

TIBCO® API Exchange

Concepts

Software Release 2.1.1
November 2014

Two-Second Advantage®



Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document contains confidential information that is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIBCO, The Power of Now, TIBCO, TIBCO ActiveMatrix, TIBCO ActiveMatrix BusinessWorks, TIBCO Administrator, TIBCO ActiveSpaces, TIBCO Designer, TIBCO Enterprise Message Service, TIBCO Hawk, TIBCO Runtime Agent, TIBCO Rendezvous, are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries. EJB, Java EE, J2EE, and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

THIS SOFTWARE MAY BE AVAILABLE ON MULTIPLE OPERATING SYSTEMS. HOWEVER, NOT ALL OPERATING SYSTEM PLATFORMS FOR A SPECIFIC SOFTWARE VERSION ARE RELEASED AT THE SAME TIME. SEE THE README FILE FOR THE AVAILABILITY OF THIS SOFTWARE VERSION ON A SPECIFIC OPERATING SYSTEM PLATFORM.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

Copyright © 2013-2014 TIBCO Software Inc. ALL RIGHTS RESERVED.

TIBCO Software Inc. Confidential Information

Figures	v
Tables	vii
Preface	ix
Changes from the Previous Release of this Guide	x
Related Documentation	xi
TIBCO® API Exchange Documentation	xi
TIBCO® API Exchange Gateway Documentation	xi
TIBCO® API Exchange Manager Documentation	xi
Other Documentation	xii
Typographical Conventions	xiii
Connecting with TIBCO Resources	xv
How to Join TIBCOCommunity	xv
How to Access TIBCO Documentation	xv
How to Contact TIBCO Support	xv
Chapter 1 Product Overview	1
What Is TIBCO API Exchange?	2
Business Scenarios	3
Enterprise API Governance	3
Partner API Gateway	4
Open API Community	4
Consumer API Gateway	5
Product Architecture	6
API Gateway	6
API Manager	7
API Analytics	7
Chapter 2 TIBCO API Exchange Gateway	9
Gateway Functional Overview	10
Design Time Components	12
Run-Time Components	13
Gateway Operational Layer	13
Gateway Management Layer	13
Analytics Layer	14
Design Concepts	15
Facade Service	15
Target Service	15
Message Processing Life Cycle	17

Mappings and Transformations	17
Throttle Policies	18
Routing	19
Message Exchange Patterns	19
Orchestration	20
Partners	20
Caching	20
Chapter 3 API Manager	21
Introduction	22
What Is an API?	22
Products and Plans	22
Applications	22
Application Developer	24
Product Catalog	24
API Explorer	25
Product Documentation	26
Product Plans	26
User Registration	26
Request Plan Subscription	27
Dashboard	27
Product Manager	30
API	30
Product	30
Product Plan	31
Analysis	31
Swagger Support	31
Partner Administrator	33
Portal Administrator	34
Chapter 4 Analytics	35
Overview of Analytics	36
Product Manager Dashboard	36
Partner Dashboard	37
Product Manager Latency and Throughput View	39
Product Manager Transactions View	40
Glossary	41
Index	43

Figures

Figure 1	How API Providers and Developers Use API Exchange	2
Figure 2	Functional Diagram of API Management	6
Figure 3	Functional View of the API Gateway	10
Figure 4	Facade Service and Target Services	16
Figure 5	Developer Portal—Product Catalog	24
Figure 6	API Explorer	25
Figure 7	Product Plans Inventory	26
Figure 8	Subscription Registration	27
Figure 9	Application Dashboard	28
Figure 10	Overview Statistics Dashboard	29
Figure 11	Documentation Repository	31
Figure 12	Documentation Detail for Post Operation	32
Figure 13	API Analytics Overview	36
Figure 14	Product Manager Dashboard Usage Dimensions	37
Figure 15	Partner Dashboard	38
Figure 16	Latency and Throughput View	39
Figure 17	Transactions View	40

Tables

Table 1 General Typographical Conventions xiii

Preface

TIBCO® API Exchange enables the building of API marketplaces, where service providers and consumers come together to create, host, manage, learn about, and use open APIs.

Topics

- [Changes from the Previous Release of this Guide, page x](#)
- [Related Documentation, page xi](#)
- [Typographical Conventions, page xiii](#)
- [Connecting with TIBCO Resources, page xv](#)

Changes from the Previous Release of this Guide

This section itemizes the major changes from the previous release of this guide.

Changes in the Document Updated: July 2014

This document has been rewritten and reorganized to add information about basic concepts, user roles, and TIBCO API Exchange Analytics. New terms have been added to the Glossary.

Related Documentation

This section lists documentation resources you might find useful.

TIBCO® API Exchange Documentation

The TIBCO API Exchange Documentation contains:

- *TIBCO API Exchange Concepts* Read this document to get an overview of API Exchange concepts, workflow, and deployment.
- *TIBCO API Exchange Getting Started*. Provides an overview of the tasks required to install, configure, and deploy TIBCO API Exchange.

These documents are included as part of the API Exchange Manager documentation.

TIBCO® API Exchange Gateway Documentation

The following documents form the TIBCO API Exchange Gateway documentation set:

- *TIBCO API Exchange Gateway Installation* Read this manual for instructions on site preparation and installation.
- *TIBCO API Exchange Gateway User's Guide* Read this manual for instructions on how to configure and use this product.
- *TIBCO API Exchange Gateway Release Notes* Read the release notes for a list of new and changed features. This document also contains lists of known issues and closed issues for this release.

TIBCO® API Exchange Manager Documentation

- *TIBCO API Exchange Manager Installation* Read this manual for instructions on site preparation and installation.
- *TIBCO API Exchange Manager Administration* Read this manual for information on how to set up users and user groups, add APIs, and manage products and plans.
- *TIBCO API Exchange Manager Release Notes* Read the release notes for a list of new and changed features. This document also contains lists of known issues and closed issues for this release.

Other Documentation

You might find it useful to read the documentation for the following:

- Joomla! - See <http://docs.joomla.org>.
- Example project hosted on GitHub: *Adapter Code for TIBCO API Exchange and Joomla!*. See <https://github.com/API-Exchange/JoomlaAdapter/wiki>.




Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
<i>ENV_NAME</i> <i>TIBCO_HOME</i> <i>ASG_HOME</i>	<p>TIBCO products are installed into an installation environment. A product installed into an installation environment does not access components in other installation environments. Incompatible products and multiple instances of the same product must be installed into different installation environments.</p> <p>An installation environment consists of the following properties:</p> <ul style="list-style-type: none"> • Name Identifies the installation environment. This name is referenced in documentation as <i>ENV_NAME</i>. On Microsoft Windows, the name is appended to the name of Windows services created by the installer and is a component of the path to the product shortcut in the Windows Start > All Programs menu. • Path The folder into which the product is installed. This folder is referenced in documentation as <i>TIBCO_HOME</i>. <p>TIBCO API Exchange installs into a directory within a <i>TIBCO_HOME</i>. This directory is referenced in documentation as <i>ASG_HOME</i>. The default value of <i>ASG_HOME</i> depends on the operating system. For example on Windows systems, the default value is C:\tibco\asg\2.1.</p>
code font	<p>Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example:</p> <p>Use MyCommand to start the foo process.</p>
bold code font	<p>Bold code font is used in the following ways:</p> <ul style="list-style-type: none"> • In procedures, to indicate what a user types. For example: Type admin. • In large code samples, to indicate the parts of the sample that are of particular interest. • In command syntax, to indicate the default parameter for a command. For example, if no parameter is specified, MyCommand is enabled: MyCommand [enable disable]

Table 1 General Typographical Conventions (Cont'd)

Convention	Use
<i>italic font</i>	<p>Italic font is used in the following ways:</p> <ul style="list-style-type: none">• To indicate a document title. For example: See <i>TIBCO ActiveMatrix BusinessWorks Concepts</i>.• To introduce new terms For example: A portal page may contain several portlets. <i>Portlets</i> are mini-applications that run in a portal.• To indicate a variable in a command or code syntax that you must replace. For example: <code>MyCommand</code> <i>PathName</i>
Key combinations	<p>Key name separated by a plus sign indicate keys pressed simultaneously. For example: Ctrl+C.</p> <p>Key names separated by a comma and space indicate keys pressed one after the other. For example: Esc, Ctrl+Q.</p>
	<p>The note icon indicates information that is of special interest or importance, for example, an additional action required only in certain circumstances.</p>
	<p>The tip icon indicates an idea that could be useful, for example, a way to apply the information provided in the current section to achieve a specific result.</p>
	<p>The warning icon indicates the potential for a damaging situation, for example, data loss or corruption if certain steps are taken or not taken.</p>

Connecting with TIBCO Resources

How to Join TIBCOCommunity

TIBCOCommunity 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. TIBCOCommunity offers forums, blogs, and access to a variety of resources. To register, go to <http://www.tibcommunity.com>.

How to Access TIBCO Documentation

You can access TIBCO documentation here:

<http://docs.tibco.com>

How to Contact TIBCO Support

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

- 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.

Chapter 1 **Product Overview**

This chapter provides an overview of TIBCO API Exchange.

Topics

- [What Is TIBCO API Exchange?, page 2](#)
- [Business Scenarios, page 3](#)
- [Product Architecture, page 6](#)

What Is TIBCO API Exchange?

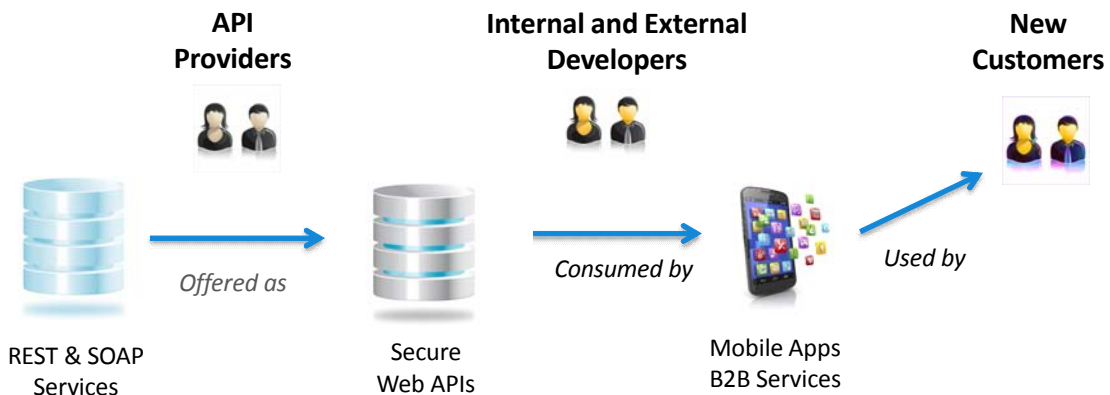
TIBCO® API Exchange is a platform that hosts and manages open APIs. TIBCO API Exchange enables the building of integrated marketplaces, where API providers and API consumers come together to create, host, manage, learn about, and use open APIs.

TIBCO API Exchange facilitates API distribution and access in the following ways:

- Provides access to enterprise data or business functionality provided by API providers and exposed as web APIs.
- Allows internal and external developers to embed *your* functionality in *their* applications.
- Takes *your* business to new channels, markets and customers.
- Enables API management by regulating access, empowering partners, and providing visibility to products.

Figure 1 shows how API providers open APIs to developers, who build applications that reach new customers.

Figure 1 How API Providers and Developers Use API Exchange



Business Scenarios

TIBCO API Exchange can be used in a variety of business scenarios. This section describes several scenarios:

- [Enterprise API Governance, page 3](#)
- [Partner API Gateway, page 4](#)
- [Open API Community, page 4](#)
- [Consumer API Gateway, page 5](#)

Enterprise API Governance

Companies are significantly lowering development costs by using service-oriented architectures, with service encapsulation, reuse, and standardization of web APIs and tools. However, as service reuse and interdependencies between organizations increase, the challenge of assuring service performance also increases.

Without controls, a single errant service consumer (an application) frequently impacts service quality for other applications, and even causes a cascade failure across an SOA environment. Changing APIs is impossible without complete knowledge of the applications that are using it. Even then, the API might need to be stable for the lifetime of the application or at least until the next upgrade. With many services, it can be difficult for enterprise developers to discover and learn about what services and APIs are available.

In this scenario, TIBCO API Exchange provides an Application Services Governance role that does the following:

- Tracks who has access
- Sets limit thresholds
- Monitors application KPIs
- Looks at usage for capacity planning
- Provides a collaborative API repository

Using Enterprise API governance, companies achieve cost reduction and improved efficiency, with consistent service uptime and performance, and benefit from the central knowledge base of APIs.

Partner API Gateway

Increasingly API consumers (applications) are running outside of a company's control. This might include mobile applications that are running on employee or customer devices and also partner-developed solutions.

Although the internal and external partners are well known and application use cases are predefined, there is much lower trust, because not only are there erratic consumers, but there is also the potential for compromised systems and networks. Changing and documenting APIs is more difficult with developers working in different organizations on different timelines and on different business problems.

In the partner API Gateway scenario, TIBCO API Exchange is used to open up business data and services to internal and external partners in a controlled and secure manner. TIBCO API Exchange does this in the following ways:

- Protects against malicious or accidental attacks.
- Sets usage limits by partner or application.
- Routes requests to different services based on version or other context.
- Maps external to internal protocols (for example, REST to SOAP, HTTP to JMS).
- Creates and monitors custom SLAs.
- Audits access.
- Allows partners to manage developers and debug API usage.
- Enables an Extranet to act as a centralized API repository.

With a partner API gateway, companies can lower the cost of onboarding new organizations and manage them more efficiently. With tighter partner integration, they can extend their digital value chain to take their business into new channels.

Open API Community

In an open API model we are encouraging third-party developers to create novel applications using our enterprise APIs. We do not know ahead of time which developers will succeed and what applications will be built, so it is important to make it easy for all developers to learn about and use APIs.

Supporting hundreds and thousands of developers requires different approaches for API governance, security, performance, and partner and contract management.

In this scenario TIBCO API Exchange supports the following:

- No cost evaluation with self-service registration, evaluation and support

- Monetization—the ability to track and meter API usage
- A scalable platform on-premise and cloud
- The ability to package SLAs as standardized product offerings
- Use of OAuth for delegated access
- Analysis of consumer and developer behavior
- Opening up enterprise services as products

Companies can inspire a developer community to create the next great application while gaining competitive advantage through innovation. An open API model fosters collaborative environments that drive successful open API initiatives.

Consumer API Gateway

As companies integrate services from third-party vendors, they face a number of challenges:

- How to apportion access between multiple internal users?
- How to change security policies without modifying business systems?
- How to monitor vendor performance?

In this scenario, TIBCO API Exchange allows an enterprise to mediate access to external web APIs. TIBCO API Exchange:

- Multiplexes access to internal applications and employees
- Apportions costs to different constituents - and enforce vendor-supplied limits
- Measures vendor performance and cost
- Standardizes security protocols, message formats, error handling, and caching
- Logs access for audit

Using an API consumer gateway, companies can implement sound internal cost management practices.

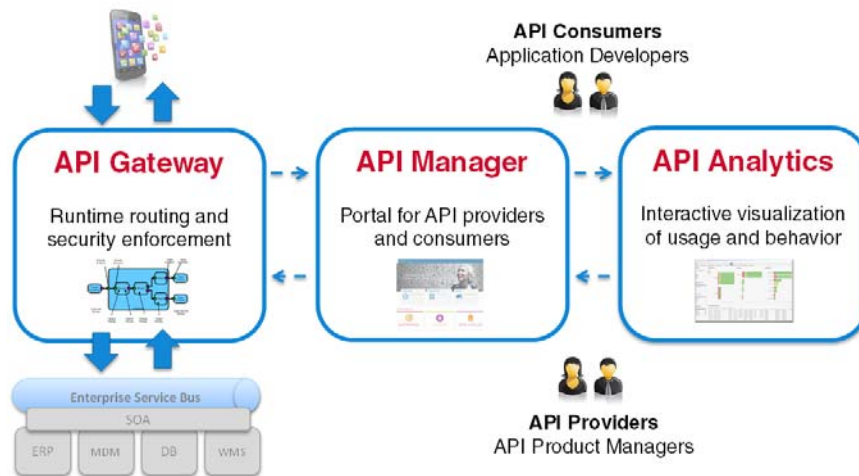
Product Architecture

Using the TIBCO API Exchange platform, you can build and test APIs, define runtime governance policies, migrate APIs between environments, and monitor and report on API usage. The platform is available as either a cloud-based or on-premise based solution.

As shown in [Figure 2](#), TIBCO API Exchange comprises three modules:

- [API Gateway](#)
- [API Manager](#)
- [API Analytics](#)

Figure 2 Functional Diagram of API Management



API Gateway

The API Gateway is a centralized access point for managing enterprise APIs, and provides mediation between internal and external services, systems, and devices.

For more information, see [Chapter 2, TIBCO API Exchange Gateway](#).

API Manager

The API Manager provides API providers with infrastructure and tooling to expose their APIs to internal and external developers.

For more information, see [Chapter 3, API Manager](#).

API Analytics

API Analytics provides insights on API usage and performance for API providers and consumers.

For more information, see [Chapter 4, Analytics](#).

Chapter 2

TIBCO API Exchange Gateway

This chapter describes the key concepts for understanding the TIBCO API Exchange Gateway component.

Topics

- [Gateway Functional Overview, page 10](#)
- [Design Time Components, page 12](#)
- [Run-Time Components, page 13](#)
- [Design Concepts, page 15](#)

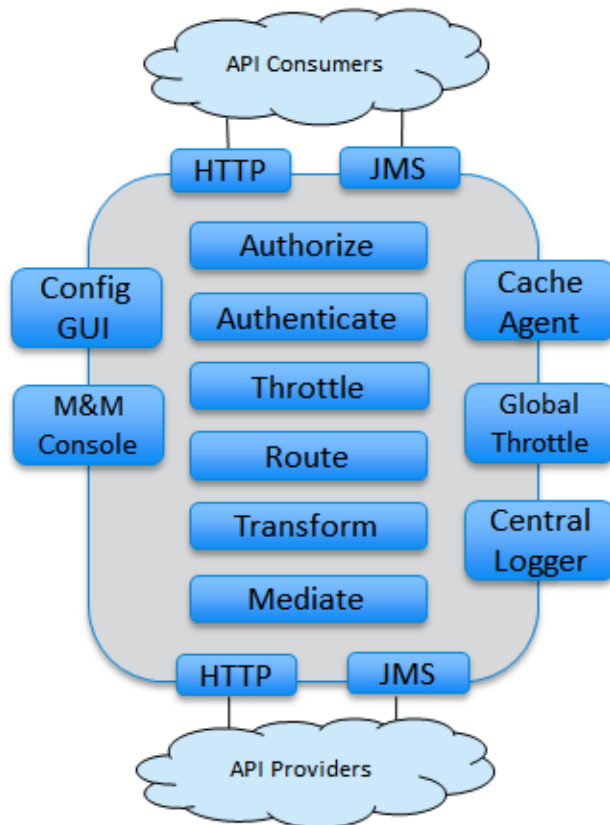
Gateway Functional Overview

TIBCO API Exchange Gateway controls API access and associated event flows by enforcing policies that determine the following:

- *Whose* requests are handled (authorization, authentication).
- *When* requests are handled (throttling)
- *Where* requests are handled (routing)
- *How* requests are handled (transformation, mediation).

Figure 3 shows a functional diagram of API Gateway.

Figure 3 Functional View of the API Gateway



Policies

TIBCO API Exchange Gateway allows API providers to enforce business and technical requirements, including security, validation, and service level agreements through declarative policies.

TIBCO API Exchange includes the following policies:

- **Security Policies** Provide authentication, authorization, encryption, validation, and digital signature and certificate management. Support is provided for WSS Username Token, SAML, X.509, Kerberos, OAuth, and LDAP.
- **Validation Policies** Check content against schemas or rules and reject invalid or hostile messages.
- **Throttle Policies** Provide limits or quotas by partner, service, or other criteria. Throttles can be based on request rate, concurrent load, or error rate and used to restrict access at the façade (such as through a commercial SLA or product plan), or against the target service (technical throttle).
- **Transformation and Mediation Policies** Transformation policies provide transformation of request, response, and fault messages.
- **Logging** Traces requests for audit or debugging.

Design Time Components

TIBCO API Exchange Gateway includes the following design time components:

- **Configuration UI** Using the configuration UI, you can configure partner data, partner operations, partner groups, services, operations, mappings, throttles, error maps, schemas and routing information.
- **Gateway Studio** The Gateway Studio is a design time environment that allows you to design and develop custom extensions. Custom extensions change the default behavior of the gateway core engine.

Run-Time Components

The TIBCO API Exchange Gateway run-time components consist of the following layers:

- Operational Layer
- Management Layer

Gateway Operational Layer

The Gateway operational layer scales as processing demands increases, primarily based on the volume of requests. The Gateway operational layer consists of the following components:

- **Apache HTTP Server** An optional Apache layer may be used in the DMZ to terminate the incoming HTTP(s) transport.
- **Core Engine** The core engine is a high-performance event-based service-request routing engine that receives requests as events and uses the rules engine to determine how requests are handled.
- **Cache Agent** The cache agent stores the cache data for all objects of the cluster.

Gateway Management Layer

The Gateway management layer has the following sub-components:

- **Central Logger** The Central Logger provides centralized messages and auditing of messages and statistics in a database.
- **Global Throttle Manager** The Global Throttle Manager manages throttle allocation across a gateway cluster.
- **Cache Clearing Manager** The Cache Clearing Manager component clears the cache based on the size and age of the cached values.
- **Monitoring and Management Server** The Monitoring and Management Server monitors the status and operational tasks for all components in the gateway cluster.

Analytics Layer

The analytics database receives runtime information and statistics from one or more Central Loggers that might represent one or more gateway clusters. Information that might be captured includes KPIs (aggregated statistics or Key Performance Indicators), logs of each request/response (transaction logs) and for message transformation and event processing steps.

Design Concepts

This section provides further details on the operational features of TIBCO API Exchange Gateway.

The core engine contains the following main sub-components:

- **Facade** The facade provides the interface for the gateway to receive requests for a given API with a given binding (for example, REST over HTTP or SOAP over JMS).
- **Router** The router receives requests from the facade and routes them to the appropriate target service handler.
- **Target** The target component calls the appropriate external service to execute the request.

Facade Service

A facade service is any application service or API that the gateway offers. Typically the service is an intermediary to one (or more) *target services* outside the gateway.

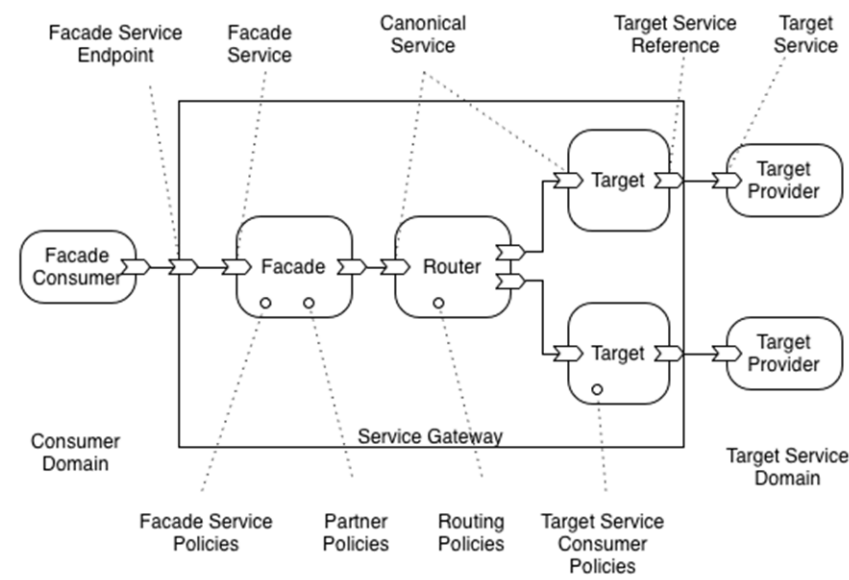
A facade service may also be known as a proxy service or virtual service, especially if the interface of the facade and target services are the same. The gateway facilitates a loose coupling between the facade and target service by managing interfaces, policies and configuration information for either the facade service or target service.

Policies applied to a facade service are enforced by the gateway and include decryption, validation, throttling, and authorization.

Target Service

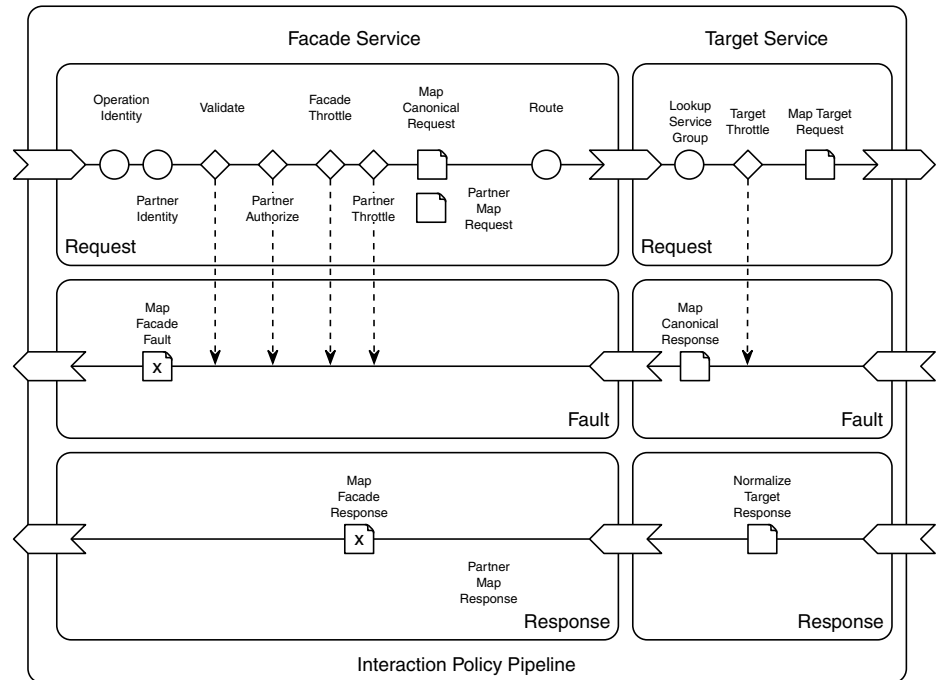
A target service is an external endpoint that the gateway may call during processing of a request. Policies applied to a target service are enforced by the gateway and include encryption, credential mapping, throttling, and load balancing.

Figure 4 Facade Service and Target Services



Message Processing Life Cycle

The gateway core engine uses a staged event-driven architecture. Request and response messages are processed internally as events with processing steps determined dynamically by the engine based on applicable policies.



Mappings and Transformations

TIBCO API Exchange Gateway provides message transformations using mappings. Combined with throttling, this allows you to implement business agreements as enforceable policies.

TIBCO API Exchange Gateway provides transformation of request and response messages at the following four points:

- **Facade to Router** After the request has been received by the facade request handler and before it is been passed to the router.
- **Router to Target** After the request has been routed but before it is passed to the service endpoint handler.

- **Target to Router** After the response has been received from the target handler and before it is passed to the router.
- **Router to Facade request handler** After the response has been routed from the router to the facade request handler and before the response is sent back to the original requestor.

Additionally, facade request and response transformations can be overridden on a partner specific basis.

For more information on mappings and transformations, see the “Mappings and Transformations” section in Chapter 5, “Transaction Pipeline Processing,” in the *TIBCO API Exchange Gateway User’s Guide*.

Throttle Policies

Using throttle policies, you can enforce usage or other limits, and define the maximum number of requests that are handled by a facade or target operation in a defined time interval. You must define the maximum count and the time interval for a throttle.

Throttles define a condition for a type and metric (entity). API Exchange Gateway checks the condition for an incoming request before processing the request. For example, you can define a condition to allow only 5 client requests within 10 seconds to the backend service for a partner request.

There are two main categories of throttles:

- **Facade throttles** Support service level agreements with consumers, for example, Partner and Partner plus Operations.
- **Target throttles** Support service level agreements with providers, and are applied on the Target Service operation.

Throttle Types

There are four kinds of throttles:

- **Rate** The rate throttle is a simple throttle that allows the requests to pass-through until a limit is reached for a time interval. The rate throttle is always increased on the request. A throttle may be incremented by a count of requests, size of a payload or it can be based on content; for example, a throttle can be based on order totals.
- **Quota** The quota throttle is similar to the rate throttle, but it uses a much larger count over much longer intervals (such as days). The quota throttles are increased on the request. For a quota throttle, you must define throttle interval (in hour) and throttle max count.

- **High Water Mark** (concurrent requestor) High Water Mark throttle is similar to the Rate Throttle, but this throttle also decrements the count after the passed-on requests are completed and the response is ready to return to the requestor. This means that the High Water Mark throttles are increased on the request and decreased on the response. You must define a throttle max count for a high water mark throttle.
- **Error** Error Throttles act as a Rate Throttle in logic, but this throttle counts the error responses, as opposed to the requests. The throttle count of an error throttle is increased on the error responses.

Routing

Routing allows the directing of requests to specific target services based on operation, partner, version or message content or operation.

Message Exchange Patterns

A message exchange pattern (MEP) defines the sequence and cardinality of messages sent between the provider and the consumer for an interaction. MEPs contain both normal and fault messages. The gateway supports several different styles of message exchange as well as the ability to mediate between them.

The service gateway supports the following patterns:

- **Request/Acknowledge (One-Way)** A consumer sends a message to a provider that provides a status response. This is also known as Fire and Forget, and In-Only.
- **Synchronous Request-Response (Sync)** A consumer sends a message to a provider with expectation of response over the same client connection. The provider sends a response message or fault and the consumer responds with a status.
- **Asynchronous Request-Response (Async)** A consumer sends a message to a provider with expectation of a callback of a response. The provider acknowledges the request. The provider then sends a response message or fault and the consumer responds with a status.
- **Consume Message** A provider sends a message to a consumer.

The service gateway supports different MEPs between (1) the facade consumer and the facade service and (2) between the target consumer and the target service.

Orchestration

TIBCO API Exchange Gateway provides the following orchestration models:

- Parallel Orchestration (also called Enumeration)

With parallel orchestration, a single inbound request is split into a set of multiple outbound sub-requests. Each sub-request may be routed differently to various service endpoints. After processing and receiving the responses for each sub-requests, all responses are recombined into a single response message for the original inbound request.

- Sequential Orchestration

Using sequential orchestration, you can access multiple target endpoints by making a number of sequential calls to fulfill or authorize a request. With sequential orchestration, there is a primary outbound target invocation, preceded by one or more secondary target invocations.

Sequential orchestration might use the associative and responses cache features to accelerate the processing of subsequent requests, which helps minimize the load on back-end systems.

Partners

Using TIBCO API Exchange Gateway, you can define partners and partner groups and specify processing for them. You can configure the following:

- Information that identifies a partner.
- A partner group that the partner belongs to.
- The throttle chain that is applied to any requests sent by a partner in that group.

Caching

Caching improves performance and reduces the load on back-end systems.

TIBCO API Exchange Gateway includes a distributed cache based on the in-memory data grid TIBCO® ActiveSpaces.

The association cache is generally used to cache service responses to improve response time and resilience and to reduce load on target services. The cache is also often used to store common reference data used to enrich a response such as to look up account information from a device identifier.

Chapter 3 **API Manager**

TIBCO® API Exchange Manager is an environment for creating and managing developer communities for TIBCO API Exchange. It includes support for the following:

- API Product Catalogs
- Developer self-service registration
- API lifecycle management
- Community management

Topics

- [Introduction, page 22](#)
- [Application Developer, page 24](#)
- [Product Manager, page 30](#)
- [Partner Administrator, page 33](#)
- [Portal Administrator, page 34](#)

Introduction

TIBCO API Exchange Manager is an environment for creating and managing API catalogs and developer communities for TIBCO API Exchange.

API Manager supports four main user roles:

- Application Developer
- Product Manager
- Partner Administrator
- Portal Administrator

What Is an API?

In the TIBCO API Exchange environment, an API is a web application programming interface that can be shared between a provider organization and a partner. TIBCO API Exchange Manager includes support for SOAP and REST APIs.

The API or service has a URL where it can be accessed as an endpoint. An API includes one or more message exchange patterns supporting different functions, called operations. An API may also include human-readable documentation and a machine-readable specification.

Products and Plans

A product is a package of one or more APIs—a commercial bundle. Products allow API providers to create and manage catalogs of APIs.

Products are offered through plans, which provide different tiers or terms of service: typically rates, thresholds, and prices, for the packaged APIs. Partners are granted access to APIs through these plans. Partners can also request custom plans.

Applications

An application is a partner component that calls or consumes the API. The API might be called from an application running under direct control of the partner; running from a data center server, for example. It might also run outside the control of the partner, for example on a mobile phone or desktop.

A partner registers each application with TIBCO API Exchange Manager, which allocates a unique application-specific key. Partners can enable or disable access to APIs on a per-application basis by revoking or resetting keys.

Application Developer

Application developers use the API management portal to do the following:

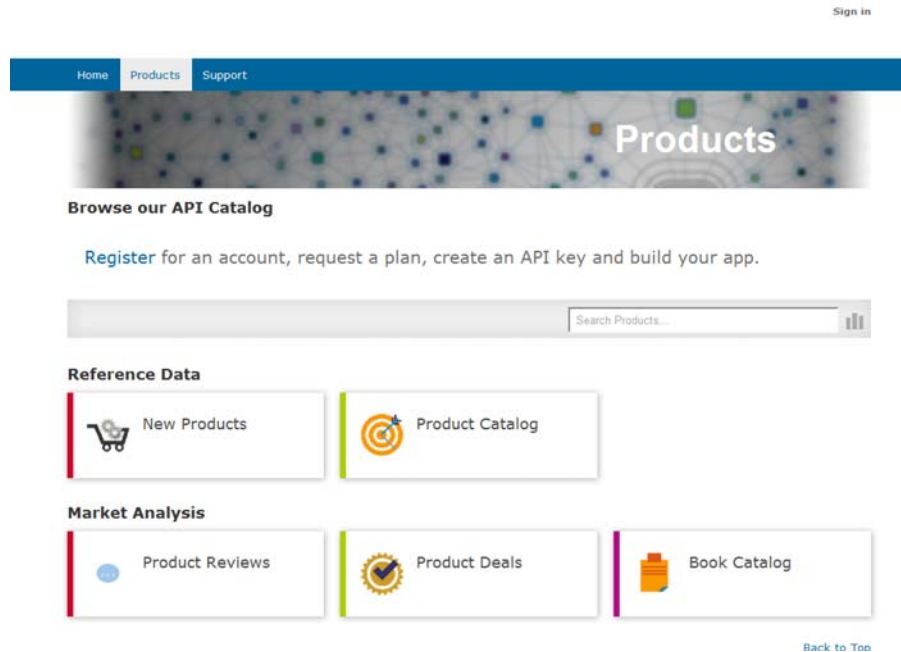
- Discover and use APIs through an API product catalog.
- Use a self-service portal for enrollment, key requests, and API testing.
- Monitor and report on API consumption.

Product Catalog

Public, partner and internal developers browse the product catalog to discover API products available to them and view detailed information about the service, including documentation, technical specifications, examples, and offered plans.

The portal administrator can customize the portal landing page and product pages, including content, colors, styles, and branding elements.

Figure 5 Developer Portal—Product Catalog



After selecting a product, users can view product details for the product. Products details include:

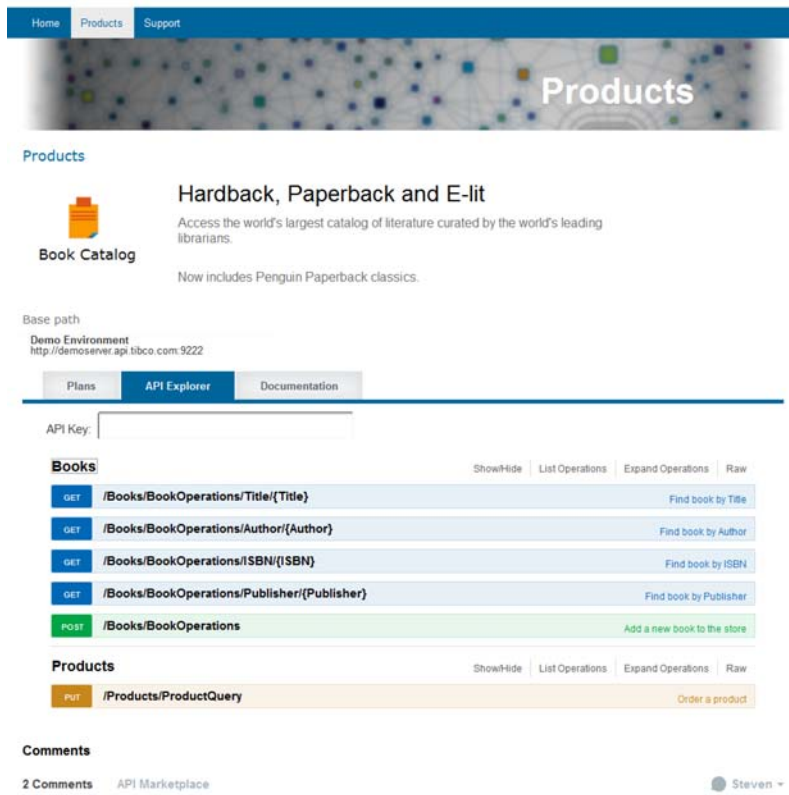
- API Explorer
- Documentation
- Product Plans

API Explorer

TIBCO API Exchange provides an API Explorer—a user interface that allows users to browse and test the APIs offered by the provider organization.

Figure 6 shows the API Explorer tab for the Book Explorer product, which is provided with the basic Adapter code installation.

Figure 6 API Explorer



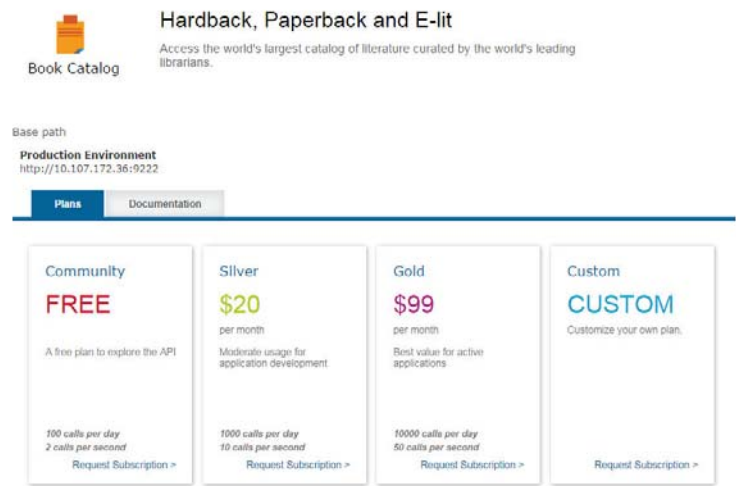
Product Documentation

The product details view for a product includes a **Documentation** tab, which displays links to the documentation for the product.

Product Plans

The Plans tab in the product page shows product plans that are available for application developers to use. By clicking **Request Plan** in a plan description, application developers can ask for access.

Figure 7 Product Plans Inventory



User Registration

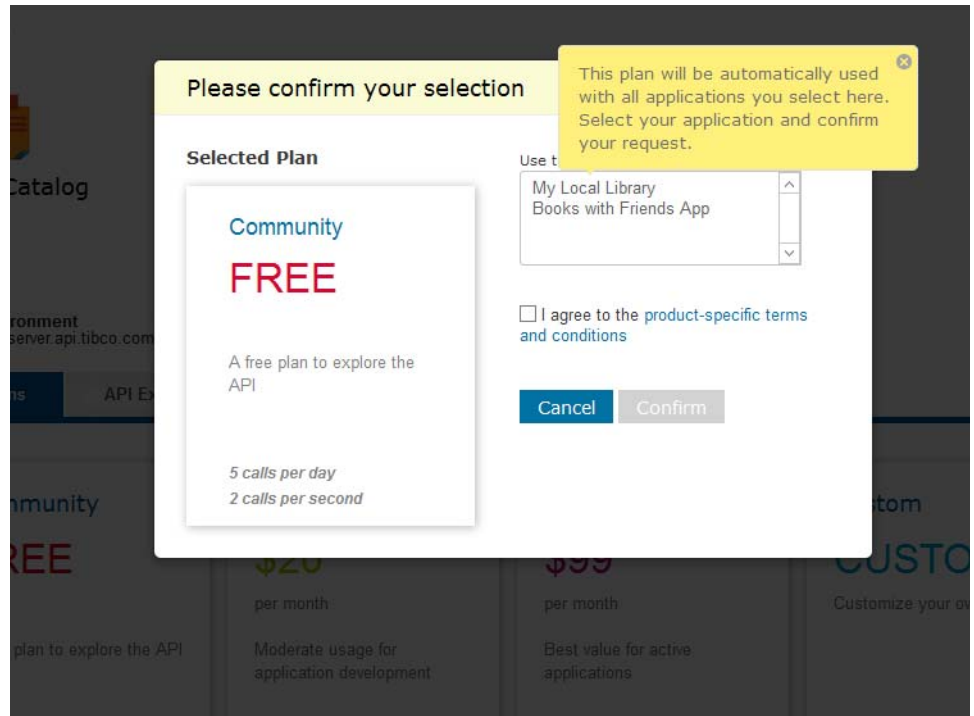
A developer can self-register through the portal to access additional features, including the ability to request access to an API and to manage application keys.

Request Plan Subscription

A developer can request access to a plan to create a product subscription. Depending on the plan setting, access may be granted immediately or the request is forwarded to a product manager for approval.

Bubble help guides developers through their first request.

Figure 8 Subscription Registration



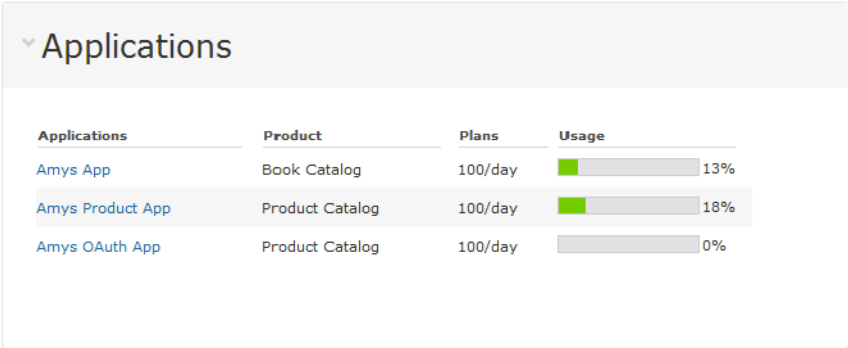
Dashboard

The Developer Portal includes a Dashboard that provides a number of reports and graphs that provide quick summaries of application and product quote usage and performance.

Applications Dashboard

The dashboard provides a summary of registered applications and quota usage.

Figure 9 Application Dashboard



Statistics Dashboard

Users can select the statistics dashboard to view detailed usage and performance statistics by application, by products or by operation. The user can view latency, throughput, and transactions statistics for all applications or for a specified application.

Figure 10 shows the Overview Statistics Dashboard.

Figure 10 Overview Statistics Dashboard



Product Manager

Product Managers use the API management portal to package APIs into products and create different product offerings or plans.

API

An API as defined within the portal is a set of operations that share a common base path and lifecycle. The product manager creates an API through the portal as either a REST or SOAP API:

- A REST API is usually represented by the set of HTTP verbs supported for a single entity. It can be represented by a Swagger resource specification.
- A SOAP API is typically the set of operations that are exposed for a single service. It can be represented by a WSDL specification.

In addition to an optional specification, documentation can also be uploaded for the API, including attachments and rich text (HTML).

Each API is associated with one or more gateway environments; for example, Staging, Production-Europe, or Production-Americas.

APIs do not become visible to application developers until they are associated with a published Product.

Product

A product is a package of one or more APIs for use by internal or external application developers. Application developers subscribe to products and thus only indirectly to APIs. APIs can be added to or removed from products without affecting existing subscriptions.

A product manager creates a product name and icon to be used in the product catalog. Product level documentation can also be uploaded including attachments and rich text (HTML).

A product is associated with one gateway environment; for example, Production-Americas. This environment must be valid for all packaged APIs.

Once created, a product must be published in order to be visible to Application Developers.

The product page is generated from the specifications and documentation of the included APIs. The API Explorer user interface is generated from the aggregate Swagger resource specifications.

Product Plan

A plan is an offering of a product for a specific capacity, which includes the following:

- **Plan name and description** Displayed within the product page.
- **Quota** Maximum calls per day.
- **Rate limit** Peak calls per second.
- **Subscription model** Automatic manual approval.
- **Price** The price for the plan.
- **Custom** One-off custom plans can also be created for specific consumers.

Analysis

The product manager has an important role in monitoring usage of APIs and Products, looking at performance of applications consuming the APIs as well as the performance of back-end target systems.

For more information on analytics capabilities see [Chapter 4, Analytics](#).

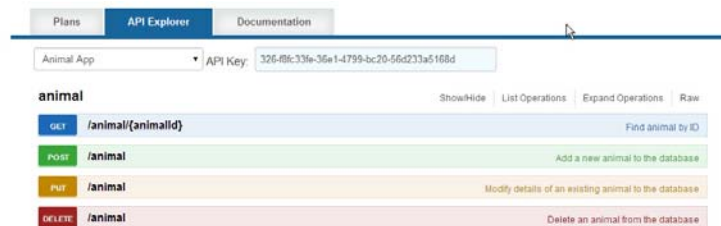
Swagger Support

To support REST, TIBCO API Exchange supports the Swagger specification, a standard for describing a REST-ful services for which interactive documentation and a test harness can be generated.

The Adapter Code for TIBCO® API Exchange and Joomla! example project, which is available on GitHub, includes an example of how the Swagger UI can be integrated into an API management portal.

[Figure 11](#) shows the documentation for an API.

Figure 11 Documentation Repository



Product managers can provide detailed documentation for each operation supported by an API.

Figure 12 Documentation Detail for Post Operation

GET /animal/{animalid}Find animal by ID

POST /animalAdd a new animal to the database

Response Class

Model Model Schema

SuccessResponse {
Code (string, optional)
Message (string, optional)
}

Response Content Type
application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
Animal	<div>{ "ID": "string", "Name": "string", "Category": "string", "Tags": "string" }</div> <div>Parameter content type: application/json</div>	Animal object that needs to be added to the database	body	Model Model Schema <div><div>{ "ID": "string", "Name": "string", "Category": "string", "Tags": "string" }</div><div>Click to set as parameter value</div></div>

Error Status Codes

HTTP Status Code	Reason
405	Invalid input

Try it out

Feedback

Partner Administrator

Partner Administrators manage user accounts and access. Administrators use the management portal to perform the following tasks:

- Manage developer and partner accounts (through Organizations).
- Set access rights by user or organization.
- Create and manage partner subscriptions to products.
- Monitor application quota usage and subscription usage.

Portal Administrator

The Portal Administrator is a superuser role for the developer portal.

The Portal Administrator performs the following tasks:

- Creates system users and user groups.
- Configures environments and gateways.
- Maintains static content of site, including menus.
- Changes the look and feel of dynamic site through style sheets and localization.
- Manages developer forums.

Chapter 4 **Analytics**

TIBCO API Exchange Analytics provides insights on how APIs are being used and performing. Using an integrated analytics server powered by TIBCO Spotfire®, API Analytics provides a range of reporting capabilities. Analytics includes the following options:

- Reporting dashboards for providers and partners.
- Service Level Agreements (SLAs) and key performance indicators (KPIs).
- Full execution auditing.

Topics

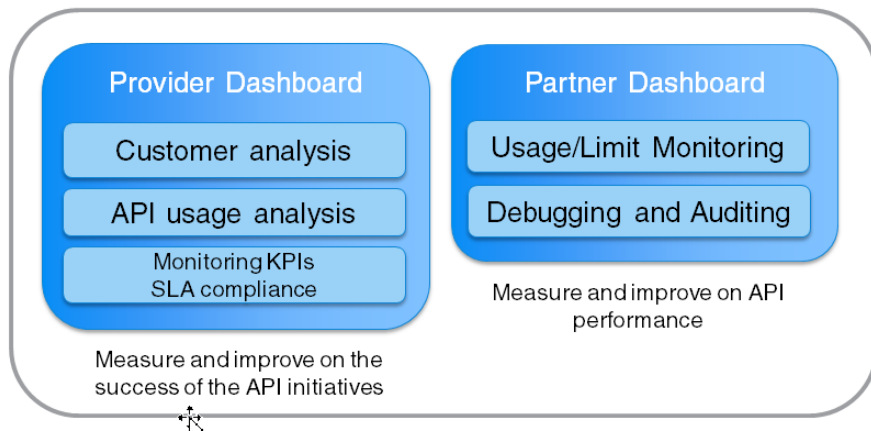
- [Overview of Analytics, page 36](#)
- [Product Manager Latency and Throughput View, page 39](#)
- [Product Manager Transactions View, page 40](#)

Overview of Analytics

API Exchange analytics enables API developers and consumers to perform end-to-end monitoring of API requests and responses. Users perform analytics using the provider dashboard or the partner dashboard.

Figure 13 shows a functional overview of the API Exchange Analytics component.

Figure 13 API Analytics Overview



The API Exchange Analytics component is powered by TIBCO Spotfire[®], available for purchase separately.

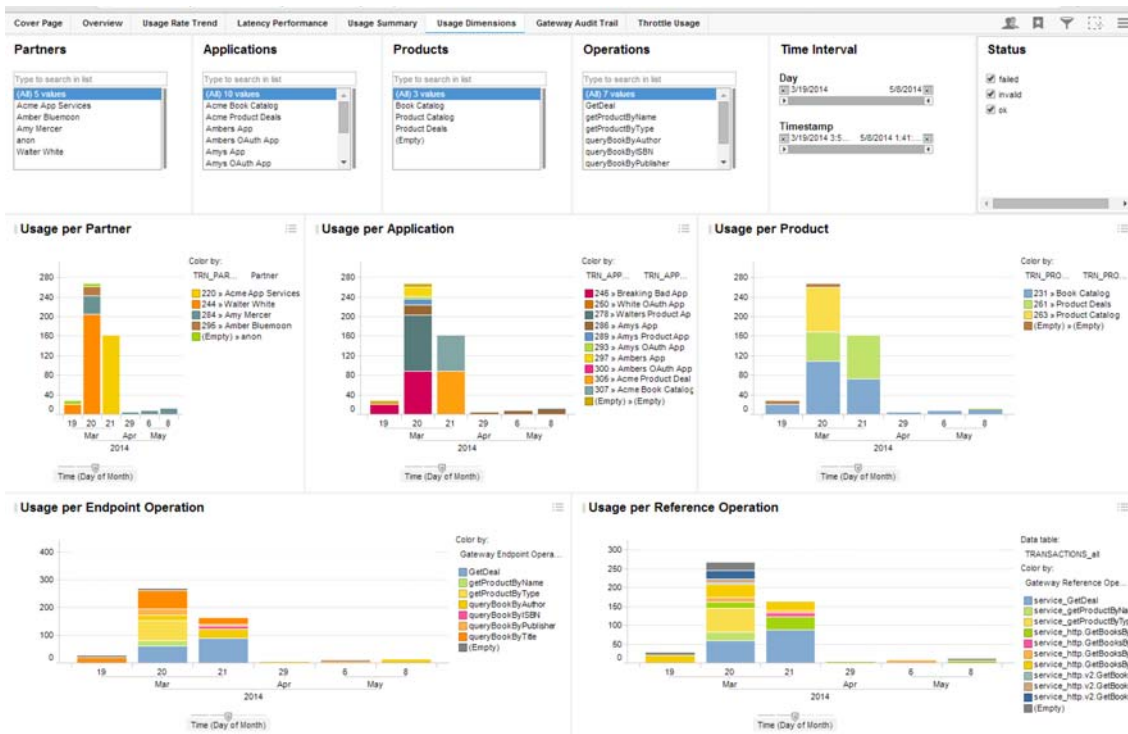
Product Manager Dashboard

The product manager dashboard enables:

- Customer analysis
- API usage analysis
- Monitoring of KPIs and SLA compliance

Figure 14 shows the Product Manager Dashboard.

Figure 14 Product Manager Dashboard Usage Dimensions



Partner Dashboard

The partner dashboard enables the following:

- Usage and limit monitoring.
- Debugging and auditing.

Figure 15 shows a page from the Partner Dashboard Overview.

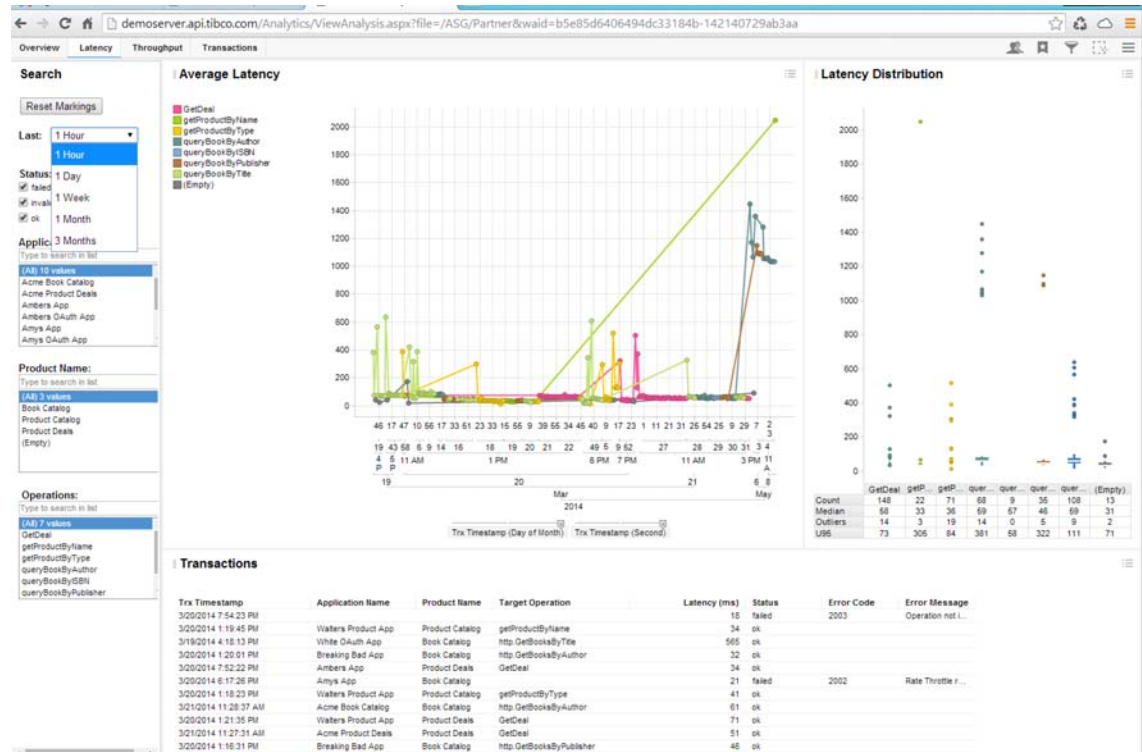
Figure 15 Partner Dashboard



Product Manager Latency and Throughput View

The Latency and Throughput view, shown in Figure 16, shows average latency for API operations.

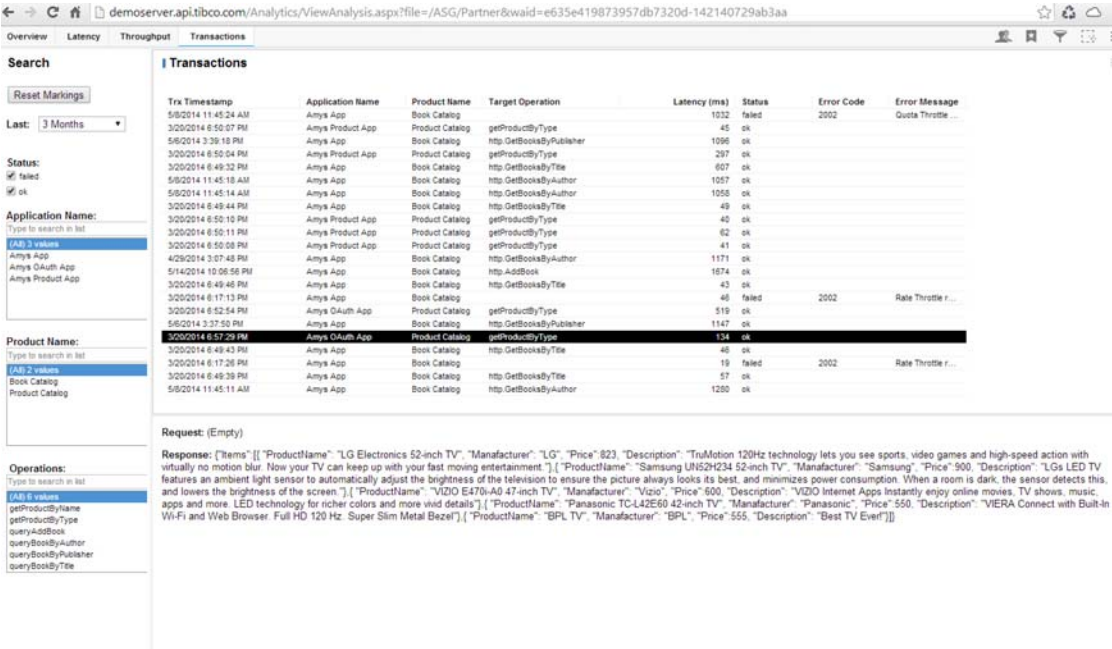
Figure 16 Latency and Throughput View



Product Manager Transactions View

The Transactions View, shown in [Figure 17](#), shows transactions statistics for API operations.

Figure 17 Transactions View



Glossary

A

API

In TIBCO API Exchange, an API is an open API that is developed by an API provider and consumed by an API consumer.

API Exchange Manager

The management component of API Exchange, which enables the API management portal.

API Exchange Gateway

The API Exchange component that enforces rules and policies that govern API processing.

API Exchange Analytics

A Spotfire Server that enables reporting on API performance and usage and display of information on a dashboard.

API Explorer

An API Exchange feature that allows users to explore the APIs that have been registered with the API exchange interface, view documentation, and review developer forum information about the API.

API Key

A token assigned to an API, which is registered with API Exchange and used by application developers when calling the API.

An API key is an opaque token passed as an HTTP header or as a URL parameter with each request. For example,

`https://api.books.co/Books/Now?apikey=195-532d7700-44fe-9175-3a9d408a7286`

When calling the API, the application usually provides an API key or OAuth token to be able to access the API.

API Management Portal

The front-end, user component of TIBCO API Exchange, a developer portal that allows API providers to add APIs and combine them to create products, and allows API consumers (partners) to subscribe to the API products.

API consumer

A user who uses APIs provided by using TIBCO API Exchange Manager: an application developer.

API provider

An organization that provides APIs and licenses them in the form of products that can be subscribed to.

Application Developer

API Exchange developer user. Registers applications, tests and learns about APIs, and requests subscriptions and keys for applications.

Authentication

TIBCO API Exchange supports a range of identity and authentication mechanisms, including API keys, OAuth, mutual SSL, WS-Security (SAML, User name, X.509, LDAP), HTTP Basic, and Kerberos, but also makes it easy to integrate with existing databases and identify management systems.

TIBCO API Exchange Manager allows partners to manage their own API keys and OAuth credentials. A partner can create multiple keys.

D**Developer Portal**

A user interface that enables API consumers (application developers) to browse and test APIs. Provides API subscription self service and developer forums.

H**Host Administrators**

API Exchange users who maintain Developer Portals, create application environments and add gateways to them, create organizations and user groups, create subscriptions, and have access to full system functionality.

M**Manager**

API Exchange user. Managers add APIs and operations, create products and plans, and run analytical tests on APIs.

O**OAuth**

An authentication mechanism that is used to provide applications with access to third party resources.

P**Partner Administrators**

Create organizations and users, and manage subscriptions.

Provider Portal

A user interface that enables API providers to define and manage products, run usage analytics, and manage partners and partner communities.

Policies

Rules created using the TIBCO Api Exchange Gateway configurations to govern the use of API resources.

S**Subscription**

A subscription to a software product developed using API Exchange.

Index

C

changes from the previous release [x](#)
customer support [xv](#)

I

instance (glossary) [42](#)

P

payload (glossary) [42](#)

S

serializer (glossary) [42](#)
support, contacting [xv](#)

T

technical support [xv](#)
TIBCO_HOME [xiii](#)