TIBCO Substation ES™ Concepts

Software Release 2.10
May 2016

Two-Second Advantage®

TIBCO®

Important Information

Contents

TIBCO Documentation and Support Services

Documentation for this and other TIBCO products is available on the TIBCO Documentation site. This site is updated more frequently than any documentation that might be included with the product. To ensure that you are accessing the latest available help topics, please visit:

https://docs.tibco.com

Product-Specific Documentation

Documentation for TIBCO products is bundled with the software. It is also available on the TIBCO Documentation site at https://docs.tibco.com/products/tibco-substation-es. To directly access documentation for this product, double-click the following file:

`TIB_substation_`*`version`*`_docinfo.html`

where this file is shipped with the software package `.zip` file.

The following documents for this product can be found on the TIBCO Documentation site or after extracting the documentation `.zip` file:

- *TIBCO Substation ES Concepts*
- *TIBCO Substation ES Installation*
- *TIBCO Substation ES Operations and Administration*
- *TIBCO Substation ES Configuration and Resources*
- *TIBCO Substation ES Messages and Codes*
- *TIBCO Substation ES Release Notes*

The following documents provide additional information and can be found on the TIBCO Documentation site:

- *TIBCO Rendezvous for z/OS Installation and Configuration*
- *TIBCO Rendezvous for z/OS COBOL Reference and TIBCO Rendezvous C Reference*
- *TIBCO Enterprise Message Service User's Guide*
- *TIBCO Enterprise Message Service C & COBOL API Reference*
- *TIBCO Mainframe RED User's Guide*
- *TIBCO Mainframe RED Installation*

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, contact TIBCO Support:

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:

  http://www.tibco.com/services/support

- If you already have a valid maintenance or support contract, visit this site:

  https://support.tibco.com

  Entry to this site requires a user name and password. If you do not have a user name, you can request one.

How to Join TIBCO Community

TIBCO Community is an online destination for TIBCO customers, partners, and resident experts. It is a place to share and access the collective experience of the TIBCO community. TIBCO Community offers forums, blogs, and access to a variety of resources. To register, go to the following web address:

https://community.tibco.com

Introduction

TIBCO Substation ES™ subscribes to and publishes data to TIBCO applications and transaction processing systems, such as Customer Information Control System (CICS) and Information Management System (IMS), which run in the z/OS operating-system environment.

TIBCO applications are those that can communicate with Substation ES through a messaging protocol supported by Substation ES, for example, TIBCO Rendezvous® and TIBCO Enterprise Message Service™. Examples of such applications include TIBCO BusinessWorks™, TIBCO BusinessEvents®, TIBCO Business Process Management (BPM), message-enabled web applications, and programs that use messaging applications APIs.

Substation ES has the following capabilities:

- Handling all communications and routing between TIBCO applications and CICS or IMS transaction processing applications. The TIBCO applications can be running anywhere on the network while the transaction processing applications are running in the z/OS environment.
- Translation data between TIBCO applications and the CICS or IMS transaction processing systems. Data mapping and configuration are specified through a series of Interactive System Productivity Facility (ISPF) panels, which define how each field is to be converted based on the direction in which the data is exchanged. Substation ES also includes the facilities for ensuring the integrity of the transactions.
- A flexible set of system startup and initialization parameters, which control the setup and configurations.
- A console interface, remote access, and a web browser application for monitoring and controlling Substation ES after startup.
- Tracing and logging facilities to track the progress of transactions and handle errors.
- Feedback on the Substation ES processing, message handling, and runtime errors by using TIBCO messaging through user data fields.

Substation ES is especially useful at sites that plan to augment the processing capabilities of z/OS applications with newer applications that execute from process engines, application servers or any network-attached devices. In that scenario, system administrators want the flexibility and ease of usage offered by popular data entry systems while retaining the robust back-end processing capabilities of their mainframe systems.

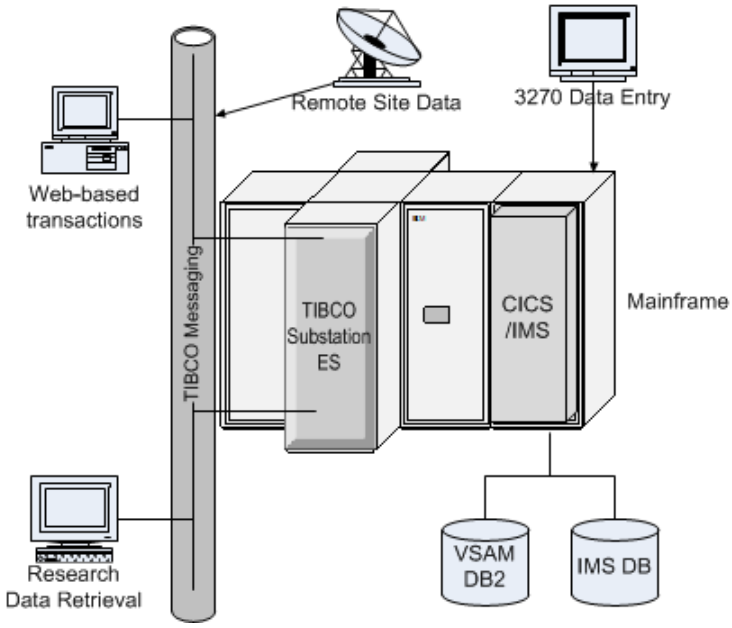
For example, suppose a financial institution wants to make the financial information stored in applications in the z/OS environment more available to its customers. The institution can use Substation ES and TIBCO applications to build a bridge by which customers can submit requests for information from PCs at branch offices or through a web browser on the Internet.

Additionally, applications in the z/OS environment can use Substation ES to publish data or access data generated by other types of applications. The information is transmitted by TIBCO application messages.

For example, a CICS or IMS application can publish a running tally of the number of widgets that have been ordered through an order processing system. With Substation ES, this information can be made available to company employees worldwide over their intranet.

As another example, the institution can use Substation ES to subscribe to events and notify transaction processing applications on z/OS when an order is processed in one of branches of the institution. The branches can be using a PC-based order-taking system and local database, but the applications on z/OS can stay informed.

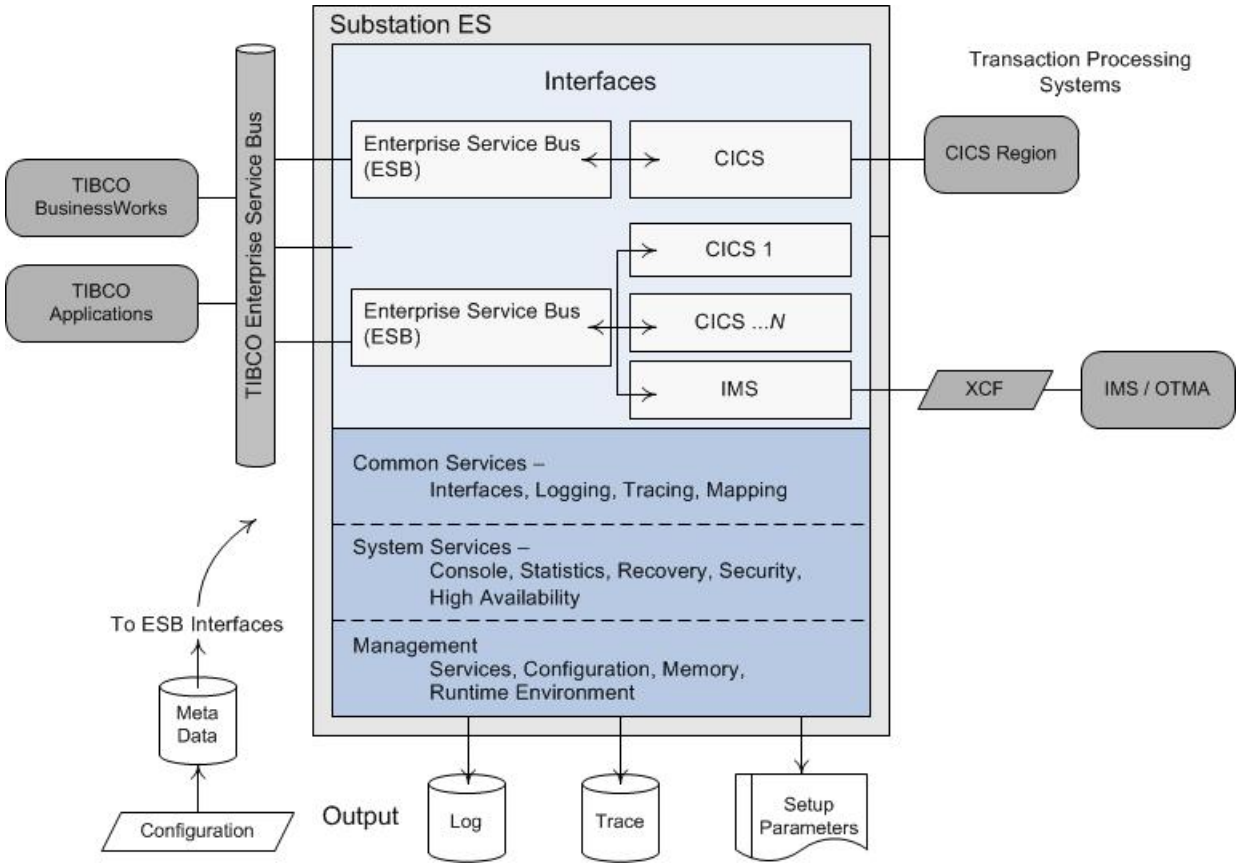
The following figure shows some of the flexibility that can be designed into a system-wide enterprise, where information must be shared between applications on different operating systems and a mix of computing platforms.


Remote Site Data
3270 Data Entry
Web-based
transactions
TIBCO Messaging
TIBCO
Substation
ES
CICS
/IMS
Mainframe
Research
Data Retrieval
VSAM
DB2
IMS DB

Architecture

Substation ES is a Multiple Virtual Storage (MVS) application that runs in a single region.

The following figure shows the basic design of Substation ES.



For information about the functional areas or layers of Substation ES, see Components.

Components

Substation ES contains functional areas or layers and each is responsible for a given set of tasks and services.

Substation ES consists of the following components:

- Interfaces
- Common Services
- System Services
- Management
- Metadata Configuration

Interfaces

Interfaces initiate, control, and terminate sessions; and manage the data flow between Substation ES and transaction processing systems.

Substation ES contains the following interfaces:

Interfaces	Description
ESB Interface	ESB Interface (formerly referred to as the Transformer) translates data types between TIBCO applications and the CICS or IMS transaction processing applications. See ESB Interface.
CICS Interface	CICS Interface passes data between TIBCO messaging applications and CICS applications in the z/OS environment. See CICS Interface
IMS Interface	IMS Interface connects TIBCO messaging applications and IMS applications in the z/OS environment. See IMS Interface.
RED (XCF) Interface	RED (XCF) Interface connects external applications with z/OS programs and transactions that are located in the batch, CICS, IMS BMP, IMS MPP or TSO regions. See RED (XCF) Interface .

Common Services

Common services are internal services that are always available for the Substation ES interfaces.

Common services are used by interface processes to perform logging, tracing, common facilitation of system monitoring, and capture of errors and statistics. Because they run in worker thread subtasks, common services are always multitasked. You can use the logging and tracing facilities to track the progress of transactions and handle errors.

Logging Agent

Substation ES Logging Agent processes requests for log entries, constructs the log header information, formats the input, and writes the input to the designated device.

Log entries represent individual events that are recorded at specific points while Substation ES is active. A log entry consists of a header and a variable data portion. Log entries are grouped into either escalated or selectable categories. An escalated category stipulates that log events are increased as the level value specified by the user is raised. The selectable category includes log entries that are written at specific focus points within the execution thread in the region where Substation ES operates.

Tracing Agent

Substation ES Tracing Agent processes requests for trace entries. It constructs the trace header information, formats the input, and writes the input to the selectable configured device.

Trace entries are individual events recorded at specific locations in the execution code path while Substation ES is active. A trace entry consists of a header and variable data. Trace entries are grouped into escalated or selectable categories. An escalated category stipulates that trace events are increased according to a value specified by the administrator. The selectable category specifies trace entries that are written at specific focus points within the execution thread of the Substation ES region.

System Services

System services are processed only by the Substation ES main task or administration interface. Substation ES provides access to all interfaces, so system services can be requested for execution when required.

System services consist of the Substation ES Unit of Work (UOW) handling, interface abend detection, the MVS system console interaction facility, statistics monitoring, and recovery from disconnection or other system failures.

The console interface offers a convenient means for administrating, monitoring, and controlling events and services. You can also check from the console the status of components and interfaces that are operating in the Substation ES region. For example, system administrators and operations personnel can identify pending requests and replies, and manage them from the console.

Substation ES contains the following configuration and administrative components:

- System startup parameters
- System Initialization Parameters (SIPs)
- Transformation Data Configuration User Interface (DCUI) panels
- Configuration and reporting utilities

System Initialization Parameters

System Initialization Parameters, which are processed as input during the creation of the Substation ES region, dictate how various components operate.

System Initialization Parameters are defined in two partitioned data set (PDS) member files that are coded as 80-byte records, as follows:

- The `SSP` member

  Specifies the configuration of the logging and tracing device that pertain to Substation ES.

- The `SSI` member

  Specifies the configuration of the identification and operational parameters.

  Substation ES contains default values for the members, which you must configure to suit the requirements of a given site. For the configuration parameters and their values, see *TIBCO Substation ES Installation*.

Interface Parameters

Configure interface parameters to request system services for Substation ES to access interfaces when required.

The configuration items that initialize and configure ESB Interface at startup are located in a PDS member. Those that describe Rendezvous or Enterprise Message Service messages and transaction input are located within a virtual storage access method (VSAM) file. Modify the following members according to your requirements:

- The `ADM` member

  Specifies the configuration of Administration and Operations Interface.

- The `XFR` member

  Specifies the configuration of ESB Interface.

- The `IMS` member

  Specifies the configuration of IMS Interface for the transaction processing system.

- The `CICS` member

  Specifies the configuration of CICS Interface for the transaction processing system.

  You must edit those members to suit the requirements of a given site. For the configuration parameters and their values, see *TIBCO Substation ES Installation*.

Management

You can use Substation ES management facilities to manage multiple interfaces in a single region, handle each interface logistics according to its configuration, respond to operational commands to allow 24x7 operations, and provide remote application users with status and real-time information.

The following are the management facilities in Substation ES:

- Reporting usage and statistics for each interface.
- Internal management of interfaces and the flexibility to specify concurrent worker tasks for managing payloads of different sizes, durations, and throughput rates.
- An active operational command interface, which listens and responds to user requests from local host consoles or to remote commands sent through ESB.
- Configuration metadata entities, which you can create, update, enable, or disable while Substation ES is in operation.
- Monitoring connectivity to TIBCO ESB components and transaction processing systems along with automatic recovery and reconnection when an external system becomes available again.

Metadata Configuration

Metadata configuration refers to the configuration items (recipes and triggers) that perform data exchanges between TIBCO applications and the transaction processing applications on z/OS.

The configuration items are entered and updated in the ESB Interface configuration panels. Those items that describe TIBCO messages and transaction input are located within a VSAM LDS file. Substation ES has a number of features for managing the flow and exchange of data between applications.

ESB Interface

ESB Interface (formerly referred to as the Transformer) translates data types between TIBCO applications and the CICS or IMS transaction processing applications.

ESB Interface performs the following tasks:

- Manages the ESB connectivity, the routing of requests and responses, marshals activities going to back-end systems, and owns the service registry located in the configuration file.
- Translates the data in the messages that are passed between TIBCO applications and the transaction processing applications on z/OS according to the data mapping parameters. You can specify those parameters in the Transformation Definition panels, which define how each field in the message is to be converted.

- Converts the data from Rendezvous and Enterprise Message Service messages according to the specified mapping, and sends it to the appropriate Substation ES interface, where it is made available for processing by a transaction processing application, such as CICS or IMS.
- Converts the data generated by transaction processing applications on z/OS into messages that contain data suitable for delivery to TIBCO messaging applications that subscribe to that data on the network.
- Ensures that the current metadata matrix objects are loaded into memory and made available for ESB Interface at Substation ES startup.

The configuration items at startup that initialize and configure ESB Interface are located in a PDS member.

Conversion Agent

The Conversion Agent performs the following tasks:

- Manage the dynamic parameter settings, which you can configure while Substation ES is running.
- Ensure that the current metadata matrix objects are loaded into memory and made available for ESB Interface at the Substation ES startup.

Substation ES Messaging Agents

When TIBCO applications issue requests to publish/subscribe to messages, the requests are processed by Substation ES through Message Receipt Agent and Message Delivery Agent. Those agents receive and deliver messages asynchronously.

Message Receipt Agent

Message Receipt Agent is a Rendezvous or Enterprise Message Service listener or a group of listeners.

Message Receipt Agent performs the following tasks:

- Listens to Rendezvous or Enterprise Message Service messages being delivered to a subject or destination, which is defined at configuration time.
- Ensures that a proper reply subject, destination, or inbox name exists so that Message Delivery Agent can route the returned data to the appropriate subscriber.
- Forwards messages received to ESB Interface for translation, if necessary.

Message Delivery Agent

Message Delivery Agent is a Rendezvous or Enterprise Message Service sender or a group of senders.

Message Delivery Agent performs the following tasks:

- Determines whether an acknowledgment is required by the calling TIBCO messaging applications upon delivery of a given message. If a message delivery fails and the service is guaranteed, Message Delivery Agent backs out all updates to recoverable resources.
- Delivers return codes, acknowledgments, condition codes, and result sets in the form of Rendezvous or Enterprise Message Service messages to TIBCO messaging applications or their clients, such as TIBCO BusinessWorks.
- Delivers acknowledgments and result sets according to the flow control protocol for the service level, for example, reliable, certified, or guaranteed, that is active for a given subject. The details of the type of service level required are specified by the messaging application, recipe, or trigger for the request.

Message Flow

Substation ES accepts Rendezvous or Enterprise Message Service messages that originate from applications anywhere on the network. Conversely, Substation ES also accepts messages or events generated by transactions that are running within the CICS or IMS regions on z/OS.

Within the scope of Substation ES, messages and requests are synonymous. Substation ES transforms the following types of messages, enabling them to be passed between applications.

- Requests from TIBCO BusinessWorks or other TIBCO messaging applications to obtain data from a transaction processing application such as CICS or IMS. In those cases, Substation ES returns a data set to the requesting application.
- Requests from TIBCO BusinessWorks or another TIBCO messaging application for data to be supplied to a transaction processing application such as CICS or IMS. No return data sets apply, but there might be a status return, that is, an acknowledgment of receipt.
- Requests (trigger events) from a transaction processing application such as CICS or IMS to supply data or events to listening TIBCO messaging applications.

There are two types of message flow:

- Requests or Replies

  Substation ES can subscribe to messages generated by external TIBCO applications, transform the messages into a z/OS data format, and pass them to CICS or IMS systems for processing. For example, a TIBCO application can publish a request for data to a CICS or IMS application that runs on z/OS. Optionally, that application responds with an acknowledgment or returns a result set.
- Triggers

  Substation ES can also publish data on behalf of CICS or IMS applications, that is, translate the data from the z/OS format and package it as an ESB message for delivery to an external application that subscribes to the data.

  When Substation ES receives a request, it does the following:

  1. Checks the authentication file of the request for authenticity and verifies if the format is valid.
  2. Sends the request to ESB Interface for data translation.

ESB Interface translates data types and formats between TIBCO applications and CICS or IMS applications according to the translation definitions specified in the transformation configuration panels. The data mapping defines how each field in a given message is translated as it is passed from a TIBCO application to the CICS or IMS application and vice versa.

Substation ES interfaces serve as the mechanism for communicating and passing information between Substation ES and transaction processing systems, databases, and other services that are available on z/OS. Substation ES interfaces give TIBCO applications on any platform a means of accessing platform-specific applications on z/OS. Substation ES interfaces for z/OS are robust, scalable, and capable of delivering a high level of transaction throughput.

Message Flow from TIBCO Applications

There are three types of message flow from TIBCO applications: request or reply, trigger, and data transformation.

The following examples illustrate the message flow scenarios:

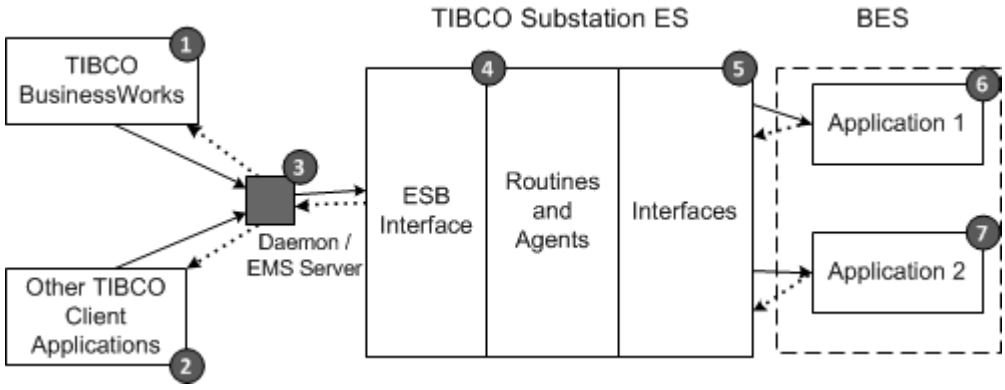
- Requests or Replies
- Triggers

- Data Transformation

Requests or Replies

Requests or replies occur in both the CICS and IMS environments.

The following figure shows the request or reply message flow:

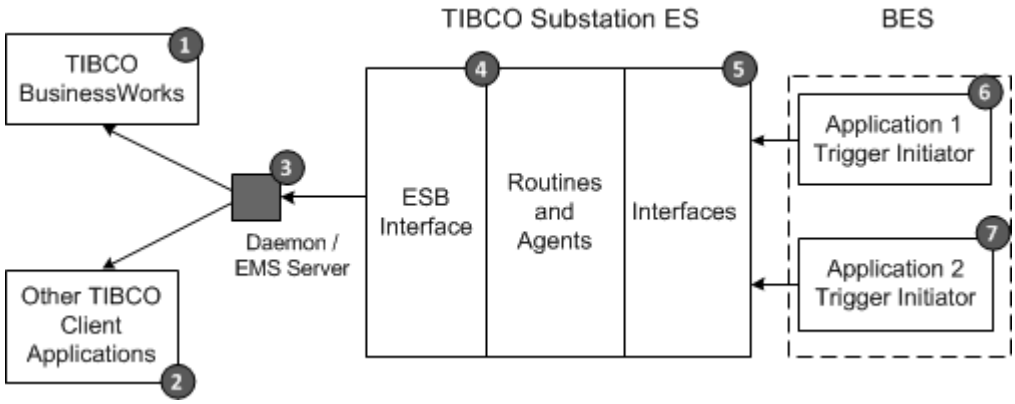


1. Either TIBCO BusinessWorks (1) or another TIBCO messaging client application (2) initiates a request that is passed to a messaging daemon or EMS server (3). The request is denoted by solid arrows.
2. The daemon or EMS server (3) forwards the request to Substation ES, where ESB Interface (4) receives and transforms the message data and sends it to the appropriate interface (5).
3. The interface (5) delivers the request to the application in the back-end system (BES). In this example, the delivery is to Applications 1 and 2 (6 and 7) within the z/OS environment.
4. Applications 1 and 2 (6 and 7) return a result set or acknowledgment, as denoted by the dotted arrows that return to the applications that initiate the request.

Triggers

Triggers occur in both the CICS and IMS environments.

The following figure shows the triggers message flow:



1. In a trigger, the request initiates within an application in the BES within the z/OS environment. In this figure, application 1 or 2 (6 or 7) initiates the request.
2. The appropriate interface (5) within Substation ES receives the request and sends it to ESB Interface (4).

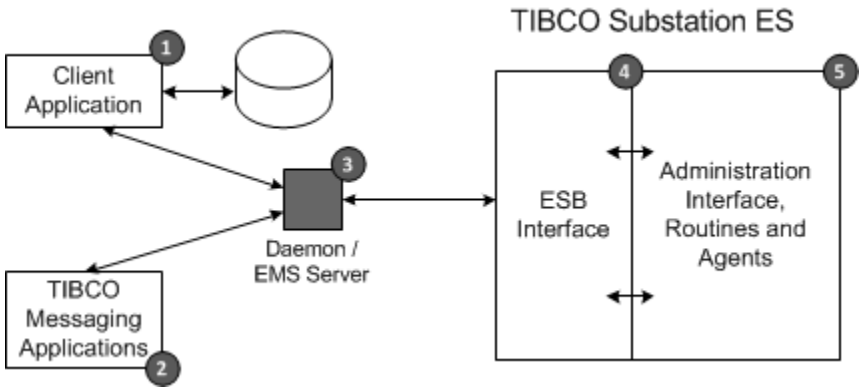
3. ESB Interface (4) transforms it to a Rendezvous or Enterprise Message Service message, which is carried by the daemon or EMS server (3). From there, the message can be sent to TIBCO BusinessWorks (1) or other TIBCO messaging client applications(2).

For triggers, no result set or acknowledgment goes back to the initiator (6 or 7), hence no dotted arrows in the figure.

Data Transformation

Substation ES effectively transforms data from different mainframe sources to TIBCO messaging applications and vice versa.

The following figure shows the message flow for data transformation:



1. A client application (1) reads records from a database or files and publishes each record to Substation ES as an opaque message. The structure of the record need not be known to the application at this point.
2. ESB Interface (4) receives the record, performs no data conversion on the input, and simply hands it to Substation ES Administration Interface (5).
3. Substation ES Administration Interface (5) hands the record back to ESB Interface (4), where an outbound conversion occurs, depending on the configuration specified.
4. The record, now converted as a Rendezvous or Enterprise Message Service message, can be published to a subscribing TIBCO messaging application (2).

Message Flow with Enterprise Message Service

Substation ES supports TIBCO Enterprise Message Service, which offers several message services, including Java Message Service (JMS), which in turn supports two messaging models.

The following list shows two messaging models:

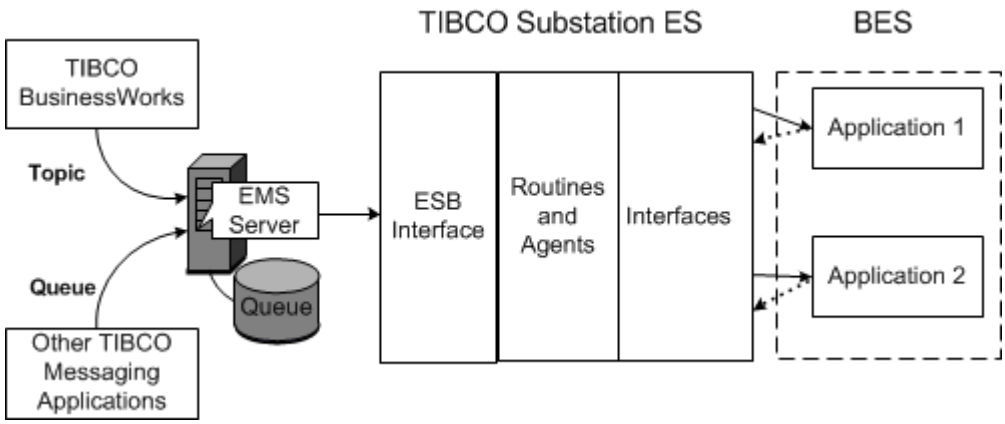
- **Queues** (point to point)

  Messages are stored in a queue until they are processed.

- **Topics** (publish and subscribe)

  Messages are addressed to a topic.

An Enterprise Message Service server acts as an intermediary between TIBCO messaging applications and ESB Interface, as shown in the following figure:


TIBCO Substation ES
BES
TIBCO BusinessWorks
Topic
EMS Server
Queue
Queue
Other TIBCO Messaging Applications
ESB Interface
Routines and Agents
Interfaces
Application 1
Application 2

Service Levels

Substation ES supports different service levels for different messaging applications.

For Rendezvous

Substation ES currently supports the reliable service level for message communication to Rendezvous applications.

You can specify the usage and implementation of the reliable service level in the Substation ES configuration, which is coordinated by Substation ES and adhered to by Substation ES Interface.

For the reliable service level, Substation ES receives a message and forwards a request to the appropriate Substation ES interface for execution of a transaction. If necessary, a receipt of the delivery, execution status, and the output result set can be returned to the TIBCO messaging applications. The execution path of this service level within Substation ES is lightweight and can be used by many applications to submit inquiries or supply information.

No context information is stored, so if a network loss occurs or if the applications on z/OS fail to respond, no results are returned to the TIBCO messaging applications. However, informative error notifications together with a correlation tag are sent to processes expecting a reply.

For Enterprise Message Service

Substation ES supports both the reliable and guaranteed service levels for message delivery for Enterprise Message Service.

The following are the service levels that Substation ES support for Enterprise Message Service:

Reliable Auto Acknowledge

The Enterprise Message Service client library automatically confirms the receipt of a message.

Reliable No Acknowledge

The Enterprise Message Service client library does not send a receipt of the message during processing.

Guaranteed All

Substation ES calls the Enterprise Message Service client library and explicitly confirms receipt of a message once after all Substation ES processes and executions controlled by Substation ES for a given message are completed successfully.

Guaranteed Substation Processing

Substation ES calls the Enterprise Message Service client library and explicitly confirms receipt of a message if the Substation ES processes are completed successfully. Processing is considered complete if a response is successfully delivered to its destination. That rule applies regardless of the type of message (error or application) that is delivered back by Substation ES.

CICS Interface

This section describes the CICS Interface used by Substation ES to pass data between TIBCO messaging applications and CICS applications in the z/OS environment.

Overview

Substation ES CICS Interface manages the communications and coordinates the data exchange between Substation ES and the CICS transaction processing system.

TIBCO messaging applications can subscribe to data and request acknowledgments or request sets of results from CICS transaction processing applications. In addition, Substation ES can publish data on behalf of CICS applications to subscribing TIBCO messaging applications.

When received by Substation ES, Rendezvous and Enterprise Message Service messages are transformed into formatted data buffers. The data buffers are then sent to CICS Interface, which combines them with a request header, depending on the method of invocation (MOI). Different MOIs are used by CICS Interface from which requests are interpreted. A reply from the CICS program can be returned to an awaiting user subscription as a Rendezvous or Enterprise Message Service message.

CICS Interface, which operates as a subtask within Substation ES region, communicates with CICS regions with IBM External CICS Interface (EXCI) protocol. CICS Interface employs a multithreaded design that can accommodate multiple sessions or pipes that communicate with each CICS region. Up to 30 CICS Interface instances can operate and coexist concurrently within the Substation ES region. The only limit is that each interface is restricted to communicating with only one CICS region at a time.

You can specify the number of CICS interfaces used by Substation ES through System Initialization Parameters, which provide a high degree of scalability for handling the data processing requirements of a given site.

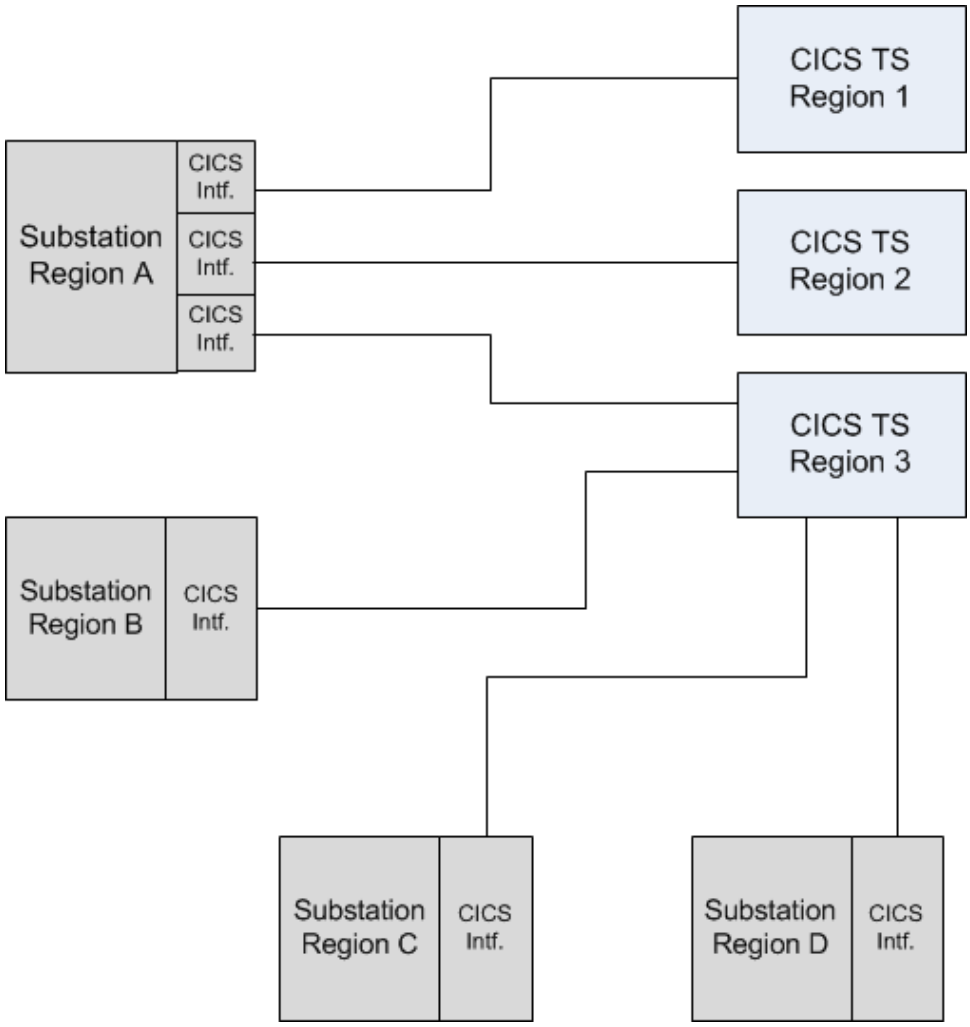
The following are some of the tasks performed by the CICS Interface:

- Enables multiple CICS interfaces to be allocated for each Substation ES, enabling communication with many CICS regions.
- Provides flexible configuration options that are specified during the startup of Substation ES.
- Requires no special APIs to perform CICS-initiated outbound events (triggered events).
- Supports reliable and guaranteed message delivery services.
- Supports simultaneous connection of up to four Substation ES interface instances to a single CICS region.
- Support alternate routing, enabling each recipe service to define both primary and secondary back-end systems in which services can be executed. If the primary system is unavailable, Substation ES automatically tries to schedule execution on the alternate, as specified.

Configurations

One or more CICS interface instances can operate and coexist concurrently within one Substation ES region. Each CICS interface is restricted to communicating with only one CICS region at a time.

The following figure shows how Substation ES is configured with CICS:

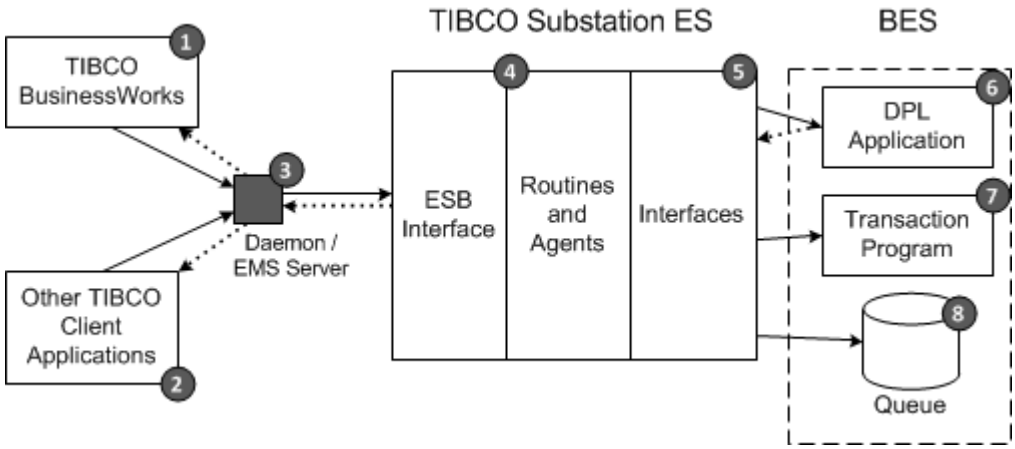


Message Flow

CICS Interface supports two types of message flow: requests or replies, and triggers.

Requests or Replies

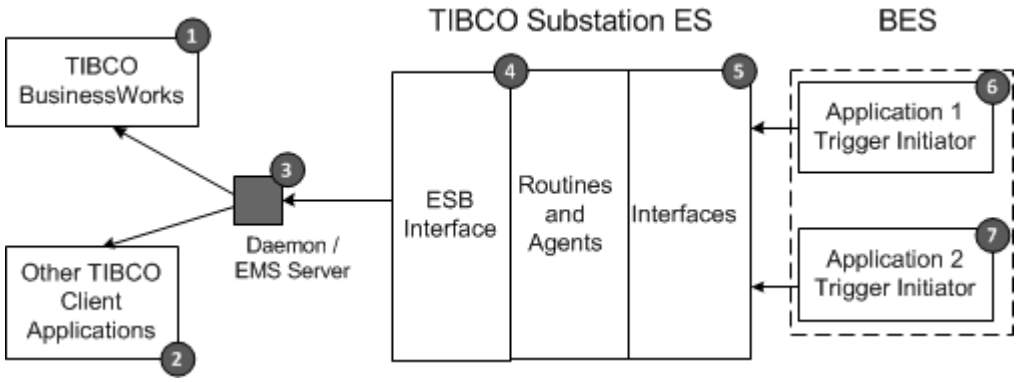
The following figure shows the message flow of the CICS interface for requests or replies:



1. Either TIBCO BusinessWorks (1) or another TIBCO messaging client application (2) initiates a request that is passed to a messaging daemon or EMS server (3). The initiating request is denoted by solid arrows.
2. The daemon or EMS server (3) forwards the request to Substation ES, where ESB Interface (4) receives and transforms the message data and then sends it to the appropriate interface (5).
3. The interface (5) delivers the request to the application in the BES. In this example, the request is sent to a Distributed Link Request (DPL) application (6), Transaction Program (7), or Queue (8).
4. Only the DPL application (6) returns a result set or acknowledgment, as denoted by the dotted arrows that return to the initiating applications.

Triggers

The following figure shows the message flow of the CICS interface for triggers:



1. In a trigger, the request initiates within an application in the BES in the z/OS environment. In this figure, Applications 1 or 2 (6 or 7) initiate the request.
2. The appropriate interface (5) in Substation ES receives the request and sends it to ESB Interface (4).
3. ESB Interface (4) transforms the message to a Rendezvous or Enterprise Message Service message, which is carried by the daemon or EMS server (3). From there, it can be sent to TIBCO BusinessWorks (1) or other TIBCO messaging clients (2).

In triggers, no result set or acknowledgment goes back to the initiator (6 or 7), hence no dotted arrows back to the initiator in this figure.

High Volume Triggers

The CICS High Volume Trigger (HVT) capability is available for use when a high volume of messages or queue records are to be sent, or when more than 32 KB of information needs to be sent from applications.

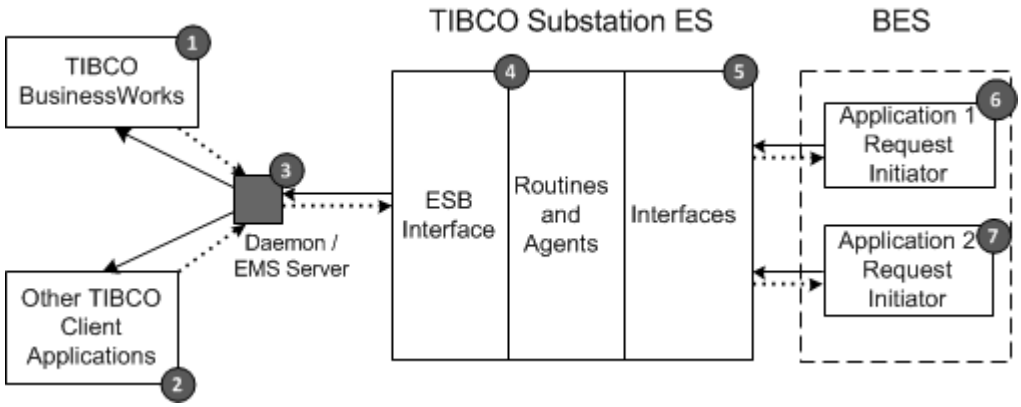
HVT also provides ordered delivery of information and guaranteed message delivery. Information located in data or transient queues can be used with this facility.

Consider using the CICS HVT capability in Substation ES in the following scenarios:

- If you expect a high volume of messages or queue records to be sent from CICS regions.
- If users require ordered delivery of information.
- If applications must send more than 32 KB of information.
- If Enterprise Message Service is your transport and if you require guaranteed message delivery for triggered information.
- For business applications that demand a higher publishing throughput and might not be concerned with message loss if CICS terminates, the HVT option can make use of Temporary Storage Queues (TSQs) or Transient Data Queues (TDQs).

For more information about HVT, see *TIBCO Substation ES Operations and Administration*.

The following figure shows the message flow for CICS-initiated requests or replies:



1. In a CICS-initiated request or reply, the initial request comes from an application in the BES in the z/OS environment. In this figure, the request initiates from Applications 1 or 2 (6 or 7). The path of the initial request is denoted by solid arrows.
2. The request is received by Substation ES CICS Interface (5), passed to ESB Interface (4), and then sent to the messaging daemon or EMS server (3).
3. The daemon or EMS server (3) sends the message to TIBCO BusinessWorks (1) or another TIBCO messaging client (2), depending on the message delivery criteria.
4. The CICS application (1 or 2) waits for a response from the external messaging application. A timeout value specified by application 1 or 2 determines the length of time applications are prepared to wait for a response.
5. In all these transactions, a result set or acknowledgment is returned, as denoted by the dotted arrows in this figure.

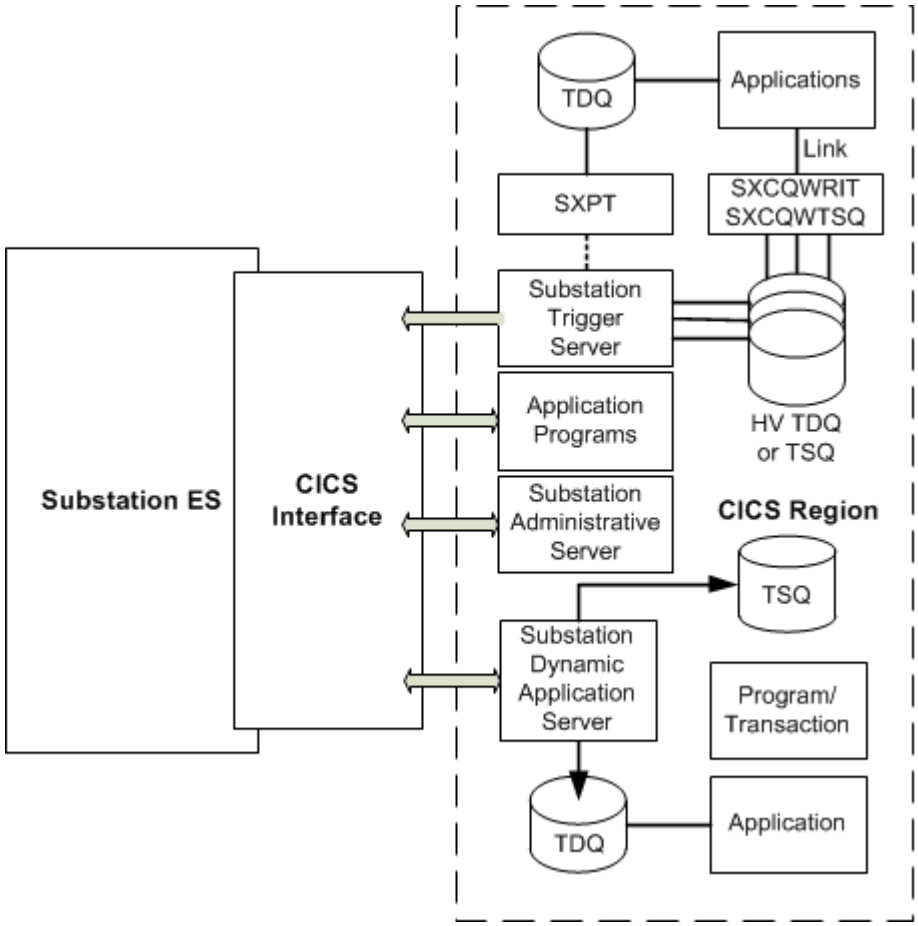
CICS Methods of Invocation

Multiple methods of invocation are available to access and work with CICS applications and resources: containers, communication areas, transient data, storage queues, and most other entities available in CICS.

The CICS Interface enables data to be communicated to the CICS region in various methods of invocation, as follows:

1. Invokes a CICS program from the CICS interface through a DPL application.
2. Writes the formatted data buffer to a CICS TDQ for subsequent processing.
3. Writes the formatted data buffer to a CICS TSQ for subsequent processing.
4. Initiates a CICS transaction from the CICS resident Substation ES server.
5. Optionally passes the communications area (COMMAREA) or a container to initiated transactions.
6. Transfers control (XCTL) to a user application from the CICS resident Substation ES server.

The following figure shows the message flow of CICS Interface:



Components

Substation ES CICS Interface components interact with Substation ES servers in the CICS region.

Substation ES CICS Interface takes advantages of three Substation ES servers in the CICS region:

- **SS Dynamic Applications Server**

  The SS Dynamic Applications server primarily extends a CICS transaction request, allowing access to other CICS resources and transactions that cannot be DPL-enabled. A request can initiate a CICS program or transaction or write the formatted data buffer content to a TDQ. You can trigger those tasks by selecting the appropriate MOI option in the transformation configuration panels. The COMMAREA can be sent to programs or transactions.

  Because they are performed asynchronously, all requests performed by the SS Dynamic Applications Server return only the results of the CICS function and not the result set from the application.

- **SS Trigger Server**

  The SS Trigger server primarily waits for events to be written to a CICS TDQ and signals Substation ES CICS Interface to extract them from the queue. CICS Interface then forwards the events or requests to ESB interface, where the data is translated according to the ESB Interface configuration. The data is published by Substation ES as a Rendezvous or Enterprise Message Service message. The name of the TDQ associated with this trigger server is specified in the CICS members of the system initialization parameters.

  The trigger server handles reliable and guaranteed triggers as well as CICS-initiated requests or replies. Each trigger facility can be disabled if it is not required by the Substation ES implementation.

- **SS Administrative Server**

  The SS Administrative server primarily performs housekeeping tasks and responds to Substation ES operational requests.

CICS TD Queue

Substation CICS Interface reads outbound events and, when requested, writes inbound data with TDQs.

Ensure that you have defined the TDQ names before starting Substation ES.

You can define a TDQ in CICS as a remote queue for multiple CICS server regions. The CICS TDQ must be a logically recoverable intrapartition TDQ that is recoverable on warm and emergency restarts.

For more information about TDQs, see *CICS System Definition Guide*.

CICS TS Queue

Substation CICS Interface reads outbound events and, when requested, writes inbound data with TSQs.

The CICS TSQ used by Substation ES must be local to the Substation CICS Routing Region (CRR). For more information about TSQs, see *CICS System Definition Guide*.

Substation ES CICS Event Post Transaction

The following are Substation ES CICS Event Post Transactions located within the CICS region:

- SXPT: reliable transaction
- SXPG: guaranteed transaction
- SXPO: ordered transaction
- SXPE: error message transaction

When events are written to the outbound TDQ, those transactions are triggered. For the CICS Interface to be notified of outbound events, the CICS administrator must declare the transaction IDs in the TDQ definition. Through a Substation ES-supplied transaction class, the transactions ensure that only one of them is active within a CICS region at any time.

Data Communications

Communications between TIBCO Substation ES and the CICS region require a Substation communication area (COMMAREA) with a maximum size of 32,000 bytes or characters. This limit is imposed by CICS for COMMAREAs.

Substation ES also supports CICS Container communications. The CICS Container is technically limited to 2 GB, but a more practical number is 2 - 3 MB. Substation ES does not support more than 4 MB for any online transaction.

When Substation ES CICS Interface issues a DPL call to a user application, it passes the formatted data buffer into the COMMAREA. From there, the data buffer is transferred to the application.

When CICS Interface communicates with TIBCO Substation ES CICS resident servers, it prefixes the COMMAREA with header and control information. If the COMMAREA must be passed to a client transaction or program, the header is stripped and the COMMAREA is transferred by way of CICS LINK, XCTL or START Transaction.

Because of the direct correlation between the TDQ record and COMMAREA, the length of a bigger TDQ record cannot exceed the supported maximum length unless HVTs are used.

IMS Interface

Substation ES IMS Interface connects TIBCO messaging applications and IMS applications in the z/OS environment.

Overview

TIBCO messaging applications are normally the source of the external messages and requests that are received by Substation ES. Substation ES-to-Substation ES communication through the messaging layer can be another source for messages.

ESB Interface transforms TIBCO messaging application requests into a buffer and forwards the buffer to Substation ES IMS Interface, which, in turn, attaches the buffer to an Open Transaction Manager Access (OTMA) header and passes the message through OTMA to IMS applications.

TIBCO clients can publish requests that initiate IMS transactions. Alternatively, IMS applications can publish events, messages, or requests/replies (synchronous callouts) from 3270 applications or from within Batch Message Program (BMP) or Message Processing Program (MPP) regions to TIBCO messaging applications that are listening to those events.

Communication between Substation ES IMS Interface and IMS is handled by the Substation ES OTMA Interface with the OTMA facility.

The preferred manner of defining OTMA destinations is using the IMS PROCLIB member DFSYDTx using the D control card. However, the previous method of using the OTMA Prerouting exit routine (DFSYPRX0) and Destination Resolution exit routine (DFSYDRU0) on M control cards is still supported.

Substation ES IMS Interface performs the following key tasks:

- Extracts the details of OTMA and IBM cross-system coupling facility (XCF).
- Submits IMS transactions and commands and receives IMS replies.
- Provides connectivity and conversations from programs that run on non-IBM platforms and other z/OS subsystems to multiple IMS servers or regions.

Substation ES IMS Interface has the following advantages:

- Flexible, scalable, and easy to install and operate, and includes implemented applications that use this facility.
- No coding (nonintrusive) is necessary or only minor changes (intrusive) are required to activate and use the interface with existing applications.
- The most direct and fastest method to add bidirectional IMS transactional information to the ESB, so that the information can be used by all SOA processes. The synchronous IMS request/reply support in Substation ES makes it even easier for IMS transactions to make requests across the ESB and have the ability to receive data.

Architecture

With Substation ES IMS Interface, external applications can communicate (publish, request, or reply) with transactions that are resident in the IMS Message Processing Program (MPP) regions.

Additionally, IMS applications, such as MPPs, Batch Message Programs (BMPs), and 3270, can also publish or redirect (trigger) output to external applications through Substation ES IMS Interface.

The Transaction Interface for IMS is part of Substation ES and couples together Substation ES and host IMS systems. With IMS Interface, multiple users can simultaneously connect and communicate with IMS Transaction Manager (TM). To understand the role and position of IMS Interface within z/OS, see Substation ES Architecture.

Substation ES IMS Interface uses IBM cross-system coupling facility (XCF) and IMS Open Transaction Manager Access (OTMA) to communicate with IMS Transaction Manager. OTMA is a transaction-based, connectionless client/server protocol.

Upon receiving a request published by TIBCO messaging clients, Substation ES determines the service level required for the message and forwards it to Substation ES IMS Interface. The interface interacts with IMS according to the desired service level. Therefore, messages sent to IMS transactions can be transactional or nonconversational.

A transaction level of service depends on the parameters, which you can configure in Substation ES, and on the type of TIBCO messaging client that initiates the request.

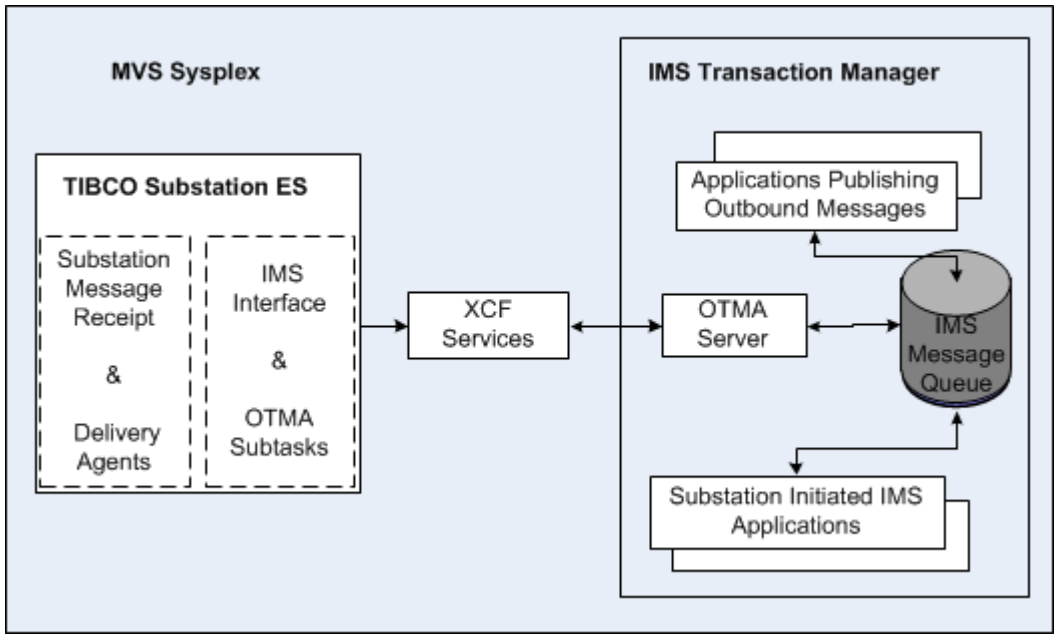
Substation ES can also receive messages initiated from IMS and will route these to the destination defined on the specified service. Also, Substation ES fully supports synchronous callout requests initiated from IMS.

Components

Substation ES IMS Interface consists of the following components:

- IMS Interface
- OTMA Interface
- XCF Services
- OTMA Server
- IMS OTMA Exits (Optional)
- IMS Exits (Optional)

The following figure shows the components of Substation ES IMS Interface:



IMS Interface

Substation ES IMS Interface establishes the connection and enables communication to IBM's IMS transaction processing system.

Substation ES IMS Interface is a multithreaded application that executes in the same address space as the host side of Substation ES. One Substation ES IMS interface can have many concurrent subtasks (OTMA clients) communicating with an IMS Transaction Manager (TM).

You set the number of threads that can be concurrently opened and other IMS connectivity settings and Substation ES required parameters. Set them at startup by specifying the values for the system initialization parameters (SIPs) for IMS Interface. For more information, see *TIBCO Substation ES Installation and Administration*.

You can configure multiple Substation ES IMS interfaces that will execute concurrently from a single Substation ES. Each such interface can simultaneously communicate with one or more IMS TMs.

ESB Interface can optionally perform data transformation when it receives a request or reply from an external application. When the ESB Interface processing is completed, it signals to IMS Interface that a request can be processed. Subsequently, the request, along with contents received from the TIBCO messaging application, is validated and packaged into an OTMA message by Substation ES OTMA Interface.

Substation ES IMS Interface can also receive data from the IMS TM applications. The data is processed as follows:

1. Forwarded as buffers (IOAREA) by the applications to an I/O Program Communication Block (IOPCB) or alternate IOPCB.
2. If used, redirected by the IMS OTMA prerouting exit.
3. Delivered to Substation ES IMS Interface by OTMA using XCF.

The OTMA message is forwarded for transformation and finally delivered to an awaiting TIBCO messaging application. When a reply is required from a TIBCO messaging application, the response is delivered as a normal inbound transaction call except the payload is delivered to the requestor data area.

OTMA Interface

Substation ES IMS Interface, contains a component known as OTMA Interface, which is the logical connection between the OTMA server and Substation ES IMS Interface (OTMA client).

OTMA Interface is a multithreaded OTMA client. Each instance of this interface operates as a subtask of the Substation ES IMS Interface task. You can determine the number of client or worker subtasks by configuring the related parameters at Substation ES startup. OTMA Interface is transparent to users.

OTMA Interface primarily performs the following tasks:

- Establishes a communications environment with IMS, such as allocating and initializing OTMA-required control blocks and Substation ES communication buffers.
- Establishes communications with XCF and connects to the IMS server.
- Joins as a member of the IMS XCF group and initiates a client bid with the IMS control region.
- Notifies Substation ES IMS Interface that initialization is completed and stands by to process requests and events.
- Relays all processing actions to Substation ES IMS Interface for logging, auditing, or both.
- Processes incoming requests by attaching buffered information to an OTMA header and forwarding the messages to IMS.
- Receives replies from IMS applications that participated in a typical request or reply transaction.
- Processes IMS-outbound events by means of an efficient Service Request Block (SRB) processing mechanism.

XCF Services

Substation ES IMS Interface connects to an XCF Group. XCF services register and announce the joining of Substation ES IMS Interface as an XCF member.

XCF, the transport layer, delivers OTMA messages between Substation ES IMS Interface and the OTMA server.

You must specify an XCF group name in the Substation ES startup parameters for connecting to the IMS server at initialization. The group name for Substation ES IMS Interface and for the targeted IMS server must be the same.

The XCF Group is created and owned by IMS. If Substation ES detects that IMS has not been started, Substation ES waits until the IMS region starts successfully and then connects. In this case, Substation ES prints messages to the log file indicating that IMS has not been started.

All members of the XCF group are clients, except IMS, which is the server. Substation ES IMS Interface, acting as a client, communicates with IMS through the XCF interface by sending OTMA messages to IMS. In turn, IMS sends the result set back to Substation ES IMS Interface.

OTMA Server

The OTMA server accepts requests from and returns results to Substation ES IMS Interface. You can make connections to one or more clients by using a server provided in the OTMA environment.

The OTMA server, located within the IMS control region, is supplied by IBM and can accept multiple connections from individual or multiple Substation ES IMS interfaces. Though easily generalized, the OTMA implementation is specific to IMS in an MVS Sysplex environment.

The key to message flow for OTMA is the transaction pipe, the logical connection between the OTMA server and Substation ES IMS Interface. IMS uses the concept of a logical terminal (LTERM) to ensure that responses are associated with the correct requesters. OTMA does not use an LTERM but still must maintain a connection between the client and IMS. This connection is the transaction pipe, or Tpipe. See Tpipes for more information.

Tpipes

Substation ES communicates with the OTMA server through a transaction pipe (Tpipe), similar to the way an LTERM communicates with IMS.

Tpipes are defined to IMS as destinations through the descriptor (D) control card of the IMS DFSYDTx PROCLIB member. The following are some of the characteristics of Tpipes and LTERMs:

- An LTERM or a Tpipe uses a queue where the transaction output is kept before it is returned to the requester.
- For each LTERM or Tpipe, IMS maintains a connection between the queue and the physical node that receives the output.
- Substation ES IMS Interface use Tpipes to associate its transactions with a transaction-pipe name.
- IMS uses the Tpipe name to associate all input and output with the particular Substation ES IMS interface. The association between the transaction output and its ultimate destination (the originating device) is not made within IMS, as is the case with LTERMs, but is handled by the Substation IMS Interface.
- When Substation ES uses a Tpipe, IMS does not know anything about the actual user of the transaction, who is often a user of the IMS application. Therefore, when a Tpipe comes into play:
  - Substation ES IMS Interface has complete control over the output of request or reply transactions.
  - For outbound requests, the IMS application must specify the same Tpipe name as that specified on Substation ES IMS Interface startup parameters.

Each Substation ES IMS interface uses at most two Tpipe names, one for request or reply requests and the other for triggered (IMS outbound) requests.

Tpipe names are supplied in the startup parameters for Substation ES. The names are also included in the message-control information segment of the OTMA message prefix for requests.

Substation ES IMS Interface supports both synchronized or unsynchronized Tpipes. For a synchronized Tpipe, all output messages are serialized through a single process and sequence numbers can be assigned to messages.

IMS OTMA Exits (Optional)

An IMS OTMA exit consists of the Pre-routing exit routine and the Destination Resolution exit routine.

The preferred method of defining OTMA destinations is through the DFSYDTx Descriptor (D) control card. The previous method of using IMS OTMA exits on the M control card can still be used, however, it is not recommended that you use this method from IMS 10 and above. Those exits, supplied by IBM, are as follows:

- OTMA Prerouting exit routine (DFSYPRX0)

  You can use the OTMA Prerouting exit routine to determine and change the destination of IMS OTMA output messages. The messages can be routed to an OTMA client or to IMS TM for processing, but this exit routine cannot determine the final destination for the message.

- OTMA Destination Resolution exit routine (DFSYDRU0)

  You can use the OTMA Destination Resolution exit routine to determine and change the final destination of IMS OTMA output messages. A Substation ES IMS interface sample exit (DFSYDRUS) is provided for use with the sample Substation ES Installation Verification Programs (IVPs).

For more information, see *IMS Exit Routines*.

IMS Exits (Optional)

The IMS exit implements back-end logic for Substation ES during terminal routing processing.

You can use Substation ES with the IMS exit, DFSMSCE0: TM/MSC Message Routing and Control User Exit, to allow for variable length transaction codes, such as those an online terminal specifies.

OTMA Super Members

You can use Substation ES with OTMA super members as targets for regular messages, triggers and requests/replies.

Consider the following points when configuring and using Substation ES with OTMA super members:

- Multiple Substation ES IMS interfaces can point to the same Tpipes under a super member. They can be in the same Substation ES instance or separate ones.
- IMS will process messages sent from an ESB to any Substation ES that references the TPIPE-DEFAULT plus the super member.
- To support conversational messages, either use the DFSYDRUn and DFSYPRXn user exits to handle `TPIPE-PREFIX=`, or define `TPIPE-PREFIX= tpipes` from 001 to 255 with the *SMEM= parameter*.

Note the following points regarding triggers and requests/replies:

- Because of IMS design, only one interface at a time processes triggers and requests/replies; the other interfaces remain idle.
- When an interface terminates, whether planned or unplanned, IMS begins sending traffic to the next interface that becomes active.
- If more than two interfaces are active, IMS will move traffic to the interfaces in the order in which they became active.
- When the first interface becomes active again, it takes its place in line behind the now-active interface.

Message Flow

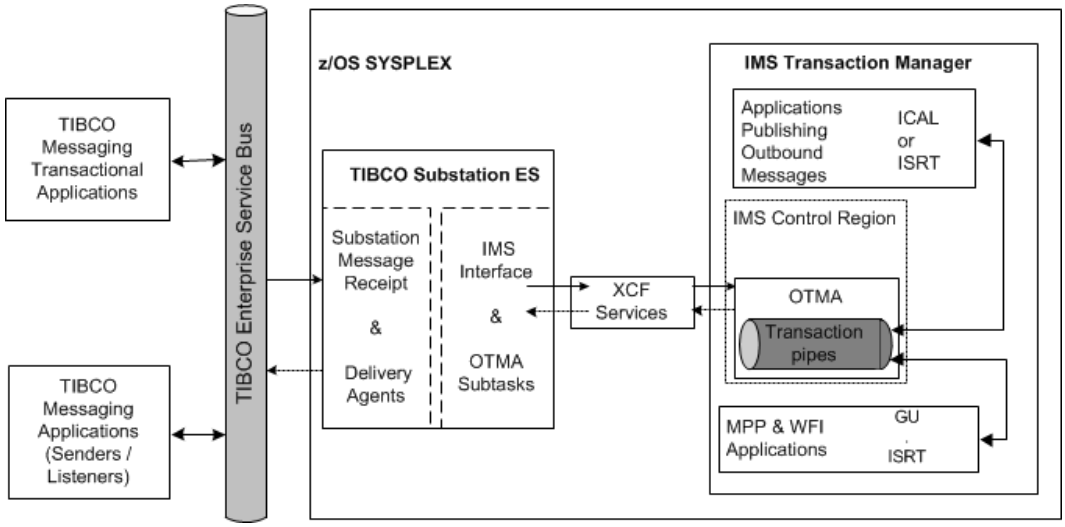
TIBCO application messages are transformed into requests with content and forwarded to Substation ES IMS Interface through XCF to the IMS OTMA server and, finally, to the IMS application for processing.

IMS Interface supports two types of message flow: requests or replies, and triggers. See the following examples for more detailed message flow scenarios:

- External Request or Reply Message Flow
- Triggered Message Flow from IMS Transactions
- Synchronous Callout Message Flow from IMS Applications

Requests or Replies

The following figure shows the message flow of IMS Interface for requests or replies:



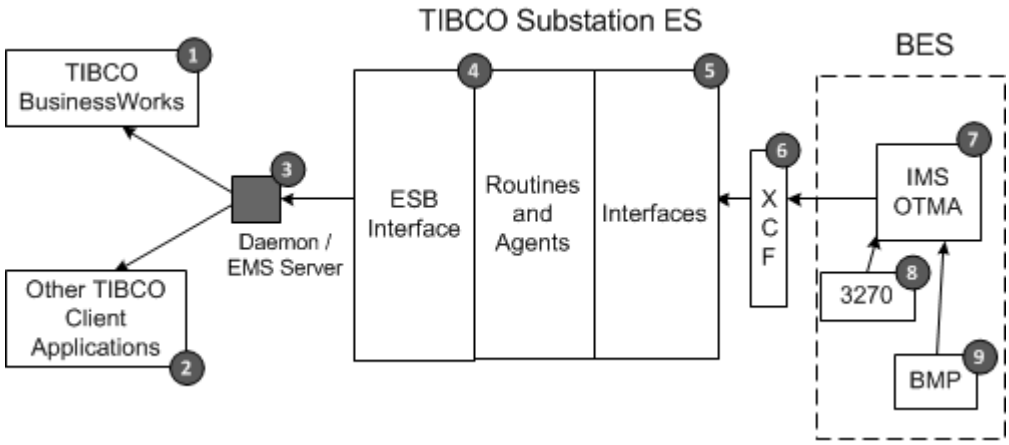
Substation ES accepts TIBCO application messages from the following sources:

- TIBCO messaging applications anywhere in the network.
- TIBCO transactional clients. Substation ES first sends the messages from those clients to the TIBCO Transactional store-and-forward daemons, which notify TIBCO Substation ES transactional agents of the messages to be processed.
- Transactions that run within the IMS BMP and MPP regions. These messages are called triggered messages.
- Other installations of Substation ES through host-side processes or transactions.

For an overview of message flow between messaging applications and Substation ES interfaces, see Substation ES Message Flow.

Triggers

The following figure shows the message flow for IMS triggers:



1. In an IMS trigger, the initial request comes from a BES application in the z/OS environment. In this figure, the request can originate from a BMP (9) or 3270 (8) device, which passes it to IMS OTMA (7).
2. IMS OTMA forwards the message to XCF (6), which sends the message to IMS Interface (5) in Substation ES.
3. IMS Interface forwards the message to ESB Interface (4) and the daemon or EMS server (3).
4. The daemon or EMS server (3) sends the message to either TIBCO BusinessWorks (1) or another TIBCO messaging client (2).

In trigger transactions, no result set or acknowledgment is returned, hence no dotted arrows in this figure.

External Request or Reply Message Flow

Implementing a request or reply solution is usually nonintrusive. You can execute existing host-side transactions and programs without changes.

TIBCO messaging applications can forward and receive messages from z/OS IMS transactions. Substation ES can initiate and execute IMS transactions by using the information received from messaging applications as input to transactions. You can configure Substation ES to determine whether the messaging application expects to receive a result or delivery notification of delivered messages.

The following is the process for a typical request or reply scenario:

1. TIBCO messaging application publishes a message intended for a Substation ES instance with an active subscription for the message.
2. Substation ES receives the message and transforms the contents of the message into a user-configured formatted buffer area.
3. Substation ES sends the buffer along with the control and execution information to Substation ES IMS Interface.
4. Substation ES IMS Interface constructs an OTMA message that contains a prefix and application information (LLZZ and buffer), and sends the OTMA message to IMS on a designated Tpipe.
5. When IMS receives the OTMA message, IMS either executes the command or schedules the applicable transaction for execution.

    - If IMS fails to execute its intended action, a message is normally returned to Substation ES IMS Interface. The message is logged and the messaging application receives an unsuccessful return code and message.
    - If IMS succeeds in the execution, the message that is created as a result of the IMS processes is returned to Substation ES IMS Interface on the originating Tpipe. By using one of the

appropriate OTMA or IMS exits, the destination name (Tpipe) can be altered by IMS applications or IMS exits.

6. Substation ES IMS Interface extracts application data from the OTMA message and returns the data in the buffer area to ESB Interface.
7. The buffer received from the host-side interface is transformed into a reply. The format, type, and destination of the message intended for the TIBCO messaging applications are configurable within Substation ES.
8. The message is published to TIBCO messaging applications based on the level of service and destinations configured on the transformation recipe.

Triggered Message Flow from IMS Transactions

Implementing a triggered message flow is normally intrusive. If you want to use an IMS application to write messages to IMS queues with an alternate IOPCB, you must first add code to the application logic.

However, if output from an application only needs to be redirected to Substation ES, the redirection of the message can be accomplished by means of IMS OTMA exits.

The following is the process for a typical flow of a triggered message:

1. An IMS application inserts a message to the IMS message queue, specifying the trigger Tpipe name.
2. Substation ES IMS Interface is notified of the messages by XCF and consumes the information from the IMS queues through the OTMA protocol.
3. If the message is valid and recognized by Substation ES IMS Interface, the message is placed on a Substation ES inbound queue, making the request eligible for transformation.
4. After transformation, ESB Interface sends a message formatted for TIBCO messaging applications, and the message is sent to Substation ES Delivery Agent.

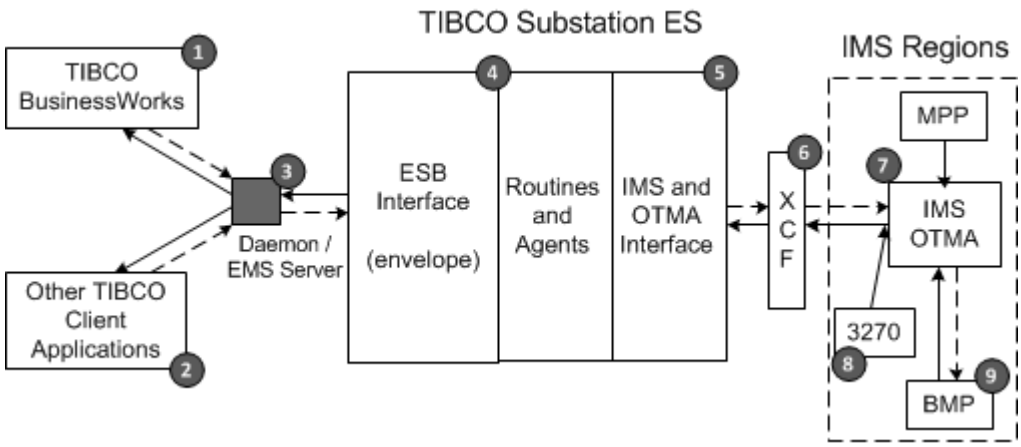
   If transformation fails, and the message is sent using a guaranteed service, the message is returned to an IMS Tpipe named SXITRNAK.
5. ESB Interface publishes the message to TIBCO messaging applications based on the level of service and destinations configured on the transformation recipe.

Synchronous Callout Message Flow from IMS Applications

Implementing a callout is normally intrusive. If you want to use IMS application to send request messages using ICAL to the awaiting Substation ES process, you must first add code to your application logic.

Synchronous callouts do not use the IMS queues, so the normal rules for trigger or outbound processing do not apply. A sample COBOL program is supplied with the product installation.

The following figure shows the message flow for IMS synchronous callout:



The following is the process for a typical flow of a triggered message:

1. An IMS application (8, or 9) executes an ICAL call with a request and a response area.
2. After Substation ES IMS Interface (5) is notified of the message from XCF (6), it consumes the information from OTMA (7).
3. If the message is valid and recognized by Substation ES IMS Interface (5), the message is placed on a Substation ES inbound queue, and the trigger service is selected by ESB Interface (4).
4. ESB Interface (4) formats the message as specified on the trigger service, which is then sent to Substation ES Delivery Agent. If formatting or delivery fails, a NAK is sent back to IMS. Substation ES envelope information is also inserted as a property or field within the message.
5. ESB Interface (4) publishes the message to the ESB based on level of service and destination defined in the trigger service.
6. TIBCO messaging application publishes the reply message to the originating Substation ES instance with an active subscription for the reply message. The Substation ES envelope information must be returned unchanged with the reply.
7. Substation ES receives the reply message and transforms the contents of the message into a user-configured, formatted buffer area.
8. Substation ES sends the buffer, along with the control and execution information, to Substation ES IMS Interface (5).
9. Substation ES IMS Interface (5) constructs an OTMA message that contains a prefix and application information (LL and buffer) and sends the OTMA message to IMS on a designated Tpipe.
10. When IMS receives the OTMA message, it places the response in the application response area specified in the ICAL call.

Communication Service Levels

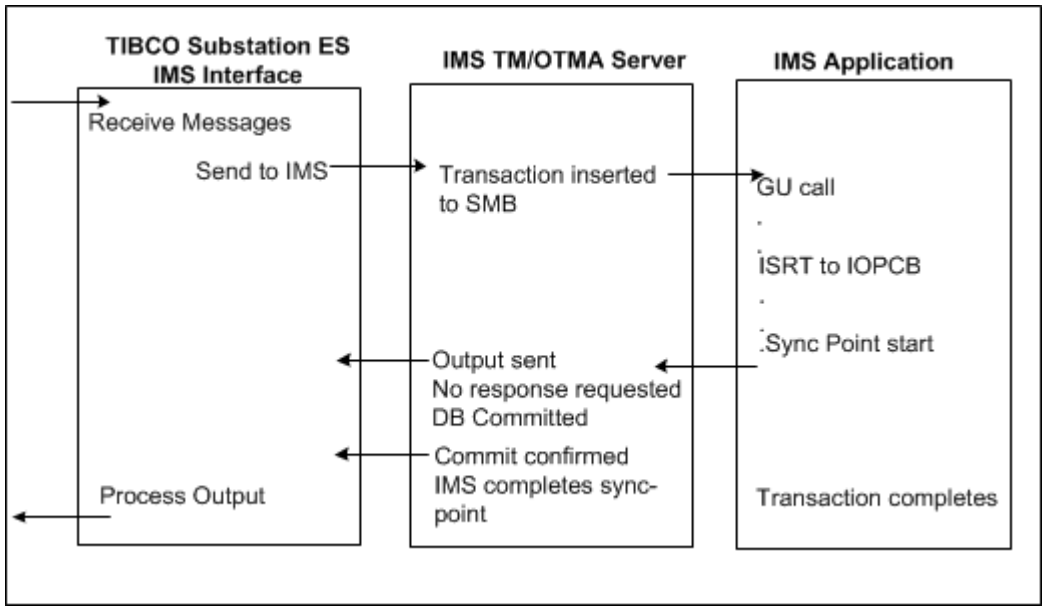
You can set different communication service levels as appropriate for communication between Substation ES IMS Interface and IMS applications.

The following service levels are supported for communication between Substation ES IMS Interface and IMS applications:

- Reliable
- Guaranteed

Reliable

The following figure shows the message flow for the reliable communication service level:



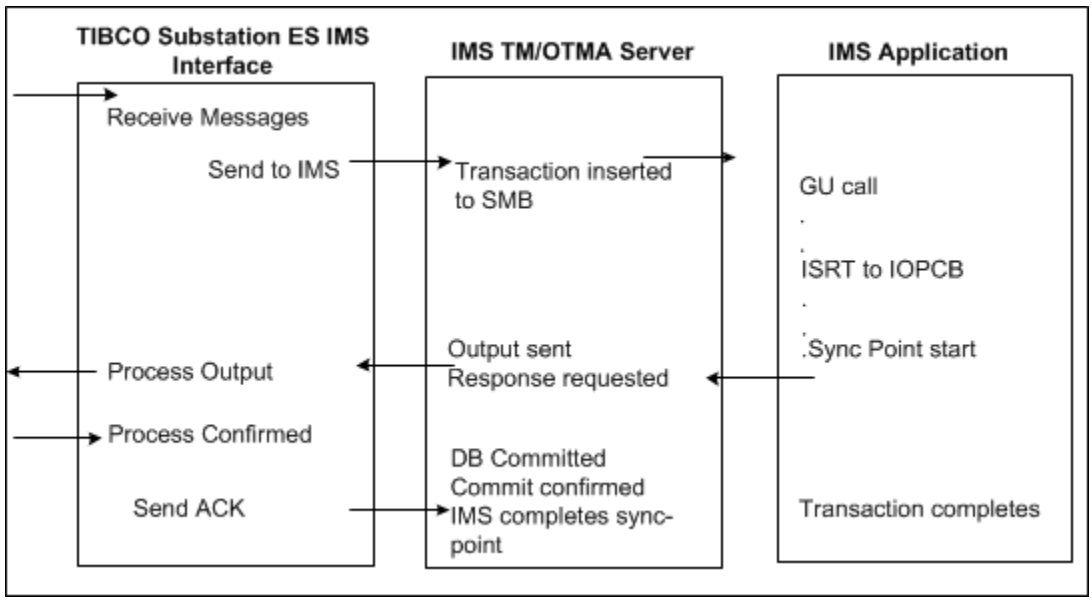
For the execution of IMS transactions in reliable mode (send-then-commit, known as Commit Mode 1), the following rules apply:

- IBM Resource Recovery Service (RRS) is not required to complete a request with Substation ES.
- Transaction pipes need not be synchronized.
- The synchronization level is specified as `None` in the OTMA message prefix.
- Message receipt from IMS by Substation ES IMS Interface is in random order because XCF does not guarantee ordering.

Currently, send-then-commit with confirmation is not supported.

Guaranteed

The following figure shows the message flow for the guaranteed communication service level:



For the execution of IMS transactions in transactional mode (commit-then-send, known as Commit Mode 0), the following rules apply:

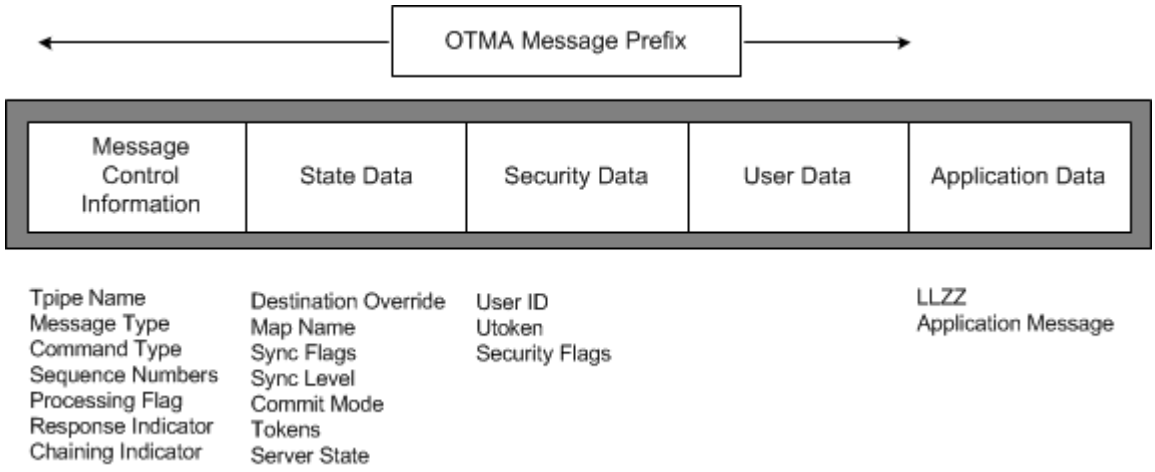
- You must set the Substation ES service level for a particular recipe to `guaranteed`.
- The synchronization level is set for the Tpipe. IMS maintains sequence numbers for recoverable input and output for the Tpipe.
- Acknowledgment is always requested by both IMS and Substation ES IMS Interface.
- TIBCO messaging applications must publish the messages with the TIBCO application transactional protocols.

If Substation ES and Substation ES IMS Interface must initiate protected conversations, Substation ES acquires and owns the unit of work (UOW) context identification. The UOW context is passed to Substation ES IMS Interface and provides the context ID in the State Data segment of the OTMA message prefix.

OTMA Messages

The communication data passed between the IMS Interface and the OTMA Server is within an OTMA message.

The following figure shows the layout of Substation ES IMS Interface OTMA messages:



An OTMA message contains the following segments:

- **Prefix**

  The prefix contains one or more of the following components:

  - A Message Control information segment is provided for every message.
  - The State Data segment contains transaction-related information and has different formats.
  - The Security Data segment is mandatory for each transaction or command.
  - The User Data segment is of variable length and is managed by the IMS OTMA Interface.

- **Application data**

  The application data can be an IMS command or a transaction. The data in this segment is unchanged by the receiver (OTMA Server) and is transmitted directly to an application that can obtain or send the data in the I/O area with DL/I calls.

For details on OTMA message segments, see *IMS Communications and Connections*.

Message Size and Constructs

IMS Interface messages include OTMA messages and application messages. Note the size of these messages.

Note the following on the size of the IMS Interface messages:

- OTMA messages

  Messages that enter IMS from OTMA contain both the OTMA message prefix and other defined IMS message prefixes. Excluding the user data section, the OTMA message prefix can become very large, sometimes over 200 bytes in length. The OTMA message prefix, including the user data section, is stored on IMS message queue data sets, which increases the usage of the queue buffer pool.

- Application messages

  The current maximum size of a Substation ES message passed from and to IMS regions and applications is 3 MB.

The following rules pertain to IMS applications:

- You can execute existing Fast Path application programs with OTMA. They must run as send-then-commit transactions.
- Parameters with the OTMA transaction that contradict this commit mode cause the transaction to be rejected.
- IMS translates in uppercase. To preserve lowercase characters, specify the ULC keyword while defining the transaction.

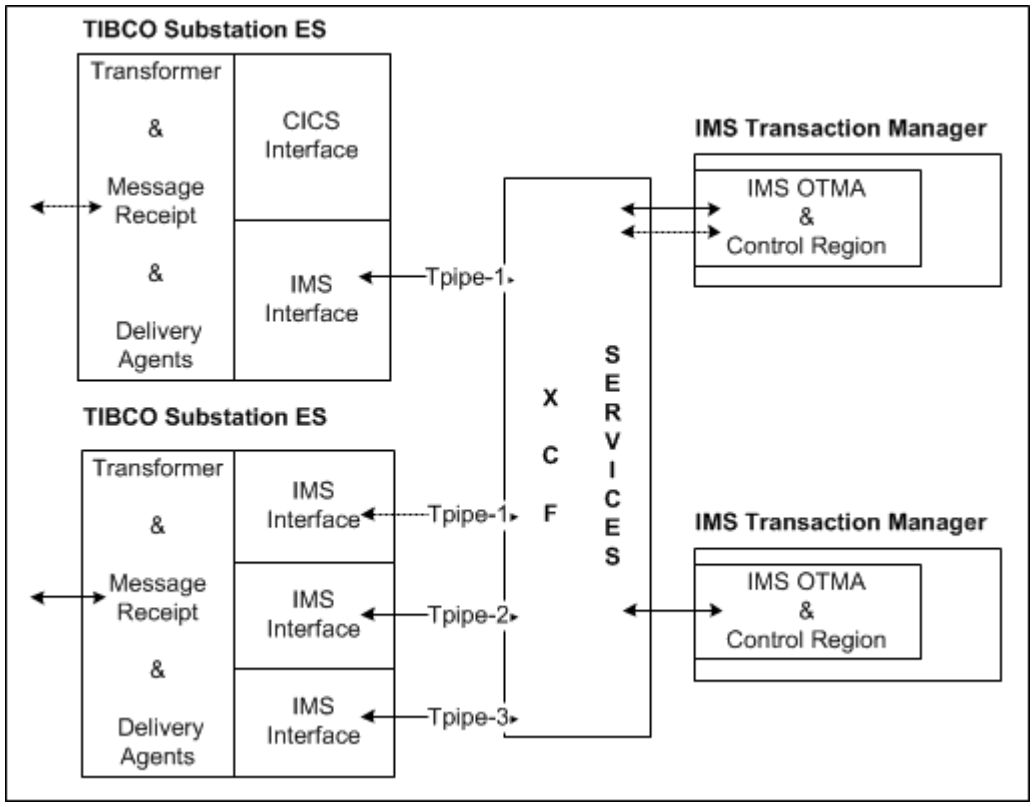
Deployment

You can deploy Substation ES and IMS Interface in various ways to meet system, application, and workload requirements.

The following are the possible configurations:

- A Substation ES region can start and communicate with various unrelated host-side interfaces. For example, a CICS and IMS interface can be active within a single Substation ES region.
- The IMS control region and Substation ES can each execute in the same or an adjacent LPAR images as long as XCF communicates with both LPARs.
- Substation ES can communicate with many IMS interfaces and, therefore, with multiple IMS control regions.
- If multiple IMS interfaces are started for the same Substation ES instance, each IMS interface can communicate with the same or different IMS control regions.

The following figure shows how you can deploy IMS Interface:



Constraints

Note the constraints of Substation ES IMS Interface.

The following are a few constraints of Substation ES IMS Interface:

IBM OTMA API

Substation ES Interface does not use the standard IBM supplied OTMA API, primarily because OTMA Callable Interface (C/I) does not support the commit-then-send option and resynchronization feature of the IMS OTMA protocol.

IMS DFS$$TOQ

Starting with IMS 11, timeout queues are now supported for outbound Tpipe messages. The default timeout queue is DFS$$TOQ.

Substation ES does not use this Tpipe, so you might have to create procedures to handle messages that might end up on the DFS$$TOQ queue.

Substation ES IMS Interface

Note the software requirements, naming conventions, and restrictions of Substation ES IMS Interface:

- Software requirements

  To initialize and install Substation ES IMS Interface, you must have z/OS 1.8 or above and IMS 8.1 or above installed.

  To use request/reply transactions, which require the IMS synchronous callout (ICAL) feature, IMS 10 or above is required. Note that IMS 10 might also require implementation PTFs.

- Naming conventions

OTMA places a restriction on the naming of clients and transaction pipes. For a list of the naming conventions you must adhere to, see *IMS Communications and Connections*.

- Restrictions

  Currently, the following restrictions apply for all versions of Substation ES IMS Interface, unless otherwise noted:

  - The commit-then-send option of IMS OTMA is not supported.
  - The message resynchronization feature of IMS OTMA is not supported.
  - Beginning with Substation ES 2.7, IMS outbound messages might now be multi-segmented.

RED (XCF) Interface

Substation ES RED Interface enables external applications to communicate (publish, request, or reply) with z/OS programs and transactions that are located in the batch, CICS, IMS BMP, IMS MPP or TSO regions.

Overview

TIBCO messaging applications are normally the source of the external messages and requests that are received by Substation ES.

However, other providers for message distribution that are suitable for the transfer of large and high volume instant messages are Substation ES to Substation ES, TSO and batch communications internally in the mainframe by way of the XCF layer.

TIBCO Mainframe RED is the product that enables XCF communications. Applications that use this natively need to use its API to enable message send and listen activities. Using Substation ES RED Interface enables applications to communicate (publish, request, or reply) with z/OS programs and transactions that are located in batch, CICS, IMS BMP, IMS MPP or TSO regions without having to be fully knowledgeable in XCF configurations or have specific knowledge about the protocols and their usage. The basic Mainframe RED API usage is needed only for batch and TSO programs.

The Substation ES RED Interface is coupled to Substation ES's host side, enabling multiple services to asynchronously communicate with each other. When CICS applications need to publish or subscribe messages, applications can continue the use of CICS TDQ and TSQ facilities. Environments should be cautious not to use the RED interface to store messages in XCF that persist for long periods such as days. Options such as storing a message for a lengthy duration should be considered only if the environment is configured to have messages stored in auxiliary storage such as DASD.

The main value of Substation RED Interface is that it provides applications with a low-latency communication option in a true multi LPAR single SYSPLEX environment. The RED interface takes the most efficient direct route between applications, and it does away with all other options. If correctly configured and used, the RED interface reduces the use of resources, minimizes the moving parts, lowers the MSU usage and lowers the TCO for its users.

Architecture

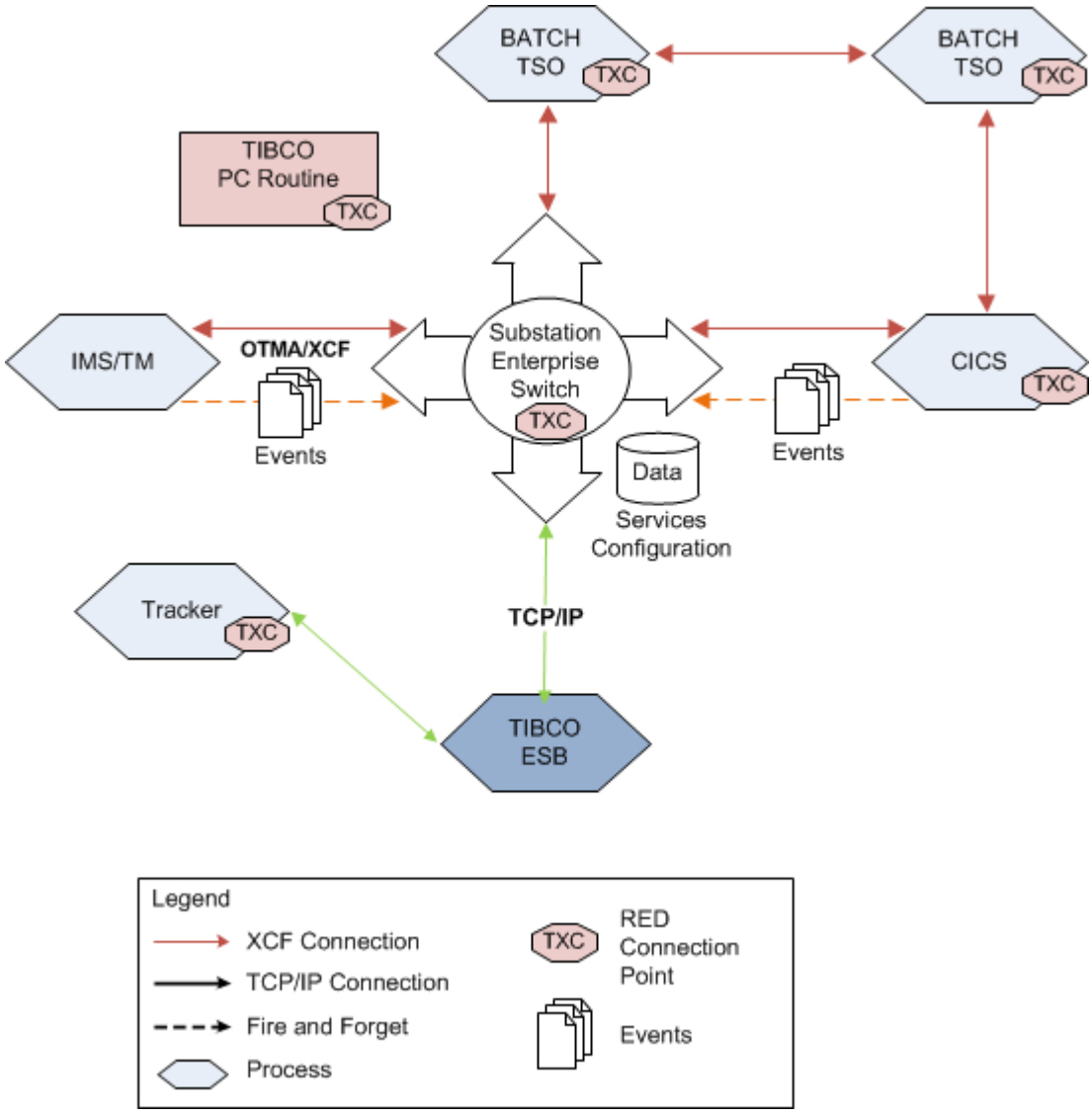
Substation ES RED Interface enables external applications to communicate (publish, request, or reply) with z/OS programs and transactions that are located in in batch, CICS, IMS BMP, IMS MPP or TSO regions.

Additionally, these z/OS applications can publish or redirect (trigger) output to external applications through the Substation ES RED Interface.

The Substation ES RED Interface is coupled to the Substation ES host side, enabling multiple services to simultaneously connect and communicate with each other bi-directionally. To fully understand the role and position of the Substation ES RED Interface within z/OS, refer to the Architecture diagrams and other topics in this chapter as well as the TIBCO Mainframe RED guides.

The Substation RED Interface uses Substation's internal service capabilities, the ESB communications as well as TIBCO Mainframe RED that provides users with an extremely efficient, low latency, high throughput and easily manageable environment for applications to converse with each other. This all together allows mainframe and ESB services to participate in a reliable or transaction-based, connectionless client-server protocol.

The Substation RED Interface does not replace the existing CICS or IMS Transactional interfaces but adds value, and provides performance and flexible features on top of those existing interfaces. It also allows batch and TSO programs to easily participate in z/OS to z/OS communications or bi-directionally in a z/OS to Enterprise Service Bus environment. The following figure shows Substation ES and TIBCO Mainframe RED architecture:



Components

Substation ES RED Interface consists of the following components:

Components	Description
RED Event Interface	RED Interface establishes a RED environment and handles all outbound events and triggered messages.
Mainframe RED	An execute-only library of Mainframe RED is supplied with Substation ES.
RED API	The Mainframe RED API can not be used with components and applications that do not communicate with Substation ES unless you have installed TIBCO Mainframe RED.

Components	Description
RED HUB	An authorized module is provided to enable cross partition (LPAR) execution and access to use XCF List structures during communications. One task for each LPAR is required to be executing.
XCF Exits (Optional)	A number of XCF system exits are provided to handle configuration, communication defaults and exception processing. For most users, the default XCF exit is optimal and no changes are required.
Samples and Configuration Definitions	RED Interface has its own configuration definitions and sample communication programs supplied for languages such as COBOL, Assembler and C.

Message Transfer Services for Applications

Within the scope of Substation ES, messages, events, and requests are synonymous.

Substation ES provides three basic types of message transfers when using RED Interface:

- Synchronous Triggers
- Synchronous Message Transfer
- Asynchronous Message Transfer

Synchronous Triggers

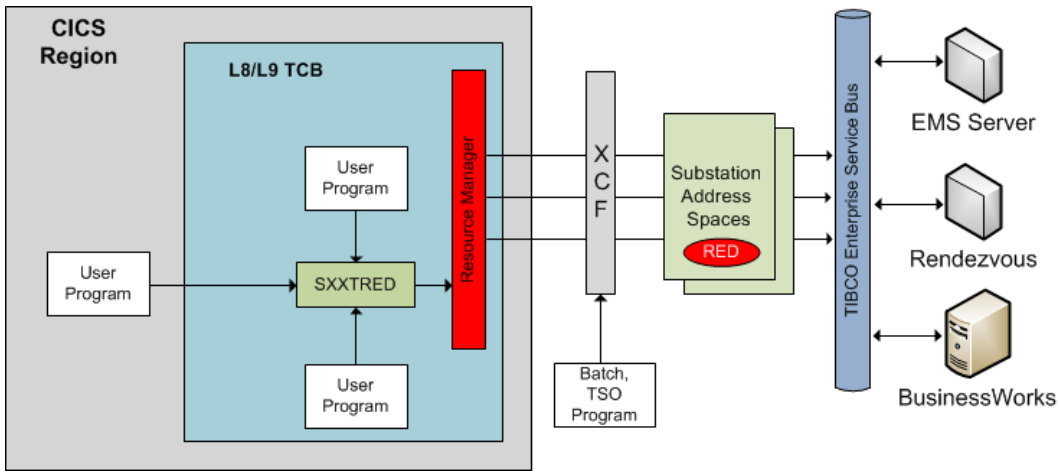
The synchronous triggers transfer starts from mainframe applications.

Consider using the synchronous triggers method in the following scenarios:

- When applications are required to send large messages or multiple single messages.
- When batch programs and CICS transactions publish continuously for a few seconds or minutes.
- When sending a message to another application.

The user application would request a dedicated listener from Substation ES and use that listener to send messages until the user application signals the end of conversation or an inactivity timeout occurs.

The following figure shows the message flow of the synchronous triggers service:



Synchronous Message Transfer

Use the synchronous message transfer method to transfer large messages (fire and forget), or initiate request/reply sequences to other applications or external services within the reach of ESB.

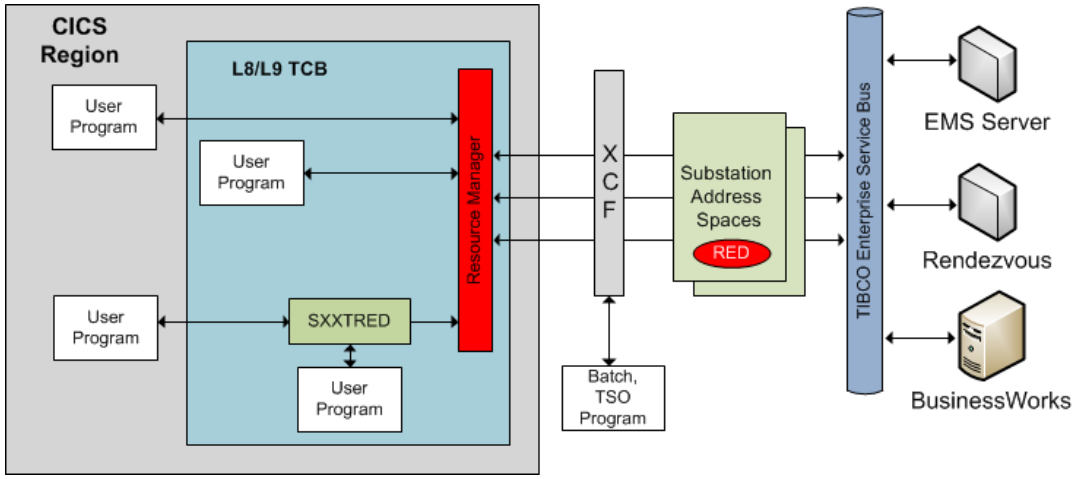
The synchronous message transfer method is typically used for request/reply scenarios. You can also use this type of message transfer for low or medium volume fire and forget scenarios.

Also, consider using the synchronous message transfer method in the following scenarios:

- When applications infrequently send large messages or are required to send messages synchronously.
- When CICS and IMS transactions publish events every few seconds apart .
- When sending a message to another application.

The user application would receive a dedicated listener from Substation ES to publish the messages on until it receives a signal from the sender or user application that a message is the last message to be sent or received, or until an inactivity timeout occurs.

The following figure shows the message flow of the synchronous message transfer service:



Asynchronous Message Transfer

Use the asynchronous message transfer method to immediately and very quickly transfer messages by way of Substation ES and Mainframe RED XCF List facility to other applications or external services within the reach of ESB.

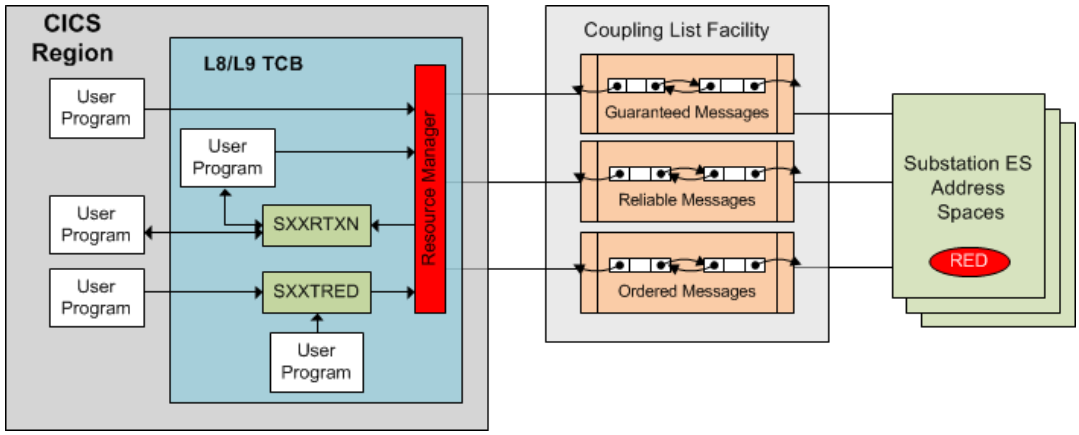
Use the asynchronous message transfer method for medium to high volume real-time messaging.

Also, consider using the asynchronous message transfer method in the following scenarios:

- When applications frequently send messages or send messages within a transactional context that are required to be committed or rolled back depending upon the transaction executions status.
- When the CICS and IMS BMP transactions publish messages and events very frequently.

User applications send messages using SXXTRED or receive messages by way of the SXXRTXN program, which can either relay or receive transaction information. Substation ES still uses the EXCI protocol for administrative functions and status updates, so only user related information is transferred using the XCF List facility.

The following figure shows the message architecture of asynchronous message transfer services:


CICS
Region
L8/L9 TCB
User
Program
User
Program
User
Program
User
Program
SXXRTXN
SXXTRED
User
Program
Resource Manager
Coupling List Facility
Guaranteed Messages
Reliable Messages
Ordered Messages
Substation ES
Address
Spaces
RED

Advanced Deployments

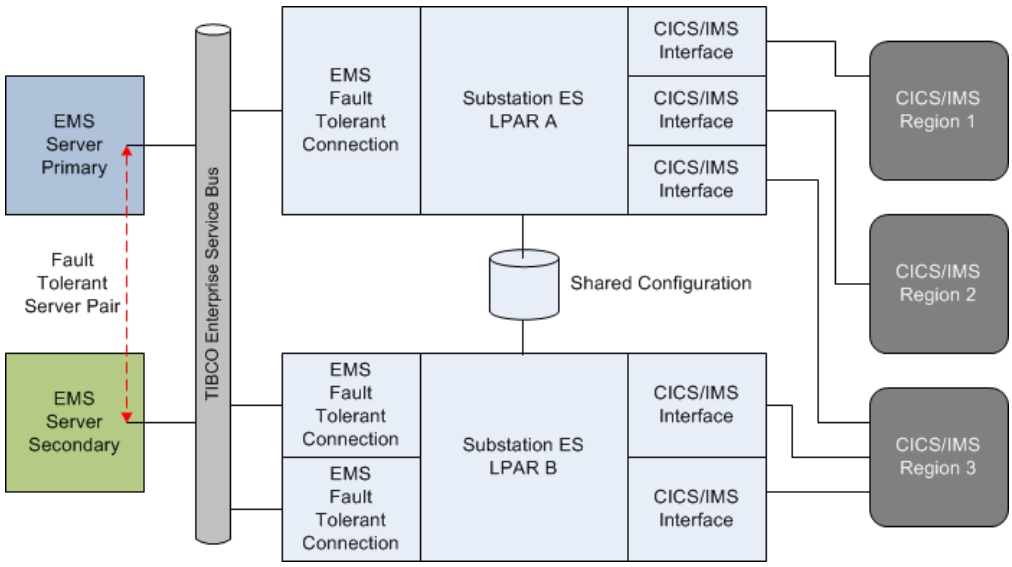
With advanced deployment, you can deploy Substation ES to use the fault tolerance and load balancing features, deploy Substation ES in a z/OS Sysplex environment, and configure Substation ES to interact with CICS in various ways.

Fault Tolerance with Load Balancing

You can configure Substation ES to talk to one or more Enterprise Message Service servers, a Fault Tolerant (FT) Enterprise Message Service pair, or both, and configure each of them separately within a single address space.

Substation ES, which runs as a separate address space in z/OS, communicates with Enterprise Message Service servers and hence operates as an Enterprise Message Service client.

The CICS and IMS interfaces enable Substation ES to talk to one or more BESs, making for many possible configuration permutations as shown in the following figure:



Configuration Tips

The configuration tips provide information about how you can configure Enterprise Message Service servers, Substation ES address spaces, and so on to achieve a high availability environment.

A pair of Enterprise Message Service servers working in parallel provides a single Enterprise Message Service server view with no loss of service if either server fails. You can place the Enterprise Message Service servers on different subnets so that failure of one subnet does not cause loss of service.

Configure multiple Substation ES address spaces in different LPARs sharing the same service configuration, each listening to the same Enterprise Message Service queues. If Substation ES A gets a message from an Enterprise Message Service queue and does not respond with a reply, as would happen if Substation ES A or LPAR A fails, the Enterprise Message Service server retains the message on the queue and gives it to the next “listen” from Substation ES B. From the business view, there is no loss of service.

The CICS regions can be Application-Owning Regions (AORs), Terminal-Owning Regions (TORs) with AORs behind them, or TORs in a CICSPlex. The following are the interface scenarios:

- If Substation ES is interfacing with AOR CICS and the CICS address space fails (not the transaction, but the address space), Substation ES disables all the recipes associated with the CICS address space and stops listening to the queues defined in the recipes.

If other recipes are in the Substation ES A or B define processing of a queue for another CICS address space, then that process continues to handle messages.

- If Substation ES is interfacing with TOR CICS or TOR in a CICSPlex, then you must build fault tolerance into the CICSPlex environments.
- Substation ES supports the notion of primary and alternate executions when services communicate with transactional management systems. When the primary system is unavailable, Substation ES automatically routes messages to the secondary systems. Once the primary system is available again, execution is routed through to the primary system again. This feature increases flexibility and reliability for the CICS or IMS interfaces and adds further dynamics to Substation ES high availability environment.

Items to Consider

Deploy Substation ES address spaces for an ACTIVE-ACTIVE setup and use a CICS TOR-AOR setup or CICSPlex to handle CICS dynamic routing.

Both Substation ES address spaces process messages. This configuration is an ACTIVE-ACTIVE setup. Substation ES does not support an ACTIVE-PASSIVE setup, where one Substation ES does all the processing and the other assumes control in the event of an outage.

Currently, Substation ES does not handle CICS dynamic routing. Even though recipes are dedicated to a BES, the ability to override the BES designation by specifying properties or information on the input message is possible. With a CICS TOR-AOR setup or CICSPlex, you can dynamically route CICS transactions and programs through Substation ES.

Load Balancing with Enterprise Message Service

Load balancing is typically achieved by combining Substation ES internal dispatching features with Enterprise Message Service load balancing techniques.

Internally, Substation ES has its own dispatching based on the **`MAXUOW`** configuration parameter and on the service payload. The **`MAXUOW`** parameter specifies the maximum concurrent activities Substation ES can perform at any point. Together with Enterprise Message Service load-balancing servers and connection factories, that is the way load balancing is typically performed. Overhead for some of those operations occurs where the payload for a service originates.

Limitations of EMS-Substation ES Load Balancing

Beware of the following two limitations of the Enterprise Message Service-Substation ES load balancing techniques:

- Do not specify load balancing in situations with durable subscribers.
  - If a client program that creates a durable subscriber, connects to Server A through a load balanced connection factory, then Server A creates and supports the durable subscription.
  - If the client program exits, restarts, and connects to Server B, then Server B creates and supports a new durable subscription. However, the pending messages on Server A remain there until the client reconnects to Server A.
- Do not specify load balancing if your application requires strict message ordering. Load balancing distributes the message load among multiple servers. This is a practice that inherently violates strict ordering.

Limit of Resources Consumed by Substation ES

Each Substation ES interface contains a worker parameter that represents the number of concurrent tasks it dispatches to perform activities for its related work. In CICS Interface, the worker parameter is associated with concurrent connections to a CICS region. Specify the appropriate number of workers for the resources available and the workload Substation ES must perform. A value between 10 and 25 is usually adequate.

Substation ES also provides a "limit" value at the Recipe Service level. When set, this limit specifies the maximum number of instances of that service that Substation ES permits in a single ESB interface. With this feature, you can easily control concurrency and cap resource usage when executions are scheduled to go to DB2 or other back-end intensive applications.

Monitoring of Transaction Execution and Queues

In case of a shortage of specified workers or if resources are not available to execute transactions, requests are queued internally and the **`MAXUOW`** limit is eventually reached. Internal Substation ES logic slows down in the asynchronous input, causing a backlog and issuing appropriate STRESS level messages. Run the operational command **`SHOW,UOW`** to determine if any excessive queue waits have occurred.

Sysplex

Sysplex is a set of z/OS systems that communicate with each other through certain multisystem hardware components and software services.

The architectural operations of Substation ES comply with the Sysplex environment but do not use the Sysplex features. For example, the Sysplex load balancing features are redundant for Substation ES, because load balancing is achieved within the configuration and operations of Substation ES communications together with the Enterprise Message Service architecture.

Enterprise Message Service supports the message queue and broadcast topic messaging structures, which are represented here in simplistic terms:

- Queues are used for higher latency applications. A message queue is exactly as it sounds: a series of messages held in a queue. Queues can be persistent or nonpersistent.
- Topics are generally used for broadcast, administrative, and low latency applications. A topic is type of message that is broadcast to all the clients who are listening to the particular topic or who have expressed interest in listening to a particular topic (known as durable subscribers). Topics are always nonpersistent, except in the case of a durable subscriber, where the message is kept until the subscriber consumes the message.

For more information about queues and topics, see the appropriate Enterprise Message Service documentation.

Substation ES supports these messaging techniques, which are activated by dynamic configuration. However, if multiple Substation ES instances share the same configuration file (repository), note the following considerations:

- **Queues**

  Where multiple Substation ES instances are picking up messages from a single queue, each message is consumed only once by an instance of Substation ES.

- **Topics**

  Topics differ from queues in that a message to a topic is processed by all of the consumers currently active in all connected Substation ES instances. Where there are multiple BESs and each must process the same message, that practice is obviously advantageous. However, where there is a single CICSPlex, the same message might be processed multiple times.

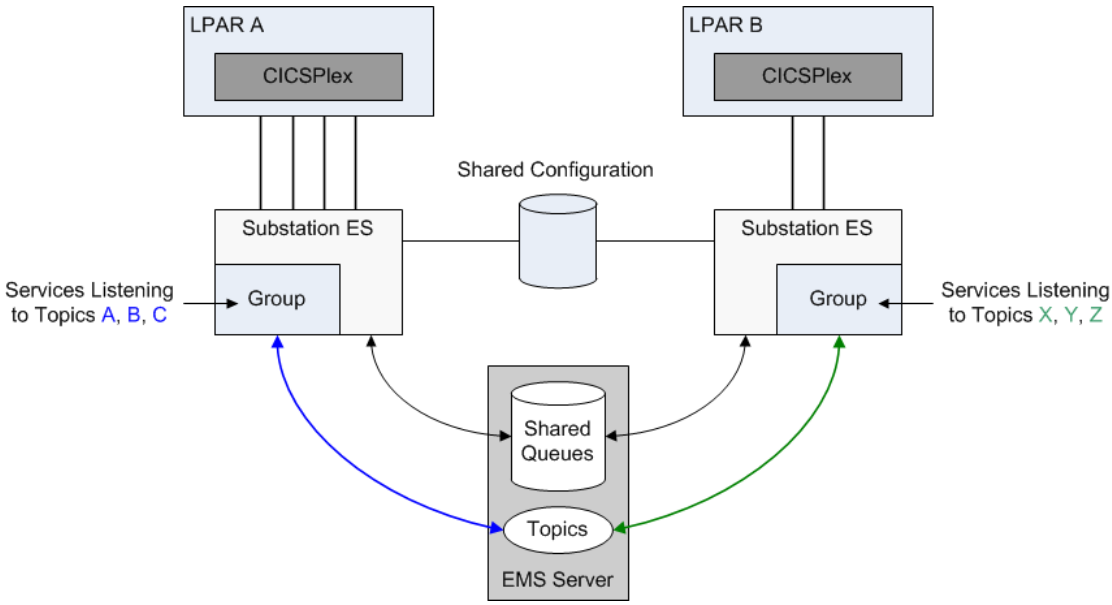
  The group construct within Substation ES provides a way for a single connection service to consume topic messages in a shared configuration environment.

Where there is a Sysplex and a CICSPlex across multiple LPARs, a common approach is to have multiple Substation ES instances processing messages with a common configuration file. You can configure these Substation ES instances in two ways: both active, each processing only its own messages; or one running and the other a standby, configured but not running. In case of a failure, the standby processes the messages that were destined for the failed Substation ES instance.

During a failover or restart of a Substation ES instance, messages that are produced through topics are not lost if the Substation ES consumer service is registered as a durable subscriber. Topic consumers that are not durable receive messages only if they are enabled to receive them.

CICSPlex Environment

The following figure shows a sample configuration of Substation ES in a CICSPlex environment:

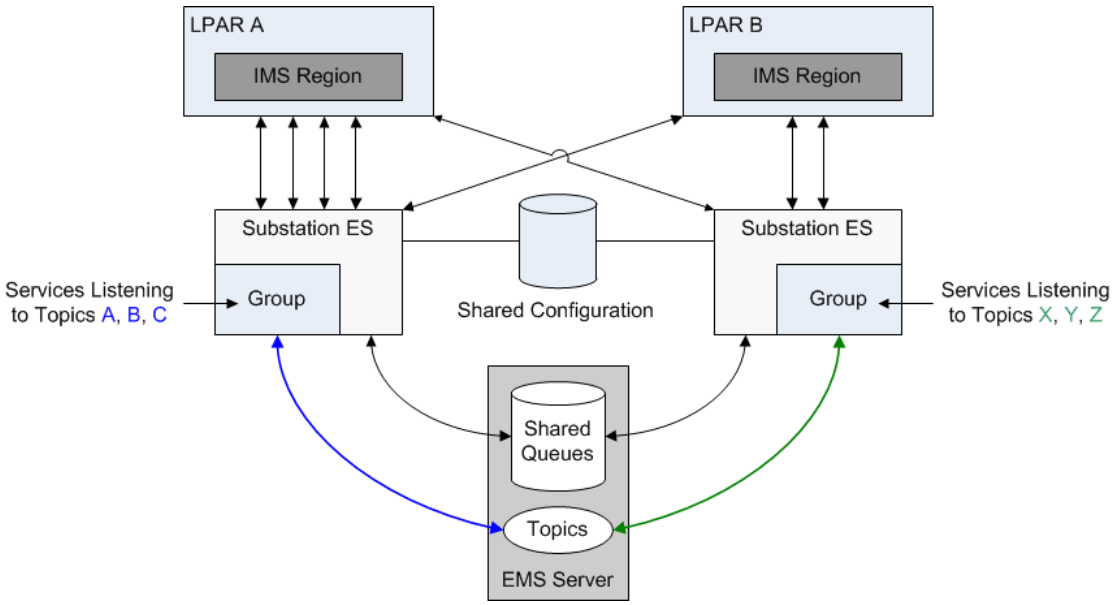


Interruption of transactional and event-driven processing usually takes less than 10 seconds during a Substation ES switchover, assuming that all the components to which Substation ES connects and with which Substation ES communicates are running and available for processing.

For more information, see the section “Substation ES in a CICSPlex Environment” in *TIBCO Substation ES Operations and Administration*.

IMS Sysplex Environment

The following figure shows a sample configuration of Substation ES in an IMS Sysplex environment:



Interruption of transactional and event-driven processing usually takes less than 10 seconds during a Substation ES switchover, assuming that all the components to which Substation ES connects and with which Substation ES communicates are running and available for processing.

Multi-CICS Environment

Substation ES interacts with CICS in many ways. The CICS features that Substation ES uses depend on how the CICS is configured.

Other considerations pertain to the applications with which Substation ES interacts and depend on whether they conform to CICSPlex.

Substation ES can operate in a CICSPlex in many scenarios, which do not depend on the features of Substation ES itself, but take advantage of the fact that Substation ES is simply an External Call Interface (EXCI) client that is using MRO capabilities. That way, Substation ES can leverage a CICS system configurations.

For more information about CICSPlex, see *TIBCO Substation ES Operations and Administration*.

For the transactions that have some form of application AOR affinity, configure Substation ES as a standalone CICS system. For the transactions that can run in a CICSPlex environment, the following are some of the scenarios that exist:

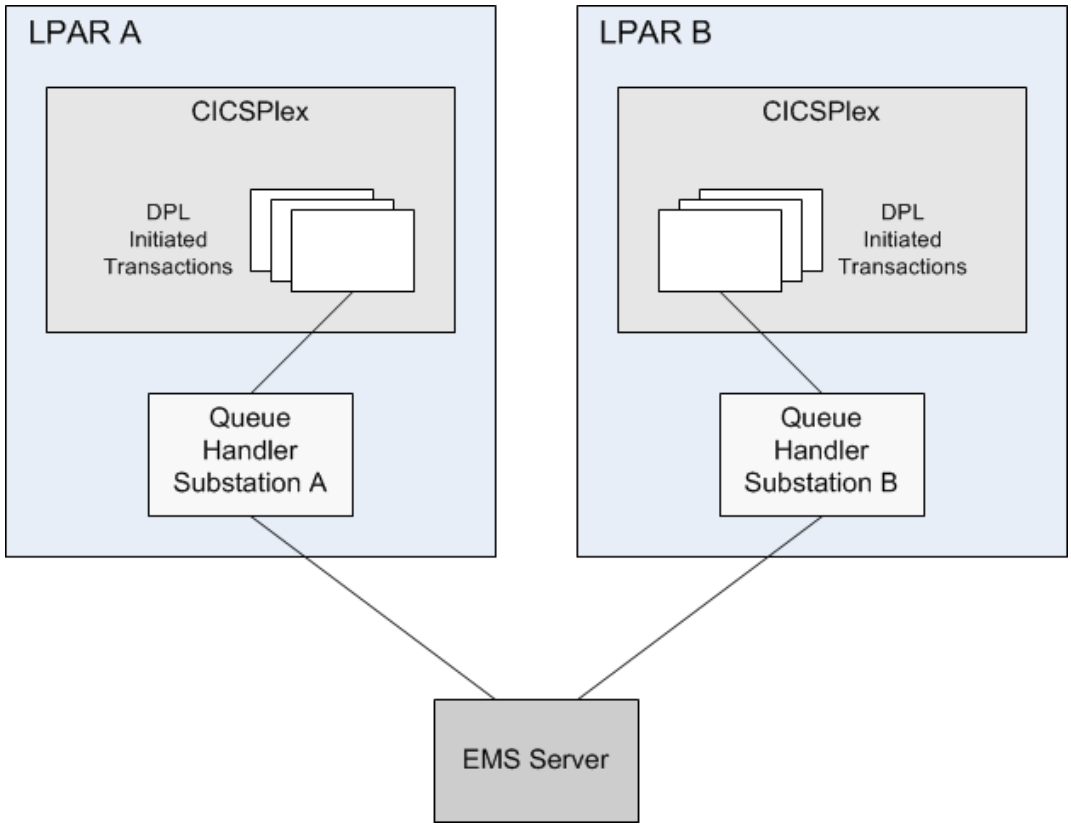
- Externally Initiated Request or Request/Reply
- Externally Initiated “Fire and Forget” (External Transaction Starts)
- Externally Initiated “Fire and Forget” to TD or TS Queue
- CICS to TD Queue (CICS-Initiated “Fire and Forget”)
- CICS Application-Initiated Request Reply

Externally Initiated Request or Request/Reply

In the externally initiated request or request/reply scenario, a Substation ES is in each z/OS LPAR, and each Substation ES instance uses Enterprise Message Service messaging by means of queues; connections to CICS use the IBM EXCI facility.

Any active Substation ES instance can pick up messages from the server for processing. If a Substation ES instance or LPAR fails, a second Substation ES instance continues to process messages from the queues.

The following figure shows the externally initiated request or request/reply scenario:



Externally Initiated “Fire and Forget” (External Transaction Starts)

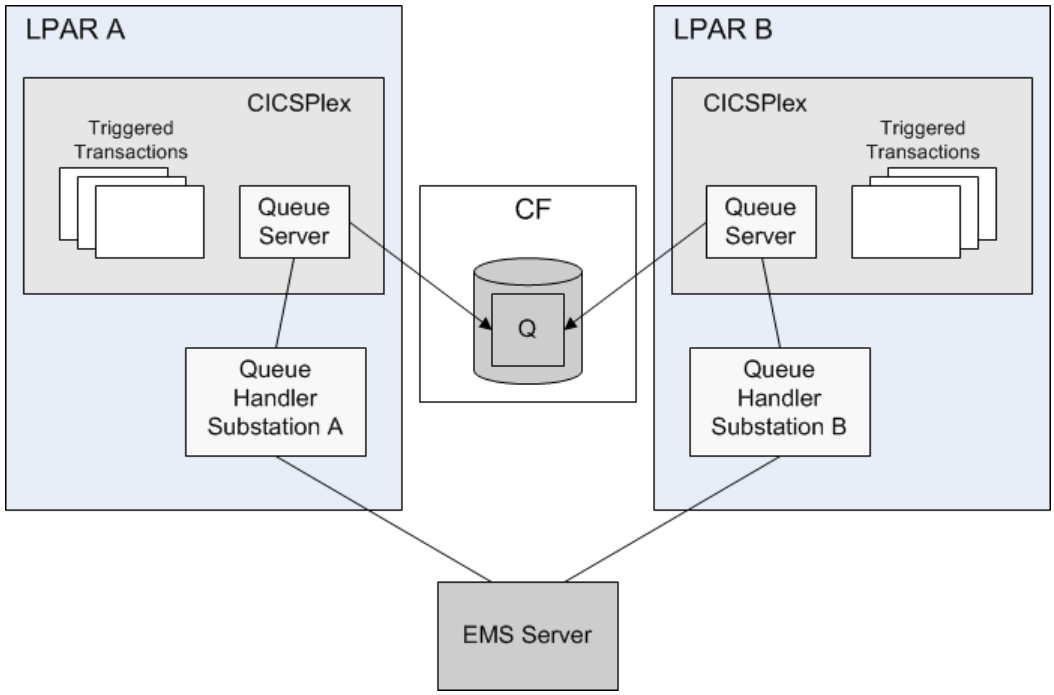
The externally initiated “fire and forget” scenario shares the same architecture with the externally initiated request or request/reply scenario. CICS transactions are invoked with a **`CICS START`** function call.

For more information, see Externally Initiated Request or Request/Reply.

Externally Initiated “Fire and Forget” to TD or TS Queue

The externally initiated “fire and forget” to TD or TS queue shares the same architecture for the preceding scenarios, except that the inbound data is written to a CICS TS or TD queue. The TD queues can be local to that AOR or shared with another AOR.

The following figure shows the externally initiated “fire and forget” to TD or TS queue scenario:



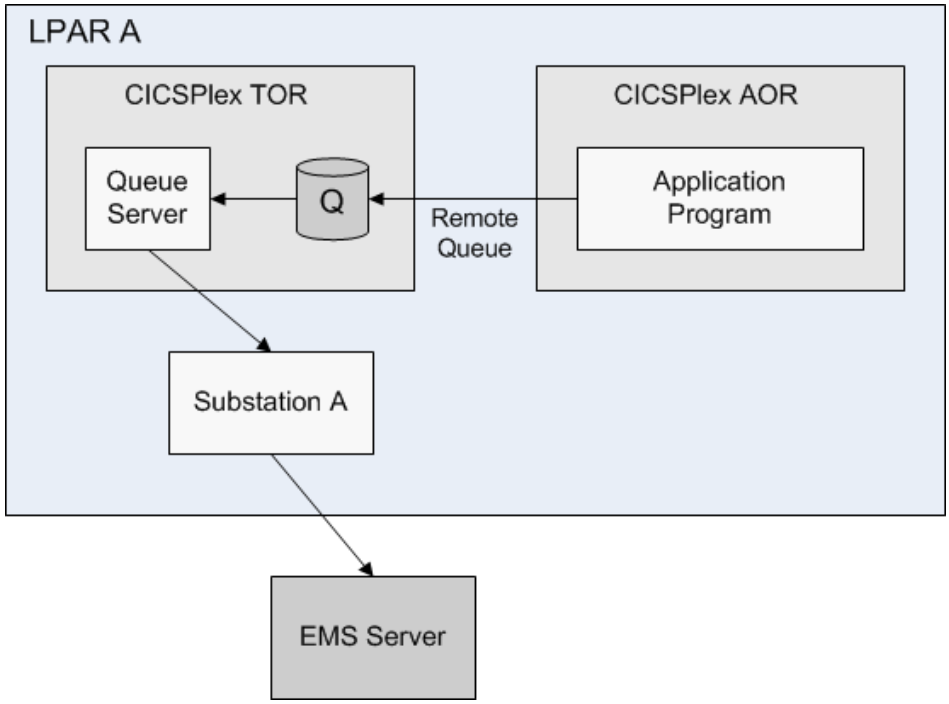
You can create a dual-purpose definition of that resource in a Circuit Switch Data (CSD) file that is shared by the local and remote systems. Such a step reduces disk storage and maintenance, because only one CSD file record is required for each shared resource.

CICS to TD Queue (CICS-Initiated “Fire and Forget”)

In the CICS to TD queue scenario, a CICS application writes data to “publish” to a TD queue. The Substation ES queue server is initiated within the CICSPlex in the region to which that Substation ES instance is connected.

Substation ES picks up the message and publishes it to Enterprise Message Service.

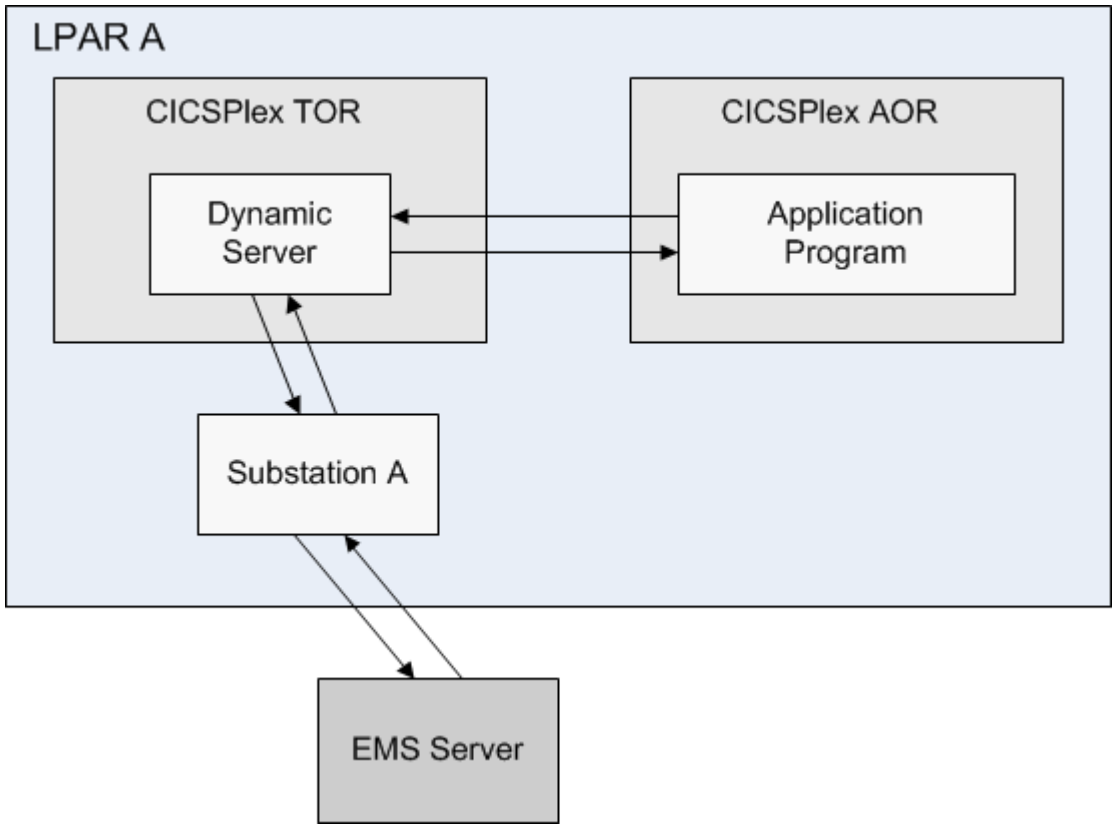
The following figure shows the CICS to TD queue scenario:



CICS Application-Initiated Request Reply

In the CICS application-initiated request reply scenario, a CICS application initiates an external request that expects a response by linking to a supplied CICS program that passes the COMAREA to Substation ES, and then waits for a response for a predefined time.

The following figure shows the CICS application-initiated request reply scenario:



Currently, this scenario causes an affinity between the initiating CICS application program and the Substation ES that handles the request.

Glossary

Asynchronous processing

A means of distributing the processing of an application between systems in an intercommunication environment. The processing in each system is independent of the session on which requests are sent and replies are received. No direct correlation can be made between requests and replies and no assumptions can be made about the timing of replies.

APF (Authorized Program Facility)

A facility that enables identification of programs that are authorized to use restricted functions.

Audit trail

A manual or computerized means for tracing the transactions affecting the contents of a record.

Back-end System

The CICS or IMS transaction processing system provided by IBM.

Batch processing

- Type of data processing in which a number of input items are grouped for processing serially with a minimum of operator intervention and no end-user interaction.
- Serial processing of computer programs.
- Pertaining to the technique of executing a set of computer programs so that each is completed before the next program of the set is started.

BMP (Batch Message Processing Program)

An IMS batch processing program that has access to online databases controlled by DBCTL and message queues. BMPs run online; however, like programs in a batch environment, they are started with job control language (JCL).

CEMT

A CICS transaction that invokes all the master terminal functions. These functions include inquiring and changing the value of parameters used by CICS, altering the status of system resources, terminating tasks, and shutting down CICS.

CICS (Customer Information Control System)

IBM's general-purpose online transaction processing (OLTP) software is an e-business, industrial-strength, server for mission-critical applications. It is a layer of middleware that integrates all the basic software services required by OLTP applications together with a rich set of resources and management services in a highly available, reliable, and scalable manner, enabling its customers to concentrate on the tasks relevant to their particular business.

COBOL (Common business-oriented language)

An English-like programming language for business data processing applications.

Cold Start

The standard initialization sequence performed by the TIBCO z/OS Substation initialization program. In a cold start, all resource definitions are refreshed. Resources dynamically installed or changed by the operations or XML Interface in a previous execution are lost.

COMMAREA (Communication area)

A CICS area that is used to pass data between tasks that communicate with a given terminal. The area can also pass data between programs within a task.

Console

An input-output device on a computer, reserved for communication between the computer operator or maintenance engineer and the computer.

Coupling Facility

A special logical partition that provides high-speed caching, list processing, and locking functions in a sysplex.

DCB (Data Control Block)

An MVS control block used by access method routines in storing and retrieving data.

DCUI (Data Configuration User Interface)

The user interface in which data transformation are configured.

DPL (Distributed Link Request)

A facility that enables a CICS client program to call a server program running in a remote CICS region and to pass and receive data using a communications area.

Dump

A representation of the contents of selected areas of the main storage that finds out whether a program is functioning as intended. A dump might serve as a helpful resource for analyzing problems.

ECB (Event Control Block)

An MVS or VSE control block that represents the status of an event.

ESTAE (Extended Specify Task Abnormal Exit)

A recovery routine to which the system passes control when an error occurs in the mainline routine. The recovery routine's objective is to intercept the error and potentially perform one or more tasks.

EXCI (External CICS Interface) (CICS Transaction Server only)

A CICS application programming interface that helps make CICS applications more easily accessible than non-CICS environments.

Function

A specific purpose of an entity or its characteristic action.

GTF (Generalized Trace Facility)

In MVS, a trace data-collection routine. GTF traces the following system events: seek addresses on START I/O records, SRM activity, page faults, I/O activity, and supervisor services. Execution options specify the system events to be traced.

Heuristic Decision

A decision that enables a transaction manager to complete a failed in-doubt unit of work (UOW) that cannot wait for re synchronization after recovery from the failure.

Under the two-phase commit protocol, the loss of the coordinator (or loss of connectivity) that occurs while a UOW is in doubt theoretically forces a participant in the UOW to wait forever for re-synchronization. While a subordinate waits in doubt, resources remain locked and, in CICS Transaction Server for z/OS, the failed UOW is shunted pending resolution.

IMS (Information Management System)

A database manager that allows access to data in DL/I databases. IMS provides for the arrangement of data in an hierarchical structure and a common access approach in application programs that manipulate IMS databases.

JCL (Job Control Language)

A control language that describes a job and its requirements to an operating system.

LUW (Logical Unit of Work)

A sequence of processing actions (database changes, for example) that must be completed before individual actions can be regarded as committed. When changes are committed (by successful completion of the LUW and recording of the syncpoint on the system log), they do not need to be backed out after a subsequent failure of the task or system. The end of an LUW is marked in a transaction by a syncpoint, issued either by the user program or by CICS/IMS at the end of task. In the absence of user syncpoints, the entire task is an LUW.

MPP (Message Processing Program)

A region for processing IMS messages.

MVS (Multiple Virtual Storage)

An operating system for processing systems consists of one or more mainframe processors.

Multitasking

Concurrent execution of application programs within a CICS region.

Multithreading

Use, by several transactions, of a single copy of an application program.

OTMA (Open Transaction Manager Access)

A transaction-based, connectionless client-server protocol in IMS. The simplified four layer model is often used in descriptions of UNIX networks. In the four layer model, OTMA is the process layer. In the Open Systems Interconnecting (OSI) model, OTMA is the session layer. It can be best to think of OTMA as a combined session and transport layer.

Parameter (ISO)

- A variable that is given a constant value for a specified application and that can denote the application.

- Data passed between programs or presented to a program at startup.

Pipe

A one-way communication path between a sending process and a receiving process.

RACF (Resource Access Control Facility)

An IBM program that provides for access control by identifying and verifying users to the system, authorizing access to protected resources, logging detected unauthorized attempts to enter the system, and logging detected accesses to protected resources.

Routine

A program or sequence of instructions called by a program. Typically, a routine has a general purpose and is frequently used.

RRS (Resource Recovery Service)

The z/OS system component that provides the services that a resource manager calls to protect resources. RRS is the z/OS system level syncpoint manager.

SAF (System Authorization Facility)

Part of the operating system and conditionally directs control to RACF, if RACF is present, or to a user-supplied processing routine, or both, when receiving a request from a resource manager.

Sysplex

A set of MVS systems communicating and cooperating with each other through certain multisystem hardware components and software services to process customer workloads. See also MVS system, parallel sysplex.

Transaction

A unit of application data processing (consisting of one or more application programs) initiated by a single request, often from a terminal. A transaction can require the initiation of one or more tasks for its execution.

UOW (Unit of Work)

A sequence of processing actions (database changes, for example) that must be completed before individual actions performed by a transaction can be regarded as committed. After changes are committed (by successful completion of the UOW and recording of the syncpoint on the system log), they become durable, and are not backed out in the event of a subsequent failure of the task or system.

UOW (In-doubt Work Unit)

In CICS and IMS, a piece of work that is pending during commit processing; if commit processing fails between the polling of subsystems and the decision to execute the commit, recovery processing must resolve the status of work units that is in doubt.

VSAM (Virtual Storage Access Method)

An access method for direct or sequential processing of fixed-and variable-length records on direct access devices.

XCF (Cross System Coupling facility)

A component of MVS that provides functions to support cooperation between authorized programs running within a sysplex.