

# Introduction

## 1. About this Document

- This document describes the information necessary to configure an HA cluster by combining two ITA all-in-one installed servers and OSS products. By turning the ITA into an HA Cluster, users can construct an ITA Server with more reliability and availability than what would be possible with a single Linux server.
- HA clusters configured with the use of this document will have the following functions.
  - The HA Cluster is constructed on 2 server systems. They can be either physical servers or vertical servers.
  - One of the server will have an ITA Server running and will have ITA Web/AP, Backyard, DBMS, Ansible Driver and other ITA Functions
  - If the Server that provides the service being used gets turned off, the other server will start ITA a minute later and will continue the service. The database data does not get deleted if this happens.
- Please see the diagram in "4. System construction example" for Deploy example.  
Make sure to change the IP Addresss, Host Name , Directory name and File name to match the users environment.
- In this document, an environment with the following will be constructed.

Main software	Application	Verified version
Red Hat Enterprise Linux release	OS	8.4
Exastro IT Automation	Exastro IT Automation	1.9.1
Pacemaker	Cluster control	2.1.0-8.el8
Corosync	Heartbeat	3.1.5
pcs	Pacemaker command	0.10.10
DRBD	Replication	9.17.0-1

## 2. Scope of responsibility

- This document is intended to provide reference information on precautions and configuration examples for clustering ITA servers, and does not guarantee the operation of said information.

## 3. Linux-HA Cluster stack

- Linux-HA Cluster stack is a software that aims to improve the server systems availability and prevent data loss from storage failure. It is constructed from the following 3 softwares and their respective packages.

### DRBD (Distributed Replicated Block Device)

Writes the same data to both of the server's storage (Partition or Logical volume) in real time (Real-time replication)

Hence, if one of the servers or the storage itself fails, the service can be continued on a normally functioning server without losing data.

### Corosync

A software that mutually monitors the normal operation of the servers that makes up the HA cluster.

### Pacemaker

Monitors the services provided by the HA Cluster and moves the service to another server if the first server shuts down.

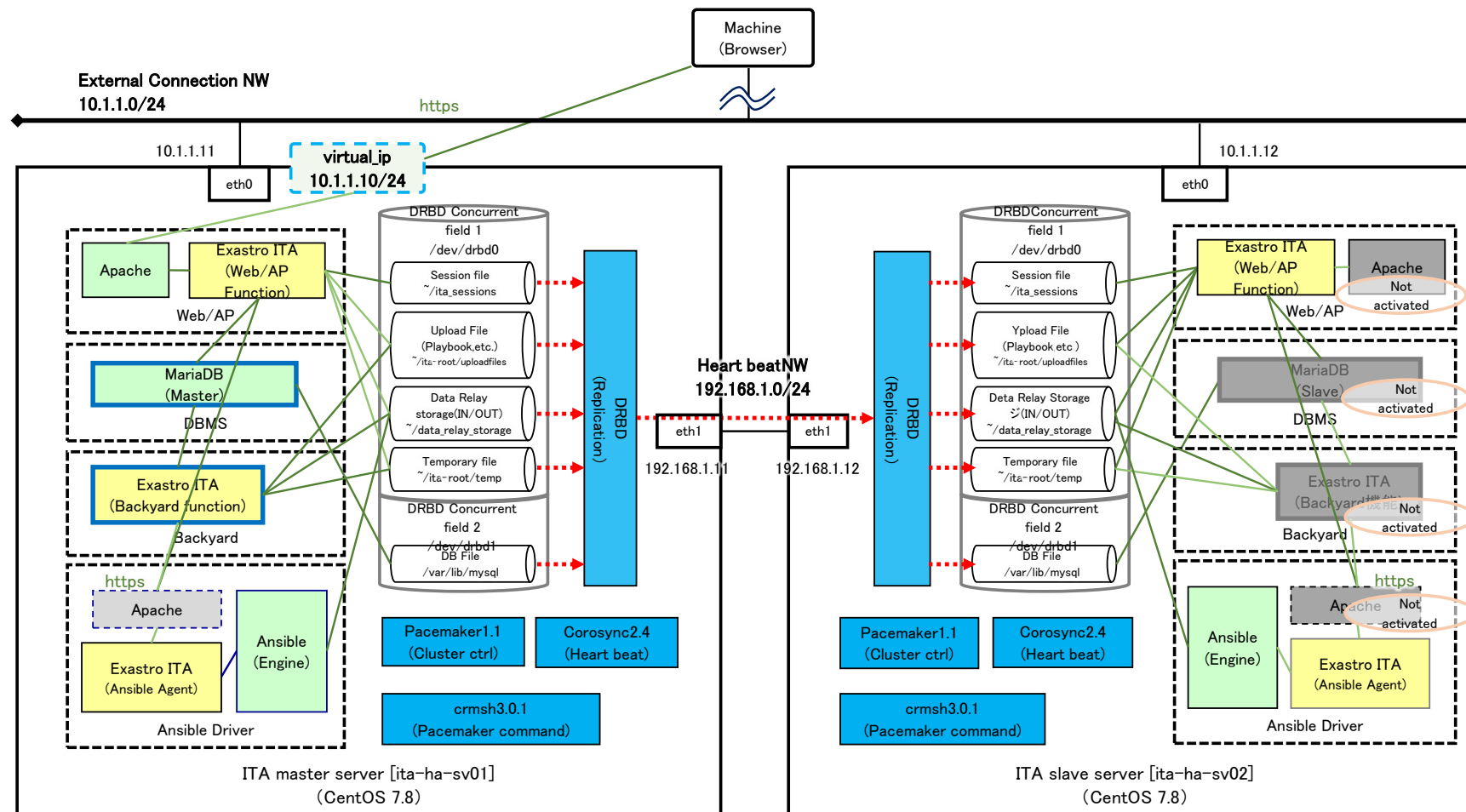
### Reference URL:

<http://linux-ha.osdn.jp/wp/>

# Introduction

## 4. System construction ex

- The disk and network configurations of the two servers that make up the ITA cluster, as well as the related programs, are shown in the figure below.



# Introduction

- Network Requirements

- Normally, each network interface has their own inherent IP Address, but in order to create an ITA Server HA Cluster, an additional IP Address (Virtual IP Address) will be needed. The virtual IP address will automatically be assigned to the master server by the Pacemaker after the cluster is constructed.

- Network settings used in this document

NetworkIF	Application	Master Server	Slave server
Virtual IP addr.	ITA Connection	10.1.1.10/24	
eth0	External Connection NW	10.1.1.11/24	10.1.1.12/24
eth1	Internal Connection NW	192.168.1.11/24	192.168.1.12/24

- The following communication ports must be available from the client to the server.

Connection type	Port number
http	80/tcp
https	443/tcp

- The cluster configuration servers must be configured on the same network, and the following communication ports must be available.

Connection type	Port Number
ssh(scp)	22/tcp
mariadb	3306/tcp
drbd	7788,7789/tcp ※1
Pacemaker	2224/tcp
corosync	5405/udp

※1 The port number on the left can be changed from the construction procedure settings.

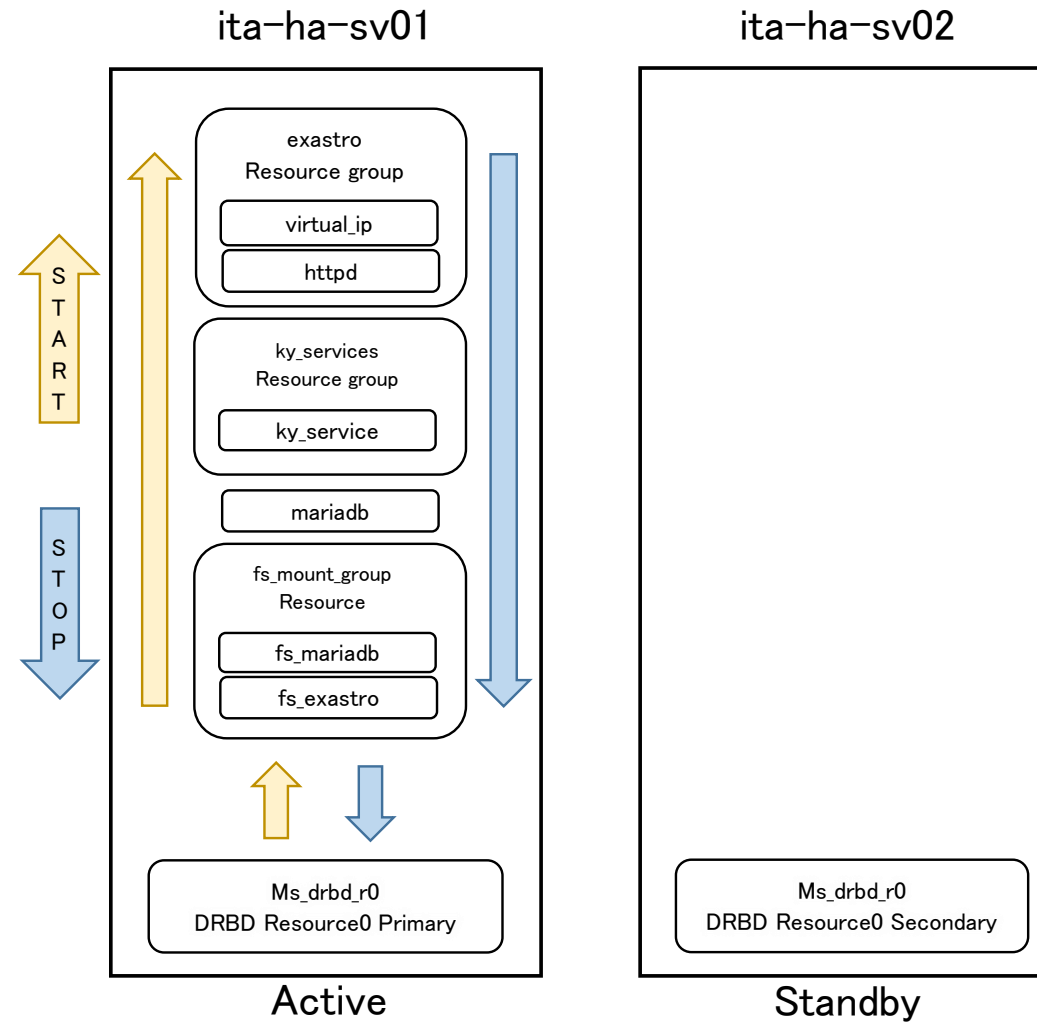
- Disc requirements

- Since the ITA Data is linked to an HA construction server different from the system disc, you must prepare a volume for DRBD synchronization and create a partition. This document describes the procedure for creating partitions and virtual block devices using "/dev/vdc" that has been created and attached in advance.
- The required volume size will vary depending on the ITA operation method and the number of tasks, so check the actual size and prepare a sizing with extra space.
- A minimum of 10GB is recommended. Be sure to prepare volumes of the same size for each of the servers that make up the HA cluster.
- DRBD Synchronization Partition construction example.

Partition	Virtual Block device name	Application	Size
/dev/vdc1	/dev/drbd0	ITA Synchronization file storage	5GB
/dev/vdc2	/dev/drbd1	MariaDB Data file storage	5GB

# Introduction

- Starting/Stopping Pacemaker cluster resources
  - The Pacemaker cluster resource start/stop procedure defined in this document is illustrated below.



## 5. Operation conditions

- This section assumes that the following construction procedure has already been done.
  - ① OS Settings (Network settings, ETC)
  - ② Creating and connecting volumes for DRBD synchronization \* For details, see “4.System construction example/Disc requirements.”
  - ③ Register RHEL (Red Hat Enterprise Linux) subscription
- The installation for each software is written in assuming that they will be installed online in an online environment  
If you want to install them on an offline environment,  
please make sure that your software dependencies match those of the server you are building before you collect the installation libraries.
- Before the construction procedure, please configure the following basic OS Settings.  
Unless stated otherwise, you will need to perform the settings on both of the servers.
  - Deactivating SELinux  
In order to operate an HA Cluster while SELinux is active, the user must have experience and knowledge regarding SELinux.  
Since the procedure in this document has SELinux deactivated, If you want to activate SELinux, do it at your own risk.
  - Configuring Name resolution  
A Name resolution must be configured between the servers that creates the HA Cluster.  
In the procedure in this document, the server's Host name and IP Address will both be registered to /etc/hosts.  
If you are going to use an DNS server in the user environment, dont register to the /etc/hosts,  
but register the HA Server's Host information to the DNS and configure an appropriate name resolution.
  - Configuring NTP  
When managing HA Cluster operations,it is very important that the system clock is correct and synchronized.  
For this reason, we use the NTP protocol for time synchronization whenever possible.

# Introduction

- Installing ITA

- For information regarding ITA Requirements, please refer to the following documents.

[https://github.com/exastro-suite/it-automation-docs/raw/master/asset/Documents/Exastro-ITA System Configuration Environment Co](https://github.com/exastro-suite/it-automation-docs/raw/master/asset/Documents/Exastro-ITA%20System%20Configuration%20Environment%20Construction%20Guide%20Basic)  
Exastro-ITA\_System\_Configuration\_Environment\_Construction\_Guide\_Basic

1.1 Server requirements

[https://github.com/exastro-suite/it-automation-docs/raw/master/asset/Documents/Exastro-ITA System Configuration Enviroment Cor](https://github.com/exastro-suite/it-automation-docs/raw/master/asset/Documents/Exastro-ITA%20System%20Configuration%20Environment%20Construction%20Guide%20Ansible-driver)  
Exastro-ITA\_System\_Configuration\_Enviroment\_Construcion\_Guide\_Ansible-driver

3. System Requirements

- Installation requirements

Refer to the following URL for the All in one construction installation manual  
ITA can be installed through either the online of offline installation procedure.

<https://exastro-suite.github.io/it-automation-docs/learn.html#deploy>

This document assumes that the following functions are activated.

ita\_answers.txt Function settings

```
ita_base:yes
createparam:yes
hostgroup:yes
ansible_driver:yes
cobbler_driver:no
terraform_driver:no
cicd_for_iac:no
```

The ITA startup service changes depending on the ITA Version and the functions the user is going to install.

If you are using an environment different version or are installing functions different to what is listed above, the following service contents must be changed appropriately.

- The services found in onstruction procedure "4. ITA Stop settings"
  - The Pacemaker cluster resources and resource group settings' target services found in construction procedure "6. Pacemaker settings".

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

№	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
1. Install ITA					
1-1	Install ITA	●	●	Please see the All in one installation manual below <a href="https://exastro-suite.github.io/it-automation-docs/learn_ia.html#deploy">https://exastro-suite.github.io/it-automation-docs/learn_ia.html#deploy</a>	
2.DRBD Synchronization settings					
2-1	Configure the environment settings for construction	●	●	【Setting line】 Change the red text to fit the user environment (In this manual, we will use the environment described in the previous section, "2. System Configuration")  export ha1_name=ita-ha-sv01 ※ Specify the Master side's Host name export ha2_name=ita-ha-sv02 ※ Specify the Slave side's Host name export ha1_addr=192.168.1.11 ※ Specify the Master side's Heartbeat IP Address export ha2_addr=192.168.1.12 ※ Specify the Slave side's Heartbeat IP Address export virtual_ip_addr=10.1.1.10 ※ Specify the ITA Access virtual IP address	If you want to re-login in the middle of the procedure, please run this command again.
2-2	Add elrepo repository for installing DRBD	●	●	yum -y install http://www.elrepo.org/elrepo-release-8.1-1.el8.elrepo.noarch.rpm	
2-3	Install DRBD	●	●	yum -y install drbd kmod-drbd90	
2-4	※Name resolution not necessary if DNS Server is used Allow HA configuration nodes to resolve each other's names	●	●	echo "\${ha1_addr} \${ha1_name}" >> /etc/hosts echo "\${ha2_addr} \${ha2_name}" >> /etc/hosts	
2-5	Use DRBD init script to define DRBD operation when starting.	●	●	vi /etc/drbd.d/global_common.conf  startup { # wfc-timeout degr-wfc-timeout outdated-wfc-timeout wait-after-sb wfc-timeout 5; ※Describe "wfc-timeout" in the startup section. }	



## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

№	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
2-6	Comment out any commands that does not exist in version 9.0 or later from the DRBD systemd settings file.	●	●	<pre>vi /usr/lib/systemd/system/drbd.service  [Service] (Display order) ~~~~~Excerpt~~~~~ #ExecStart=/usr/sbin/drbdadm sh-b-pri all    ※Comment out the "sh-b-pri" subcommand. # disconnect and detach all resources ExecStop=/usr/sbin/drbdadm down all</pre>	
2-7	Create /dev/vdc1	●	●	<pre>echo -e "\n\np1\n1\n+(Partition size)\nw"   fdisk /dev/vdc</pre>	(Partition size) equals Number+Measure unit. K = KibiByte, M = MebiByte, G = GibiByte. E.g.) 5 GibiByte = 5G
2-8	Create /dev/vdc2 ※When using all remaining free space on the DRBD Synchronization volume.	●	●	<pre>echo -e "\n\np2\n2\nnw"   fdisk /dev/vdc</pre>	
2-9	Fill the created partition with "0" and reset the file system.	●	●	<pre>dd if=/dev/zero of=/dev/vdc1 bs=1M count=1 dd if=/dev/zero of=/dev/vdc2 bs=1M count=1</pre>	
2-10	Create DRBD settings file	●	●	<pre>cat &gt; /etc/drbd.d/r0.res &lt;&lt; DRBD resource r0 {   device  /dev/drbd0;   disk    /dev/vdc1;   meta-disk internal;   on \$ha1_name {     address \$ha1_addr:7788;   }   on \$ha2_name {     address \$ha2_addr:7788;   } } DRBD</pre>	Install the commands together
		●	●	<pre>cat &gt; /etc/drbd.d/r1.res &lt;&lt; DRBD resource r1 {   device  /dev/drbd1;   disk    /dev/vdc2;   meta-disk internal;   on \$ha1_name {     address \$ha1_addr:7789;   }   on \$ha2_name {     address \$ha2_addr:7789;   } } DRBD</pre>	Install the commands together
2-11	※Only if using firewall Allow communication to the used port.	●	●	<pre>firewall-cmd --add-port=7788/tcp --zone=public --permanent firewall-cmd --add-port=7789/tcp --zone=public --permanent firewall-cmd --reload</pre>	
2-12	Create DRBD resource meta data	●	●	<pre>drbdadm create-md r0 drbdadm create-md r1</pre>	
2-13	Run DRBD service	●	●	<pre>systemctl start drbd</pre>	

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

№	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
2-14	Configure the mount for ITA synchronization on the master side.				
	Initial Synchronize the DRBD synchronization devices from Master to Slave.	●		drbdadm primary --force all	
	Format the DRBD synchronization devices using xfs.	●		mkfs -t xfs /dev/drbd0 mkfs -t xfs /dev/drbd1	
	Configure ITA Synchronization file mount (drbd0)				
	Create a directory of ITA Synchronization and mount the DRDB Synchronization device	●		mkdir -p /mnt/(ITA Install path) mount /dev/drbd0 /mnt/(ITA Install path)	Please change the red text to fit your environment.
	Store the ITA Installation directory in array variables.	●		dirs=( "/(ITA Install path)/data_relay_storage/symphony" "/(ITA Install path)/data_relay_storage/conductor" "/(ITA Install path)/data_relay_storage/ansible_driver" "/(ITA Install path)/ita_sessions" "/(ITA Install path)/ita-root/temp" "/(ITA Install path)/ita-root/uploadfiles" "/(ITA Install path)/ita-root/webroot/uploadfiles" "/(ITA Install path)/ita-root/webroot/menus/sheets" "/(ITA Install path)/ita-root/webroot/menus/users" "/(ITA Install path)/ita-root/webconfs/sheets" "/(ITA Install path)/ita-root/webconfs/users" )	Install the commands together Please change the red text to fit your environment.
	Move the data from the ITA Synchronization directory to the DRBD Synchronization device and create a symbolic link to the ITA Installation path.	●		for dir in "\${dirs[@]}"; do ## directory backup if [ -d \${dir} ]; then mv \${dir}{,.org} fi ## Create directory under the mounted directory. if [ ! -d `dirname /mnt\${dir}` ]; then mkdir -p `dirname /mnt\${dir}` fi ## Copy ITA Data to under the Mounted directory. if [ ! -d /mnt\${dir} ]; then cp -pr \${dir}.org /mnt\${dir}/ fi ##Create symbolic link if [ -d /mnt\${dir} ]; then ln -s "/mnt\${dir}" \${dir} fi ## Comment in if you want to delete the backed up directory. # if [ -d \${dir}.org ]; then # rm -rf \${dir}.org # fi done	Install the commands together
	Configure MariaDB Datafile device (drbd1)				
	Save the source MariaDB data	●		cp -pr /var/lib/mysql /var/lib/mysql.org	
	Mount /var/lib/mysql to the DRBD Synchronization device	●		mount /dev/drbd1 /var/lib/mysql chown -R mysql:mysql /var/lib/mysql	
	Copy the MariaDB Data to the mount (Synchronize) and delete the saved source data	●		rsync -a --delete /var/lib/mysql.org/ /var/lib/mysql/ rm -rf /var/lib/mysql.org	
	Unmount the DRBD Synchronization device.	●		umount /dev/drbd0 umount /dev/drbd1	
	Demote the DRBD Resource to Secondary.	●		drbdadm secondary all	

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH |

№	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
2-15	Configure the Slave server's ITA Synchronization mount.				
	Initial synchronize the Slave storage to the Master storage.			● drbdadm primary --force all	
	Format the DRBD Synchronization partition using xfs			● mkfs -t xfs /dev/drbd0 mkfs -t xfs /dev/drbd1	
	Configure ITA Synchronization file mount (drbd0)				
	Create a directory for ITA Synchronization and mount it to DRBD Synchronization device			● mkdir -p /mnt/(ITA Install path) mount /dev/drbd0 /mnt/(ITA Install path)	Please change the red text to fit your environment.
	Store the ITA Synchronization directory in an array variable.			● dirs=( "/(ITA Install path)/data_relay_storage/symphony" "/(ITA Install path)/data_relay_storage/conductor" "/(ITA Install path)/data_relay_storage/ansible_driver" "/(ITA Install path)/ita_sessions" "/(ITA Install path)/ita-root/temp" "/(ITA Install path)/ita-root/uploadfiles" "/(ITA Install path)/ita-root/webroot/uploadfiles" "/(ITA Install path)/ita-root/webroot/menus/sheets" "/(ITA Install path)/ita-root/webroot/menus/users" "/(ITA Install path)/ita-root/webconfs/sheets" "/(ITA Install path)/ita-root/webconfs/users" )	Install the commands together Please change the red text to fit your environment.
	Move the data from the ITA Synchronization directory to the DRBD Synchronization device and create a symbolic link to the ITA Installation path.			● for dir in "\${dirs[@]}" ; do ## back-up directory if [ -d \${dir} ]; then mv \${dir}{,.org} fi ## Create directory under the mounted directory if [ ! -d `dirname /mnt\${dir}` ]; then mkdir -p `dirname /mnt\${dir}` fi ## Copy ITA Data to under the mounted directory if [ ! -d /mnt\${dir} ]; then cp -pr \${dir}.org /mnt\${dir}/ fi ## Create symbolic link if [ -d /mnt\${dir} ]; then ln -s "/mnt" \${dir} \${dir} fi ## Comment in if you want to delete the backed up directory. # if [ -d \${dir}.org ]; then # rm -rf \${dir}.org # fi done	Install the commands together
	Configure Device for MariaDB Data files (drbd1)				
	Mount /var/lib/mysql to DRBD Synchronization device.			● mount /dev/drbd1 /var/lib/mysql chown -R mysql:mysql /var/lib/mysql	
	Unmount the DRBD Synchronization partition.			● umount /dev/drbd0 umount /dev/drbd1	
	Demote the DRBD Resource to Secondary.			● drbdadm secondary all	
2-16	Stop DRBD Service			● systemctl stop drbd	

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

№	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
3.Maria DB Stop settings					
3-1	Stop Maria DB	●	●	systemctl disable mariadb systemctl stop mariadb	
4.ITA Stop settings					
4-1	Stop ITA services (ky_*)	●	●	systemctl stop ky_activatedirectory_roleuser_replication-workflow.service systemctl stop ky_ansible_execute-workflow.service systemctl stop ky_ansible_towermasterSync-workflow.service systemctl stop ky_bulk_excel-workflow.service systemctl stop ky_change_col_to_row.service systemctl stop ky_cmdbmenuanalysis-workflow.service systemctl stop ky_create_er-workflow.service systemctl stop ky_create_param_menu_execute.service systemctl stop ky_data_portability_execute-workflow.service systemctl stop ky_hostgroup_check_loop.service systemctl stop ky_hostgroup_split.service systemctl stop ky_legacy_role_valautostup-workflow.service systemctl stop ky_legacy_role_varsautolistup-workflow.service systemctl stop ky_legacy_valautostup-workflow.service systemctl stop ky_legacy_varsautolistup-workflow.service systemctl stop ky_mail.service systemctl stop ky_pioneer_valautostup-workflow.service systemctl stop ky_pioneer_varsautolistup-workflow.service systemctl stop ky_std_checkcondition-linklist.service systemctl stop ky_std_synchronize-Collector.service systemctl stop ky_std_synchronize-Conductor.service systemctl stop ky_std_synchronize-regularly2.service systemctl stop ky_std_synchronize-regularly.service systemctl stop ky_std_synchronize-regularly.service systemctl stop ky_terraform_execute-workflow.service systemctl stop ky_terraform_checkcondition-workflow.service systemctl stop ky_terraform_varsautolistup-workflow.service systemctl stop ky_terraform_valautosetup-workflow.service	Some services might not exist depending on the ITA Version. In that case, there will be no need to stop said service.

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

№	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
4-2	Deactivate automatic startup for the different ITA services (ky_*)	●	●	systemctl disable ky_activatedirectory_roleuser_replication-workflow.service systemctl disable ky_ansible_execute-workflow.service systemctl disable ky_ansible_towermasterSync-workflow.service systemctl disable ky_bulk_excel-workflow.service systemctl disable ky_change_col_to_row.service systemctl disable ky_cmdbmenuanalysis-workflow.service systemctl disable ky_create_er-workflow.service systemctl disable ky_create_param_menu_execute.service systemctl disable ky_data_portability_execute-workflow.service systemctl disable ky_hostgroup_check_loop.service systemctl disable ky_hostgroup_split.service systemctl disable ky_legacy_role_valautostup-workflow.service systemctl disable ky_legacy_role_varsautilistup-workflow.service systemctl disable ky_legacy_valautostup-workflow.service systemctl disable ky_legacy_varsautilistup-workflow.service systemctl disable ky_mail.service systemctl disable ky_pioneer_valautostup-workflow.service systemctl disable ky_pioneer_varsautilistup-workflow.service systemctl disable ky_std_checkcondition-linklist.service systemctl disable ky_std_synchronize-Collector.service systemctl disable ky_std_synchronize-Conductor.service systemctl disable ky_std_synchronize-regularly2.service systemctl disable ky_std_synchronize-regularly.service systemctl disable ky_std_synchronize-symphony.service systemctl disable ky_terraform_execute-workflow.service systemctl disable ky_terraform_checkcondition-workflow.service systemctl disable ky_terraform_varsautilistup-workflow.service systemctl disable ky_terraform_valautosetup-workflow.service	Some services might not exist depending on the ITA Version. In that case, there will be no need to stop said service.
5.Apache resource settings					
5-1	Create Apache Server status	●	●	cat > /etc/httpd/conf.d/server_status.conf << STAT ExtendedStatus On  <Location /server-status> SetHandler server-status Order deny,allow Deny from all Allow from localhost </Location> STAT	Install the commands together
5-2	Stop/Deactivate Apache service.	●	●	systemctl disable httpd systemctl stop httpd	
5-3	Copy the https server certificate/secret key from the master server to the slave server.	●	●	scp /etc/pki/tls/certs/(https server certificate name ).cert \${ha2_addr}:/etc/pki/tls/certs/ scp /etc/pki/tls/certs/(https server certificate name ).key \${ha2_addr}:/etc/pki/tls/certs/	Make sure to change the https server certificate/secret key to the file name used when installing the ITA Server.

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

№	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallell)	Remarks
		master	slave		
6.Pacemaker settings					
6-1	Install HA Software	●	●	yum --enablerepo=rhel-8-for-x86_64-highavailability-rpms install pacemaker pcs drbd-pacemaker NetworkManager NetworkManager-config-server systemctl restart NetworkManager	
6-2	※Do only if under a proxy. Add main machine/secondary machine IP to no_proxy	●	●	sed -i 's/^export NO_PROXY/export no_proxy=\${no_proxy},\${ha1_name},\${ha2_name}\nexport NO_PROXY/' /etc/profile.d/proxy.sh  sed -i 's/^setenv NO_PROXY/setenv no_proxy=\${no_proxy},\${ha1_name},\${ha2_name}\nsetenv NO_PROXY/' /etc/profile.d/proxy.csh  source /etc/profile.d/proxy.sh source /etc/profile.d/proxy.csh	※Run one time after pasting the text
6-3	※Do only if firewalld is used add firewalld settings	●	●	firewall-cmd --add-service=high-availability --permanent firewall-cmd --reload	
6-4	Modify corosync.service settings	●	●	cp -p /usr/lib/systemd/system/corosync.service /etc/systemd/system/ sed -i 's/^#Restart=on./Restart=on-failure/' /etc/systemd/system/corosync.service sed -i 's/^#RestartSec=.*/RestartSec=70/' /etc/systemd/system/corosync.service	
6-5	Treat Pacemaker internal process failure as a node failure	●	●	sed -i 's/^#\ PCMK_fail_fast./PCMK_fail_fast=yes/' /etc/sysconfig/pacemaker	
6-6	Modify pacemaker.service settings	●	●	cp -p /usr/lib/systemd/system/pacemaker.service /etc/systemd/system sed -i 's/^#\ ExecStopPost=VbinVsh./ExecStopPost=VbinVsh -c 'pidof crmd \\\ killall -TERM corosync/' /etc/systemd/system/pacemaker.service	
6-7	Generate/deploy cluster certification setting file	●	●	corosync-keygen -l scp -p /etc/corosync/authkey root@\${ha2_addr}:/etc/corosync/authkey	
6-8	Configure password for the "hacluster" user ID (pcs admin account).	●	●	passwd hacluster New password:xxxxxx Re-input the new password:xxxxxx passwd: Successfully updated all authentication tokens.	Please change the red text to fit your environment. Make sure the password for both master and slave are the same
6-9	Start pcsd service and configure pcsd so it is active when the system starts.	●	●	systemctl enable pcsd systemctl start pcsd systemctl status pcsd	

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH |

№	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
6-10	Authenticate cluster	●		pcs host auth \${ha1_name} \${ha2_name} Username: hacluster Password:(Password set with "passwd hacluster")	Please change the red text to fit your environment.
6-11	Set up cluster	●		pcs cluster setup hacluster --start \${ha1_name} addr=\${ha1_addr} \${ha2_name} addr=\${ha2_addr}	
6-12	Deactivate STONITH (Shoot The Other Node In The Head) option	●		pcs property set stonith-enabled=false	
6-13	Configure the clusters so that no special actions are carried and all nodes are managed if the minimum required number of nodes are not reached.	●		pcs property set no-quorum-policy=ignore	
6-14	Configure attribute wait time.	●		pcs property set transition-delay="0s"	
6-15	Deactivate automatic failback.	●		pcs resource defaults resource-stickiness="INFINITY" migration-threshold="1"	
6-16	Configure the clusters so they run automatically	●		pcs cluster enable --all	
6-17	Configure vip resources	●		pcs resource create virtual_ip ocf:heartbeat:IPaddr2 ip=\${virtual_ip_addr} cidr_netmask=24 \ op monitor interval=10s timeout=30s \ op start interval=0s timeout=20s \ op stop interval=0s timeout=20s	
6-18	Register r0 DRBD device as resource	●		pcs resource create drbd_r0 ocf:linbit:drbd drbd_resource=r0 \ op monitor interval=10s role=Master \ op monitor interval=30s role=Slave \ op notify interval=0s timeout=90s \ op promote interval=0s timeout=90s \ op reload interval=0s timeout=30s \ op start interval=0s timeout=240s \ op stop interval=0s timeout=100s	
6-19	Register r1 DRBD device as resource	●		pcs resource create drbd_r1 ocf:linbit:drbd drbd_resource=r1 \ op monitor interval=10s role=Master \ op monitor interval=30s role=Slave \ op notify interval=0s timeout=90s \ op promote interval=0s timeout=90s \ op reload interval=0s timeout=30s \ op start interval=0s timeout=240s \ op stop interval=0s timeout=100s	
6-20	Configure resource settings for DRBD block devices	●		pcs resource create fs_exastro ocf:heartbeat:Filesystem device=/dev/drbd0 directory=/mnt/exastro fstype=xfst pcs resource create fs_mariadb ocf:heartbeat:Filesystem device=/dev/drbd1 directory=/var/lib/mysql fstype=xfst	
6-21	Configure httpd resources	●		pcs resource create httpd systemd:httpd \ op monitor interval=60s timeout=100s \ op start interval=0s timeout=100s \ op stop interval=0s timeout=100s	

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

No	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
6-22	Configure ITA service (ky_*) resources	●		<pre> pcs resource create ky_activatedirectory_roleuser_replication-workflow systemd:ky_activatedirectory_roleuser_replication-workflow \ op monitor interval=60s timeout=100s \ op start interval=0s timeout=100s \ op stop interval=0s timeout=100s  pcs resource create ky_ansible_execute-workflow systemd:ky_ansible_execute-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_ansible_towermasterSync-workflow systemd:ky_ansible_towermasterSync-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_bulk_excel-workflow systemd:ky_bulk_excel-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_change_col_to_row systemd:ky_change_col_to_row \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_cmdbmenuanalysis-workflow systemd:ky_cmdbmenuanalysis-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_create_er-workflow systemd:ky_create_er-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_create_param_menu_execute systemd:ky_create_param_menu_execute \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_data_portability_execute-workflow systemd:ky_data_portability_execute-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s </pre>	



## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

No	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
6-22	Configure ITA service (ky_*) resources	●		<pre> pcs resource create ky_hostgroup_check_loop systemd:ky_hostgroup_check_loop \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_hostgroup_split systemd:ky_hostgroup_split \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_legacy_role_valautostup-workflow systemd:ky_legacy_role_valautostup-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_legacy_role_varsautolistup-workflow systemd:ky_legacy_role_varsautolistup-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_legacy_valautostup-workflow systemd:ky_legacy_valautostup-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_legacy_varsautolistup-workflow systemd:ky_legacy_varsautolistup-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_pioneer_valautostup-workflow systemd:ky_pioneer_valautostup-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_pioneer_varsautolistup-workflow systemd:ky_pioneer_varsautolistup-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_std_checkcondition-linklist systemd:ky_std_checkcondition-linklist \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s </pre>	

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

№	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
6-22	Configure ITA service (ky_*) resources	●		<pre> pcs resource create ky_std_synchronize-Conductor systemd:ky_std_synchronize-Conductor \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_std_synchronize-regularly2 systemd:ky_std_synchronize-regularly2 \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_std_synchronize-regularly systemd:ky_std_synchronize-regularly \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_std_synchronize-symphony systemd:ky_std_synchronize-symphony \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_terraform_execute-workflow systemd:ky_terraform_execute-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_terraform_checkcondition-workflow systemd:ky_terraform_checkcondition-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_terraform_varsautolistup-workflow systemd:ky_terraform_varsautolistup-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s  pcs resource create ky_terraform_valautosetup-workflow systemd:ky_terraform_valautosetup-workflow \ op monitor interval=30s timeout=60s \ op start interval=0s timeout=60s \ op stop interval=0s timeout=60s </pre>	
6-23	Configure mariadb resources	●		<pre> pcs resource create mariadb systemd:mariadb \ op monitor interval=60s timeout=100s \ op start interval=0s timeout=100s \ op stop interval=0s timeout=100s </pre>	

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

№	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
6-24	Configure fs resource group	●		pcs resource group add fs_mount_group fs_exastro fs_mariadb	
6-25	Configure vip and httpd resource groups	●		pcs resource group add exastro virtual_ip httpd	
6-26	Configure ITA service (ky_*) resource groups	●		pcs resource group add ky_services ky_activedirectory_roleuser_replication-workflow ky_ansible_execute-workflow ky_ansible_towermasterSync-workflow ky_bulk_excel-workflow ky_change_col_to_row ky_cmdbmenuanalysis-workflow ky_create_er-workflow ky_create_param_menu_execute ky_data_portability_execute-workflow ky_hostgroup_check_loop ky_hostgroup_split ky_legacy_role_valautostup-workflow ky_legacy_role_varsautolistup-workflow ky_legacy_valautostup-workflow ky_legacy_varsautolistup-workflow ky_pioneer_valautostup-workflow ky_pioneer_varsautolistup-workflow ky_std_checkcondition-linklist ky_std_synchronize-Conductor ky_std_synchronize-regularly2 ky_std_synchronize-regularly ky_std_synchronize-symphony	
6-27	Assign the added DRBD resources to Master/Slave. Start up 2 nodes and specify one of the nodes to be the master node (DRBD primary node)	●		pcs resource promotable drbd_r0 master-max=1 master-node-max=1 clone-max=2 clone-node-max=1 notify=true pcs resource promotable drbd_r1 master-max=1 master-node-max=1 clone-max=2 clone-node-max=1 notify=true	
6-28	Add constraint so fs resource and ms_drbd start on the same node	●		pcs constraint colocation add fs_exastro with drbd_r0-clone INFINITY with-rsc-role=Master pcs constraint colocation add fs_mariadb with drbd_r1-clone INFINITY with-rsc-role=Master	
6-29	Add constraint so the resources are run in the following order: Resource ms_drbd → fs_exastro	●		pcs constraint order promote drbd_r0-clone then start fs_exastro pcs constraint order promote drbd_r1-clone then start fs_mariadb	
6-30	Specify all created resources and then the resource group start/stop order.	●		pcs constraint order set fs_mount_group mariadb ky_services exastro	
6-31	Specify a constraint that forces all set resources to be started on the same node.	●		pcs constraint colocation set fs_mount_group mariadb ky_services exastro	
6-32	Reload resources	●		pcs resource cleanup	
6-33	Reload settings file	●	●	systemctl daemon-reload	
6-34	Start HA program services	●	●	systemctl enable corosync systemctl enable pacemaker systemctl start corosync systemctl start pacemaker	

## Construction procedure

### User/Connection requirements

Run as root user  
the root user must be able to login through SSH

No	Operation	Server		Contents (※When "Master" and "Slave" server configuration is required, please configure them in parallel)	Remarks
		master	slave		
6-35	Check that the Cluster is running properly	●		<p>pcs status</p> <p>(Display example)</p> <p>~~~~~Excerpt~~~~~</p> <p>2 nodes configured ←Check that this is displayed</p> <p>32 resource instances configured</p> <p>Online: [ ita-ha-sv01 ita-ha-sv02 ] ←Check that both Master and Slave are online</p> <p>Active resources:</p> <p>  mariadb (systemd:mariadb): Started ita-ha-sv01 ←Check that "master" is "Started"</p> <p>Resource Group: fs_mount_group</p> <p>  fs_mysql (ocf::heartbeat:Filesystem): Started ita-ha-sv01</p> <p>  fs_httpd (ocf::heartbeat:Filesystem): Started ita-ha-sv01</p> <p>Resource Group: ky_services</p> <p>  ky_xxx (systemd:ky_xxx): Started ita-ha-sv01</p> <p>Resource Group: exastro</p> <p>  virtual_ip (ocf::heartbeat:IPaddr2): Started ita-ha-sv01</p> <p>  httpd (systemd:httpd): Started ita-ha-sv01</p> <p>~~~~~Excerpt~~~~~</p>	Install the commands together
6-36	Check ITA Connection	●		<p>Access the login screen with the URL below</p> <p><a href="https://10.1.1.10">https://10.1.1.10</a></p>	Install the commands together
作業終了					