HiCentral Optimus

White Paper

Legal Information

About this Manual

The Manual includes instructions for using and managing the Product. Pictures, charts, images and all other information hereinafter are for description and explanation only. The information contained in the Manual is subject to change, without notice, due to firmware updates or other reasons. Please find the latest version of this Manual at the Hikvision website (https://www.hikvision.com/).

Please use this Manual with the guidance and assistance of professionals trained in supporting the Product.

Trademarks

HIKVISION and other Hikvision's trademarks and logos are the properties of Hikvision in various jurisdictions.

Other trademarks and logos mentioned are the properties of their respective owners.

Disclaimer

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THIS MANUAL AND THE PRODUCT DESCRIBED, WITH ITS HARDWARE, SOFTWARE AND FIRMWARE, ARE PROVIDED "AS IS" AND "WITH ALL FAULTS AND ERRORS". HIKVISION MAKES NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, MERCHANTABILITY, SATISFACTORY QUALITY, OR FITNESS FOR A PARTICULAR PURPOSE. THE USE OF THE PRODUCT BY YOU IS AT YOUR OWN RISK. IN NO EVENT WILL HIKVISION BE LIABLE TO YOU FOR ANY SPECIAL, CONSEQUENTIAL, INCIDENTAL, OR INDIRECT DAMAGES, INCLUDING, AMONG OTHERS, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, OR LOSS OF DATA, CORRUPTION OF SYSTEMS, OR LOSS OF DOCUMENTATION, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), PRODUCT LIABILITY, OR OTHERWISE, IN CONNECTION WITH THE USE OF THE PRODUCT, EVEN IF HIKVISION HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES OR LOSS.

YOU ACKNOWLEDGE THAT THE NATURE OF THE INTERNET PROVIDES FOR INHERENT SECURITY RISKS, AND HIKVISION SHALL NOT TAKE ANY RESPONSIBILITIES FOR ABNORMAL OPERATION, PRIVACY LEAKAGE OR OTHER DAMAGES RESULTING FROM CYBER-ATTACK, HACKER ATTACK, VIRUS INFECTION, OR OTHER INTERNET SECURITY RISKS; HOWEVER, HIKVISION WILL PROVIDE TIMELY TECHNICAL SUPPORT IF REQUIRED.

YOU AGREE TO USE THIS PRODUCT IN COMPLIANCE WITH ALL APPLICABLE LAWS, AND YOU ARE SOLELY RESPONSIBLE FOR ENSURING THAT YOUR USE CONFORMS TO THE APPLICABLE LAW. ESPECIALLY, YOU ARE RESPONSIBLE, FOR USING THIS PRODUCT IN A MANNER THAT DOES NOT INFRINGE ON THE RIGHTS OF THIRD PARTIES, INCLUDING WITHOUT LIMITATION, RIGHTS OF PUBLICITY, INTELLECTUAL PROPERTY RIGHTS, OR DATA PROTECTION AND OTHER PRIVACY RIGHTS. YOU SHALL NOT USE THIS PRODUCT FOR ANY PROHIBITED END-USES, INCLUDING THE DEVELOPMENT OR PRODUCTION OF WEAPONS OF MASS DESTRUCTION, THE DEVELOPMENT OR PRODUCTION OF CHEMICAL OR BIOLOGICAL WEAPONS, ANY ACTIVITIES IN THE CONTEXT RELATED TO ANY NUCLEAR EXPLOSIVE OR UNSAFE NUCLEAR FUEL-CYCLE, OR IN SUPPORT OF HUMAN RIGHTS ABUSES.

IN THE EVENT OF ANY CONFLICTS BETWEEN THIS MANUAL AND THE APPLICABLE LAW, THE LATER PREVAILS.

TABLE OF CONTENTS

Chapter 1 Introduction

1.1 Product Scopes

This document introduces the positioning, design, and functionality of HikCentral Optimus, as well as software and hardware requirements.

This document is for pre-sale and technical support personnel to understand the features of this platform.

1.2 Terminology and Abbreviations

Terminology	Description
ISAPI	Communication protocols in RESTful style, and messages are in JSON and XML format.
HCP	Abbreviation of HikCentral Professional, which is an integrated platform providing the functions of video management, access control, and so on. It also provides Control Client, Web Client, Mobile Client, and Server for accessing or managing modules and resources.
Connector	Optimus creates connectors to exchange data with Integrated Systems. As defined by Connector API, Optimus must connect to a specific connector developed for a system if you want to integrate it with Optimus.
ROSE	Rose is a high availability cluster product developed by Datasystem. It monitors the running state of resources, implements automatic failover in case of resource failure, and ensures the continuity of the running critical data and Internet services.
MongoDB	Classified as a distributed NoSQL database program. Its advantages include flexible mode, strong scalability, and excellent performance, but its transactional support is relatively weaker and the functionality of joining table data is also weaker than traditional databases.
TPI	A configuration file describing information of external systems and the required parameters.
Flow	Determines how specific event(s) from specific Integrated System(s) trigger specific actions in other specific Integrated System(s) under specific condition(s).

Chapter 2 Product Description

2.1 Product Positioning

Optimus, an independent platform from HCP, allows multiple different systems including HCP to communicate with each other by providing connectors, flows, and data synchronization.
Optimus Core is responsible for creating and executing rules. For example, a door of system A is linked with a camera of system B, and the information of system B is provided to system A on a regular basis. Connector is responsible for passing the capability information of systems to Optimus Core.

2.2 Design Principles

1. Modular Design
   Optimus contains several modules including Dashboard, Connectors, Flows, and Data Synchronization. These modules can interact with each other.
2. Multi-level Security Design
   Optimus adopts various security control strategies from network, data, storage, and application levels.
   - Network level: The platform supports HTTPS access, and sensitive data is transmitted after security-approved encryption. Access to the platform by non-LAN external networks is realized through a small number of external ports mapped by the platform.
   - Data level: Data encryption meets the encryption standards of current industry security requirements, and the use of outdated and insecure encryption algorithms is prohibited in the product development process.
   - Storage level: Sensitive information is encrypted and stored in the database, and the encryption method meets the security requirements of the industry.
   - Application level: Unified user authentication is provided in the product design. Login authentication is required and login password is encrypted with tamper-proof and irreversible algorithms to prevent password leakage and tampering.

2.3 Reference

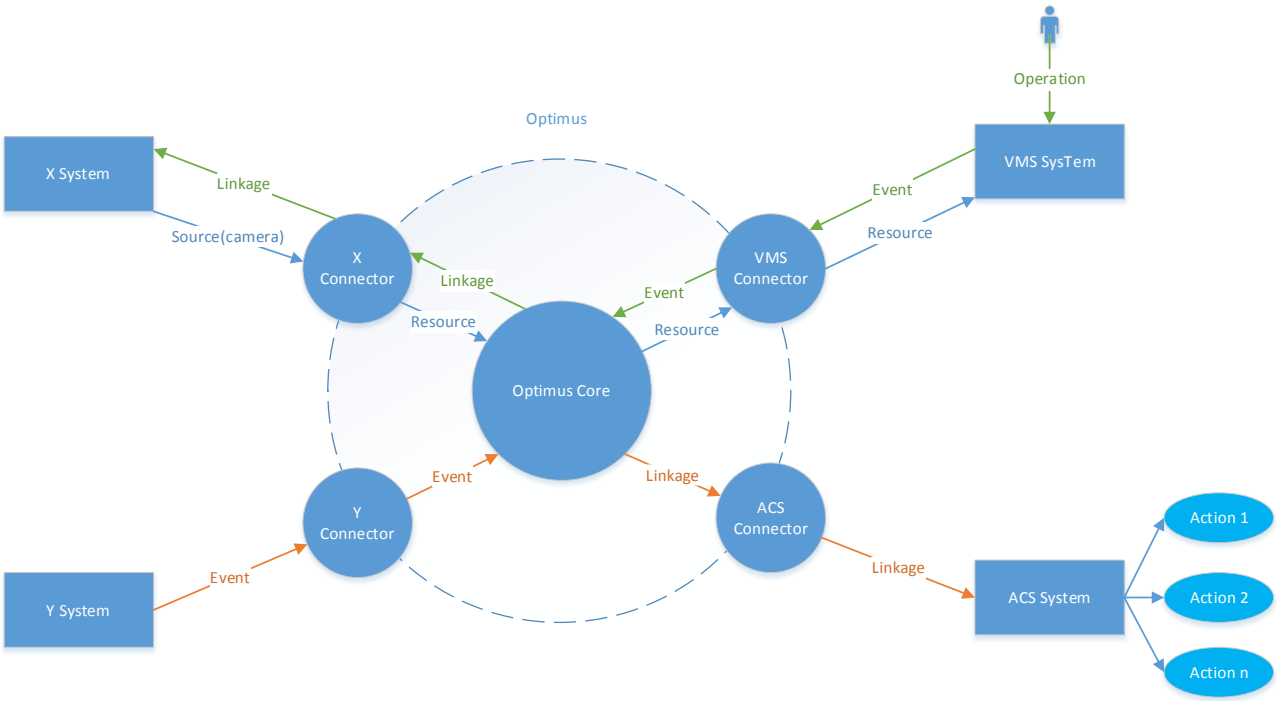
- RESTful API Development Practices
- Open Source Software Practices

Chapter 3 Product Architecture

3.1 System Description

Optimus is positioned as a middleware product, which allows two different systems to communicate with each other. Optimus allows systems to share resources, events, and operations.

The following flow chart displays how Optimus works.



How Optimus Works

3.2 Logical Architecture

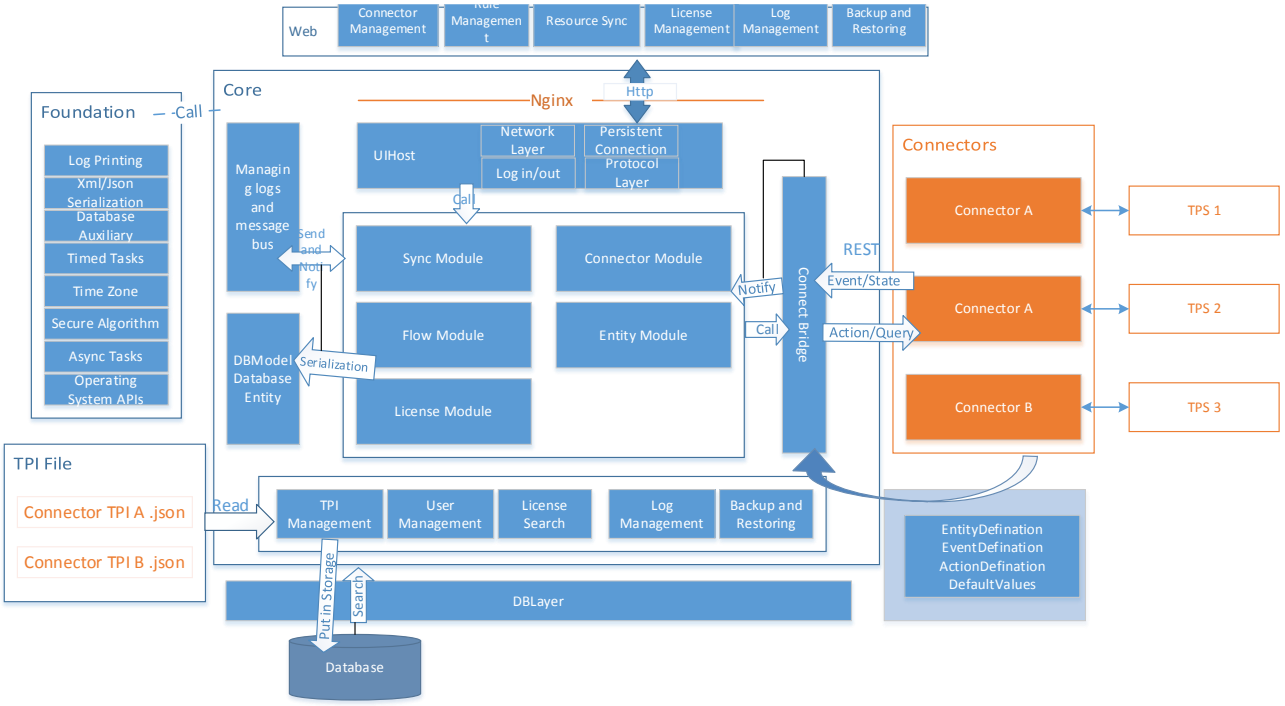
Optimus is composed of Core and Connectors. Connectors can communicate with integrated Systems through a unified protocol and sending their capability information to Core, so that Core can use the resources of multiple systems and receive notifications from Connectors.

The Core can be divided into application layer, basic service, functional modules, and other auxiliary functions.

The application layer is responsible for receiving web requests, processing data of administrator's operations, and sending back the responses.

The basic service provides the basic functions and is responsible for the processing of TPI management, user management, etc.

The functional modules handle core applications.



Logical Architecture

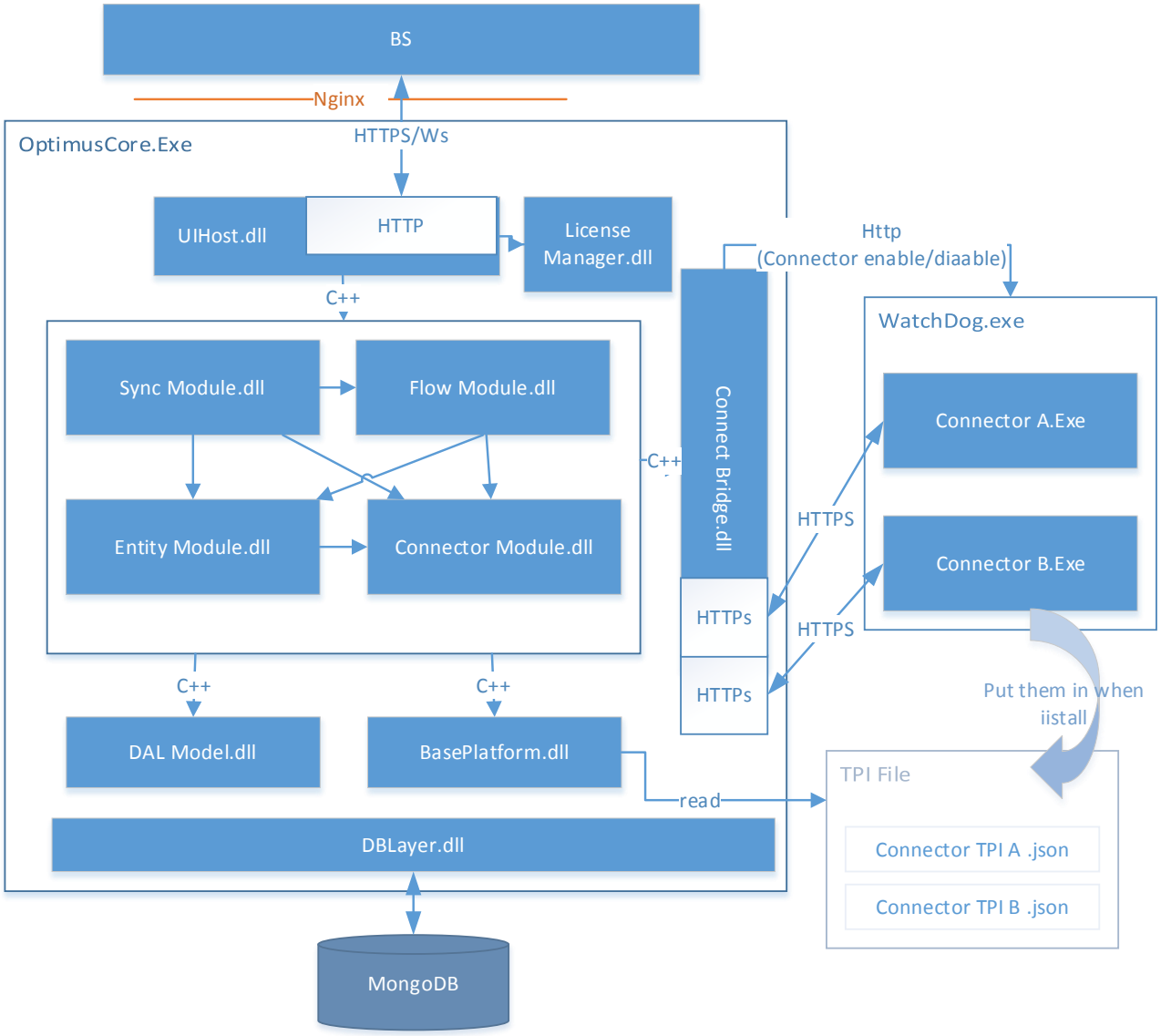
3.3 Physical Architecture

Core runs one process in general, and the DLL of each functional module exposes C++ virtual API in the form of header file and call each other. The functional modules and the general platform modules are called directly through the exporting API.

Among them, UIHost provides HTTP API for processing web requests, and the requests will go through the Nginx reverse proxy first.

The watchdog is not only responsible for starting and stopping the basic services, but also responsible for providing the HTTP API so that the ConnectorBridge can start multiple Connector processes or stop the Connector processes depending on the connection configuration. ConnectorBridge initiates multiple HTTPs ports to communicate with Connectors.

The Core is connected to MongoDB through DBLayer for database reading and writing.



Physical Architecture

3.4 Technical Architecture

- Server Programming Language: C++
- Front-End: Vue
- Web Service: Nginx
- Database: MongoDB
- License Activation Service: License Service Developed by Hikvision
- Message Forwarding: Internal Message Bus

Chapter 4 Function List

4.1 Maintenance and Management

Maintenance and management data backup and restoring, License activation/deactivation, password changing, and wizard.

4.1.1 Backup and Restore

- Supports backup system data by day/week/month
- Supports backup up to 5 files
- Supports setting the storage path for backup files
- Supports restoring the system data via the backup files.

4.1.2 License Activation/Update/Deactivation

- Supports one-month free trial. After one month, you should purchase a License if you want to continue using Optimus.
- Supports online/offline activation
- Supports online/offline update
- Supports online/offline deactivation
- Supports adding multiple activation codes for activation/deactivation
- In hot spare mode, supports activating or deactivating two servers of the hot spare system at the same time by only using an activation code.

4.1.3 Change Password

- Supports changing the password of *admin* user by old password.
- Supports downloading a new encrypted file for getting back the password if you forgot password.

4.2 Connector

Connectors are created to integrate third-party systems so that data can be exchanged between Optimus and systems.

4.2.1 Connector Management

- Supports adding, deleting, editing, and searching connectors (including fuzzy search by connector name)
- Supports displaying the real-time online/offline status of connectors
- Supports enabling/disabling connectors
- Supports displaying connectors in thumbnail and list mode
- Supports sorting connectors in ascending or descending order

4.2.2 Connector Process

- When adding a connector, supports starting the process according to the executable file path defined by the connector, and establishing network communication with the process
- When deleting or disabling a connector, supports stopping the connector's process and disabling the listen port of the connector on Optimus Core.
- When the connector's process crashed, supports checking the heartbeat of the connector and restart the connector if it is offline

4.3 Flow

A flow connects two or more integrated systems and includes one or more triggers, conditions, and responses. Under specific conditions, when all the triggers are satisfied in specific integrated system(s), with stable network, responses will be made in other specific integrated systems(s) accordingly.

4.3.1 Flow Management

- Supports adding, deleting, editing, and searching the flows (including fuzzy search by flow name)
- Supports quickly creating one or more flows by copying the settings of existing flows
- Supports enabling/disabling flows
- Supports setting network timeout for all flows on different networks
- Supports displaying flows in thumbnail and list mode
- Supports sorting flows in ascending or descending order

4.3.2 Flow Configuration

- Supports configuring responses for a single trigger

- Supports configuring AND condition for multiple triggers
- Supports setting the occurrence order of multiple events (triggers) and specify the time interval between each events (triggers).
- Supports setting conditions, including ==, !=, <=, >=, <, and >, for flows

4.4 Data Synchronization

Data synchronization rules specify what data and how the data is synchronized between two integrated systems. You can push or pull data from one integrated system to another in an adapted format.

4.4.1 Rule Management

- Supports adding, deleting, editing, and searching the synchronization rule (including fuzzy search by rule name).
- Supports enabling/disabling rules
- Supports displaying rules in thumbnail and list mode
- Supports sorting rules in ascending or descending order
- Supports manually triggering rules to synchronize data
- Supports displaying the last synchronization result.

4.4.2 Rule Configuration

- Supports manual/auto/scheduled synchronization
- Supports setting auto and scheduled synchronization at the same time

4.5 System

- Supports self-watching Optimus Core to restart the process if it stopped
- Supports smooth upgrade
- When installing Optimus, supports selecting the installation mode to Mirror Hot Spare for building a Rose hot spare system

4.6 Log

- Supports log storage
- Supports searching logs by time and component type
- Supports viewing log details

Chapter 5 Other Designs

5.1 Maintainability

- Supports printing system logs and operation logs
- Supports monitoring the service status in health monitoring module
- Supports tracking the execution time of web service and connectors by log

5.2 Security

- **Web**
  Use HTTPS to encrypt the communication links and ensure the anti-replay.
- **Connector**
  Use HTTPS to encrypt the communication links and ensure the anti-replay.
- **TPI Configuration Files**
  Connector acts as an intermediary between Core and integrated systems, and determines capabilities of Optimus. As a result, we have provided HASH signature verification for the TPI configuration files to ensure the validity of TPI information and prevent users from calling inactivate integrated system by tampering TPI files.


See Far, Go Further