ITA_System Configuration / Environment Construction Guide

Basic

—Version 1.9—

Disclaimer

Trademark

- Linux is registered trademark or trademark of Linux Torvalds, registered in the U.S. and other countries.
- Red Hat is registered trademark or trademark of Red Hat, Inc., registered in the U.S. and other countries.
- Apache、Apache Tomcat and Tomcat are registered trademarks or trademarks of the Apache Software Foundation.
- Oracle and MySQL are registered trademarks of Oracle Corporation and its subsidiaries and affiliates in the U.S. and other countries.
- MariaDB is a registered trademark or trademark of the MariaDB Foundation.

The names of other systems, company name and products mentioned in this document are registered trademarks or trademarks of their respective companies.

The® mark and TM mark are not specified in this document.

※”Exastro IT Automation” is written as ”ITA” in this document.

Table of contents

Introduction

This document explains the system configuration and environment construction for ITA system operation.

1. System requirements

1.1 Server requirements

The system operates on a Linux server and is accessed from a client PC via browser.
When installing the system, please prepare a server that meets the following requirements.

■1.1.1 Server configuration

Table 1.1.1 Server configuration list

Category	Required/ Select	Product name	Version
OS	Either of	RHEL ※1	7.0 or higher
		CentOS	7.0 or higher
Web server	Required	Apache	2.4 series
DataBase	Required	MariaDB	10.3 or higher
language	Required	PHP	7.2
PHP library	Required	PhpSpreadsheet	1.10.1 or higher
		php-yaml	2.1.0 or higher
Pear library	Required	HTML_AJAX	0.5.7 or higher

※1 Red Hat Enterprise Linux

■1.1.2 Server minimum specifications

Table 1.1.2 List of minimum server specifications

Category	Minimum specification	Remarks
CPU	2Core	
Memory	4GB	
Disk space	1GB ※1	※1 Capacity of ITA system. Excluding OS and log storage capacity.

■1.1.3　Sizing

The following is the recommended spec for server.

① Number of records in 1 menu

The number of records (columns) inside a single menu which is created in menu creation function.

Table 1.1.3-1 Number of items in 1 menu and server spec

Number of menu items	CPU	Memory
～ 10,000	2Core	4GB
1,000 ～ 20,000	4Core	8GB

② Number of parallel execution of Ansible operations

Maximum number of parallel execution can be set in “Ansible Common” > “Interface information” > “Number of parallel executions”.

Table 1.1.3-2 Number of parallel execution of Ansible operations and server spec

Number of parallel executions	CPU	Memory
～ 50	2Core	4GB
50 ～ 100	4Core	8GB

③ Number of simultaneous login and operation

The number of the users that logged in to the system at the same time, and perform operations such as screen moving, filter searching or registration in login stat.

Table 1.1.3-3 Number of simultaneous login and operation and server spec

Number of simultaneous login and operation	CPU	Memory
～ 200	2Core	4GB
200 ～ 300	4Core	8GB

The setting of ITA after installation is set to the minimum spec (CPU: 2 core / Memory: 4GB) for ITA to operate on ITA system server.

Please change the setting value to improve the performance for the system to work above minimum spec.

Please refer to the “[Reference] Configuration settings during installation” manual for details of setting value.

※1　ITA system server … A basic ITA configuration that server of associated driver such as Ansible server is constructed in individual server .

1.2 Client requirements

While using the functions of this system, the following requirements are recommended for client PCs.

Table 1.2.1 Requirements of client PC

Category	Product name	Version
Software	Excel（※）	MS Office 2010 or higher
Browser	Google Chrome	72 or higher
	FireFox	41 or higher
	Edge	20 or higher

※Required when downloading Excel files (because the format of download file is Excel).

2. System configuration

2.1 System configuration pattern

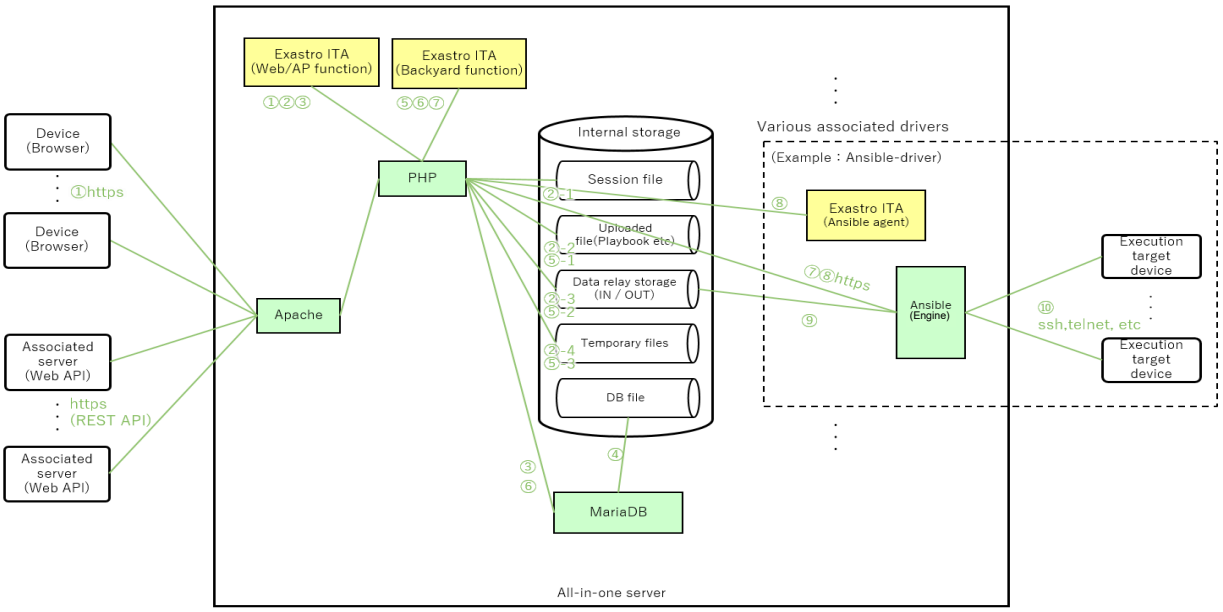
The Web / AP function, BackYard function, database and data storage of this software can be operated with following server configurations.

Table 2.1 System configuration patterns

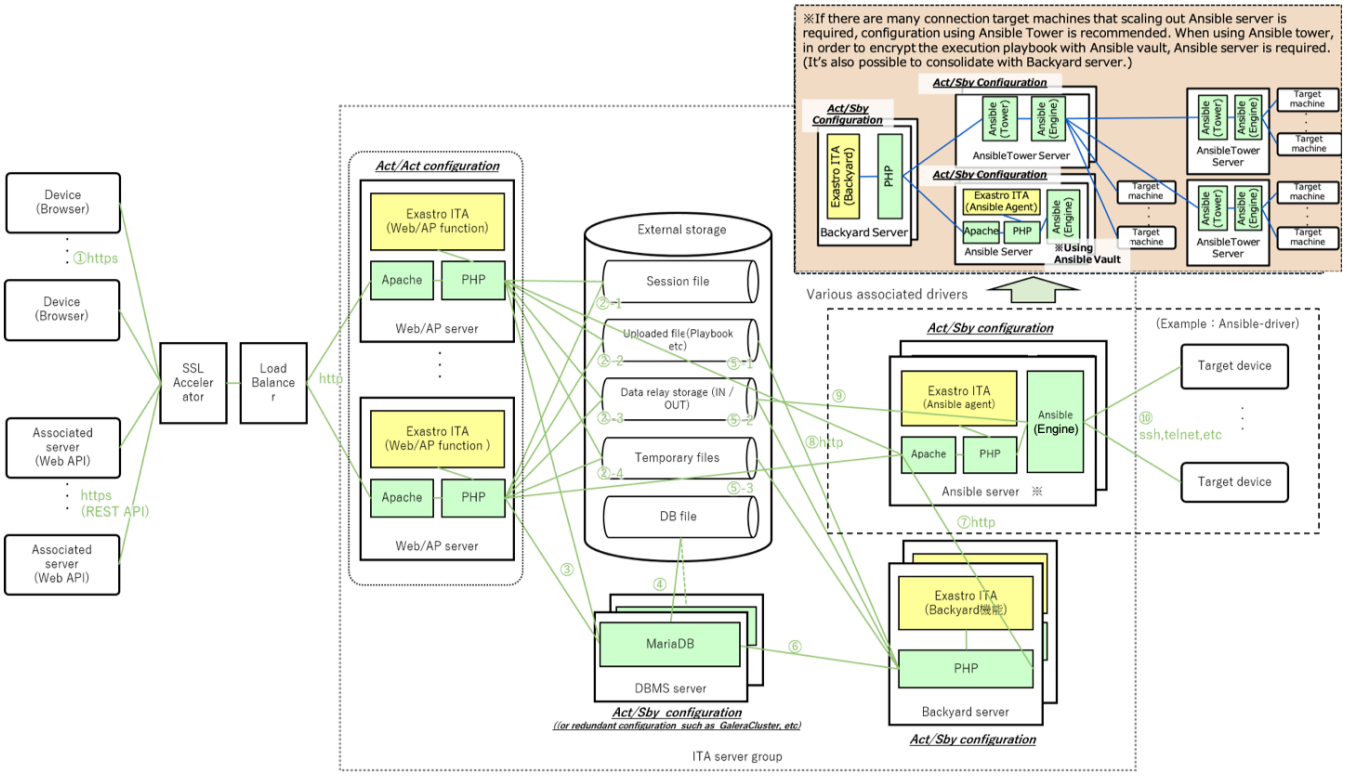
No	Configuration	Description	Remarks
1	All-in-one configuration	A configuration pattern that assembles the system on a single server.	Association driver that is possible to be configured in All-in-one configuration with ITA-BASE function. ・Ansible-driver ・Cobbler-driver
2	HA configuration	A configuration pattern in which all systems are separated into individual servers to create a redundant configuration, and data files and DataBase files are stored in external storage.	Web/AP server (Act/Act configuration) DBMS server (Act/Sby configuration) Backyard server (Act/Sby configuration)

The following is a representative example image of a system using the Ansible driver

■ All-in-one configuration



■ HA configuration



2.2 System communication requirements

In this system configuration, the communication requirements between each service are as follows.

Table 2.2 List of communication requirements

Communication number ※1	FROM	TO	Protocol [port number ※2]	Main Applications
①	Terminal	Web/AP server	http(s) [80(443)/tcp]	Access to Exastro ITA Web content
②-1	Web/AP server	Storage device (session file)	File access (tcp or storage I / O)	Store / view web session files
②-2		Storage device (uploaded file)		Store / view uploaded files (Playbook,etc)
②-3		Storage device (data relay storage)		Store execution information (Playbook, host_vars, etc.) in Symphony execution.
②-4		Storage device (temporary file)		Store/ view temporary files (upload files, etc.)
③		DBMS server	tcp (DB access) [3306 / tcp]	Access to DB server (Data processing according to view/ registration / update /discard / restore on ITA screen)
④	DBMS server	Storage device (DB file)	File access (tcp or storage I / O)	Write DB file
⑤-1	Backyard server	Storage device (uploaded file)	File access (tcp or storage I / O)	Refer to uploaded file (Playbook, etc.)
⑤-2		Storage device (data relay storage)		Store information and logs during Symphony execution
⑤-3		Storage device (temporary file)		Store / view temporary files (upload files, etc.)
⑥		DBMSserver	tcp (DB access) [3306 / tcp]	Access to DB server (View/update/discard)
⑦		Ansible server	http(s) [80(443)/tcp]	Submitting REST API requests to Ansible (process execution,etc)
⑧	Web/AP server	Ansible server	http(s) [80(443)/tcp]	Submitting REST API requests to Ansible (Emergency stop)
⑨	Ansible server	Storage device	File access (tcp or storage I / O)	Refer to the execution information (Playbook, host_vars, etc.) when executing Ansible command
⑩		Target device	Any (※3 ssh [22/tcp] telnet [23/tcp] , etc.)	Execute command to target device from Ansible.
⑪ ※4	Web/AP server	Terraform Enterprise	http(s) [80(443)/tcp]	Registers ITA's Organization/Workspace to the Terraform Enterprise side Gathers information from ITA's Organization/Workspace/Policy/PolicySet
⑫ ※4	Backyard server	Terraform Enterprise	http(s) [80(443)/tcp]	Executes and gather results from Plan/PolicyCheck/Apply to Terraform Enterprise when running.
⑬ ※4	Backyard server	git	http(s) [80(443)/tcp]	Uses CI/CD for IaC to connect to the Git repository and gathers file information.

※1 Describe the communication number associated with the above number in the configuration image of "2.1 System Configuration Pattern".

※2 The port number is the standard port number

※3 Typical examples are described .Usage protocol differs depending on Ansible module.

※4 No description linked with the configuration figure in "2.1 System Configuration Patterns".

2.3 Server scalability affecting points

In this system configuration, the points that affect server scalability and the configuration are as follows. The numbers in the table below indicates the following:

① What component is most affected (Memory, Disc or CPU)
② What effect it has when running out of resources
③ How to solve

Table 2.3 Affecting points of server scalability

	Web/AP server	DBMS server	Backyard server	External storage	Ansible server
	ACT/ACT	ACT/SBY	ACT/SBY	-	ACT/SBY
Increase in the number of web accesses (combining various requirements)	①Memory ②Exhausts memory and returns system error when searching, registering or updating takes too much time or when processing a large amount of data ③Scale up or Scale out	① CPU/Memory(Depends on MariaDB specs) ② Searching/Registering/Updating takes more time(Depends on MariaDB specs) ③Scale up	No effect	①Disc ② Registering/Updating the database and writing to files returns an error. ③Scale up or Scale out	No effect
Increasing number of Symphony/Conductor to be executed simultaneously	No effect	① CPU/Memory(Depends on MariaDB specs) ② Searching/Registering/Updating takes more time(Depends on MariaDB specs) ③Scale up	①CPU ②Sets the processing Symphony/Condoctur to “Finished (Error) when processing large amounts of data or when the process is taking too much time to finish. ③Scale up	①Disc ② Registering/Updating the database and writing to files returns an error. ③Scale up or scale out	① CPU/Memory(Depends on Ansible specs) ②(Depends on Ansible specs) ③Scale up or implement Tower
Increase in work pattern (Movement, Playbook, parameter sheet, etc.)	No effect	①Memory ②Exhausts memory and returns system error when searching, registering or updating takes too much time or when	①CPU/Memory ②Outputs an error to the log when processing large amounts of data or when the process is taking too much time to finish. ③Scale up	①Disc ② Registering/Updating the database and writing to files returns an error. ③Scale up or Scale out	No effect

		processing a large amount of data ③Scale up			
Increase in the number of target devices.	No effect	No effect	No effect	No effect	① CPU/Memory(Depends on Ansible specs) ②(Depends on Ansible specs) ③Scale up or implement Tower