uCosminexus Application Server and BPM/ESB Platform Terminology Guide

3020-3-Y99(E)

■ Relevant program products

See the manual *uCosminexus Application Server Overview*.

■ Export restrictions

If you export this product, please check all restrictions (for example, Japan's Foreign Exchange and Foreign Trade Law, and USA export control laws and regulations), and carry out all required procedures.

If you require more information or clarification, please contact your Hitachi sales representative.

■ Trademarks

AIX is a trademark of International Business Machines Corporation in the United States, other countries or both.

CORBA is a registered trademark of Object Management Group, Inc in the United States.

Gzip is software distributed by FSF (Free Software Foundation) in the United States.

HP-UX is a product name of Hewlett-Packard Development Company, L.P. in the U.S and other countries.

Linux is the registered trademark of Linus Torvalds in the U.S and other countries.

Microsoft is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

OMG, CORBA, IIOP, UML, Unified Modeling Language, MDA, Model Driven Architecture are either registered trademarks or trademarks of Object Management Group, Inc. in the United States and/or other countries.

Oracle and Java are registered trademarks of Oracle Corporation and/or its affiliates.

SOAP is an XML based protocol for sending messages and making remote procedure calls in a distributed environment.

UNIX is a registered trademark of The Open Group in the United States and other countries.

W3C is a trademark (registered in numerous countries) of the World Wide Web Consortium.

WebSphere is a trademark of International Business Machines Corporation in the United States, other countries, or both.

Windows is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

Windows Server is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

Windows Vista is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

XSL is a trademark of the World Wide Web Consortium.

The other company names and product names are either trademarks or registered trademarks of the respective companies.

Eclipse is an open development platform for tools integration provided by Eclipse Foundation, Inc., an open source community for development tool providers.

This product includes software developed by the Apache Software Foundation (http://www.apache.org/).

■ Microsoft product name abbreviations

This manual uses the following abbreviations for Microsoft product names.

Abbreviation			Full name or meaning
Microsoft IIS	Microsoft IIS 7.0		Microsoft(R) Internet Information Services 7.0
	Microsoft IIS 7.5		Microsoft(R) Internet Information Services 7.5
Windows	Windows Server 2008	Windows Server 2008 x86	Microsoft(R) Windows Server(R) 2008 Standard 32-bit
			Microsoft(R) Windows Server(R) 2008 Enterprise 32-bit#
		Windows Server 2008 x64	Microsoft(R) Windows Server(R) 2008 Standard
			Microsoft(R) Windows Server(R) 2008 Enterprise#
		Windows Server 2008 R2	Microsoft(R) Windows Server(R) 2008 R2 Standard
			Microsoft(R) Windows Server(R) 2008 R2 Enterprise#
			Microsoft(R) Windows Server(R) 2008 R2 Datacenter
	Windows Server 2012	Windows Server 2012 Standard	Microsoft(R) Windows Server(R) 2012 Standard

Abbreviation			Full name or meaning
Windows	Windows Server 2012	Windows Server 2012 Datacenter	Microsoft(R) Windows Server(R) 2012 Datacenter
	Windows XP		Microsoft(R) Windows(R) XP Professional Operating System
	Windows Vista	Windows Vista Business	Microsoft(R) Windows Vista(R) Business (32 bit)
		Windows Vista Enterprise	Microsoft(R) Windows Vista(R) Enterprise (32 bit)
		Windows Vista Ultimate	Microsoft(R) Windows Vista(R) Ultimate (32 bit)
	Windows 7	Windows 7 x86	Microsoft(R) Windows(R) 7 Professional (32 bit)
			Microsoft(R) Windows(R) 7 Enterprise (32 bit)
			Microsoft(R) Windows(R) 7 Ultimate (32 bit)
		Windows 7 x64	Microsoft(R) Windows(R) 7 Professional (64 bit)
			Microsoft(R) Windows(R) 7 Enterprise (64 bit)
			Microsoft(R) Windows(R) 7 Ultimate (64 bit)
	Windows 8	Windows 8 x86	Windows(R) 8 Pro (32 bit)
			Windows(R) 8 Enterprise (32 bit)
		Windows 8 x64	Windows(R) 8 Pro (64 bit)
			Windows(R) 8 Enterprise (64 bit)
Windows Server Failover Cluster			Windows Server(R) Failover Cluster

Note that Windows 32 bit and Windows 64 bit are sometimes respectively referred to as Windows x86 and Windows x64.

#: Note that these product names are referred to with a general name as Windows Server 2008 Enterprise.

■ Restrictions

Information in this document is subject to change without notice and does not represent a commitment on the part of Hitachi. The software described in this manual is furnished according to a license agreement with Hitachi. The license agreement contains all of the terms and conditions governing your use of the software and documentation, including all warranty rights, limitations of liability, and disclaimers of warranty.

Material contained in this document may describe Hitachi products not available or features not available in your country.

No part of this material may be reproduced in any form or by any means without permission in writing from the publisher.

Printed in Japan.

■ Issued

Aug. 2013: 3020-3-Y99(E)

■ Copyright

Preface

For details on the prerequisites before reading this manual, see the manual *uCosminexus Application Server Overview*.

■ Non-supported functionality

Some functionality described in this manual is not supported. The non-supported functionality includes:

- Audit log functionality
- Compatibility functionality
- Cosminexus Component Transaction Monitor
- Cosminexus DABroker Library
- Cosminexus Reliable Messaging
- Cosminexus TPBroker and VisiBroker
- Cosminexus Web Service - Security
- Cosminexus XML Security - Core functionality
- JP1 linkage functionality
- Management portal functionality
- Migration functionality
- SOAP applications complying with specifications other than JAX-WS 2.1
- uCosminexus OpenTP1 linkage functionality
- Virtualized system functionality
- XML Processor high-speed parse support functionality

■ Non-supported compatibility functionality

"Compatibility functionality" in the above list refers to the following functionality:

- Basic mode
- Check of JSP source compliance (cjjsp2java) with the JSP1.1 and JSP1.2 specifications
- Database connection using Cosminexus DABroker Library
- EJB client application log subdirectory exclusive mode
- J2EE application test functionality
- Memory session failover functionality
- Servlet engine mode
- Simple Web server functionality
- Switching multiple existing execution environments
- Using EJB 2.1 and Servlet 2.4 annotation

Contents

1 Terminology Guide

1.1 Terminology guide for Application Server and BPM & ESB platform

This chapter describes the terminology given in the manuals of Application Server and BPM & ESB platform.

1.4 mode

A server operation mode that supports the functions of J2EE 1.4 or later. This mode enables transaction management for multiple resources, including databases.

1-to-1 node switching system

System with 1-to-1 mapping of Application Server on the executing node and Application Server on standby node. When an error occurs in the Application Server on the executing node, the Application Server on the standby node inherits the process.

Abstract parameter

A parameter that combines mutually related parameters as one parameter. The Smart Composer functionality uses this parameter. The Smart Composer functionality provides a command that expands an abstract parameter to the multiple effective parameters.

Action

A term that describes the operation during transition of Web windows.

Action definition file

A collective name for the common definition file and the process-wise definition file. Defined to set up the operations of the JAX-WS engine such as log output and timeout.

activity

A component that defines the processing flow of business processes. Activities are classified into basic activities and structured activities. The user defines the processing flow of a business process by arranging multiple activities and connecting them using a window.

Active node

A node initially operated as the executing node is called an *active node*. The invocation method does not change even when the executing node becomes a standby node due to node switching.

Address

This is a system that serves as the message sending destination.

Administration target object

An object (AdminObject) defined in the Connector 1.5 specifications that is used for transmitting the messages using a queue and a topic. The administration target object is used for the message conversion during the dispatch of messages, and also during the synchronous or asynchronous reception of messages.

The administration target object is provided as a unique object in each message provider. The administration target object used in a resource adapter is specified with the `<adminobject>` tag of the resource adapter DD (`ra.xml`).

Adapter

An application that receives request messages from service requester and sends request messages to service components defined in the adapter, and invokes the service component.

There are two types of adapters; standard resource adapters (standard adapter) and custom adapters used by users as and when required (such as DB adapters or TP1 adapters). Service adapters invoke service components of Web Services, SessionBeans, MDB (WS-R), and MDB (Database queue). Custom adapters invoke other service components.

Alias

Any name given to a J2EE resource and the EJB home object reference registered in the JNDI name space. An alias is added when the user-specified name space functionality is used. It is also called an *Optional Name*.

Annotation

A coding format for embedding the additional information of a class and method into a source code. By specifying an annotation, you can acquire a reference to a resource and other Enterprise Beans on which Enterprise Beans depend.

Application

A program implemented by the user according to a business. Cosminexus RM applications are implemented using the API provided by Cosminexus RM.

Application Development Plug-in

This is the plug-in used at the time of development and testing of an application.

Application server

Middleware used for building an application layer, to serve as an interface between the user requests (presentation layer) and the processing of a business system (data layer) such as the database, by setting the application server in the middle of information system.

Application Server is a middleware that is located at the center of a business system, and it not only serves as an interface for mutual requests and responses exchange, but also provides the functionality required for implementing, managing and controlling business processes. Application Server is the base of a business process located in the application layer, among three layers such as the presentation layer, application layer and data layer.

Application Server mainly indicates a server having functionality for executing a Java application conforming to the Java EE specifications and provides Application Server equipped with the operation functionality linked with the following functionality, based on the support to the Java EE specifications:

- EJB flow-volume and priority control by the scheduler
- Load balancing
- Batch application scheduling function
- Function for RPC connection reception from OpenTP1 Virtual platform

Archive format

A J2EE application that assembles the programs and DDs constituting a J2EE application, and compiles them in a WAR file, EAR file, or ZIP file.

A program or DD that configures a J2EE application is packaged in the WAR file, EAR file or ZIP file.

Assemble

An assembly operation used for positioning an EJB-JAR that does not operate independently, as a configuration element of an application.

Asynchronous parallel processing of threads

The functionality for implementing asynchronous parallel processing of threads, such as the asynchronous timer processing and asynchronous thread processing, in the Java EE environment. The Cosminexus Application Server provides APIs based on the specifications of Timer and Work Manager for Application Servers stipulated by `CommonJ`. As a result, you can execute asynchronous timer processing using TimerManager and asynchronous thread processing using WorkManager.

Audit trail

Audit trail is a task record that is output by database functionality, and is related to the database. The audit trail is used for checking that the authentication operation for the database was appropriately performed. The database audit trail integration functionality is provided by Application Server to output the information of the system that is configured by Application Server in the audit trail.

Audit log

File to output the log of operations and the behavior of programs as per the operations executed by the person setting up the system or by the system operator for the programs of Application Server. The auditor can know as to when, who has executed what by checking the audit log, and can verify whether the system operations are compliant to various standards, such as laws and regulations, security evaluation criteria, or business standards.

Automatic node switching

One of the methods for node switching, in which cluster software automatically switches nodes.

Auto-generated class

Java class generated when the `cjwsimport` command and `apt` command are executed.

Basic activity

An activity is a component that defines the process flow of a business process. An activity is classified into a basic activity and a structured activity. It defines a processing flow of a business process, by scheduling various activities on a window and linking those activities.

Basic mode

One of the operation modes of a server to secure compatibility with a previous version.

Transaction management with a single resource can be performed.

Batch application

A Java application that implements batch processing. Use batch execution commands to execute batch applications on a batch server.

Batch Job Execution Server

This is the product that provides a batch job execution platform for executing/operating the base job batch operations in the open environment. Use it by combining with JP1/AJS.

Batch server

A server used for executing a batch application. Apart from providing the functionality for executing a batch application, the batch server also provides the functionality for establishing a connection from a batch application to the database, and for accessing the EJB.

binary format definition file

A message format definition file that is used to handle binary data (other than XML data). It is created as an XML schema file. It defines the storage format for binary data values, and the defined information is then added to the XML schema file (extension: .fdx).

Binding

Specifies the data correspondence performed by JAXB between the XML document defined in the schema document and the Java object.

Bind address specification functionality

Functionality to explicitly specify the IP addresses that are used in Web server integration and simple Web server, in the Web container. If you execute this functionality in a host that has either multiple physical network interfaces or a single physical network interface, you can set any IP address by using this functionality.

BJEX

uCosminexus Batch Job Execution Server. For details, see *Batch Job Execution Server*.

BMP

This is an abbreviation of *Bean-Managed Persistence*. For details, see *Bean-Managed Persistence*.

BMP Entity Bean

The word BMP in a BMP Entity Bean is an abbreviation for Bean-Managed Persistence, and implies that a Bean itself executes the persistence-related processing. All access codes to a resource need to be coded by an application developer. Implementation of BMP takes more time and effort as compared to CMP, BMP has a wider operation scope and higher level of freedom than CMP.

BMT

This is an abbreviation of *Bean-Managed Transaction*. For details, see *Bean-Managed Transaction*.

BPEL

This is an abbreviation of *Business Process Execution Language*. For details, see *Business Process Execution Language*.

BPMN

This is an abbreviation of *Business Process Modeling Notation*. For details, see *Business Process Modeling Notation*.

Business interface

Interface to define a business method to invoke Enterprise Beans.

Business-oriented tree

One of the monitoring trees displayed in the integrated scope when the system is integrated with JP1/IM. The business-oriented tree is configured mainly of the J2EE applications running on the system that is built using Application Server. The business-oriented tree enables the user to monitor which problem has occurred in which business, and monitor the trouble from the business point of view.

business process

A business process defines the processing order and conditions of multiple service components as a business series. In the development environment of Cosminexus service platform, a business process deploys multiple activities in a window, connects and defines them.

Business process base

This is functionality that calls appropriate service components in accordance with the business process.

Business Process Execution Language

This is workflow coding language based on XML. With service platform, you can define the business process in this format, by using a GUI.

Business Process Modeling Notation

This is the notation in which the description rules for plotting and visualization of the business process in easily understandable format are consolidated. This is also called Business process modeling notation method.

business-processing program

Enterprise Bean with business logic implemented.

CA

This is an abbreviation of *Certificate Authority*. For details, see *Certificate Authority*.

Catalog function

The catalog function executes mapping (association) based on the XML Catalogs 1.1 specifications. The mapping (association) has the following two types:

- Mapping of a namespace URI of an XML schema specified in the `namespace` property of the `xsd:import` element and the URI reference that indicates the location of the XML schema
- Mapping of a URI reference that indicates the location of a WSDL or XML schema and the URI reference that indicates the location of a different WSDL or XML schema

Catalog file

The catalog file is scheduled when you use the *Catalog* function. This file defines the mapping (association) information.

Certificate Authority

This is function for issuing certificate for using SSL is also called as authentication authority.

This is entity in reliability that issues a valid certificate for other entity. The types of this certificate are:

- Root CA
  Correctness of self is certified by self, without receiving the certification of the upper level CA.
- Intermediate CA
  Receives the certificate from the upper level CA for certifying the correctness of self.

Cache file

A file that exists on a disk to save a disk cache.

Cache table

An internal table to save a memory cache. A cache table is created when a user starts a J2EE server.

CA

CA is an abbreviation of Certification Authority. CA is the certification authority that issues the certificate to use SSL.

Central console

A mechanism for integrated system management using JP1 events to centrally manage the events occurred system when the system is integrated with JP1/IM.

Central scope

A mechanism for monitoring a target-oriented system by providing a window suitable to the administrator's purpose as the system monitoring window, when a system is integrated with JP1/IM.

Character code conversion UOC

An external program for character code conversion. The character code conversion UOC is developed by using the jar files provided by Cosminexus service platform.

CJMSP Broker

A process used by the Cosminexus JMS provider. CJMSP Broker manages the sending of messages.

CJMSP resource adapter

A resource adapter used by Cosminexus JMS provider.

The CJMSP resource adapter is used for connecting J2EE server to CJMSP Broker. Complies with the Connector 1.5 specifications.

Client application information

Information output to the performance analysis trace. The following information is set in the unit that invokes the Enterprise Bean.

- Invoking the EJB container from the Web container
- Invoking the EJB container from the EJB client
- Invoking the EJB container from the EJB container

Client definition file

A file used in the SOAP Communication Infrastructure. The file is used for defining the operations of the SOAP Communication Infrastructure when sending a SOAP message.

Cluster

A cluster is a set of servers that provide common functions. You can perform operations and management by considering one entire set as one server.

Cluster software

A program for switching a system with the purpose of improving the reliability and operating state of a system.

If there is a failure in the running server system, you can automatically and immediately switch to the standby server system arranged beforehand. Therefore, even without the special attention of the operator, the reliability and operating state of the system can be improved.

In Application Server, you can implement a 1-to-1 node switching system, mutual node switching system, and node switching system for a host unit management model using cluster software.

If you are using AIX, HP-UX, or Linux, you can use HA monitor as the cluster software. In Windows, you can use Microsoft Cluster Service and Windows Server Failover Cluster as the cluster software.

CMP

This is an abbreviation of *Container-Managed Persistence*. For details, see *Container-Managed Persistence*.

CMP Entity Bean

The word CMP in a CMP Entity Bean is an abbreviation for Container-Managed Persistence, and implies that an EJB container executes the persistence-related processing. In the case of CMP, a developer need not write the code for accessing a resource (such as a database), and the EJB container executes all transactions with the resource.

CMT

This is an abbreviation of *Container-Managed Transaction*. For details, see *Container-Managed Transaction*.

Common definition file

Definition file used to set up operations common to the system.

Common folder

A folder created on the HCSC server to store files that can be accessed from multiple business processes during FTP integration.

Command

In this manual, the `cjwsimport` command and `apt` command used for developing a Web Service are collectively called command.

Commit option

Option that specifies the cache method of a CMP field and the cache model of the Entity Bean status transition.

In the case of Full caching, an Entity Bean is started with the same status as that at the time of the last transaction commit.

In the case of Caching, an Entity Bean is started with the status same as the latest status of the database.

In the case of No caching, an Entity Bean is started with the status same as the latest status of the database and becomes passive at the time of transaction commit.

Component Container Administrator

Users other than the super users who can set up, start, and stop J2EE servers and batch servers. A Component Container Administrator can be set in a UNIX system.

connection

A component for linking activities and defining their execution order. In addition to a normal connection, there is a fault connection for linking fault processing, and a link connection for setting links during parallel processing.

Connection sweeper

Functionality to discard unused connections in the connection pool at fixed intervals.

Connection pooling

The functionality for pooling the resource connection in memory based on the access to the resource from the J2EE components such as servlets, JSPs, and Enterprise Beans, and batch applications. The Resource connections include JDBC connections and resource adapter connections. By pooling a connection, you can process a request for a resource connection from an application at high speed.

container extension library

A library that extends a user-created library interface and function in such a way that it can be used from a J2EE container, Web container, and batch server.

Container-Managed Persistence

This is the model that EJB container users for managing the persistence of the data.

Container-Managed Transaction

This is the model that the EJB container uses to manage the transaction. Target is Session Bean, Entity Bean, and Message-driven Bean.

Context

Management unit that compiles the Web applications.

Context path

See *Context root*.

Context root

Root path of a context. Specify the context root in a URL when accessing a Web application present inside a context. A context root is also called a context path.

cookie switching functionality

One of the server switching functionalities provided by load balancers. When the cookie switching functionality is used, a session is redirected to the same server based on the cookie information in the request packet from the client.

correlation set

A character string used to uniquely identify a request message that has been sent from a service requester. A request message including a correlation set is sent from a service requester to an HCSC server. The service component's HCSC server sends the response message containing the correlation set back to the service requester. As a result, the service requester can identify the request message to which the received response message corresponds.

CORBA Naming Service

A Naming Service that conforms to the CORBA specifications for managing the location for storing the remote objects. The CORBA Naming Service is a function provided by Cosminexus TPBroker. Cosminexus TPBroker is component software of Application Server.

cosminexus.xml

Property file used for defining information unique to Cosminexus Application Server.

Cosminexus Application Development Plug-in

A plug-in used when developing and testing an application.

Cosminexus Component Container

Configuration software for executing a business-processing program (business logic) as a component on the server machine. The Cosminexus Component Container also provides the functionality such as Management Server for managing the application server, integrated user management, and snapshot log collection.

Cosminexus Component Transaction Monitor(CTM)

Component software for scheduling requests from a client.

Cosminexus JMS provider

A JMS provider provided by Application Server. Cosminexus JMS provider provides functionality based on JMS1.1 specifications and is configured by CJMSP broker and CJMSP resource adapter.

Cosminexus JPA Provider

JPA provider provided with Application Server. Apart from the functionality based on the JPA1.0 specifications, Cosminexus JPA Provider provides the cache functionality and the pessimistic lock functionality of the entity objects.

Cosminexus Manager

The management functionality for systems running on Application Server.

Cosminexus Performance Tracer

Configuration software used to output the performance analysis output by each functionality information to a file at predetermined points when a request is processed.

Cosminexus Application Server

The base product (with the core of an application server) for building a system for developing and executing applications with high performance and reliability.

Cosminexus standard login module

Login module used in the user authentication process of integrated user management. It is the implementation class of the `LoginModule` interface that is included in `javax.security.auth.spi`.

The login modules are as follows:

- `WebPasswordLoginModule`
  Uses a password to authenticate the user.
- `WebCertificateLoginModule`
  Uses the client certificate to authenticate the user.
- `WebPasswordLDAPLoginModule`
  Uses the authentication functionality of LDAP directory server to authenticate the user.
- `WebPasswordJDBCLoginModule`
  Uses the information stored in RDB to authenticate the user.
- `DelegationLoginModule`
  Invokes the custom login module.
- `WebSSOLoginModule`

Invokes other login modules during single sign-on.

CTM

This is an abbreviation of *Component Transaction Monitor*. For details, see *Component Transaction Monitor*.

CTM daemon

A process for managing the schedule queue that controls requests from the client. The CTM daemon functions as a scheduler for scheduling requests. If you are using Management Server, you can handle CTM daemon as a logical server.

CTM domain

A CTM domain is composed of multiple CTM daemons having shared information and a load balancer. A CTM daemon distributes the requests by exchanging the information of a business processing program that is registered in the respective CTM domain, and by exchanging the load balancer information of the schedule queue within the CTM domain.

CTM domain manager

A process for managing a CTM domain. The information about the CTM daemons on a host is distributed to another CTM domain manager within the CTM domain.

If you are using Management Server, you can handle CTM domain manager as a logical server.

CTM regulator

A process for distributing and consolidating the requests collected in a CTM daemon from the EJB client. The CTM regulator also provides the gateway functionality for enabling direct invocation of EJB applications running on J2EE servers from the TPBroker client (CORBA client).

Custom reception

A functionality (interface) used for receiving execution requests from the service requester. Execution requests from any protocol can be received by operating user-created reception processing on the custom reception framework.

Custom reception framework

A development framework provided on Cosminexus Service Platform for creating a custom reception.

Custom binding declaration

A customization instruction described in the schema document to be input when generating a Java source file using the schema compiler. The custom binding declaration is used when you want to change the default JAXB binding.

Custom function

One of the functions used in the data transformation definition. Using this function, any user-created Java program can be invoked and the data can be processed.

Custom login module

Implementation class created when you want to use a module other than the Cosminexus standard login module to perform user authentication of each application.

Data source

Functionality for connecting to a database by using JDBC.

data transformation

Conversion of the format of a standard message to that of a service component message according to the data transformation definition, when the two types of message have different message formats.

Data transformation base

Data transformation base is the function that transforms the message formats of request message, response message, in accordance with the data transformation definition.

data transformation definition

The data transformation definition is used to resolve differences in the message format between standard messages and service component messages. Each element is mapped between the message format definition files for standard messages and those for service component messages, and if necessary, functions are set.

data element

Data that is an element of an XML document. The term data element indicates, for example, the string enclosed within tags or tag attributes.

DB adapter

A service component that defines the SQL statement to be executed and database to be operated when using the database operations in the service integration environment as a service module.

DB Connector

A resource adapter used for connecting to a database.

DD

This is an abbreviation of *Deployment Descriptor*. For details, see *Deployment Descriptor*.

Dependency Injection

Functionality used to make the EJB container automatically set up the reference to an EJB and a resource by specifying the EJB and Resource annotation in the EJB and interceptor class field and setter method. If you use Dependency Injection, you do not need to use JNDI to look up the reference to the EJB and resources.

Deploy

In the case of J2EE application, to enable execution of the J2EE application imported in the J2EE server, from the client.

In the case of J2EE resource adapter, to enable usage of the J2EE resource adapter imported in the J2EE server from all the J2EE applications running on the J2EE server.

deployment

An action that deploys HCSC components at an appropriate HCSC server. An HCSC component created in the development environment is deployed in the execution environment (HCSC server) via the operating environment.

deployment definition

Deployment definition involves defining how HCSC components are deployed at the HCSC server and cluster in the development environment, and then updating the system configuration definitions. When a HCSC component is deployed from the operating environment, it is deployed according to the system configuration definitions that contain the deployment definitions created in the development environment.

Deployment Descriptor

Codes the definition information when allocating an application to the operation environment.

Deployment directory format

This is a J2EE application format that has the entity of applications such as EJB or servlet, in a file or directory that conforms to certain rules outside the J2EE server.

Developer

An infrastructure for creating the development environment of Application Server. Developer is a collective term for Developer Standard and Developer Professional.

development environment

An environment used to create HCSC components that are needed for calling service modules in the execution environment. The development environment is also used to define how the HCSC components, once created, are deployed at the HCSC server and cluster (deployment definition).

Development Environment Instant Setup

Sets up the server environment required to develop J2EE applications. This functionality is provided by Developer. If you execute this functionality, the Management Server, host, performance tracer, J2EE server, and embedded database will be set up.

DII

This is an abbreviation of *Dynamic Invocation Interface*. For details, see *Dynamic Invocation Interface*.

Disk cache

A schema cache that is saved on a disk's

DI target

Target for injecting the reference by using the Dependency Injection functionality. Dependency Injection is executed for those fields whose names match with the DI target name or for the setter method that consists the DI target name as property.

document/literal

Interface type for sending and receiving XML documents with the types defined in the schema, as child elements of the SOAP Body.

DOMDocument Object Model

An acronym for Document Object Model. DOM describes the content of the XML document parsed by the DOM parser, and data is structured in a tree format that is resident in memory. This tree structure is called a *DOM tree*.

DOM

This is an abbreviation of *Document Object Model*. For details, see *Document Object Model*.

DOM API

An API to access a DOM generated by the DOM parser. The DOM parser can process data based on the data structure in the XML document.

DOM tree

Structure of the XML document displayed in a tree format. A DOM tree is generated by parsing by using the DOM parser.

DTD

This is an abbreviation of *Document Type Definition*. For details, see *Document Type Definition*.

Dynamic contents

Contents generated dynamically according to the request from the client such as a servlet and JSP.

Dynamic Invocation Interface

This is the function that calls SOAP service with the standard API defined in JAX-RPC, without using the service interface and stub. As it does not use the stub, calling destination is not fixed to a specific service and you can dynamically change the calling destination.

EADs

This is an abbreviation of *Elastic Application Data store for Application Server*. EADs is a product for managing data in the memory, in a simple format like a key and a value. With EADs, you can manage multiple servers (EADs servers) as a cluster, process the large amount of data for using the memory area of individual servers within the cluster as a single memory area, and can access the data without considering the individual server from the client.

With Application Server, you can use the EADs session failover functionality that uses EAD as the session information storage destination server.

EADs client

This is an abbreviation of *Elastic Application Data store Client for Application Server*. This is the client software for executing EADs functionality by accessing the EADs server.

EADs server

This is the server process for managing data with EAD. You can include this in the system configuration when using the EADs session failover functionality.

EAR file

This is an abbreviation of *Enterprise ARchive file*. For details, see *Enterprise ARchive file*.

Eclipse

An open source integrated development environment provided by the Eclipse project (eclipse.org). It provides various functions for improving application development efficiency, such as a function that supports source code editing, and a debugging function.

The service platform provides the functions required for the development of an SOA applied system, as plug-in.

Eclipse Web Tools Platform

This is a development environment of the Java application.

effective parameter

A parameter that is set in definition files of application servers. The Smart Composer functionality enables the user to set values in multiple related effective parameters in the batch mode by setting a single abstract parameter.

EIS

A back-end system such as the database and transaction server, built in an enterprise.

EJB

This is an abbreviation of *Enterprise JavaBeans*. For details, see *Enterprise JavaBeans*.

EJB-JAR file

Compresses an Enterprise Bean and EJB-JAR DD in the JAR format.

EJB client

Client program invoking the Enterprise Beans started on the J2EE server. The three types of EJB clients are as follows:

- EJB client applications
- Web applications, such as servlets or JSPs
- Other Enterprise Beans

EJB client application

A Java application invoking an Enterprise Bean.

EJB container

An execution environment that controls an Enterprise Bean. An EJB container also provides system level services such as communication and transaction management. The entity of the Enterprise Bean is executed in an EJB container.

EJB timer

A timer for managing timeouts by using the Timer Service functionality of an EJB container. An EJB timer calls back the timeout method at a specified time or intervals and executes the events.

Embedded database

A standard database provided by Developer. You can use the embedded database for developing and testing an application on a local machine.

Enterprise Archive file

This file is the file in which multiple files (EJB-JAR file, WAR file, and DD) that configure the J2EE application are packaged in the EAR file format.

The Cosminexus Service Platform creates the EAR file by packaging the HCSC components.

Enterprise Bean

Business logic created according to the EJB architecture. Applicable to business-processing programs.

Enterprise JavaBeans

An entity that stores business logic in which the business logic is coded as a program and persistent data is converted to Java components.

entity

An entity contains files that are used in XML documents (such as image data or PDF files) or character strings that are substituted.

Entity Bean

A bean indicating the persistent data. An Entity Bean is further classified into CMP Entity Bean and BMP Entity Bean.

Event listener

An object that receives and processes an event. In Application Server, this object receives the status change event of a servlet context object or a session object for each Web application.

execution environment

An environment where appropriate service modules are called and applications are executed via HCSC server, according to execution requests accepted by a service requester.

execution log

The two types of execution logs are the process instance execution log of a business process and the message execution log. The user can use these logs to check the execution and response status of business processes, and the delivery route and status of service module information and messages.

execution log information table

A table used to manage the process instances of business processes and message execution logs of messages.

Error page customization

Functionality used to change and display an error page that is returned from Web container to any page created by the user when a non-existing resource is accessed from a Web client.

Explicit heap

A memory space used in explicit management heap functionality. An explicit heap is not considered for full garbage collection as it is outside the Java heap.

Explicit management heap functionality

The functionality using an explicit heap as place for deploying a Java object. An explicit heap is not considered for full garbage collection as it is outside the Java heap. You can use the explicit management heap functionality to avoid occurrence of a full garbage collection.

Extensible Markup Language

This is the W3C specified markup language and expresses the data exchange on the internet. In this manual, document coded in XML format is called *XML document*.

Extensible Stylesheet Language

This is the stylesheet of XML and it can code not only the display method; but also the data processes such as data copy, bifurcation, and extraction.

FIFO

A unit for grouping messages. Messages are managed for each FIFO.

A FIFO is broadly classified into the following three types:

1. FIFO for registration or releasing
   A unit for messages that exist in the queue to be registered or released from the application.
2. FIFO for sending
   A unit for messages sent to another system in the communication layer.

3. FIFO for receiving

   A unit for messages received from another system in the communication layer.

   2 and 3 are sometimes generically called *FIFO for communication*. The FIFO for communication has the same meaning as a group.

file adapter

An adapter used for direct input and output of files from the execution environment of cosminexus Service Platform to the local disk on a HCSC server.

File operation adapter

An adapter used for layout transformation, character code transformation, duplication, and deletion of files.

Forced termination of J2EE applications

Method used to forcibly interrupt the processing, even if requests are being processed in J2EE applications.

Forced termination of J2EE applications

Method used to forcibly interrupt the processing, even if requests are being processed in J2EE applications.

Framework

A frame to stipulate a function and service. By using a framework, you can easily create applications that you want to use in a business.

Front-end EJB

The Enterprise Bean that is used as a receiver of requests.

FTP adapter

An adapter used to send and receive files between cosminexus Service Platform and an FTP server.

FTP inbound adapter

A resource adapter used in FTP integration. Conforms to Connector 1.5 specifications.

FTP reception

Functionality (interface) that receives execution requests from the FTP client via the FTP inbound adapter.

Files can be transmitted between the FTP client and FTP server on cosminexus Service Platform by integrating with the file operation adapter and FTP adapter.

FTP client

Client software that uses FTP to send and receive files.

FTP server

Server software that sends and receives files based on the requests from the FTP client.

FTP integration

Functionality used to transmit files between the FTP client and FTP server on cosminexus Service Platform.

function

A definition of the data transformation processing method used when the value of the source node is processed and mapped to the target node.

Supported functions include the conversion of character strings and numeric values, the acquisition of node names, and the definition of selection and iteration processing according to the conditions that have been specified.

Functionality Layer

A layer of functionality to output the PRF trace when performance analysis trace is collected. This layer includes the EJB client, redirector, Web container, CTM, EJB container, JNDI, JTA, and JCA/JDBC.

Functionality for monitoring the J2EE application execution time

Functionality for monitoring the request execution time for EJB and Web applications. If a failure is detected while the execution time is being monitored, the method timeout functionality and method cancellation functionality are activated.

Garbage collection control functionality

Functionality used for setting the execution of garbage collection in the waiting state, during the mutual exclusion of the resources. You can avoid the occurrence of the full garbage collection and the termination of processes for a longer period, during the mutual exclusion of the resources. You can use the garbage collection control functionality on the batch servers.

Gateway specification functionality

Functionality used to send gateway information to a Web container and to enable correct redirection of the gateway information to a welcome file and Form authentication window when a gateway, such as SSL accelerator or load balancer, is deployed between the client and the Web server.

General custom adapter

This adapter calls the service components of a system that has a protocol.

The user can develop an adapter corresponding to any protocol, by using the custom adapter development framework provided by the system platform.

Global CORBA Naming Service

A Naming Service for common management of the information about the business-processing programs (Stateless Session Bean) registered in multiple J2EE servers of a CTM domain, when requests are scheduled with CTM.

Global session

The session failover functionality treats an HTTP session before a failure and an HTTP session inherited after a failure as one session. This is called a global session.

Global session ID

An ID that manages a global session. This ID is added to a global session to uniquely identify the global session information within a system.

Global session information

The information retained by a global session is the global session information. The global session information is managed with a database, when using the database session failover functionality. When using EADs session failover functionality, the global session information is managed by the EADs server and when using the memory session failover functionality, it is managed by the SFO server.

Global transaction

Transaction managed by the transaction manager provided by the J2EE server. A two-phase commit protocol can be used, and therefore, you can involve multiple resources in a transaction.

Grammar object

A grammar object is an internal object that can be acquired when parsing a schema document for displaying schema grammar definitions.

Group

A unit for grouping the messages on the communication layer. The messages are managed for each group. QoS, such as order guarantee, is provided in the group scope.

HA monitor

A type of cluster software.

A program that monitors and switches the system when a failure occurs. By integrating with HA monitor, you can improve the reliability and operation rate of a system built using Application Server. A system can be integrated with HA monitor when using AIX, HP-UX, or Linux.

By integrating with the HA monitor you can implement a 1-to-1 node switching system, mutual node switching system, or node switching system for the host management model.

Handler

A handler object of the Java logging API that is used to output the log of an application received from a logger to a file and console. You can use a handler object with the J2EE applications, batch applications, and EJB client applications.

CJMessageFileHandler, the file handler to output the log of HNTRLib2 format to a file, is provided in Application Server.

Handler framework

Functionality (framework) that intercepts and adds processing in the JAX-WS engine when a SOAP Message is sent and received. Multiple handlers can be added between the Web Service client and Web Service Implementation Class to extend the Web Service functionality.

HCSC

See *Service Coordinator*.

HCSC-Business Process

This implies the business process platform. For details, see *Business process base*.

HCSC-Business Process (business process infrastructure)

A function that calls appropriate service components according to the business process definitions.

HCSC component

A collective name for the service adapters, DB adapters, business processes, and user-defined receptions that are created in the development environment.

HCSC component validation

A function for verifying whether or not the adapters, business processes, and data transformation definitions that have been defined are valid. It enables the user to determine at any time whether or not there are required definitions, and whether or not the defined relationship is correct.

HCSC-Data Transform

This implies the *data transformation platform*. For details, see *Data transformation base*.

HCSC-DB Adapter (DB adapter)

A function that calls appropriate service components (database operations) according to the DB adapter definitions.

HCSC-Manager

An operations management function used in the operating environment of a cosminexus Service Platform.

It provides the HCSC server setup function, HCSC component deployment function, and process instance execution-log management function.

HCSC-Manager command definition file

A file (csccmd.properties) used to define the default values of items required in commands that are used in the operating environment. It is created as a J2SE property-format file.

By defining default values in this file, the user can omit the corresponding options during command entry.

HCSC-Manager definition file

A file (cscmng.properties) used to define the information required for customizing the operating environment. It is created as a J2SE property-format file.

The information that can be defined in this file includes the repository import target directory, the size of log files, and the number of log files.

HCSC-Messaging

This implies the *messaging platform*. For details, see *Messaging base*.

HCSCTE

A development support function used in a cosminexus Service Platform. The functions provided by HCSCTE are used as Eclipse plug-ins.

They include functions for creating and defining adapters and business processes.

HCSC server

A server facility that manages the execution of service components in the execution environment. It includes functions required for the execution environment, such as the messaging infrastructure, the business process infrastructure, and the data transformation infrastructure. It also includes the request reception function or all types of adapter functions such as the standard reception and user definition reception.

HCSC server configuration definition file

A file (cscsvsetup.xml) that defines the configuration of an HCSC server. It is created as an XML file.

The information that can be defined includes an HCSC server, a cluster, and Cosminexus Manager.

HCSC server runtime definition file

A file (cscsvconfig.properties) that defines the runtime information required at HCSC server startup. It is created as a J2SE property-format file.

The information that can be defined includes whether or not execution logs are to be collected, the levels of trace files, and their output destination.

HCSC server setup definition file

A file (cscsvsetup.properties) that defines the information required for setting up an HCSC server. It is created as a J2SE property-format file.

The cluster names to be set up and the standard reception types that can be used in an HCSC server can be defined.

Host unit management model

A system configuration model where Management Server is deployed on the same host as that of Application Server when you configure the system with the Smart Composer functionality or the management portal.

HTTP

HTTP is an abbreviation for *Hypertext Transfer Protocol*. For details, see *HyperText Transfer Protocol*.

HTTP adapter

This is the adapter for calling Web Services (RESTful Web Services) published in the REST style. You can also use this adapter to send and receive files between the service platform and service components.

By linking with the HTTP reception, you can forward the files sent from the client to the Web server.

HTTP reception

This is the function (interface) for receiving an execution request from the HTTP client.

You can use the HTTP reception to send and receive the text data or multimedia data between the Web browser and service components without passing through the Web front system.

HTTP session

The `HttpSession` object acquired using the `getSession` method of the `javax.servlet.http.HttpServletRequest` class. Shows the entire HTTP session, including the objects referenced as members by the HTTP session, in addition to the "Objects stored in an HTTP session".

HTTP response compression function

A function to compress an HTTP response for an HTTP request to a servlet, JSP, and static contents, into the gzip format with a servlet filter. You can use this function to reduce the communication time between a Web container and a Web client (such as a Web browser) and consequently reduce the response time of a system.

HyperText Transfer Protocol

This is the communication protocol for sending and receiving the HTML document between the Web server and Web client, by using Internet.

Hyper-V

This is one of the types of hyperviser. This is the name of the Microsoft provided function for implementing the server virtualization. By using this, you can attain the high performance, availability and reliability,with the 64 bit virtualization technique.

IANA

This is an abbreviation of *Internet Assigned Numbers Authority*. For details, see *Internet Assigned Numbers Authority*.

IDE

This is an abbreviation of *Integrated Development Environment*. For details, see *Integrated Development Environment*.

Inbound

One of the communication models between a J2EE server and resource adapter as stipulated in the Connector 1.5 specifications. Inbound is a communication model used for accessing the J2EE server from resource adapters.

Instance document

A XML document that is a target for schema verification.

In-process

A method for invoking a process. When in-process is used, a high-performance system can be achieved because the started process is optimized to be executed within the J2EE server process. Naming services, transaction services, and HTTP server function can be invoked by in-process. In this case, the CORBA Naming Service, in-process OTS, and the in-process HTTP server are started automatically when the J2EE server is invoked.

In-process HTTP server

A Web server function that is operated by the in-process on J2EE servers. The in-process HTTP server is provided as a part of the Web container functionality.

Integrated console

A window that displays information such as error messages for events occurring in a system when integrated with JP1/IM.

Integrated Development Environment

This is the integrated development environment in which you can execute the Java program or create a WAR file by using GUI. It supports Eclipse on Application Server and ESB & BSM platform.

Internet Assigned Numbers Authority

This is the organization for registering the internet names or numbers.

Integrated scope

A window that displays a monitoring tree matching the objectives of the administrator as a system-monitoring screen when integrated with JP1/IM.

Integrated naming scheduler server

A host that deploys global the CORBA Naming Service without deploying a J2EE server, when you use request scheduling as per CTM. You can create a replica of an integrated naming scheduler server and can improve the availability. Moreover, the CTM daemon is also required on the integrated naming scheduler server.

Integrated naming scheduler server model

A system configuration model that creates the host that deploys the Global CORBA Naming Service without deploying J2EE server when you use request scheduling as per CTM.

Integrated user management

Mechanism for integrated management of the user who logs on to the system that has been developed in the application server. You can use integrated user management framework that implements user authentication.

Integrated user management configuration file

This is the setting file (ua.conf) for using the JAAS supported user management, and single sign on function.

Integrated user management framework

A framework that realizes user authentication using integrated user management. APIs are provided according to Java standard specifications (JAAS).

J2EE application

An application configured by JSPs, servlets, and Enterprise Beans. The J2EE application format handled by Application Server includes the applications of the archive format that are packaged in the EAR file format, and the applications in the exploded archive format that are not archived. The applications packaged in the EAR file format are configured from multiple EJB-JAR files, multiple WAR files, and one DD.

J2EE container

A server platform for running J2EE applications. It is configured with a Web container and EJB container that provide the various APIs to the J2EE components.

J2EE component

A user application program such as a servlet, JSP, or Enterprise Bean.

J2EE resource

A resource that can be used by a server. For a J2EE server, you can use resources such as a database, OpenTP1, and SMTP server. You can also use a database for a batch server.

J2EE resource adapter

Resource adapter provided as a shared stand-alone module for a J2EE server.

J2EE server

An environment used to create and execute J2EE containers, which are the server infrastructure for executing applications. The Cosminexus service platform uses Cosminexus Component Container as a J2EE server.

J2EE server mode

The J2EE server mode is one of the operation modes of the J2EE server. In this mode, you can use J2EE-related APIs from an application running in the J2EE container. The J2EE server mode includes a 1.4 mode and a basic mode that is the operation mode for compatibility.

J2EE service

As a function of a J2EE container. The APIs of JNDI, JDBC, JTA, RMI-IIOP, and JavaMail are provided in J2EE components.

JAAS

This is an abbreviation of *Java Authentication and Authorization Service*. For details, see *Java Authentication and Authorization Service*.

jaas.conf

This implies the configuration file of JAAS. For details, see *JAAS configuration file*.

JAAS configuration file

This is a settings file for JAAS authentication, which is required for using the functions of the user authentication library and single sign-on library.

JAAS login module

This is a LoginModule interface mounted class of JAAS API, which is used for user authentication of the JAAS compliant integrated user management.

JAR file

This is the abbreviation of *Java ARchive file*. For details, see *Java ARchive file*.

Java

This is an object oriented programming language having high portability and reliability. As a software coding language, this is used in a wide range of application areas such as backend processes at the server side.

Java API for XML Processing

This is standard XML API for Java language. The reference implementation itself of DOM, SAX, and XSLT might be called *JAXP*.

JAXP includes APIs that generate a DOM parser and a SAX parser. JAXP can generate cross-platform programs because JAXP is independent of the implementation and coding of a parser. Also, JAXP can use an XSLT transformer from Java programs because JAXP includes the API of the XSLT transformer.

Java ARchive file

This is the archive for consolidating the class files or data files required for executing a Java program to one file.

Java Authentication and Authorization Service

These are the specifications of the Java standard user authentication. Application Server and BPM & ESB platform provides the JAAS compliant user management function, with the integrated user management framework of Cosminexus Manager. The user authentication by API of JAAS is called the *JAAS authentication*.

JavaBeans

A method used to build an application by embedding programs that are developed as components using the Java language.

JavaBeans

A method used to build an application by embedding programs that are developed as components using the Java language.

Java Database Connectivity

This is an API for accessing, and operating data in the database, with Java programs.

Java Naming and Directory Interface

This is the standard extended functionality for the Java platform and provides an interface (integrated to multiple naming and directory services within an enterprise) to the application corresponding to the Java technology.

In Application Server, the CORBA naming service of Cosminexus TPBroker is used to implement the JNDI functionality.

JavaServer Pages

A template file of the transformation function invoked from the custom function. Codes the configuration of the packages, classes, and methods (including arguments).

Java Transaction API

An API that provides a transaction processing service in a J2EE architecture. JTA defines the APIs required among various components participating in a transaction, when the transaction manager executes a transaction processing.

Java template file

A template file of the transformation function invoked from the custom function. Codes the configuration of the packages, classes, and methods (including arguments).

Java logging API

Class and interface of core logging functionality provided in J2SE. With Application Server, functionality is provided to output the user log in the HNTRLib2 format by using the Java logging API.

JAX-RS engine

This is the engine that serves as a communication platform of RESTful Web Services (Web resource), complying with the JAX-RS specifications. It is allocated on the Web resource side, receives the messages received from the client through HTTP protocol and executes the process. The JAX-RS engine performs the required injection based on the annotation included in the resource file, at the time of method calling.

JAX-RS function

This is the JAX-RS specification compliant function that executes RESTful Web Services (Web resource). This function provides the communication base for executing Web resource.

JAX-WS API

The API provided in the JAX-WS specifications. Used for adding functionality such as Handler framework.

JAX-WS engine

An engine that forms the Communication Infrastructure of a Web Service complying with the JAX-WS specifications. The JAX-WS engine is deployed on the Web Service client and the Web Service and plays the role of marshalling/un-marshalling the sent and received SOAP Messages.

JAX-WS functionality

The functionality for developing and executing the SOAP Web Services compliant with the JAX-WS specifications. It provides the command line interface for Web Service development and the Communication Infrastructure for the SOAP Web Services execution.

JAXP

This is an abbreviation of *Java API for XML Processing*. For details, see *Java API for XML Processing*.

JAXR

JAXR is the Java API specification created by Sun Microsystems, Inc., under the organization of Java Community Process that defines the standard specifications of Java technology. The latest specifications of JAXR1.0 are incorporated as configuration elements of J2EE1.4.

JCA container

A JCA container provides the system level plug-in functionality for connecting the J2EE server or batch server with EIS.

JDBC

This is an abbreviation of *Java Database Connectivity*. For details, see *Java Database Connectivity*.

JDBC connection

A connection for accessing the database from an application implemented on Java. Generated using the standard interface published by Sun Microsystems, Inc.

Cosminexus RM uses the JDBC connection through the DB Connector to access a database.

JMS client

One of the messaging system components that use the JMS interfaces. A program coded in the Java language and used for sending and receiving messages.

The application is the JMS client for using Cosminexus RM is used.

JMS interface

An API for sending and receiving messages through a queue, between the applications implemented on Java. The specifications are defined according to the Java Message Service Specification.

The JMS interface provided by Cosminexus RM is based on the PTP (point-to-point) model of JMS Version 1.0.2b.

JMS provider

One of the messaging system components that use the JMS interfaces. A product that implements the management functionality, the control functionality, and the JMS interfaces required for the messaging system.

Cosminexus RM is the JMS provider for using Cosminexus RM.

JNDI

This is an abbreviation of *Java Naming and Directory Interface*. For details, see *Java Naming and Directory Interface*.

JNDI name space

Name space managing the name information accessed by JNDI. In an Application Server, the EJB object reference is bound with the name of the JNDI name space when J2EE application is deployed.

You can add an optional name in the name of JNDI name space used in the application server. By doing this, the Enterprise Beans and J2EE resources can be look up with the optional name.

JNDI Naming Service

A standard extension functions for the Java platform that provides interfaces for the Naming Service and directory service in the application.

In Cosminexus RM, the JNDI Naming Service is used for generating the queue objects in the application.

Job

Each operation in the system operations is handled using JP1/AJS2.

Job-net

Links the execution sequence of the jobs that are handled using JP1/AJS.

JP1

An integrated system operations management software provided by Hitachi.

JP1/AJS

Program suite used for automating the routine or periodic operations from among the daily business operations, thereby reducing the cost incurred for system operations, and implementing reliable operations using fewer people. Configured from the products JP1/AJS - Manager, JP1/AJS - Agent, and JP1/AJS - View.

JP1/AJS3

A group of programs used for automating routine or periodic operations among daily business operations, reducing the cost incurred for system operations and implementing reliable operations using fewer people. In a virtual system, JP1/AJS3 - Manager, JP1/AJS3 - Agent, and JP1/AJS3 - View are used and integrated.

JP1/Audit Management - Manager

This is the program product for batch management of the audit log output by the products within the system, by automatically collecting the audit log. You can link with JP1/Base to automatically collect the audit log output by Cosminexus Application Server, Operation management server, and open middleware products to be linked, with the Audit log management server. Also, you can perform source management of the collected audit log, with the database of Audit log management server.

JP1/Base

This is the program, which provides the base functions of JP1 product, such as JP1/IM, JP1/AJS. You can manage the events of JP1 users, and control the service startup within the system in which JP1 products are used.

JP1/Cm2/NC - Manager

This is the program product for setting and controlling the load balancing function.

JP1/Cm2/NC - Manager for Cosminexus

This is the program product for controlling the load balancing function by linking with Application Server. Management Server can automatically control the load balancing function.

JP1/IM

A group of key programs for integrating and managing an entire corporate information system. JP1/IM is configured from three products such as JP1/IM-Manager, JP1/IM-View, and, JP1/IM-CM.

In the virtual system, you can link by using JP1/IM - Manager and JP1/IM - View.

JP1/IM - CM

Program used for consolidating and managing various resources configuring a business system (such as servers, storage) or the information related to the system configuration. Since the resources to be managed can be grouped randomly, the system monitoring (performance, failure) and operations (distribution of settings, changing the configuration) can be managed and operated in groups.

JP1/ITRM

This program is allocated from current acquisition of IT resources in various system environments and is used for primary management of the IT resource operation cycle till volume checking or optimization.

JP1/NETM/Audit

Program product used for automatic collection and batch management of the audit log output by products in the system. You can integrate the system with JP1/Base, and automatically collect the audit log output by Cosminexus Application Server, Management Server, and the integrated Hitachi open middleware products on the audit log management server. Also, you can uniformly manage the collected audit log using the database of the audit log management server.

JP1/PFM

Program suite that monitors the operating state of the databases and applications configuring enterprise system. In a virtual system, JP1/PFM - Manager, JP1/PFM - Web Console, and JP1/PFM - Agent for Cosminexus are used and integrated.

JP1/SC/DPM

Software used for executing the deployment operations. Enables the back up of the disk images and the batch installation of the OS and applications.

JP1 account manager

This is the user, who manages the account by operating JP1/Base, which is the authentication user. With the virtual system, you can manage virtual server manager administrator account and system configuration user account.

JP1 event

Information about an event that is sent to JP1/Base when the event occurs in the system.

JP1 job operation management server

Host that deploys JP1/AJS2 - Manager, when automatic system operations are performed by integrating with JP1/AJS2.

JP1 Integrated Management server

Host that deploys JP1/IM - Manager for the centralized monitoring of a system by integrating with JP1/IM.

JPA

A specification provided for mapping between Java objects and relational database (O/R mapping), with the aim of simplifying the design and coding of database-related processing. In Developer, you can develop applications using JPA, with the use of MyEclipse.

JPA Provider

JPA implementation that provides the functionality for mapping the Java objects and database and the API functions encapsulating database processing. With the Application Server, Cosminexus JPA provider is provided as a JPA provider.

JP1 permission level

Defines in JP1/Base with the name that indicates the operations (with permissions) that the JP1 user can perform for the management target (resources). With the virtual system, you can set the names for virtual server manager administrator account and system configuration user account.

JPQL

Query language defined in the JPA specifications. The language does not depend on the database and operates on the entity class. With the Application Server, database operations can be performed using JPQL.

JP1 resource group

This is the group in which management targets (resources) are divided and is defined in JP1/Base. With the virtual system, you can set the management unit name for controlling the access of a JP1 user, for the management unit.

JSF

A framework for developing the user interface of the Web application. In addition to the functionality of managing Web applications with a servlet that acts as a controller of MVC (Model View Controller) architecture, the event processing and component rendering elements, such as those in the user interface component Swing, are handled as components, and the user interface can be built by arranging these components in a form.

JSP

This is an abbreviation of *JavaServer Pages*. For details, see *JavaServer Pages*.

JSP EL

Formula language defined in JSP 2.0. You can describe the access to attributes of JavaBeans in JSP file or tag file. API is also provided.

JSP pre-compile

A function to compile the un-deployed JSP files and tag files included in a Web application. By compiling these files before their deployment, you can reduce the response time of the first request for a Web application.

You can also generate Java source files, and specify files that are not to be compiled, when you execute the JSP pre-compile function.

JSP tag library

Library used for creating JSP tag and is described in JSP specifications. The application server provides the JSP tag library for integrated user management for authenticating the user by using the Cosminexus standard login module.

JSP debug

A functionality for creating a class file in which the SMAP information that describes the mapping between the rows of the JSP file and java file is embedded during JSP compilation. This enables the execution of the functionality of the debug tool, such as setting up a breakpoint, in the JSP file.

In Developer, you can use MyEclipse as the debug tool.

JTA

This is an abbreviation of *Java Transaction API*. For details, see *Java Transaction API*.

library JAR

A common library packaged in JAR file format that can be used from all J2EE components in J2EE applications.

light transaction

A function that enables local transactions to be used in the optimal status.

Load balancer

A device that receives unique request from Web browser, transmits the requests to multiple servers having the same functionality and distributes the load balance of each server.

Local interface

Interface to invoke an Enterprise Bean by invoking the Java method. There is no communication.

Includes the local home interface and local business interface

Local transaction

Transaction managed by a resource manager of the connection destination. You can include only one resource in the transaction.

Logger

The logger object of Java logging API that acquires the log of J2EE application, batch application, or EJB client application.

Login module

A mechanism conforming to the Java standard specifications (JAAS) that is used for user authentication in the integrated user management.

Logical connection

A connection operated directly by J2EE components, such as a servlet and Enterprise Beans.

The logical connection used in the case of a resource adapter is `javax.resource.cci.Connection`, and in the case of JDBC is `java.sql.Connection`.

Logical naming service

A multiple CORBA Naming Service where JNDI is considered as a target of the round robin search.

Logical server

A server or cluster that is managed by Management Server. The server includes Web servers and J2EE servers. A cluster is a set of multiple servers that provide some common functions.

Logical user server

A logical server for which any service or process is defined as the management target of Management Server. It can be started and terminated and its status monitored by the Management Server.

Mail adapter

You can use this adapter to call the mail server that supports SMTP protocol as a service component on the Cosminexus service platform.

Management action

A process that is automatically executed when a Management event occurs. The type of process to be executed as a Management action is defined in advance in the Management Server.

Management agent

An agent program to monitor the operational status of logical servers running on the host and to collect the statistics.

Management command

The management command indicates `mngsvrutil`. For details, see *mngsvrutil*.

Management domain

A set of logical servers managed by the Management Server. The same management policy is applied to all.

Management event

An event that is issued in a Management Server in response to the phenomenon occurred in the logical server in the operation management domain when operating with a Management Server. This event can be issued when the message is displayed in the logical server. For example, it can be issued when the threshold value set during resource monitoring is exceeded.

If a Management action is defined for a Management event, it is possible to automatically execute the action in response to the occurred phenomenon.

management policy

Operations management-related policy, such as operations, monitoring, and security.

Management portal

A GUI for operating the Management Server. The management portal is displayed in the Web browser.

MBean

This is an abbreviation of *Managed Bean*. For details, see *Managed Bean*.

Management Server

This is a server program that configures the management domain. One server is allocated for one management domain.

Management Server executes the management of the entire management domain by performing the output of the instructions to the Administration Agent instances that are allocated to each host within the management domain.

management server model

A system configuration model employed when the Smart Composer functionality or management portal is used to implement a system in which the Management Server is placed on a separate operations management host from an Application Server.

Marshal

The task of saving a Java object in an XML document using a JAXB API is called *marshalling*. Marshal refers to serialization of the Java object in an XML document.

MBean (Managed Bean)

A Java object that represents a managed resource, and that is implemented according to the JMX specifications. It functions as a resource interface for the manager and provides the manager with resource-related information and operations.

Member resource adapter

Resource adapter connected to the database node when the connection pool is configured in a cluster. Member resource adapter is always accessed through the root resource adapter.

Memory cache

A schema cache that is saved in the memory.

MDB

This is an abbreviation of *Message-driven Bean*. For details, see *Message-driven Bean*.

Message

A unit of data sent and received between applications that is implemented using the JMS interface.

The messages used in the Cosminexus RM applications are implemented using the Message interfaces and its sub-interfaces.

Messaging

Messaging means sending and receiving any data between client and server as a message. In the SOAP applications of the Messaging-Type, use the SOAP message that is XML format data while sending and receiving data between client and server.

Messaging base

This function calls an appropriate service component in accordance with the definition of a service adapter.

Message-driven Bean

This is Message driven Bean for linking with JMS. EJB container starts Bean in receiving a JMS message from the Destination of JMS.

With Cosminexus RM, you can execute the methods mounted in Message-driven Bean, when a message is registered in the reception queue.

Message Queue adapter

An adapter used for sending and receiving messages from the execution environment of Cosminexus Service Platform to the existing message queue (IBM WebSphere MQ system).

Message Queue registration

This indicates that the application stores a message in a queue.

Message Queue extraction

This indicates that the application extract a message from a queue.

Message Queue delivery

This indicates the storing of the received message to the queue.

Message receipt

This indicates receiving of the message by transfer between servers.

Message transfer

This indicates sending and receiving of messages between multiple systems.

Method cancellation function

One of the functionality used to monitor the execution time of the J2EE application. This function executes cancellation of a method after a timeout is notified by the method timeout function. The transaction is also terminated forcibly if you access a database during cancellation.

Method timeout function

One of the functionality used to monitor the execution time of the J2EE application. Among the requests present in the monitoring base, the method processing that does not terminate within a fixed time period is notified to the user as a timeout.

method trace

Information that is required when a failure occurs and it is difficult to determine its cause.

MIME header

Header that defines the data format that can be handled in an E-mail or message. When sending and receiving a SOAP message with file attachment, this header defines the correlation between SOAP messages and file attachments within the MIME header.

mngsvrutil

This is CUI for operating Management Server.

Model/View/Controller

The term MVC consists of the first letters of the words Model, View, and Controller. Model manages the logic in a system. View manages the display and I/O, while Controller controls View and Model.

Monitoring tree

Tree displayed in the integrated scope when a system is integrated with JP1/IM. A system built using the application server provides the functionality to create two types of trees as the monitoring trees; the Cosminexus business-oriented tree and the server-oriented tree.

Mutual node switching system

One of the systems that can be implemented using the cluster software. In this system, the executing nodes and standby nodes on the Application Servers are arranged in a 1-to-1 ratio. The mutual node switching system has a standby system by operating the respective Application servers as the executing node.

Mutual standby

A configuration in which the executing nodes and standby nodes on the Application Servers are arranged in a 1-to-1 ratio when the operation is performed using the cluster software.

The mutual standby has a standby system by operating the respective Application Servers as the executing node.

MVC

This is an abbreviation of *Model/View/Controller*. For details, see *Model/View/Controller*.

N-to-1 recovery system

System where one recovery server is deployed as a standby node for an executing node Application Server in the cluster configuration. The recovery server concludes the transactions for the failed executing node Application Server.

Namespace

Identifier defined to indicate different meanings of elements and attributes with the same name in an XML document.

Namespaces in XML

A modifier that is used to avoid name collisions between elements or attributes when more than one language defined in XML is handled in a single document.

Naming Service

A service that names objects and manages their storage locations so that an object can be accessed by its name even when its storage location is unknown. Application Servers use the CORBA Naming Service.

The Management Server enables a Naming Service to be handled as a logical server.

Naming service switching

Switching of CORBA naming service that is connected to JNDI when registration, deletion, and search of the EJB home objects are performed through JNDI.

Node

A node is a general term for all the systems that includes programs to be executed or the communication devices besides the hardware required for business processes. There are two types of nodes: executing nodes and standby nodes.

Node failure

Indicates the failures occur in nodes, excluding servers. This includes the node hardware failure and HA monitoring failure. Note that a failure that occurs in a server is called the *server failure*.

Node switching

You can switch the system when error occurs or when you want to perform system maintenance. When you perform node switching, the standby node inherits the business processing, and becomes the executing node. On the other hand, the executing node becomes the standby node that is prepared for the error occurrences. Hereafter, for the node switching, you can execute the business operations normally on any of the nodes by switching the executing and standby nodes.

Node switching system for host unit management model

One of the systems that can be built using cluster software. A system where multiple executing node Application Servers and one standby node Application Server is allocated, and a Management Server and Management Agent is allocated respectively for each executing node Application Server and standby node Application Server.

Normal mode

Operation mode of a J2EE application. Normal mode is a mode for running a J2EE application that performs actual operations.

Normal termination of J2EE applications

Method to safely terminate J2EE applications by sequentially implementing the termination processing of Web application or EJB.

Number of dedicated threads

Number of threads that can ensure the execution of requests that are controlled by the number of concurrently executed threads. You can specify the number of dedicated threads for each Web application deployed on the Web container or for each business processing logic in a Web application.

Object Access adapter

An adapter used to call a service component in an existing TPBroker system (Object Wrapper system) from the execution environment of Cosminexus service platform.

Object Transaction Service

This is the service required for using the distributed transaction by TPBroker OTS, in 1.4 mode.

operating environment

The operating environment reads the EAR file created in the development environment from the repository, and deploys the file to the execution environment. The operating environment also sets up the HCSC servers that will be used in the execution environment. After the applications begin running in the execution environment, the startup, termination, and status of the system are monitored.

Objects used for managing HTTP sessions

Object that is used in a Web container and manages an HTTP session.

Object stored in an HTTP session

A user object stored in an HTTP session by the `setAttribute` method of the `javax.servlet.http.HttpSession` class.

Also called an *HTTP session attribute*.

Objects related to HTTP sessions

A collective term for "Objects used for managing HTTP sessions" and "HTTP session".

Operation mode

An operation mode of a server. The operation mode consists of the J2EE server mode and the servlet engine mode. Furthermore, there is a 1.4 mode and a basic mode in the J2EE server mode.

However, the basic mode and the servlet engine mode are the operation modes for compatibility with the previous versions.

Oracle RAC

A cluster functionality of Oracle database. With an application server, you can connect to a clustered Oracle database with the Oracle RAC functionality.

ORB gateway

A gateway functionality that can directly call the EJB application operated in J2EE server from CORBA client. ORB gateway is provided by CTM regulator process.

OTM gateway

Gateway functionality that can directly invoke an EJB application that is operating in J2EE server, from the TPBroker OTM client. It is provided by OTM gateway process.

OTS

This is an abbreviation of *Object Transaction Service*. For details, see *Object Transaction Service*.

Outbound

One of the communication models between a J2EE server and resource adapter as stipulated in the Connector 1.5 specifications. Outbound is a communication model for accessing the resource adapter from the J2EE server.

Out-process

A method to activate a process. If out-process is activated, it is executed external to the J2EE server processes. The user must activate the CORBA Naming Service when using a Naming Service in out-process.

packaging

Assembling an EAR file such that a HCSC component created in the development environment can be deployed in the execution environment.

Percent encoding

Percent encoding includes a percent sign with two character hexadecimals after that, and expresses the characters, such as the Japanese characters that cannot be used in a URL.

Performance analysis trace

A trace information for analyzing the performance of the system configured on Application Server.

Performance analysis trace file

A text file that edits and outputs the trace information in the CSV format, for analyzing the performance of the system configured on Application Server.

Physical connection

Connection of the connection destination resource. Usually a J2EE container operates the physical connection. The physical connection used in a resource adapter is `javax.resource.spi.ManagedConnection`, and that used in JDBC is `javax.sql.XAConnection`.

physical tier

A set of server machines to which the same deployment (process arrangement, deployment of J2EE applications and resource adapters, and definitions) is applied.

This concept is used when a system is implemented using the Smart Composer functionality.

Physical machine

A physical computer.

Planned node switching

One of the methods of node switching. When operator maintains nodes, this method switches nodes in planned manner.

Planned termination

Planned termination is a method to stop running server processes or server threads after execution completes.

Plain Old Java Object or Plain Ordinary Java Object

This is the terminology that indicates the Java object, created in the specification range of Java.

POJO

This is an abbreviation of *Plain Old Java Object or Plain Ordinary Java Object*. For details, see *Plain Old Java Object or Plain Ordinary Java Object*.

PRF daemon

An IO process that outputs a PRF trace from buffer to file.

In a Management Server, PRF daemon can be treated as a logical server. When supported by a Management Server, PRF daemon is called as performance tracer.

PRF trace

Trace information that has been output by each function of application servers when requests are processed. The trace information is output to a buffer; when a specified amount of trace information has accumulated, it is then output to a PRF trace file.

PRF trace output library

Library embedded in each functionality of the application server system. It provides a functionality to output the PRF trace to a buffer for each function layer.

PRF trace collection level

Level to collect the PRF trace. The levels are 'Standard level', 'Advance level', and 'Maintenance level'. If 'Advance level' or 'Maintenance level' is specified, there are more trace collection points as compared to the 'Standard level'.

PRF trace file

A binary format file to which PRF trace is output. This is a base file for converting to trace based performance analysis file (CSV format).

Private key

Private key is used in public key cryptography. The private key is not released to the communication partner and is saved only by the sender.

process instance

An application entity that is executed by a business process. It corresponds to a workflow item.

Public key

Public key is used by public key cryptography. The public key releases information to the communication partner.

Queue

A storage destination of the messages sent and received by applications.

In Cosminexus RM, a queue is created on the database.

Queue creation file

A file used for defining the queues used in a non-persistent resource adapter. In a persistent resource adapter, a queue is created using commands, but in a non-persistent resource adapter, by defining the queue attributes in the queue creation file beforehand, the queue information is enabled when the resource adapter starts.

Queue definition file

A file that associates the queue display name registered in the JNDI Naming Service and the actual queue name. If the full path of the queue definition file is specified in the QueueConfigFileName property, the defined queue display name is registered in the JNDI Naming Service when Cosminexus RM starts.

Queue definition statement

A statement used for defining the queue attribute and the queue name within the queue creation file. The queue information is restored when the resource adapter starts according to the contents coded in the queue definition statement.

Real parameter

This parameter is to be set in the definition file of Application Server. With Smart Composer functionality, you can set up the values as a batch, to the mutually connected multiple real parameters, by setting one abstract parameter.

RealServer

This implies the real server of the load balancing function. For details, see *Real server of load balancing function*.

Real server of load balancer

This is the object that indicates an actual server which is the classification destination of the requests received by VirtualServer, on the load balancer. One real server of the load balancer is created for each virtual server of the virtual system.

Reception process

An API that codes the process for receiving the service component invocation requests and for transmitting messages. A reception supporting any protocol can be developed by operating the user-created reception process on the custom reception framework.

Redirector

A plug-in component (library) registered in the Web server to enable to connect with a Web container. A redirector registered in the Web server enables requests to be distributed and processed in multiple Web containers using a URL pattern or round-robin method.

redeploy functionality

A functionality used for replacing J2EE applications. It is applied when the running J2EE application is to be replaced with a corrected version of the J2EE application, such as when a J2EE application is being tested.

Remote interface

Interface that invokes an Enterprise Bean with the help of RMI-IIOP communication based on the interface rules of Java RMI.

Includes the remote home interface and the remote business interface.

Remote Method Invocation over Internet Inter-ORB Protocol

This is API in which JavaRMI and JavaIDL is integrated. The EJB component and respective containers implement the calling between distributed objects by using Remote Method Invocation (RMI) technology.

Remote Procedure Call

This is the procedure for executing the process on a different machine in the network. SOAP application of RPC format provides the service in the format of specifying the argument in the client side and receiving the return value from the server side.

repository

A directory for storing defined HCSC components and system configuration definitions. The stored information is transferred between the development and operating environments using repository management functions (repository import/export functions).

Request reception

A common name for standard receptions and user-defined receptions. This is a functionality (interface) to receive execution requests from a service requester.

Request processing thread

Thread generated in the Web server for receiving and processing the requests from clients, such as a Web browser. In the case of the in-process HTTP server, the thread is generated in a Web container.

request trace

Information that determines which business process or service adapter is targeted by a request accepted by a standard reception, business process acceptance, or server-to-server transfer acceptance. The request trace also determines the request result.

Request scheduling

Functionality to properly distribute the servers that execute the requests from the client, on the basis of the load, and restrict the number of requests sent to the server.

Resource adapter

A function for connecting a J2EE server or batch server to EIS based on the J2EE Connector Architecture.

The Application Server system provides a DB Connector that is a resource adapter for connecting to the database and a DB Connector for Cosminexus RM.

You can also use the uCosminexus TP1 Connector that is a resource adapter to connect the OpenTP1 and SPP, the TP1/Message Queue - Access that is a resource adapter to connect to the TP1/Message Queue, and the Cosminexus RM that is a resource adapter to connect to the queues on the database.

Reload functionality

A functionality to switch the programs and files that constitute an application, without terminating the application that is running. You can execute the reload functionality for an application deployed in the exploded archive format.

Realm

Range that applies the same authentication policy in a system performing integrated user management.

Resource manager

A function to manage the resources. DBMS is a resource manager.

response message

A message used to return service component execution results. The two types of response messages are those that are sent from service components to the HCSC server, and those that are sent from the HCSC server to service requesters.

RMI-IIOP

This is an abbreviation of *Remote Method Invocation over Internet Inter-ORB Protocol*. For details, see *Remote Method Invocation over Internet Inter-ORB Protocol*.

Role

A unit used when controlling access to a context. Role is defined for each group. The roles required for accessing the contexts, wherein the access is to be controlled, are also defined. The access to the context becomes successful when the role possessed by the user accessing the context matches the role defined in the context.

Root application information

Information output in the Trace based Performance Analysis. This information is acquired by the first process in the series of processing executed in an event.

Root context

Root path in a WAR file wherein the context root is a null character (name is not specified in the context root).

Root resource adapter

Adapter to cluster the member resource adapters and receive the requests from a user program, when the connection pool is clustered.

Round-robin search

Search functionality to look up multiple EJB home objects with the same name existing on the CORBA naming service, based on the round robin policy.

Round-robin policy

A policy applied when multiple EJB home objects with the same name existing in the CORBA Naming Service are looked up.

The EJB home object reference can be acquired by round-robin search, and the load balancing of the requests sent to the J2EE server without being identified in the EJB client can be implemented.

RPC

This is an abbreviation of *Remote Procedure Call*. For details, see *Remote Procedure Call*.

Reverse proxy server

A Web server deployed at DMZ between an external network such as an internet and internal network where an Application Server is deployed and has a function to broadcast the requests from an external network to the internal network.

In the Application Server, you can use the Hitachi Web Server as a reverse proxy server. The reverse proxy module is embedded and operates in the reverse proxy server.

SAAJ

This is an abbreviation of *SOAP with Attachments API for Java*. For details, see *SOAP with Attachments API for Java*.

SAX

This is an abbreviation of *Simple API for XML*. For details, see *Simple API for XML*.

Scale in

A process of reducing the number of servers when reducing the size of the system.

Scale out

A process of increasing the number of servers with the aim of improving the processing performance of an entire system.

Schedule queue

Schedule queue is a queue that holds the requests, when performing the scheduling for the requests with CTM. The schedule queue is created for each J2EE application, and is managed by the CTM daemon.

Schedule policy

Policy for deciding the distribution standards, when performing the scheduling for the requests with CTM. You can select either a method to distribute the requests to the CTM having lesser load or a method to distribute the requests on priority to the CTM daemon that accepts the requests.

Schema cache

Grammar objects that are saved for reusing schema documents.

Depending on the specifications of a schema definition file, the grammar objects are saved either in the memory or on a disk.

schema compiler

A program used to generate a Java source code in the JavaBean format, when schema document is input and the JAXB mapping annotations are added. If the schema document includes user instructions given through the custom binding declaration, bind the schema elements to the Java source code based on the instructions.

Schema definition file

A file for specifying schema documents to be cached and the save locations of cache.

schema document

An XML document that defines a logical data structure within an XML document, and that is written according to the XML Schema specifications.

Schema generator

A program used to generate a schema document from the declaration part of the class and field, when a Java source file is input. If the Java class includes user instructions given through the JAXB mapping annotations, mapping is done to the schema components based on the instructions.

Secure Sockets Layer

This is one of the protocols for sending and receiving the encrypted information through the network. This is the topmost layer of TCP. This layer performs authentication, key exchange, encryption and message authentication by certification between the client and Web Server.

Security identity

Authentication information transmitted to a Web container and an EJB container, when the user authentication by the J2EE server is successful.

SecurityManager

A J2EE server's function that provides Web container runtime protection based on the security policy.

Server virtual technique

A technique that shows one server as if there were multiple servers. Using this technique, you can divide or integrate the resources of one physical server to show that multiple virtual servers are operating and you can show multiple physical servers as one server. Also, the OS and applications are allocated for each of these virtual servers.

Server-oriented tree

One of the monitoring trees displayed in the integrated scope when integrated with JP1/IM. This is a monitoring tree that mainly consists of the server machine (host).

Server management command

A group of commands used to set the applications and resources managed in the server.

Server start/stop hook function

A function to automatically invoke the initialization process and termination process for using a container extension library, when starting or stopping a J2EE server or Web server.

Server communication agent

An agent program invoked on the virtual server to set up and operate Application Server on the virtual server from the virtual server manager.

Service [SOA]

Unit of applications and data handled in the SOA concept. Service corresponds to business processes implemented in single or multiple J2EE applications. In SOA, services are combined for building a business system.

service adapter

The service adapter is a component for linking with various types of services, on the service platform. It provides adapters for connecting with various systems such as Web Services, databases, Mainframe systems, and online systems. You can also create a general custom adapter for calling the system that has optional protocols, as and when required.

service component

Application processing that can be executed from service adapters and database adapters. Web Services, SessionBean, MDB (WS-R), MDB (database queue), and database operations can be used as service components.

service component message

A message with a format unique to the service component in question. If the service component is database manipulation, the service component message is called a database adapter message.

Software as a Service

A software distribution form for distributing the functionality that a user requires, among the functionality of software, so as to enable the user to use the required functionality. This is also known as *Saas* (<u>S</u>oftware <u>a</u>s <u>a</u> <u>S</u>ervice).

Service requester

The service requester is a client application for calling the reception on the HCSC server.

Note that protocols or the message format to be used differs depending on the type of the reception.

service unit

A closed part of a system that provides application services consisting of J2EE and Web servers is called a service unit. The Smart Composer functionality enables the user to implement a system and perform batch start and termination for each service unit.

This concept is used when a system is implemented using the Smart Composer functionality.

Servlet

One of the methods to run Java on the server machine. A servlet integrates with a Web server, and not just sends HTML documents and image files to a Web Server, but executes an application, and provides a function to return the result as an HTML document.

Servlet engine mode

One of the server operation modes. The servlet engine is operated separately using only the Web container. The Enterprise Beans do not operate.

A servlet engine is a server equipped with the servlet execution functionality and JSP execution functionality.

- Note that the servlet engine mode is an operation mode for compatibility.

Servlet and JSP service method

`service` method of `javax.servlet.Servlet` and method invoked from the `service` method. The method invoked from the `service` method is the `do*` method (such as `doGet` and `doDelete`) for servlet, and the `_jspService` method for JSP.

Servlet filter

A function to filter requests to a servlet or JSP, and responses from a servlet or JSP. With the help of a servlet filter, you can wrap the requests before executing the servlet or JSP, or responses after executing the servlet or JSP, change the data, and acquire the trace for a resource.

The session failover filter, HTTP response compression filter, and client performance filter are provided as the servlet filters for implementing the Application Server functionality.

Session

Unit indicating the working series of a Web application. A session usually consists of a set of multiple requests from the Web client to the Web server.

Session Bean

Bean mainly for executing the business process. Session Bean can have only one client. If the client is terminated, the corresponding Session Bean also terminates. The Bean operates only between the sessions with the client and, therefore, it is temporary and non-persistent. Session Bean is classified into Stateful Session Bean and Stateless Session Bean.

Session failover functionality

Functionality used to manage the objects registered in an HTTP session object of a running J2EE application as the session information and to pass the managed session information to another J2EE server when an error occurs in the J2EE server. Even when an error occurs in J2EE server and the requests are transmitted to another J2EE server, the business continues as prior to the occurrence of an error.

There are three types of session failover functionality such as the database session failover functionality, EADs session failover functionality, and the memory session failover functionality. The SFO server manages the session information with the memory session failover functionality. The database manages the session information with the database session failover functionality. The EADs server manages the session information, with the EADs session failover functionality. The SFO server manages the session information with the memory session failover functionality.

Session failover server

A host that deploys and uses an SFO server without deploying a J2EE server when using the session failover functionality.

Setup Wizard

A function for configuring a system, using a wizard program that has an interactive format. The Setup Wizard sets only the minimum required items, and can configure a system that can execute J2EE applications. The Setup Wizard is functionality recommended to configure the execution environment with minimum required procedures. However, the Setup Wizard can be used to configure only one system on a single machine.

SFO server

A J2EE server that manages the global session information generated in a J2EE application of a J2EE server in a system. An SFO server is included in the system configuration when you use the memory session failover function.

SFO server application

A J2EE application running on the SFO server is called as *SFO server application*. The SFO server application is implemented in EJB. The SFO server application is provided by Application Server.

Simple API for XML

This is callback API for accessing the XML document contents by perceiving and notifying the configuration property within the XML document. In case of SAX, passes only the information related to locations being analyzed to the user program and does not create a tree as in case of DOM. Due to this, less memory space compared to DOM, is used.

Simple Web server

Functionality used to download a stub by using a Web container as an HTTP server. Unlike the in-process HTTP server, the simple Web server cannot be used for any purpose other than downloading stubs during actual operation.

When using the Management Server, this is referred to as the 'server for management'.

Single sign-on

Functionality to allow a user ID to seamlessly use multiple different systems by logging on once.

Single sign-on information repository

Executes user mapping of integrated user management framework and stores the authentication information and mapping information of each system for user authentication by single sign-on.

Single sign-on library

Library to perform user mapping on the basis of the information about the single sign-on information repository that stores the user mapping information, and then to implement single sign-on.

Skeleton

Static interface for the SOAP service. Deployed on the Web Service and hides the send and receive processing with the Web Service client.

Smart Agent

A process that provides a dynamic distribution directory service when you use a global transaction in the 1.4 mode, or when you use CTM.

When using Management Server, you can handle the Smart Agent as a logical server. CTM is managed by the Smart Agent.

Note that even for a global transaction, the Smart Agent is not required if you use the in-process transaction service.

Smart Composer functionality

The Smart Composer functionality supports the easy setup and operations of the systems running on Application Server for a general 3-tier model, when setting up systems using the Application Server. The system can be set up easily and quickly, since setting the entire system and deploying the J2EE application and resource adapter can be performed together.

The operation functionality such as starting the built system in a batch and changing the settings of the system in a batch are supported by this functionality.

snapshot log

A log file that displays the status of a J2EE server or batch server at a certain point.

The information necessary for troubleshooting is automatically output to the log file.

In the case of using Management Server, this log file is output at the following times:

- When an error occurs in the logical server
- When the collection of snapshot log by management commands or the management portal is requested

When Management Server is not used, a snapshot log is output when the command (snapshotlog) is executed. The information of the host who is executing this command can only be output by this command.

SOAP

This is an XML based protocol that brings a communication protocol such as the HTTP to the lower level and specifies the communication between systems through a network.

SOAP 1.1 mode

This mode indicates the standard specifications and communication platform of the Web Services to be used, on a service platform. Specify the SOAP 1.1 mode to use service components and service requesters for the SOAP 1.1 specifications. With the SOAP 1.1 mode, use the SOAP communication infrastructure provided by Cosminexus, as the communication infrastructure.

SOAP1.1/1.2 combined mode

This mode indicates the standard specifications and communication platform of the Web Services to be used, on the service platform. Specify the SOAP 1.1/1.2 combined mode to use service components and service requesters for the SOAP 1.1 specifications and the SOAP 1.2 specifications. With the SOAP1.1/1.2 combined mode, use the JAX-WS engine provided by Cosminexus, as the communication infrastructure.

SOAP with Attachments API for Java

This is one of the functions included in Java XML Pack and provides an API for dealing with SOAP messages having attachment.

SOAP attachment

A part containing multiple body information of SOAP messages. The Binary data might also be included in a SOAP attachment.

SOAP application

This is the application for Web Services developed by using the SOAP application development support function, which is the pre-existing function of Cosminexus Application Server.

SOAP application development support function

Functionality provided in Cosminexus to develop SOAP applications. With the SOAP application development support function, you can develop SOAP applications while using development support commands.

For details about the SOAP application development support function, see the manual *uCosminexus Application Server SOAP Application Development Guide*.

SOAP Body

Element of a SOAP Message. In the SOAP Body, describe the contents of the outgoing message.

SOAP client library

A library that mediates the processing implemented in the client and data transmission of the server. The SOAP client library receives the message from the processing implemented in the client, and invokes the SOAP service. After the processing is performed in the SOAP service, the returned message is analyzed, and returned to the invoking source as the process result.

SOAP Communication Infrastructure

Communication infrastructure of Web Services (SOAP applications) provided by Cosminexus. SOAP Communication Infrastructure corresponds with the SOAP 1.1 specifications.

SOAP engine

An engine that mediates sending and receiving between client and SOAP service. The SOAP engine analyzes the message sent from client and sends it to the SOAP service. After the processing is performed in the SOAP service, the returned message is sent to the client.

SOAP Envelope

Element of a SOAP Message. It is the most external element of the message and has child elements such as SOAP Header and SOAP Body.

SOAP Fault

This indicates SOAP messages used to code errors that occur during SOAP message processing. SOAP fault can be divided into messages created by a SOAP engine and messages created by a SOAP service developer.

SOAP Header

Element of a SOAP message. In the SOAP Header, specify the address of the message processing and specify whether message processing is required.

SOAP Messages

The messages used for sending and receiving data between systems by using the SOAP protocol. The SOAP Message consists of elements such as SOAP Envelope, SOAP Header, and SOAP Body.

SOAP Messages with attachments

A SOAP Message with any file attachment in addition to the SOAP Header and SOAP Body.

SOAP Service

A program (service) among the programs that build a SOAP application, deployed at the server machine to perform the processing requested from the client.

Source code generating function

Development support command functionality that automatically generates the source code required to develop SOAP applications from WSDL.

SQL operation definition file

A file that defines SQL statements to execute when a database adapter is used to access the database. It is the base file used to create an XML-format definition file for a database adapter.

SSL

This is an abbreviation of *Secure Sockets Layer*. For details, see *Secure Sockets Layer*.

Standard message(Standard message format)

This message (message format) is commonly used within a business process. When linking with a service having a different message format, you can link by changing the message format of a standard message according to the connection destination service, by using a data transformation and use the changed message format as a common message within a business process.

standard reception

This is a fixed interface that receives the service component execution requests from the service requester. It is provided as a function of the HCSC server. If a standard reception starts, the HCSC server can accept request messages from a service requester.

A requester service can accept synchronous acceptance (Web Services/SessionBean) and asynchronous acceptance (MDB (WS-R)/MDB (database queue)) requests.

standby node

Specifies the standby node that is prepared for errors (standby server).

Stateful Session Bean

A Bean that can maintain an internal status. Unlike a Stateless Session Bean, after this Bean is instantiated by generating from the home interface, the Stateful Session Bean cannot be created or destroyed each time a method is invoked. This Bean is appropriate in the case of continuous processing for each session.

Stateless Session Bean

A Bean that does not maintain an internal status. An EJB container independently manages the creation and destruction of an instance of this Bean. Even if you consecutively invoke methods of the same Bean, there is no guarantee that the methods will be of the same instance. As a result, in a Stateless Session Bean, you cannot perform continuous processing with multiple methods while maintaining the internal status. This Bean is appropriate when coding a process that can be finished with a single method.

Static contents

Contents that are always the same without any impact on the request contents among the files, such as HTML files and image files, used for responding to the requests from the client.

Static contents cache function

A function that caches the static contents in the memory. By caching those static contents into the memory that have been accessed once, you can reduce the frequency of accessing the file system when you access these static contents for the second time and thereafter, and improve the response speed.

Statistics collection functionality

Functionality to periodically collect the statistics of each functionality in the J2EE server and batch server, and output the same to a file. The output file is called a *statistics file*.

StAX

This is an abbreviation of *Streaming API for XML*. For details, see *Streaming API for XML*.

Streaming API for XML

This is event based API included in the JAXP specifications and is the standard API of Java defined in JSR173 of Java Community Process. In case of StAX, a handler need not be used when parsing the XL document and hence you can process the XML document, as per the procedure.

Stub

An interface that passes the request from the Web Service client to the JAX-WS engine, and returns the return value from the JAX-WS engine. A stub is deployed on the Web Service client and hides the send and receive processing with the Web Service.

structure activity

An activity that defines the structure of processing in a business process, such as branch, parallel, and loop iteration processing.

Struts

A framework for developing the user interface of a Web application. In addition to the functionality of managing Web applications with a servlet that acts as a controller of MVC (Model View Controller) architecture, the user interface components are handled as components, and the user interface can be built by arranging these components in a form.

system configuration definition

Definition information that includes setup information for HCSC servers and clusters and information about how HCSC components are deployed at HCSC servers and clusters.

The HCSC server and cluster setup information is updated in the operating environment. In the development environment, information is defined about how HCSC components are deployed at the HCSC server and cluster.

Test mode

An operation mode of an application. This is a mode for operating an application to be tested in a manner similar to the operations in the real environment.

Thread asynchronous parallel process

Thread asynchronous parallel process is a function that executes the thread asynchronous parallel process such as asynchronous timer process or asynchronous thread process in the Java EE environment. Cosminexus Application Server provides APIs based on the specifications of Timer and Work Manager for Application Servers laid down by CommonJ. You can use this for executing the asynchronous timer process in which TimerManager is used and the asynchronous thread process in which WorkManager is used.

Timer Service

Functionality provided by an EJB container. The EJB container calls back the timeout method in the specified time, lapsed time, or at intervals.

Timer Service

A function provided in an EJB container. The EJB container calls back the timeout method at the specified time, elapsed time, or interval.

Trace collection point

Outputs PRF trace in each function layer of the application server system. The number of trace collection points differs with the specification of the PRF trace collection level.

TLS

This is an abbreviation of *Transport Layer Security*. For details, see *Transport Layer Security*.

TP1/EE

An option product used along with TP1/Server Base. TP1/EE provides the functionality for sending and receiving messages between Cosminexus RM through the database. Communication between different components can be implemented between Cosminexus and OpenTP1 using TP1/EE.

TP1/RPC reception

Functionality (interface) that receives service component invocation requests from the service requester on an existing OpenTP1 system.

TP1 adapter

An adapter used to call a service component in an existing OpenTP1 system from the execution environment of Cosminexus Service Platform.

TP1 inbound adapter

A resource adapter used in the TP1 inbound integrated function. Complies with the Connector 1.5 specifications.

TP1 inbound integration functionality

The functionality for invoking a message-driven Bean in Application Server from SUP of OpenTP1 system via inbound using TP1 inbound adapters.

TP1 integrated system

A system configuration used to connect the service requester and service components existing in an OpenTP1 system.

Transmit

Transmitting and receiving messages between multiple systems.

Transmit destination queue

A queue to which a message is transmitted.

Transmit destination queue address

An address of the queue to which a message is transmitted. Indicated by the destination address and queue name.

Transformation API for XML

Ths is an API included in the JAXP specification and is the standard API of the XSLT transformer.

If you use TrAX, you can execute XML conversion process from the Java program. If you want to perform simple XML document conversion, you can execute only TrAX.

Transformation function

Java program invoked from the custom function.

Transformation function definition file

File that defines the configuration of the transformation function invoked from the custom function.

Transaction manager

Functionality to manage a transaction. Provided with the Application Server. In the case of global transactions, TPBroker OTS is used by the back-end of transaction manager.

Transaction Service

A service that manages a transaction when using a global transaction. Indicates all the services provided by TPBroker OTS. A transaction service is invoked by the in-process of J2EE server.

Transport Layer Security

This is one of the protocols for sending and receiving the encrypted information through the network. This protocol has been made better based on SSL.

TrAX

This is an abbreviation of *Transformation API for XML*. For details, see *Transformation API for XML*.

Translet

Java byte code that the compiler making up the XSLTC transformer generates in memory by parsing an XSLT stylesheet. A translet enables high-speed transformation of an XML document.

uCosminexus Batch Job Execution Server

Product that provides the batch job execution infrastructure for executing and operating the key system batch tasks in an open environment. Used along with JP1/AJS2.

uCosminexus Client

A product used to build an EJB client environment.

uCosminexus Service Architect

A product that allows the user to configure and run the development environment of a Cosminexus service platform.

uCosminexus Service Platform

A product that allows the user to configure and run the execution and operating environments of a Cosminexus service platform.

UDDI

UDDI defines the APIs used for publishing and searching for information related to SOAP Web Services and the UDDI registry that stores that information.

UML

This is an abbreviation of *Unified Modeling Language*. For details, see *Unified Modeling Language*.

Unified Modeling Language

This is a modeling language laid down by OMG. This provides a notation method to express the structure of application, based on the object trend in more structural and formal format, which is even more abstract than the source code. The notation method provided by UML is further divided in the use case diagram, class diagram, and sequence diagram depending on the contents of objective of the analysis.

Uniform Resource Identifier

This is the standard that has the method of specifying the position of data on the Web and method of specifying a certain data on the Web with a specific name, as the method of URL. It has the combination of URL and URN and you can use the same for expressing the internet resources, and identifying the name space in an XML document. URI is a generic concept and holds a position above URL. It is prescribed with RFC1630.

Uniform Resource Locator

This is the standard for specifying a position of a certain data on the Web.

URI

This is an abbreviation of *Uniform Resource Identifier*. For details, see *Uniform Resource Identifier*.

URL

This is an abbreviation of *Uniform Resource Locator*. For details, see *Uniform Resource Locator*.

usrconf.properties

For details, see *User property file*.

Unconcluded transaction

Transaction that is not concluded and is being processed.

Unmarshal

The task of reading an XML document and converting the XML document to a Java object using a JAXB API is called *unmarshalling*. An unmarshal is deserializing an XML document.

URI(Uniform Resource Identifier)

An acronym for Uniform Resource Identifier. URI is a standard for specifying the location of data on the Web like a URL, and for specifying specific data on the Web with a unique name.

URL(Uniform Resource Locator)

An acronym for Uniform Resource Locator. URL is a standard for specifying the location of specific data on the Web.

user log

A log that output by a J2EE application, batch application, or client application.

User definition reception

This is a function (interface) for receiving an execution request from a service requester. Unlike the standard reception included in the HCSC server, user can define any interface. When you start by deploying the defined user definition reception to the HCSC server, this function can receive the request messages from the service requester.

User definition reception random definition file

This is the file (cscurcptnconfig.properties) that defines the SOAP reception information. It is created as a file in the J2SE property format.

You can define the Max threads, the number of dedicated threads, and the size of the pending queues in this file.

usrconf.properties (User property files)

A file used to define the system properties of JavaVM used by a J2EE server, batch server, Web container server, or server management commands.

User property file

The user property file is named `usrconf.properties`. The user property file specifies the system properties of the J2EE server, Web container server, or JavaVM executing the server management commands. The storage location and the keys that can be specified differ for the user property files used by the J2EE server, Web container server, and server management commands.

User-specified name space functionality

Functionality for the user to give an optional name to the EJB home object reference names and to the J2EE resource names registered in the JNDI name space. By giving an optional name, lookup can be performed with a name that does not depend on the server name and application name, when the requests are distributed by CTM and the load is balanced using a load balancer.

User information

Information for login of a user to a system configured with Application Server.

User information repository

Repository that stores the information used for user authentication during logon, when performing integrated user management. You can use an LDAP directory server or a database (RDB).

User thread

Threads created explicitly by the user in a servlet and JSP.

User mapping

Correlation of the information of the same user saved in different user information repositories.

User-defined reception

A function (interface) for accepting execution requests from the service requesters. Unlike a standard reception included in an HCSC server, a user can define any interface using this function. If you deploy a defined user-defined reception in an HCSC server and start the server, the HCSC server can accept the request messages sent from a service requester.

User-defined exceptions

An exception implemented independently in the SOAP service processing by the SOAP application developers. Based on the requirement, this exception is thrown in the SOAP service processing.

Version setup functionality of Web applications

Functionality to execute Web applications compliant with the specified version without changing the version of Web applications defined in `web.xml`. You can use this functionality to move the applications created with earlier versions.

Virtual environment

An environment that uses the virtual machine implemented by the virtual software.

Virtual software

A collective term for products implementing the virtual machine using the software.

Virtual server (virtual software)

This is a logical machine that is implemented by the virtual software. You can perform the operation management of the application execution environment (Application Server) for the machine on which the OS or the product is installed, by using the virtual server manager.

Virtual server of load balancer

This is a virtual server for managing multiple real servers with the load balancer. The load balancer automatically distributes the traffic that accesses the virtual server from the client, to the real server.

Virtual software

A collective term for products implementing the virtual machine using the software.

Virtage

This is one of the hyperwiser types. It indicates the server logical classification setup. It provides the most suitable high performance and highly reliable server virtualization technique, in case of simple server integration and also in case of core system.

VirtualServer

This is an abbreviation of a *virtual server for the load balancing function*. For details, see *Virtual server of load balancing function*.

Virtual server manager

A process used for setting up the virtual system and managing virtual server operations.

W3C

This is an abbreviation of *World Wide Web Consortium*. For details, see *World Wide Web Consortium*.

WAR file

This is an abbreviation of *Web ARchive file*. For details, see *Web ARchive file*.

Web application

Application created for the client provided by the Web browser. It is the cluster of servlet programs, JSP file, HTML/XML documents.

Web Archive file

This is the file in which components of a Web application are compressed in a WAR format. It includes the HTML file, JSP file, Java class file, JAR file, and Web application scheduling identifier, required for executing the Web application.

Web container

An execution environment that controls Web application. Web container also provides all types of services such as security and transaction. A Web application runs on Web container.

Web application conforming to the specifications of Java Servlet2.5 and JavaServer Pages Specification v2.1 can be executed.

Web container server

Web container server is the server running in servlet engine mode.

Web Service Description Language

This is one of the language specifications for coding SOAP Web Services. XML is the base for the same.

Web Services Interoperability Organization

WS-I is a group that lays does the standard specifications for improving the mutual connectivity between Web Services products and platforms created by different vendors.

Web Services Reliability

This is an open asynchronous communication protocol for internet that is standardized by the standardization group OASIS.

Web server

A program to execute the processing related to requests receptions from the Web browser and data transmission to Web browser. The Hitachi Web Server, Microsoft IIS, or in-process HTTP server can be used on the Application Server. The in-process HTTP server is the Web servers running in J2EE server processes.

If the Management Server is used, Hitachi Web Server can be treated as a logical server.

Web server integration

A method for using Hitachi Web Server or Microsoft IIS as a Web server used in Application Server.

To use Web server integration, the redirector module provided by Cosminexus Component Container is embedded into Hitachi Web Server or Microsoft IIS.

Web Services

This program (service) is provided through a network by using the Web related technology. With Application Server, you can develop and execute SOAP Web Services and RESTful Web Services.

Web Services client

This is the client program that uses the Web Services. It codes the process for calling the Web Services by using the classes automatically created by commands, in the development of the client of SOAP Web Services. The client of RESTful Web Services is developed by using the standard Java API.

Web Service Implementation Class

A program for implementing SOAP Web Services. During the SOAP Web Services development, the classes that are auto-generated by the command are used to code the implementation of the Web Service.

Web system

A Web cluster system that is configured with a single load balancer and multiple Web servers and application servers. It provides a single application service.

WFA

This is an abbreviation of *Work-Flow Architecture*. For details, see *Work-Flow Architecture*.

Windows Server Failover Cluster

A type of cluster software.

Program that monitors and switches the system when a failure occurs. By integrating the system with the Windows Server Failover Cluster, the reliability and operation rate of a system built using Application Server can be improved. Windows Server Failover Cluster can be used on Windows Server 2008 and Windows Server 2012.

Work-Flow Architecture

This is the notation about business flow proposed by the Ministry of Economy, Trade and Industry, in the enterprise architecture You can manifest the business related organization or in-charge, business flow and the communicated information.

World Wide Web Consortium

This is the non-profit making group that proposes the Web related technology standardization including XML or DOM.

WS-I

This is an abbreviation of *Web Services Inter-operability Organization*. For details, see *Web Services Inter-operability Organization*.

Wizard

A type of Help functionality provided with package software products. In general, an interactive navigator functionality that supports the setup operation is called a wizard. Complex parameter setup operations requiring detailed know-how, such as formatting of tables and documents, and creation of cross-tabulation tables, are navigated in an interactive manner.

Worker process

Process of a Web container operating in the background of a Web server. It is the Web container execution process as viewed from the redirector. It specifically applies to the J2EE server or the Web container server.

WS-R

This is an abbreviation of *Web Services Reliability*. For details, see *Web Services Reliability*.

WS-Security

The specification related to SOAP Web Services security defined by OASIS. The WS-Security uses the XML security technology.

WSDL

This is an abbreviation of *Web Service Description Language*. For details, see *Web Service Description Language*.

WSDL generating function

The functionality for automatically generating WSDL using the development support commands.

WTP

This is an abbreviation of *Eclipse Web Tools Platform*. For details, see *Eclipse Web Tools Platform*.

WTP connector

Eclipse plug-ins are used for starting J2EE servers from WTP and deploying J2EE applications on the J2EE servers.

XML

This is an abbreviation of *Extensible Markup Language*. For details, see *Extensible Markup Language*.

XML document

A document written in XML format.

XML format definition file

A message format definition file used for handling XML data. The XML format definition file is created as an XML schema file.

XML parser

A program that parses an XML document.

There are two types of parsers: DOM parsers that use a DOM API, and SAX parsers that use SAX. DOM parsers output the parsing result for XML documents as a DOM tree. SAX parsers report the parsing result as an event rather than outputting DOM trees.

These parsers are collectively called *XML parsers* in this manual.

XML Schema

A schema defining a logical data structure in XML documents by using specifications defined in the W3C Recommendation.

XML specifications

Specifications defined in the XML 1.0 and XML 1.1 documents of the W3C Recommendation. The XML parser for Cosminexus XML Processor can process XML documents that follow XML specifications.

XPath

Xpath defines the syntax that indicates a specific part in an XML document. Xpath enables you to specify any location in a document without specifying an anchor in the XML document.

XSL

This is an abbreviation of *Extensible Stylesheet Language*. For details, see *Extensible Stylesheet Language*.

XSLT

This is an abbreviation of *XSL Transformations*. For details, see *XSL Transformations*.

XSLTC transformer

An XSLTC transformer offers the same functionality as the XSLT transformer provided by Cosminexus XML Processor, with improved performance in XML document transformation. An XSLTC transformer consists of the compiler that parses each XSLT style sheet and the runtime processor that performs transformation.

XSL Transformations

These are specifications for converting the structure of XML document such as converting the XML document to the XML document, HTML, or text conforming to the style sheet.

XSLT transformer

A program that implements XSLT. The XSLT transformer enables you to transform an XML document into an XML document, HTML, or text that follows a style sheet.

Index