
  password = ""  
  
[dest]  
  driver = "postgres"  
  address = "/var/run/postgresql"  
  dbname = "intdash-micro-auth"  
  username = "intdash"  
  password = ""
```

8. Run the migration tool.

```
# sudo -u intdash ./auth-mig run -c /tmp/auth-mig.conf
```

9. Update the database schema.

```
# sudo -u intdash authd db migrate -c /etc/intdash/authd.conf
```

10. Execute the following command to restart the Auth Service.

```
# systemctl start intdash-micro-auth.service
```

6.4.4 Perform migrations required for software updates

1. Execute the following commands to update the database schema except for the Auth Service.

```
# sudo -u intdash intdashd db migrate -c /etc/intdash/intdashd.conf
# sudo -u intdash brokerd db migrate -c /etc/intdash/brokerd.conf
# sudo -u intdash configd db migrate -c /etc/intdash/configd.conf
# sudo -u intdash measurementd db migrate -c /etc/intdash/measurementd.conf
# sudo -u intdash mediad db migrate -c /etc/intdash/mediad.toml
```

2. Execute the following command to migrate the video files in Media Service.

```
# sudo -u intdash mediad migrate -c /etc/intdash/mediad.toml
```

6.4.5 Configure your own visual parts for the Data Visualizer (only if necessary)

Note: This step is only required if you are using your own visual parts developed using the Data Visualizer Visual Parts SDK.

In AMI version 202112, the structure of the Data Visualizer configuration file has been changed. If you were using custom visual parts in a previous version, you can still use the visual parts, but you will need to create a new configuration file.

To use custom visual parts, add a visual part configuration file like the following to the `/etc/vm2m-dataviz-backend/visual-parts-plugins/` directory.

For ease of management, it is recommended to name the configuration file `{number}.{simple_name}.json`.

Example: `/etc/vm2m-dataviz-backend/visual-parts-plugins/99.visual-parts-sample.json`

```
{
  "items": [
    {
      "url": "/vm2m/data-visualizer/plugins/visual-parts-sample/app.js",
      "avoidSendingPartSpecificMetadata": true
    }
  ]
}
```


url Enter the URL of your own visual part (which was set in the `pluginURLs` field in `vm2m-2nd-visual-parts.production.json` in previous versions).

avoidSendingPartSpecificMetadata

When set to `true`, information specific to this visual part (such as the visual part name) will be excluded from the collection of usage information. Normally, you should set this to `true`.

For more information on visual part configuration files, see the deployment instructions in [Creating visual parts for Data Visualizer with Visual Parts SDK](#).

6.4.6 Configure your own links in Data Visualizer (only if necessary)

Note: This step is only necessary if you have customized the links displayed in the Link screen () of Data Visualizer.

In AMI version 202112, the structure of the Data Visualizer configuration file has been changed. If you have configured your own links in previous versions, you will need to create a new configuration file.

To use your own links, add a link configuration file like the following to the `/etc/vm2m-dataviz-backend/links/` directory.

For ease of management, it is recommended to name the configuration file `{number}.{simple_name}.json`.

Example: `/etc/vm2m-dataviz-backend/links/900.100.vm2m-dataviz.json`

```
{
  "items": [
    {
      "label": "",
      "href": "",
      "type": "separator"
    },
    {
      "label": "Visual M2M Online Manual",
      "href": "https://docs.intdash.jp/manual/data-visualizer-operation/ed8/ja/data-visualizer-operation-japdf",
      "type": "manual"
    },
    ...
  ]
}
```

label

Label

href URL or path of the target

type If you want to display a separator, or if you want to display the link in an error occur message, set the following.

- separator: Display a separator. Set { "label": "", "href": "", "type": "separator" } to show the separator.
- support: Displayed as contact information. This link will also be displayed when an error occurs.

If strings other than these are set, it will not affect the display or functionality.

The configuration files are loaded in the order of their file names, and the links are displayed in that order.

6.4.7 Configure application links (only if necessary)

Note: This step is only necessary if you have customized the links to other applications ().

In AMI version 202112, the structure of the link settings file has been changed. If you had your own link configuration in previous versions, you will need to create a new configuration file.

To add your own links, add a configuration file like the following to the `/etc/intdash-web-widget-app-links/` directory.

Example: `/etc/intdash-web-widget-app-links/100.100.vm2m-dataviz.json`

```
{
  "items": [
    {
      "iconUrl": "/share/image/icon_vm2m.svg",
      "href": "/vm2m/",
      "label_ja-JP": "Visual M2M Data Visualizer",
      "label_en-US": "Visual M2M Data Visualizer"
    }
  ]
}
```

iconUrl

URL or path to the icon

href URL or path of the target

label_ja-JP

Label in Japanese

label_en-US

Label in English

The configuration files are loaded in the order of their file names, and the links are displayed in that order.

6.4.8 Configure service-related settings

1. Execute the following commands to configure the added services to start automatically. (Newly added services will not be started automatically by default.)

```
# systemctl enable intdash-api-auth.service
# systemctl enable intdash-api-gateway.service
# systemctl enable intdash-micro-media.service
# systemctl enable intdash-web-admin.service
```

2. Execute the following commands to reload and restart the services.

```
# systemctl reload nginx.service
# systemctl restart intdash-api.service
# systemctl restart intdash-api-auth.service
# systemctl restart intdash-api-gateway.service
# systemctl restart intdash-micro-auth.service
# systemctl restart intdash-micro-broker.service
# systemctl restart intdash-micro-config.service
# systemctl restart intdash-micro-measurement.service
# systemctl restart intdash-micro-media.service
```

(continues on next page)

(continued from previous page)

```
# systemctl restart intdash-web-admin.service
# systemctl restart intdash-web-edges.service
# systemctl restart intdash-web-measurements.service
# systemctl restart intdash-web-me.service
# systemctl restart intdash-web-oauth2-redirector.service
# systemctl restart intdash-web-oauth2.service
# systemctl restart intdash-web-signin-redirector.service
# systemctl restart intdash-web-widget-app-links.service
# systemctl restart vm2m-dataviz-backend.service
```

This completes the procedure to update the software of AMI version 202106 to the version 202112 equivalent.

6.5 Update AMI version 202112 software to the version 202203 equivalent

If you are using an instance created using AMI version 202112, you can update the software to the version 202203 equivalent by following the steps below. The following steps should be performed on the intdash instance connected via SSH.

Note: The following procedure will update the software in the instance, but will not change the version name of the AMI displayed in the EC2 console.

6.5.1 Update the repository definition file

1. Open the repository definition file in a text editor.

```
# vi /etc/yum.repos.d/intdash.repo
```

2. Edit the following. Then close the configuration file.

- [intdash] and [intdash-testing] sections
- baseurl

```
[intdash]
...
baseurl=https://private-repository.aptpod.jp/intdash/202203/linux/rpm
...

[intdash-testing]
...
baseurl=https://private-repository.aptpod.jp/intdash/202203/testing/linux/rpm
...
```

3. Remove package management metadata.

```
# yum clean metadata
```


6.5.2 Update and delete software

1. Install the new version of the software and new software to be added.

```
# yum install \
  intdash-api \
  intdash-ffmpeg \
  intdash-micro-auth \
  intdash-micro-broker \
  intdash-micro-config \
  intdash-micro-measurement \
  intdash-micro-media \
  intdash-nginx \
  intdash-web-admin \
  intdash-web-edges \
  intdash-web-email \
  intdash-web-me \
  intdash-web-measurements \
  intdash-web-media \
  intdash-web-oauth2 \
  intdash-web-password \
  intdash-web-shared-assets \
  intdash-web-signin \
  intdash-web-widget-app-links \
  vm2m-assets \
  vm2m-dataviz \
  vm2m-dataviz-backend \
  vm2m-dataviz-plugins-core
```

If the configuration file needs to be updated, a warning will be displayed.

2. Delete software that is no longer needed.

```
# yum remove intdash-web-utility-tools
```

3. Uninstall previously installed FFmpeg that is no longer needed.

Remove the ffmpeg command installed when setting up AMI version 202112 or earlier, as it is no longer needed.

```
# rm -i /usr/local/bin/ffmpeg
```

4. Update the configuration file.

If you need to update the configuration file, a new default configuration file with the extension `.rpmnew` will be created at the same level as the existing configuration file. Compare the existing configuration file with the `.rpmnew` file and apply the new settings to the configuration file. Refer to [Set up the intdash server](#) (p. 10) for more information about the configuration.

Note: For example, if a new configuration file named `intdash.conf.rpmnew` is created in the same directory as the configuration file `intdash.conf`, you can do the following

1. Check the differences between the current configuration file `intdash.conf` and the new configuration file `intdash.conf.rpmnew`.
2. Check the changes made in `intdash.conf.rpmnew` and reflect them in `intdash.conf`.
3. Remove `intdash.conf.rpmnew`.

6.5.3 Perform migrations required for software updates

1. Execute the following commands to update the database schema.

```
# sudo -u intdash intdashd db migrate -c /etc/intdash/intdashd.conf
# sudo -u intdash authd db migrate -c /etc/intdash/authd.conf
# sudo -u intdash brokerd db migrate -c /etc/intdash/brokerd.conf
# sudo -u intdash configd db migrate -c /etc/intdash/configd.conf
# sudo -u intdash measurementd db migrate -c /etc/intdash/measurementd.conf
# sudo -u intdash mediad db migrate -c /etc/intdash/mediad.toml
```

2. Execute the following command to migrate the video files in Media Service.

```
# sudo -u intdash mediad ts2fmp4 -c /etc/intdash/mediad.toml
```

6.5.4 Configure service-related settings

1. Execute the following commands to configure the added services to start automatically. (Newly added services will not be started automatically by default.)

```
# systemctl enable intdash-web-media.service
```

2. Execute the following commands to reload and restart the services.

```
# systemctl reload nginx.service
# systemctl restart intdash-api.service
# systemctl restart intdash-api-auth.service
# systemctl restart intdash-api-gateway.service
# systemctl restart intdash-micro-auth.service
# systemctl restart intdash-micro-broker.service
# systemctl restart intdash-micro-config.service
# systemctl restart intdash-micro-measurement.service
# systemctl restart intdash-micro-media.service
# systemctl restart intdash-web-admin.service
# systemctl restart intdash-web-edges.service
# systemctl restart intdash-web-me.service
# systemctl restart intdash-web-measurements.service
# systemctl restart intdash-web-media.service
# systemctl restart intdash-web-oauth2-redirector.service
# systemctl restart intdash-web-oauth2.service
# systemctl restart intdash-web-signin-redirector.service
# systemctl restart intdash-web-widget-app-links.service
# systemctl restart vm2m-dataviz-backend.service
```

This completes the procedure to update the software of AMI version 202112 to the version 202203 equivalent.

6.6 Update AMI version 202203 software to the version 202206 equivalent

If you are using an instance created using AMI version 202203, you can update the software to the version 202206 equivalent by following the steps below. The following steps should be performed on the intdash instance connected via SSH.

Note: The following procedure will update the software in the instance, but will not change the version name of the AMI displayed in the EC2 console.

6.6.1 Update the repository definition file

1. Open the repository definition file in a text editor.

```
# vi /etc/yum.repos.d/intdash.repo
```

2. Edit the following. Then close the configuration file.

- [intdash] and [intdash-testing] sections
 - baseurl

```
[intdash]
...
baseurl=https://private-repository.aptpod.jp/intdash/202206/linux/rpm
...

[intdash-testing]
...
baseurl=https://private-repository.aptpod.jp/intdash/202206/testing/linux/rpm
...
```

3. Remove package management metadata.

```
# yum clean metadata
```

6.6.2 Update software

1. Install the new version of the software.

```
# yum install \
  intdash-api \
  intdash-ffmpeg \
  intdash-micro-auth \
  intdash-micro-broker \
  intdash-micro-config \
  intdash-micro-measurement \
  intdash-micro-media \
  intdash-nginx \
  intdash-web-admin \
  intdash-web-edges \
  intdash-web-email \
  intdash-web-me \
  intdash-web-measurements \
  intdash-web-media \
  intdash-web-oauth2 \
  intdash-web-password \
```

(continues on next page)

(continued from previous page)

```
intdash-web-shared-assets \
intdash-web-signin \
intdash-web-widget-app-links \
vm2m-assets \
vm2m-dataviz \
vm2m-dataviz-backend \
vm2m-dataviz-plugins-core
```

If the configuration file needs to be updated, a warning will be displayed.

2. Update the configuration file.

If you need to update the configuration file, a new default configuration file with the extension `.rpmnew` will be created at the same level as the existing configuration file. Compare the existing configuration file with the `.rpmnew` file and apply the new settings to the configuration file. Refer to [Set up the intdash server](#) (p. 10) for more information about the configuration.

Note: For example, if a new configuration file named `intdash.conf.rpmnew` is created in the same directory as the configuration file `intdash.conf`, you can do the following

1. Check the differences between the current configuration file `intdash.conf` and the new configuration file `intdash.conf.rpmnew`.
2. Check the changes made in `intdash.conf.rpmnew` and reflect them in `intdash.conf`.
3. Remove `intdash.conf.rpmnew`.

6.6.3 Perform migrations required for software updates

1. Execute the following commands to update the database schema.

```
# sudo -u intdash intdashd db migrate -c /etc/intdash/intdashd.conf
# sudo -u intdash authd db migrate -c /etc/intdash/authd.conf
# sudo -u intdash brokerd db migrate -c /etc/intdash/brokerd.conf
# sudo -u intdash configd db migrate -c /etc/intdash/configd.conf
# sudo -u intdash measurementd db migrate -c /etc/intdash/measurementd.conf
# sudo -u intdash mediad db migrate -c /etc/intdash/mediad.toml
```

6.6.4 Configure service-related settings

1. Execute the following commands to reload and restart the services.

```
# systemctl reload nginx.service
# systemctl restart intdash-api.service
# systemctl restart intdash-api-auth.service
# systemctl restart intdash-api-gateway.service
# systemctl restart intdash-micro-auth.service
# systemctl restart intdash-micro-broker.service
# systemctl restart intdash-micro-config.service
# systemctl restart intdash-micro-measurement.service
# systemctl restart intdash-micro-media.service
# systemctl restart intdash-web-admin.service
# systemctl restart intdash-web-edges.service
# systemctl restart intdash-web-me.service
# systemctl restart intdash-web-measurements.service
# systemctl restart intdash-web-media.service
# systemctl restart intdash-web-oauth2-redirector.service
```

(continues on next page)

(continued from previous page)

```
# systemctl restart intdash-web-oauth2.service
# systemctl restart intdash-web-signin-redirector.service
# systemctl restart intdash-web-widget-app-links.service
# systemctl restart vm2m-dataviz-backend.service
```

This completes the procedure to update the software of AMI version 202203 to the version 202206 equivalent.