

ICE iPush[®] Communication Server V2

Datasheet

Production version: 2.0-2.1

Document version: 1.6

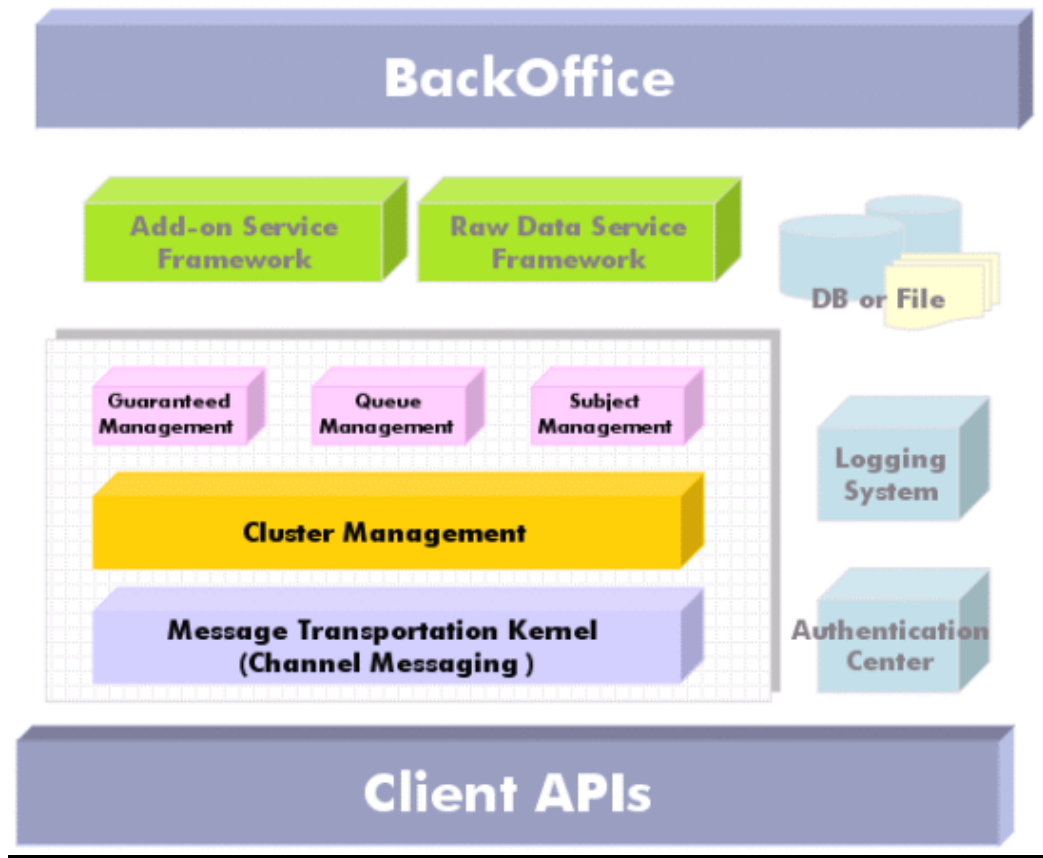
I. Introduction

ICE iPush[®] Communication Server V2 (iPush[®] Server V2) is a message-oriented middleware (MOM) to serve as a universal message delivery platform based on TCP or UDP (over IP) communication protocols.

It is suitable for the message delivery service for 1-to-n, n-to-n, n-to-1 and 1-to-1. With bi-directional communication and cross platform service ability, iPush[®] Server V2 has become state-of-the-art massive connection messaging platform.

As a middleware, iPush[®] Server V2 provides all kinds of application programming interface (API) for programmer to develop their applications for real-time message service.

II. iPush[®] Server V2 System Architecture



▲ iPush[®] Server V2 System Architecture

- ▶ **Message Transportation Kernel:** It's the message delivery kernel of iPush Server; also handles channel addressing delivery.
- ▶ **Subject Management:** Handles subject addressing delivery.
- ▶ **Queue Management:** Handles Point-to-Point (P2P) message delivery management.
- ▶ **Guaranteed Management:** Handles guaranteed message delivery functions such as persistent message, durable subscription, store-and-forward, and once-and-only-once characteristics.
- ▶ **Cluster Management:** Handles iPush Server Cluster / Cluster Gold clustering function management.
- ▶ **Authentication Center:** Handles user access authentication and rights management.

- ▶ Logging System: Handles iPush Server, Authentication Center, and License Daemon operating logs.
- ▶ BackOffice: Provide iPush Server management functions and interfaces.
- ▶ Add-on Service Framework: Option module architecture on server side to enhance iPush Server message process functionality.
- ▶ Raw Data Service Framework: Optional module architecture to simplify multiple data source connection with iPush Server.

III. iPush[®] Server V2 Edition Differentiation

iPush[®] Server V2 series includes three versions: Standalone, Cluster, and Cluster Gold. Each version contains Enterprise Edition and Standard Edition depends on whether it includes “Guarantee Message Delivery” feature.

The comparison of all iPush[®] Server V2 versions as follows:

The comparison of all iPush[®] Server V2 versions

Edition Features	Standalone		Cluster		Cluster Gold	
	Standard	Enterprise	Standard	Enterprise	Standard	Enterprise
Publish/Subscribe Messaging Model	V	V	V	V	V	V
Point-to-Point Messaging Model*	V	V	V	V	V	V
Channel Addressing	V	V	V	V	V	V
Subject Addressing	V	V	V	V	V	V
Message Queue*	V	V	V	V	V	V
Store-and-Forward*		V		V		V
Persistent Message*		V		V		V
Durable Subscription*		V		V		V
Message Priority*	V	V	V	V	V	V
Quality of Service (QoS)*	V	V	V	V	V	V
Time-To-Live (TTL)*	V	V	V	V	V	V
Connection ID*	V	V	V	V	V	V
Message ID*	V	V	V	V	V	V

Subject Message Scrambling*	V	V	V	V	V	V
Authentication Center	V	V	V	V	V	V
BackOffice	V	V	V	V	V	V
Logging System	V	V	V	V	V	V
Load Balancing			V	V	V	V
Fail Over			V	V	V	V
Scalability			V	V	V	V
HTTPS Proxy Tunneling	V	V	V	V	V	V
TCP and UDP Transmission	V	V	V	V	V	V
Large Size Message	V	V	V	V	V	V
Message Uplink			V	V	V	V
Full Connection Redundancy					V	V

▲ Above chart with * marker are operating with subject addressing only.

Feature description

- ◆ Publish/Subscribe Messaging Model: One of message delivery models. Server will actively push message to subscribers without additional request after subscription been conducted. From single message point of view, this is the method to exercise one-to-many message delivery model.
- ◆ Point-to-Point Messaging Model: One of message delivery models. Message will be transported through virtual channel so call **Queue** listing. Every delivered message will be received by one recipient only. From single message point of view, this is the method to exercise one-to-one message delivery model.
- ◆ Channel Addressing: Every channel ID is 4-Byte, support ASCII or hexadecimal naming and group subscription.
- ◆ Subject Addressing: Hierarchical oriented, support maximum up to 223 characters and group subscription.
- ◆ Message Queue: A virtual channel in P2P messaging model. All messages addressed for specific queue message will be line up in queue listing. Every listing queue message

will be delivered to one subscriber only.

- ◆ Store-and-Forward: Server will store messages then try delivering these messages to recipient. This is the most important key characteristic in guaranteed message delivery system.
- ◆ Persistent Message: Once server receive, store, delivery message with persistent tag incorporated, client side will require responding ACK confirmation to ensure per message can be transport on once-and-only-once basis.
- ◆ Durable Subscription: Server will store message intended for delivery once subscriber is offline. Redeliver message once subscriber is back online.
- ◆ Message Priority: Adding predefined priority sequence order in the process of message generation. Resulting earlier message deliverable to subscriber with queue jump capability.
- ◆ Quality of Service (QoS): To ensure transport operation liquidity, every message can be classified with importance. The least important message class can be drop by server when system transport encounter heavy traffic.
- ◆ Time-To-Live (TTL): Message preservation time in system. Every message will have related header field indicate per message validation period during process of message generation.
- ◆ Connection ID: Every established user connection with sever will be given a length of 12-bit Connection ID to apply differentiation. Per Connection ID will be valid during each connection. Every new connection will be given a new Connection ID.
- ◆ Message ID: Ever published subject message to server will be given a length of 32-bit Message ID to apply differentiation.
- ◆ Subject Message Scrambling: Message system will scramble and encode for each subject message. Then decode back to original format after transport to receiver. This is the encryption method to protect subject message security.
- ◆ Large Size Message: When transporting over default size message or file, transporting side's API will automatically slice, transport message, then reassemble sliced message by subscribed receiving side's API.
- ◆ TCP and UDP Transmission: Every client side can choose TCP or UDP connection when establishing connection with server.

- ◆ HTTPS Proxy Tunneling: iPush is HTTPS tunneling compliance. Message can successfully tunnel through firewall and gateway to fit in enterprises network infrastructure.
- ◆ Authentication Center: This is user authentication center server (AC Server), process user authentication operation for all connections.
- ◆ BackOffice: Provide iPush Server management functions and interfaces.
- ◆ Logging System: Handles logging for iPush Server, AC Server, and License Daemon operation.
- ◆ Load Balancing: Server is capable to conduct concurrent multiple message service procedures. With its unique design to balance single iPush Sever when facing massive client side concurrent connection payload to enhance overall system transportation performance and communication quality.
- ◆ Fail Over: When deploying two or more servers operating concurrently with manual or automated monitoring system. Fail Over can relocate online operating tasks between different message servers with least time to keep tasks continuation.
- ◆ Scalability: Through distributed deployment, it's very convenient to serve thousands of clients with server farms.
- ◆ Message Uplink: Single unit of server can be configured to uplink another in order to make entire cluster or inter-cluster linking with same messages.
- ◆ Full Connection Redundancy: Under circumstances of user authentication server redundant cluster deployment, as long as one of authentication server still operating properly, system can full guarantee user connection amount up to cluster authorization.

IV. Server System Requirement

Hardware Requirement

H/W component	Requirement
CPU	Pentium III or above
Memory	256 MB ~ 1 GB (depends on the number of concurrent connection)
HDD	10 GB or above (full program installation need approximately 50 MB, others will vary upon guaranteed message storing volume)
NIC	Ethernet 100 Mb/Sec. ~ Gb/Sec.

Operation System

- Windows 2000 Family / XP / Server 2003 (NTFS suggested)
- Linux (Red Hat / Fedora Core distributions)

TCP/IP

- Assign server a fix IP address (127.0.0.1 prohibited)

BackOffice operating environment requirement

- J2SDK (Java 2 Software Development Kit v1.3.x or above)

License Key

- To start iPush Sever, you must obtain a valid license key (*.lic) from ICE Technology Corp.

Installation components

Installation components	Component brief description
iPush kernel	iPush Server messaging kernel. Service and application execution modes are both provided in Windows version.
Auth. Center	iPush Server user authentication and authorization management service system (AC Server).
License Daemon	iPush Server license management system.
BackOffice	Web-based (Java Servlet / JSP) management tool of iPush Server.
MySQL	Database system for iPush Server BackOffice.
MyODBC	MySQL ODBC driver.
Tomcat	Java application server for iPush Server BackOffice.

V. Client side application programming interfaces (APIs)

- **ActiveX Control** for Win32 (OCX)
Support Windows application development
- **ActiveX Control** (OCX) for Windows CE.Net / CE 4.x / PocketPC
Support Windows embedded system application development
- **Java Package** (Class)
Support Java application and Java applet development
- **Linux C Library**
Support static linking and dynamic linking
- **MS .NET Framework Library**
Support MS .NET Framework variety of program development tools
- **MS .NET Compact Framework Library**
Support Pocket PC and Smartphone application development
- **uClinux C Library**
Support Embedded System application development

VI. BackOffice Management Interface

- System Monitor
- License Management
- Admin. Management
- Service Management
- User Management
- CGI Access Management
- Report Generator
- Database Backup
- Cluster Management (only for Cluster or Cluster Gold version)

Contact

Global headquarter

Address: 12F-1, No. 9, Sec. 2, Roosevelt Rd., Taipei, Taiwan 100
Telephone: +886-2-2396-1880
Facsimile: +886-2-2396-1881
Website: <http://www.icetechnology.com>
Developer site: <http://www.icetechnology.com/icedc>
E-Mail: service@icetechnology.com

ICE China

Address: Room 1607, 16F, No. 933, Jian She Avenue - Hankou, Wuhan, 430015 China
Telephone: +86-27-8265-6360
Facsimile: +86-27-8265-6378

About ICE Technology Corporation

ICE Technology Corp. is the 1st software company in Asia to develop MOM (message-oriented middleware) as core business. Aggressively market product internationally, vowed to be: "[The World-class Business Communication Software Provider](#)". Current products have been deployed and integrated by world class enterprises, Future Exchange, financial service institutions, online game design company, military agency, environmental bureau and agency, weather bureau and agency, water resources bureau and agency, academic agency, and IT technology vendor.

iPush Server is a registered trademark of ICE Technology Corp.

Copyright © 2003-2005 ICE Technology Corporation. All Rights Reserved.