JIRA Service Desk Data Model

for use with Invantive SQL

23.0

invantive
the SQL company

Copyright

Important Safety and Usage Information

Intended Use and Limitations: This software, developed by Invantive, is designed to support a variety of business and information technology data processing functions, such as accounting, financial reporting an sales reporting. It is important to note that this software is not designed, tested, or approved for use in environments where malfunction or failure could lead to life-threatening situations or severe physical or environmental damage. This includes, but is not limited to:

- Nuclear facilities: The software should not be used for operations or functions related to the control, maintenance, or operation of nuclear facilities.
- Defense and Military Applications: This software is not suitable for use in defense-related applications, including but not limited to weaponry control, military strategy planning, or any other aspects of national defense.
- Aviation: The software is not intended for use in the operation, navigation, or communication systems of any aircraft or air traffic control environments.
- Healthcare and Medicine Production: This software should not be utilized for medical device operation, patient data analysis for critical health decisions, pharmaceutical production, or medical research where its failure or malfunction could impact patient health.
- Chemical and Hazardous Material Handling: This software is not intended for the management, control, or operational aspects of chemical plants or hazardous material handling facilities. Any malfunction in software used in these settings could result in dangerous chemical spills, explosions, or environmental disasters.
- Transportation and Traffic Control Systems: The software should not be used for the control, operation, or management of transportation systems, including railway signal controls, subway systems, or traffic light management. Malfunctions in such critical systems could lead to severe accidents and endanger public safety.
- Energy Grid and Utility Control Systems: This software is not designed for the control or operation of energy grid systems, including electrical substations, renewable energy control systems, or water utility control systems. The failure of software in these areas could lead to significant power outages, water supply disruptions, or other public utility failures, potentially endangering communities and causing extensive damage.
- Other High-Risk Environments: Any other critical infrastructure and environments where a failure of the software could result in significant harm to individuals or the environment.

User Responsibility: Users must ensure that they understand the intended use of the software and refrain from deploying it in any setting that falls outside of its designed purpose. It is the responsibility of the user to assess the suitability of the software for their intended application, especially in any scenarios that might pose a risk to life, health, or the environment. Disclaimer of Liability: Invantive disclaims any responsibility for damage, injury, or legal consequences resulting from the use or misuse of this software in prohibited or unintended applications.

Contents

1 SQL Driver for JIRA Service Desk API

Invantive SQL is the fastest, easiest and most reliable way to exchange data with the JIRA Service Desk API.

Use the "Search" option in the left menu to search for a specific term such as the table or column description. When you already know the term, please use the "Index" option. When you can't find the information needed, please click on the Chat button at the bottom or place your question in the user community. Other users or Invantive Support will try to help you to our best.

JIRA is cloud software for managing business processes and sharing knowledge. JIRA Cloud offers tickets and customized workflows. JIRA Service Desk extends those by integration of external parties in business processes, such as service desk customers.

The JIRA Service Desk driver covers 27 tables and 745 columns.

JIRA Service Desk API Clients

Invantive SQL is available on many user interfaces ("clients" in traditional server-client paradigma). All Invantive SQL statements can be exchanged with a close to 100% compatibility across all clients and operating systems (Windows, MacOS, Linux, iOS, Android).

The clients include Microsoft Excel, Microsoft Power BI, Microsoft Power Query, Microsoft Word and Microsoft Outlook. Web-based clients include Invantive Cloud, Invantive Bridge Online as OData proxy, Invantive App Online for interactive apps, Online SQL Editor for query execution and Invantive Data Access Point as extended proxy.

The JIRA Service Desk Power BI connector is based on the Invantive SQL driver for JIRA Service Desk, completed by a high-performance OData connector which works straight on Power BI without any add-on. The OData protocol is always version 4, independent whether the backing platform uses OData, SOAP or another protocol.

For technical users there are command-line editions of Invantive Data Hub running on iOS, Android, Windows, MacOS and Linux. Invantive Data Hub is also often used for enterprise server applications such as ETL. High-volume replication of data taken from the JIRA Service Desk API into traditional databases such as SQL Server (on-premise and Azure), MySQL, PostgreSQL and Oracle is possible using Invantive Data Replicator. Invantive Data Replicator automatically creates and maintains JIRA Service Desk datawarehouses, possibly in combination with data from over 70 other (cloud) platforms. Data Replicator supports data volumes up to over 1 TB and over 5.000 companies. The on-premise edition of Invantive Bridge offers an JIRA Service Desk ADO.net provider.

Finally, online web apps can be build for JIRA Service Desk using App Online of Invantive Cloud.

Monitor API Calls

When a query or DML-statement has been executed on Invantive SQL a developer can evaluate the actual calls made to the JIRA Service Desk API using a query on sessionios@DataDictionary. As an alternative, extensive request and response logging can be enabled by setting log-native-calls-to-disk to true. In the %USERPROFILE%\Invantive\NativeLog folder Invantive SQL will create log files per API request and response.

Specifications

The SQL driver for JIRA Service Desk does not support partitioning. Define one data container in a database for each company in JIRA Service Desk to enable parallel access for data from multiple companies.

An introduction into the concepts of Invantive SQL such as databases, data containers and partitioning can be found in the Invantive SQL grammar.

The configuration can be changed using various attributes during log on and use. A full list of configuration options is listed in the driver attributes [2].

The catalog name is used to compose the full qualified name of an object like a table or view. The schema name is used to compose the full qualified name of an object like a table or view. On JIRA Service Desk the comparison of two texts is case sensitive by default.

Changes and bug fixes on the JIRA Service Desk SQL driver can be found in the release notes. There is currently no specific section on the Invantive forums for JIRA Service Desk. Please reach out to other users of JIRA Service Desk by leaving a question or contact request.

Driver code for use in settings.xml: `JiraServiceDesk`

Alias: `jiraservicedesk`

Recommended alias: `jsd`

Status: Non-production

More technical documentation as provided by the supplier of the JIRA Service Desk API on the native APIconnection used can be found at https://developer.atlassian.com/jiradev/jira-apis/jira-rest-apis.

General documentation on JIRA Service Desk is available at https://jira-apps.com

AuthenticationAuthentication can be done using one of the following two alternatives:1. Using the user log on code, token and server name from the JIRA URL (basic authentication).2. Using OAuth code grant flow.Authentication using user log on code and token is recommended for on-premise use. The basic authentication token can be defined on https://id.atlassian.com/manage/api-tokens. The OAuth code grant flow is recommended for online use; register a client at https://developer.atlassian.com/apps.

Updated: 15-06-2022 20:37 using Invantive SQL version 22.0.232-PROD+3445.

2 SQL Driver Attributes for JIRA Service Desk API

The SQL driver for JIRA Service Desk has many attributes that can be finetuned to improve handling in scenarios with unreliable network connections to the API server of JIRA Service Desk or high-volumes of data. Also, many drivers have driver-specific attributes to finetune actual behaviour or handle data not matching specifications.

The JIRA Service Desk driver attributes are assigned a default value which seldom requires change. However, changes can be applied when needed on four levels, which are reflected in the table below by separate checkmarks:

- Connection string: the connection string from the settings*.xml file and applied during log on.

- Set SQL statement: a set SQL-statement to be executed once connection has been established.

- Drivers file: the providers.xml file (obsolete starting release 17.32).

- Log on: value to be specified interactively by user during log on in a user interface.

The connection string for JIRA Service Desk can be found in the settings*.xml file used for the database. Settings*.xml files are typically located in the `%USERPROFILE%\invantive` folder in most deployment scenarios. The reference manuals contain instructions how to relocate the settings*.xml files. Each data container of a database in the connection string can have a `connectionString` element specifying the name and values of attributes. Both name and value must be properly escaped according to XML-semantics. Actual application of the value is solely done during log on. A new connection must be established to change the value of a driver attribute using a connection string.

The set SQL statement can be executed after log on. The syntax is: `set NAME VALUE`, or for a distributed database: `set NAME@ALIAS VALUE`. In some scenarios you may need to enclose the driver attribute name in square brackets to escape it from parsing, for instance when a reserved SQL keyword is part of the name. The new value takes effect straight after execution of the set-statement. The set-statement can be executed as often as needed during a session.

Driver attributes that can be interactively set to a value are typically presented in the log on window. Depending on the platform and design decisions of the user interface designer, some or all of the available driver attributes can have been made available.

The JIRA Service Desk driver can be configured using the following attributes:

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
add-odata-mandatory-filters	Whether to automatically add OData filters deemed necessary by the platform.	OData	False	✓	✓	✓	
analysis-enforce-row-uniqueness	Use for analysis only! Enforce rows to be unique.	Shared	False	✓	✓	✓	
api-access-token	Access Token is a security token for multiple OAuth2 Flows. With an Access Token you can access protected resources. An Access Token must be stored securely since once compromised allows access to your protected resources.	OData		✓		✓	✓
api-client-id	The client ID is a unique identifier of your application. It is generated by registering an application.	OData		✓		✓	✓
api-client-secret	The client secret is to be kept confidential. Such as a password for a logon code, the client secret is the confidential part of an app identified by a client ID. It is needed during the OAuth2 Code Grant Flow together with the refresh token to get access.	OData		✓		✓	✓
api-password-authentication	Use username/token basic authentication when true. OAuth otherwise.			✓		✓	
api-pre-expiry-refresh-sec	The number of seconds before the token expires to acquire a new	OData		✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
	token.						
api-redirect-url	The redirect URI is the w ebsite a brow ser session is redirected to after the OAuth2 authentication process has been completed.	OData		✓		✓	✓
api-refresh-token	Refresh Token is a security token for the OAuth2 Code Grant Flow . With a Refresh Token and client secret you can retrieve a renew ed access token to access protected resources. A Refresh Token and client secret must be stored securely since once compromised allow s access to your protected resources.	OData		✓		✓	✓
api-scope	The scope to request an OAuth token for.	OData		✓		✓	
api-token-url	The token URI is the OAuth2 endpoint to exchange tokens.	OData		✓		✓	
api-url	URL to access the API.	OData		✓		✓	
bulk-delete-page-size-row s	Number of row s to delete per batch w hen bulk deleting	Shared	10000	✓	✓	✓	
bulk-insert-page-size-bytes	Approximate maximum size in bytes of batch w hen bulk inserting	Shared	10000000	✓	✓	✓	
bulk-insert-page-size-row s	Number of row s to insert per batch w hen bulk inserting	Shared	250	✓	✓	✓	
dow nload-error-400-bad-request-max-tries	Maximum number of tries w hen OData server reports bad format during retrieval of data.		3	✓	✓	✓	
dow nload-error-400-bad-request-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen OData server reports that the API server is unavailable during retrieval of data.		500	✓	✓	✓	
dow nload-error-400-bad-request-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen OData server reports that the API server is unavailable during retrieval of data.		5000	✓	✓	✓	
dow nload-error-400-bad-request-sleep-multiplicator	Multiplication factor for sleep betw een retries OData server reports that the API server is unavailable during retrieval of data.		2	✓	✓	✓	
dow nload-error-408-request-timeout-max-tries	Maximum number of tries w hen the w ebsite reports a HTTP status 408.		10	✓	✓	✓	
dow nload-error-408-request-timeout-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen the w ebsite reports a HTTP status 408.		10000	✓	✓	✓	
dow nload-error-408-request-timeout-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen the w ebsite reports a HTTP status 408.		300000	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Conne ction String	Set from Set SQL-Statem ent	Set from Driver s File	Set from Log On
dow nload-error-408-request-timeout-sleep-multiplicator	Multiplication factor for sleep betw een retries w hen the w ebsite reports a HTTP status 408.		2	✓	✓	✓	
dow nload-error-422-bad-request-max-tries	Maximum number of tries w hen OData server reports unprocessable entity during retrieval of data.		30	✓	✓	✓	
dow nload-error-422-bad-request-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen OData server reports unprocessable entity during retrieval of data.		10000	✓	✓	✓	
dow nload-error-422-bad-request-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen OData server reports unprocessable entity during retrieval of data.		300000	✓	✓	✓	
dow nload-error-422-bad-request-sleep-multiplicator	Multiplication factor for sleep betw een retries OData server reports unprocessable entity during retrieval of data.		2	✓	✓	✓	
dow nload-error-429-too-many-requests-max-tries	Maximum number of tries w hen the w ebsite reports that too many requests have been made during a timeslot of one minute or one day.		10	✓	✓	✓	
dow nload-error-429-too-many-requests-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen the w ebsite reports that too many requests have been made during a timeslot of one minute or one day.		10000	✓	✓	✓	
dow nload-error-429-too-many-requests-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen the w ebsite reports that too many requests have been made during a timeslot of one minute or one day.		300000	✓	✓	✓	
dow nload-error-429-too-many-requests-sleep-multiplicator	Multiplication factor for sleep betw een retries w hen the w ebsite reports that too many requests have been made during a timeslot of one minute or one day.		2	✓	✓	✓	
dow nload-error-502-server-unavailable-max-tries	Maximum number of tries w hen OData server reports a bad gatew ay during retrieval of data.		30	✓	✓	✓	
dow nload-error-502-server-unavailable-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen OData server reports a bad gatew ay during retrieval of data.		10000	✓	✓	✓	
dow nload-error-502-server-unavailable-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen OData server reports that a bad gatew ay during retrieval of data.		300000	✓	✓	✓	
dow nload-error-502-server-	Multiplication factor for sleep betw een retries OData server		2	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
unavailable-sleep-multiplicator	reports a bad gateway during retrieval of data.						
download-error-503-server-unavailable-max-tries	Maximum number of tries when OData server reports that the API server is unavailable during retrieval of data.		30	✓	✓	✓	
download-error-503-server-unavailable-sleep-initial-ms	Initial sleep in milliseconds between retries when OData server reports that the API server is unavailable during retrieval of data.		10000	✓	✓	✓	
download-error-503-server-unavailable-sleep-max-ms	Maximum sleep in milliseconds between retries when OData server reports that the API server is unavailable during retrieval of data.		300000	✓	✓	✓	
download-error-503-server-unavailable-sleep-multiplicator	Multiplication factor for sleep between retries OData server reports that the API server is unavailable during retrieval of data.		2	✓	✓	✓	
download-error-504-gateway-timeout-max-tries	Maximum number of tries when the website reports a gateway timeout.		10	✓	✓	✓	
download-error-504-gateway-timeout-sleep-initial-ms	Initial sleep in milliseconds between retries when the website reports a gateway timeout.		10000	✓	✓	✓	
download-error-504-gateway-timeout-sleep-max-ms	Maximum sleep in milliseconds between retries when the website reports a gateway timeout.		300000	✓	✓	✓	
download-error-504-gateway-timeout-sleep-multiplicator	Multiplication factor for sleep between retries when the website reports a gateway timeout.		2	✓	✓	✓	
download-error-590-network-connect-timeout-max-tries	Maximum number of tries when the website reports a HTTP status 590.		10	✓	✓	✓	
download-error-590-network-connect-timeout-sleep-initial-ms	Initial sleep in milliseconds between retries when the website reports a HTTP status 590.		10000	✓	✓	✓	
download-error-590-network-connect-timeout-sleep-max-ms	Maximum sleep in milliseconds between retries when the website reports a HTTP status 590.		300000	✓	✓	✓	
download-error-590-network-connect-timeout-sleep-multiplicator	Multiplication factor for sleep between retries when the website reports a HTTP status 590.		2	✓	✓	✓	
download-error-599-network-	Maximum number of tries when the website reports a HTTP status 599.		10	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
connect-timeout-max-tries							
dow nload-error-599-netw ork-connect-timeout-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen the w ebsite reports a HTTP status 599.		10000	✓	✓	✓	
dow nload-error-599-netw ork-connect-timeout-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen the w ebsite reports a HTTP status 599.		300000	✓	✓	✓	
dow nload-error-599-netw ork-connect-timeout-sleep-multiplicator	Multiplication factor for sleep betw een retries w hen the w ebsite reports a HTTP status 599.		2	✓	✓	✓	
dow nload-error-argument-exception-max-tries	Maximum number of tries w hen an argument exception is returned w hen dow nloading a blob.		10	✓	✓	✓	
dow nload-error-argument-exception-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen an argument exception is returned w hen dow nloading a blob.		10000	✓	✓	✓	
dow nload-error-argument-exception-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen an argument exception is returned w hen dow nloading a blob.		300000	✓	✓	✓	
dow nload-error-argument-exception-sleep-multiplicator	Multiplication factor for sleep betw een retries w hen an argument exception is returned w hen dow nloading a blob.		2	✓	✓	✓	
dow nload-error-internet-dow n-max-tries	Maximum number of tries w hen the Internet connection seems dow n during retrieval of data.		10	✓	✓	✓	
dow nload-error-internet-dow n-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen the Internet connection seems dow n during retrieval of data.		10000	✓	✓	✓	
dow nload-error-internet-dow n-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen the Internet connection seems dow n during retrieval of data.		300000	✓	✓	✓	
dow nload-error-internet-dow n-sleep-multiplicator	Multiplication factor for sleep betw een retries w hen the Internet connection seems dow n during retrieval of data.		2	✓	✓	✓	
dow nload-error-io-exception-max-tries	Maximum number of tries w hen a netw ork I/O connection failure occurs during retrieval of data.		10	✓	✓	✓	
dow nload-error-io-exception-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen a netw ork I/O connection failure occurs during retrieval of data.		10000	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
dow nload-error-io-exception-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen a netw ork I/O connection failure occurs during retrieval of data.		300000	✓	✓	✓	
dow nload-error-io-exception-sleep-multiplicator	Multiplication factor for sleep betw een retries w hen a netw ork I/O connection failure occurs during retrieval of data.		2	✓	✓	✓	
dow nload-error-json-exception-max-tries	Maximum number of tries w hen an invalid JSON body is returned.		3	✓	✓	✓	
dow nload-error-json-exception-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen an invalid JSON body is returned.		1000	✓	✓	✓	
dow nload-error-json-exception-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen an invalid JSON body is returned.		10000	✓	✓	✓	
dow nload-error-json-exception-sleep-multiplicator	Multiplication factor for sleep betw een retries w hen an invalid JSON body is returned.		2	✓	✓	✓	
dow nload-error-other-exception-max-tries	Maximum number of tries w hen an unqualified error occurs during retrieval of data.		3	✓	✓	✓	
dow nload-error-other-exception-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen an unqualified error ocurrs during retrieval of data.		10000	✓	✓	✓	
dow nload-error-other-exception-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen an unqualified error ocurrs during retrieval of data.		300000	✓	✓	✓	
dow nload-error-other-exception-sleep-multiplicator	Multiplication factor for sleep betw een retries w hen an unqualified error ocurrs during retrieval of data.		2	✓	✓	✓	
dow nload-error-socket-exception-max-tries	Maximum number of tries w hen the netw ork connection is forcible dropped during retrieval of data.		10	✓	✓	✓	
dow nload-error-socket-exception-sleep-initial-ms	Initial sleep in milliseconds betw een retries w hen the netw ork connection is forcible dropped during retrieval of data.		10000	✓	✓	✓	
dow nload-error-socket-exception-sleep-max-ms	Maximum sleep in milliseconds betw een retries w hen the netw ork connection is forcible dropped during retrieval of data.		300000	✓	✓	✓	
dow nload-error-socket-exception-sleep-multiplicator	Multiplication factor for sleep betw een retries w hen the netw ork connection is forcible dropped during retrieval of data.		2	✓	✓	✓	
dow nload-error-w eb-exception-max-tries	Maximum number of tries w hen a w eb connection failure occurs during retrieval of data.		10	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
download-error-web-exception-sleep-initial-ms	Initial sleep in milliseconds between retries when a web connection failure occurs during retrieval of data.		10000	✓	✓	✓	
download-error-web-exception-sleep-max-ms	Maximum sleep in milliseconds between retries when a web connection failure occurs during retrieval of data.		300000	✓	✓	✓	
download-error-web-exception-sleep-multiplicator	Multiplication factor for sleep between retries when a web connection failure occurs during retrieval of data.		2	✓	✓	✓	
download-error-web-not-implemented-max-tries	Maximum number of tries when the connection reports not implemented.		1	✓	✓	✓	
download-error-web-not-implemented-sleep-initial-ms	Initial sleep in milliseconds between retries when the connection reports not implemented.		10000	✓	✓	✓	
download-error-web-not-implemented-sleep-max-ms	Maximum sleep in milliseconds between retries when the connection reports not implemented.		300000	✓	✓	✓	
download-error-web-not-implemented-sleep-multiplicator	Multiplication factor for sleep between retries when the connection reports not implemented.		2	✓	✓	✓	
download-error-web-timeout-max-tries	Maximum number of tries when the connection reports a timeout.		10	✓	✓	✓	
download-error-web-timeout-sleep-initial-ms	Initial sleep in milliseconds between retries when the connection reports a timeout.		1000	✓	✓	✓	
download-error-web-timeout-sleep-max-ms	Maximum sleep in milliseconds between retries when the connection reports a timeout.		30000	✓	✓	✓	
download-error-web-timeout-sleep-multiplicator	Multiplication factor for sleep between retries when the connection reports a timeout.		2	✓	✓	✓	
download-error-web-unauthorized-max-tries	Maximum number of tries when the connection reports an unauthorized error.		1	✓	✓	✓	
download-error-web-unauthorized-sleep-initial-ms	Initial sleep in milliseconds between retries when the connection reports an unauthorized error.		10000	✓	✓	✓	
download-error-web-unauthorized-sleep-max-ms	Maximum sleep in milliseconds between retries when the connection reports an unauthorized error.		300000	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
download-error-web-unauthorized-sleep-multiplicator	Multiplication factor for sleep between retries when the connection reports an unauthorized error.		2	✓	✓	✓	
force-case-sensitive-identifiers	Consider identifiers as case-sensitive independent of the platform capabilities.	Shared	False	✓	✓	✓	
forced-casing-identifiers	Forced casing of identifiers. Choose from Unset, Lower, Upper and Mixed.	Shared		✓	✓	✓	
http-disk-cache-compression-level	Compression level for the HTTP disk cache, ranging from 1 (little) to 9 (intense). Default is 5.	Shared	5	✓	✓	✓	
http-disk-cache-directory	Directory where HTTP cache is stored.	Shared	C:\Users\gle3.WS212\Invantive\Cache\http\gle3\shared	✓	✓	✓	
http-disk-cache-ignore-write-errors	Whether to ignore write errors to disk cache.	Shared	False	✓	✓	✓	
http-disk-cache-max-age-sec	Maximum acceptable age in seconds for use of data in the HTTP disk cache.	Shared	2592000	✓	✓	✓	
http-get-timeout-max-ms	HTTP GET maximum timeout on retry (ms).		300000	✓	✓	✓	
http-get-timeout-ms	HTTP GET timeout (ms).		60000	✓	✓	✓	
http-memory-cache-compression-level	Compression level for the HTTP memory cache, ranging from 1 (little) to 9 (intense). Default is 5.	OData	5	✓	✓	✓	
http-memory-cache-max-age-sec	Maximum acceptable age in seconds for use of data in the HTTP memory cache.	OData	14400	✓	✓	✓	
http-post-timeout-max-ms	HTTP POST maximum timeout on retry (ms).		300000	✓	✓	✓	
http-post-timeout-ms	HTTP POST timeout (ms).		300000	✓	✓	✓	
ignore-http-400-errors	Ignore HTTP 400 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
ignore-http-401-errors	Ignore HTTP 401 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
ignore-http-402-errors	Ignore HTTP 402 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
ignore-http-403-errors	Ignore HTTP 403 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
ignore-http-404-errors	Ignore HTTP 404 errors when exchanging results with the OData		False	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
	endpoint.						
ignore-http-422-errors	Ignore HTTP 422 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
ignore-http-429-errors	Ignore HTTP 429 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
ignore-http-500-errors	Ignore HTTP 500 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
ignore-http-502-errors	Ignore HTTP 502 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
ignore-http-503-errors	Ignore HTTP 503 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
invalid-json-on-get-max-tries	Maximum number of tries when the JSON received on GET is invalid.		10	✓	✓	✓	
invalid-json-on-get-sleep-initial-ms	Initial sleep in milliseconds between retries when the JSON received on GET is invalid.		10000	✓	✓	✓	
invalid-json-on-get-sleep-max-ms	Maximum sleep in milliseconds between retries when the JSON received on GET is invalid.		300000	✓	✓	✓	
invalid-json-on-get-sleep-multiplicator	Multiplication factor for sleep between retries when the JSON received on GET is invalid.		2	✓	✓	✓	
invalid-json-on-post-max-tries	Maximum number of tries when the JSON received on POST is invalid.		1	✓	✓	✓	
invalid-json-on-post-sleep-initial-ms	Initial sleep in milliseconds between retries when the JSON received on POST is invalid.		10000	✓	✓	✓	
invalid-json-on-post-sleep-max-ms	Maximum sleep in milliseconds between retries when the JSON received on POST is invalid.		300000	✓	✓	✓	
invalid-json-on-post-sleep-multiplicator	Multiplication factor for sleep between retries when the JSON received on POST is invalid.		2	✓	✓	✓	
invantive-sql-compress-sparse-arrays	Whether to compress sparse arrays in result sets during compression.	SQL Engine V1	True	✓	✓	✓	
invantive-sql-correct-invalid-date	Whether to correct dates considered invalid since they are before 01-01-1753. When nullable, they are removed. Otherwise they are replaced by 01-01-1753.	SQL Engine V1	False	✓	✓	✓	
invantive-sql-forward-filters-to-data-containers	Whether to forward filters to data containers.	SQL Engine V1	True	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
invantive-sql-share-byte-arrays	Whether to share the memory used by identical byte arrays in result sets during compression.	SQL Engine V1	True	✓	✓	✓	
invantive-sql-share-strings	Whether to share the memory used by identical strings in result sets during compression.	SQL Engine V1	True	✓	✓	✓	
invantive-sql-shuffle-fetch-results-data-containers	Whether to shuffle results fetched from data containers.	SQL Engine V1	False	✓	✓	✓	
invantive-use-cache	Whether to cache the results of a query.	SQL Engine V1	True	✓	✓	✓	
join-set-points-per-request	Maximum number of values in a request when executing a join set.	OData	60	✓	✓	✓	
limit-partition-calls-left	Minimum number of remaining API calls on a partition towards a hard limit. When below, an error is raised.	OData	500	✓	✓	✓	
log-native-calls-to-disk-max-events	Maximum number of events to register from last activation.	Shared		✓	✓	✓	
log-native-calls-to-disk-max-seconds	Maximum number of seconds to register from last activation.	Shared		✓	✓	✓	
log-native-calls-to-disk-on-error	Registers native calls to data container backend as disk files when an error occurred.	Shared	False	✓	✓	✓	
log-native-calls-to-disk-on-success	Registers native calls to data container backend as disk files when successful.	Shared	False	✓	✓	✓	
log-native-calls-to-trace	Log native calls to data container backend on the trace.	Shared	False	✓	✓	✓	
maximum-length-identifiers	Non-default maximum length in characters of identifier names.	Shared		✓	✓	✓	
max-odata-filters	The maximum number of OData filter elements.	OData	100	✓	✓	✓	
max-url-length-accepted	The maximum accepted URL length before raising an error.	Shared	8000	✓	✓	✓	
max-url-length-desired	The maximum desired URL length.	Shared	8000	✓	✓	✓	
metadata-cache-max-age-sec	Maximum acceptable age in seconds for re-use of metadata.	OData		✓	✓	✓	
oauth-unauthorized-max-tries	Maximum number of tries when an OAuth exception occurs.	OData	2	✓	✓	✓	
oauth-unauthorized-sleep-initial-ms	Initial sleep in milliseconds between OAuth reauthentication tries when the OAuth authentication fails.	OData	10000	✓	✓	✓	
oauth-unauthorized-sleep-max-ms	Maximum sleep in milliseconds between OAuth reauthentication tries when the OAuth authentication fails.	OData	1000	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
oauth-unauthorized-sleep-multiplicator	Multiplication factor for sleep between OAuth reauthentication tries when the OAuth authentication fails.	OData	2	✓	✓	✓	
partition-slot-based-rate-limit-length-ms	Total length in ms across all slots of a partition-based rate limit.	Shared	60000	✓		✓	
partition-slot-based-rate-limit-slots	Number of slots per partition-based rate limit. Null means no slot-based rate limit	Shared		✓		✓	
pre-request-delay-ms	Pre-request delay in milliseconds per request.	Shared	0	✓	✓	✓	
requested-page-size	Preferred number of rows to exchange per round trip; only effective on limited platforms such as AFAS Online	Shared		✓	✓	✓	
requests-parallel-max	Maximum number of parallel data requests from individual partitions on the data container.	Shared	32	✓	✓	✓	
server	Server name (basic authentication only).			✓		✓	✓
simulate-http-400-errors	Simulate HTTP 400 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
simulate-http-400-errors-percentage	Percentage of simulated HTTP 400 errors when exchanging results with the OData endpoint.		0	✓	✓	✓	
simulate-http-401-errors	Simulate HTTP 401 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
simulate-http-401-errors-percentage	Percentage of simulated HTTP 401 errors when exchanging results with the OData endpoint.		0	✓	✓	✓	
simulate-http-403-errors	Simulate HTTP 403 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
simulate-http-403-errors-percentage	Percentage of simulated HTTP 403 errors when exchanging results with the OData endpoint.		0	✓	✓	✓	
simulate-http-408-errors	Simulate HTTP 408 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
simulate-http-408-errors-percentage	Percentage of simulated HTTP 408 errors when exchanging results with the OData endpoint.		0	✓	✓	✓	
simulate-http-429-errors	Simulate HTTP 429 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
simulate-http-429-errors-percentage	Percentage of simulated HTTP 429 errors when exchanging results with the OData endpoint.		0	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
simulate-http-500-errors	Simulate HTTP 500 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
simulate-http-500-errors-percentage	Percentage of simulated HTTP 500 errors when exchanging results with the OData endpoint.		0	✓	✓	✓	
simulate-http-502-errors	Simulate HTTP 502 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
simulate-http-502-errors-percentage	Percentage of simulated HTTP 502 errors when exchanging results with the OData endpoint.		0	✓	✓	✓	
simulate-http-503-errors	Simulate HTTP 503 errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
simulate-http-503-errors-percentage	Percentage of simulated HTTP 503 errors when exchanging results with the OData endpoint.		0	✓	✓	✓	
simulate-http-protocol-errors	Simulate HTTP protocol errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
simulate-http-protocol-errors-percentage	Percentage of simulated HTTP protocol errors when exchanging results with the OData endpoint.		0	✓	✓	✓	
simulate-http-timeout-errors	Simulate HTTP timeout errors when exchanging results with the OData endpoint.		False	✓	✓	✓	
simulate-http-timeout-errors-percentage	Percentage of simulated HTTP timeout errors when exchanging results with the OData endpoint.		0	✓	✓	✓	
slot-based-rate-limit-length-ms	Total length in ms across all slots of a slot-based rate limit.	Shared	60000	✓		✓	
slot-based-rate-limit-slots	Number of slots of a slot-based rate limit. Null means no slot-based rate limit	Shared		✓		✓	
standardize-identifiers	Rewrite all identifiers to the preferred standards as configured by standardize-identifiers-casing and maximum-length-identifiers.	Shared	True	✓	✓	✓	
standardize-identifiers-casing	Rewrite all identifiers to the recommended standard platform-specific casing when changing a data model on a case-dependent platform.	Shared	True	✓	✓	✓	
use-batch-insert	Whether to use batch insert.	OData	True	✓	✓	✓	
use-http-disk-cache-read	Whether to use HTTP responses from previous queries stored on disk to answer the current query.	Shared	True	✓	✓	✓	
use-http-disk-cache-write	Whether to memorize HTTP responses on disk.	Shared	True	✓	✓	✓	

Code	Description	Origin	Default Value	Set from Connection String	Set from Set SQL-Statement	Set from Drivers File	Set from Log On
use-http-memory-cache-read	Whether to use HTTP responses from previous queries stored in memory that can answer the current query.	OData	True	✓	✓	✓	
use-http-memory-cache-write	Whether to memorize HTTP responses from previous queries for use by future queries.	OData	True	✓	✓	✓	

3 Schema: Native
3.1 Tables
3.1.1 NATIVEPLATFORMSCALARREQUESTS: JIRA Service Desk Native Platform Scalar Requests

Direct access to native API.

Catalog: JIRA

Schema: Native

Alias: `npt`

Label: Native Platform Scalar Requests

Documentation:

The NativePlatformScalarRequests table provides direct access to the native API protocol over an established connection to the JIRA Service Desk API server. It will contain a new row for every row inserted with a native API request in PAYLOAD_TEXT with the results of unaltered forwarding of the payload to the JIRA Service Desk API server.

Retrieve: true

Insert: true

Update: false

Delete: false

View Columns

The columns of the view NATIVEPLATFORMSCALARREQUESTS are shown below. Each column has an SQL data type. A new non-null value must be provided for every required column at all times during insert.

Name	Data Type	Label	Required	Documentation
BLOB_PREFERRED	boolean	BLOB Preferred	☑	Indicator whether a BLOB result is preferred over text.
BOL_RESPONSE_CACHE_MAX_AGE_SEC	int32	Response Cache Maximum Age (sec)	☐	Maximum age in seconds of Bridge Online response cache entries to be used.
CONTENT_TYPE	string(240)	Content Type	☐	

Name	Data Type	Label	Required	Documentation
DATE_ENDED	datetime	End Date	☑	
DATE_STARTED	datetime	Start Date	☑	
DRY_RUN	boolean	Run without Actions	☑	
DURATION_MS	int32	Duration (ms)	☑	
ERROR_MESSAGE_CODE	string(30)	Error Message Code	☐	
ERROR_MESSAGE_TEXT	string(32000)	Error Message Text	☐	
FAIL_ON_ERROR	boolean	Fail on Error	☑	Whether to raise an exception when processing the native request triggered an error from the provider.
HTTP_DISK_CACHE_MAX_AGE_SEC	int32	HTTP Disk Cache Maximum Age (sec)	☐	Maximum age in seconds of HTTP disk cache entries to be used.
HTTP_DISK_CACHE_SAVE	boolean	Save HTTP Disk Cache	☐	Whether results can be stored in HTTP disk cache.
HTTP_DISK_CACHE_USE	boolean	Use HTTP Disk Cache	☐	Whether results can be fetched from HTTP disk cache.
HTTP_MEMORY_CACHE_MAX_AGE_SEC	int32	HTTP Memory Cache Maximum Age (sec)	☐	Maximum age in seconds of HTTP memory cache entries to be used.
HTTP_MEMORY_CACHE_SAVE	boolean	Save HTTP Memory Cache	☐	Whether results can be stored in HTTP memory cache.
HTTP_MEMORY_CACHE_USE	boolean	Use HTTP Memory Cache	☐	Whether results can be fetched from HTTP memory cache.
HTTP_METHOD	string(30)	HTTP Method	☐	
HTTP_STATUS_CODE	int16	HTTP Status Code	☐	
ORIG_SYSTEM_GROUP	string(4000)	Original System Group	☐	
ORIG_SYSTEM_REFERENCE	string(4000)	Original System Reference	☐	
PAYLOAD_TEXT	string	Payload	☐	
RESULT_BLOB	byte[]	Result BLOB	☐	
RESULT_DATE_TIME_UTC	datetime		☐	
RESULT_NUMBER	decimal		☐	
RESULT_TEXT	string	Result Text	☐	
SUCCESSFUL	boolean	Succesful	☑	
TIMEOUT_SEC	int32	Timeout (sec)	☐	Timeout in seconds.
TRANSACTION_ID	int32	Transaction ID	☑	Incrementing ID of the transaction.
URL	string(4000)	URL	☐	

4 Schema: Service
4.1 Tables
4.1.1 ArticlesByQuery: JIRA Service Desk Articles by Query

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Articles by Query

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/knowledgebase/article`

Insert JIRA Service Desk API URL: `/knowledgebase/article`

Update JIRA Service Desk API URL: `/knowledgebase/article`

Delete JIRA Service Desk API URL: `/knowledgebase/article`

Field Selection Method: NotRequired

Base Path: `values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function ArticlesByQuery. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
highlight	boolean	☐	False	If set to true matching query term in the title and excerpt will be highlighted using the {@code @@@hl@@@term@@@endhl @@@} syntax. Default: false.
highlight	boolean	☐	False	If set to true matching query term in the title and excerpt will be highlighted using the {@code @@@hl@@@term@@@endhl @@@} syntax. Default: false.
query	string	☑		The string used to filter the articles.

Table Function Columns

The columns of the table function `ArticlesByQuery` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
content_iframeSrc	string		☐	URL containing the body of the article (w ithout title), suitable for rendering in an iframe.
excerpt	string		☐	Excerpt of the article w hich matches the given query string.
source_type	string		☐	Type of the know ledge base source.
title	string		☐	Title of the article.

4.1.2 ArticlesByServiceDeskIdAndQuery: JIRA Service Desk Articles by Service Desk ID or Query

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Articles by Service Desk ID or Query

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/servicedesk/{serviceDeskId}/knowledgebase/article`

Insert JIRA Service Desk API URL: `/servicedesk/{serviceDeskId}/knowledgebase/article`

Update JIRA Service Desk API URL: `/servicedesk/{serviceDeskId}/knowledgebase/article`

Delete JIRA Service Desk API URL: `/servicedesk/{serviceDeskId}/knowledgebase/article`

Field Selection Method: NotRequired

Base Path: `values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function ArticlesByServiceDeskIdAndQuery. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
query	string	☑		The string used to filter the articles.
serviceDeskId	int64	☑		The ID of the service desk to return. This can alternatively be a project identifier.

Table Function Columns

The columns of the table function `ArticlesByServiceDeskIdAndQuery` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
content_iframeSrc	string		☐	URL containing the body of the article (without title), suitable for rendering in an iframe.
excerpt	string		☐	Excerpt of the article which matches the given query string.
source_type	string		☐	Type of the knowledge base source.
title	string		☐	Title of the article.

4.1.3 CustomersByServiceDeskId: JIRA Service Desk Customers by Service Desk ID

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Customers by Service Desk ID

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: /servicedesk/{serviceDeskId}/customer

Insert JIRA Service Desk API URL: /servicedesk/{serviceDeskId}/customer

Update JIRA Service Desk API URL: /servicedesk/{serviceDeskId}/customer

Delete JIRA Service Desk API URL: /servicedesk/{serviceDeskId}/customer

Field Selection Method: NotRequired

Base Path: `values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function CustomersByServiceDeskId. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
serviceDeskId	int64	☑		The ID of the service desk to return. This can alternatively be a project identifier.

Table Function Columns

The columns of the table function `CustomersByServiceDeskId` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk

Name	Data Type	Label	Required	Documentation
active	boolean		☐	Indicates w hether the user is active.
displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
key	string		☐	This property is no longer available and w ill be removed from the documentation soon
locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
name	string		☐	This property is no longer available and w ill be removed from the documentation soon.
timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.

4.1.4 Information: JIRA Service Desk Information

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Information

This is a read-only table. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/info`

Insert JIRA Service Desk API URL: `/info`

Update JIRA Service Desk API URL: `/info`

Delete JIRA Service Desk API URL: `/info`

Field Selection Method: NotRequired

Table Columns

The columns of the table `Information` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
buildChangeSet	string		☐	Reference of the change set included in the build.
buildDate	datetime		☐	Date of the current build.
isLicensedForUse	boolean		☐	Indicates whether the instance is licensed (true) or not (false).
platformVersion	string		☐	Jira Platform version upon which Service Desk is based.
version	string		☐	Jira Service Desk version.

4.1.5 Organizations: JIRA Service Desk Organizations

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Organizations

This is a read-only table. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/organization`

Insert JIRA Service Desk API URL: `/organization`

Update JIRA Service Desk API URL: `/organization`

Delete JIRA Service Desk API URL: `/organization`

Field Selection Method: NotRequired

Base Path: `values[*]`

Table Columns

The columns of the table `Organizations` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
id	string		☐	A unique system generated ID for the organization.
name	string		☐	Name of the organization.

4.1.6 QueuesByServiceDeskId: JIRA Service Desk Queues by Service Desk ID

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Queues by Service Desk ID

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/servicedesk/{serviceDeskId}/queue`

Insert JIRA Service Desk API URL: `/servicedesk/{serviceDeskId}/queue`

Update JIRA Service Desk API URL: `/servicedesk/{serviceDeskId}/queue`

Delete JIRA Service Desk API URL: `/servicedesk/{serviceDeskId}/queue`

Field Selection Method: NotRequired

Base Path: `values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function QueuesByServiceDeskId. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
includeCount	boolean	☐	False	Specifies whether to include each queue's customer request (issue) count in the response.
serviceDeskId	int64	☑		The ID of the service desk to return. This can alternatively be a project identifier.

Table Function Columns

The columns of the table function `QueuesByServiceDeskId` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
id	string		☐	ID for the queue.
issueCount	int64		☐	The count of customer requests in the queue.
jql	string		☐	JQL query that filters requests for the queue.
name	string		☐	Short name for the queue.

4.1.7 RequestAttachments: JIRA Service Desk Request Attachments

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Attachments

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request?`
`expand=requestType,serviceDesk,action,attachment`

Insert JIRA Service Desk API URL: `/request?`
`expand=requestType,serviceDesk,action,attachment`

Update JIRA Service Desk API URL: `/request?`
`expand=requestType,serviceDesk,action,attachment`

Delete JIRA Service Desk API URL: `/request?`
`expand=requestType,serviceDesk,action,attachment`

Field Selection Method: NotRequired

Base Path: `values[*].attachments.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestAttachments. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
approvalStatus	string	☐		Filters results to customer requests based on their approval status:MY_PENDING_APPROVAL returns customer requests pending the user's approval.MY_HISTORY_APPROVAL returns customer requests where the user was an approver.Note: Valid only when

Name	Data Type	Required	Default Value	Documentation
				used with requestOwnership = APPROVER.
organizationId	int32	☐		Filters customer requests that belong to a specific organization (note that the user must be a member of that organization). Note: Valid only when used with requestOwnership=ORGANIZATION.
requestOwnership	string	☐		Filters customer requests using the following values: OWNED_REQUESTS returns customer requests where the user is the creator. PARTICIPATED_REQUESTS returns customer requests where the user is a participant. ORGANIZATION returns customer requests for an organization of which the user is a member when used in conjunction with organizationId. ALL_ORGANIZATIONS returns customer requests that belong to all organizations of which the user is a member. APPROVER returns customer requests where the user is an approver. Can be used in conjunction with approvalStatus to filter pending or complete approvals.
requestStatus	string	☐		Filters customer requests where the request is closed, open, or either of the two where: CLOSED_REQUESTS returns customer requests that are closed. OPEN_REQUESTS returns customer requests that are open. ALL_REQUESTS returns all customer requests.
requestTypeId	int32	☐		Filters customer requests by request type. Note that the serviceDeskId must be specified for the service desk in which the request type belongs.
searchTerm	string	☐		Filters customer requests where the request summary matches the searchTerm. Wildcards can be used in the searchTerm parameter.
serviceDeskId	int32	☐		Filters customer requests by service desk.

Table Function Columns

The columns of the table function `RequestAttachments` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
author_accountId	string		☐	The account ID of the user, w hich uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID w ith value unknow n is returned w hen there is missing data for a user, w hich only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
author_accountType	string		☐	The user account type. Can take the follow ing values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
author_active	boolean		☐	Indicates w hether the user is active.
author_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
author_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
author_key	string		☐	This property is no longer available and w ill be removed from the documentation soon
author_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
author_name	string		☐	This property is no longer available and w ill be removed from the documentation soon.
author_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
created	datetime		☐	Date the attachment w as added.
filename	string		☐	Filename of the item attached.
mimeType	string		☐	MIME type of the attachment.
request_actions_addAttachment_a llow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
request_actions_addComment_allo w ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).

Name	Data Type	Label	Required	Documentation
request_actions_addParticipant_allow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
request_actions_removeParticipant_allow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on w hich the request w as created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCate gory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on w hich the status w as attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, w hich uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID w ith value unknow n is returned w hen there is missing data for a user, w hich only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the follow ing values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
request_reporter_active	boolean		☐	Indicates w hether the user is active.
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and w ill be removed from the documentation soon

Name	Data Type	Label	Required	Documentation
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.
size	int64		☐	Size of the attachment in bytes.

4.1.8 RequestAttachmentsByIdOrKey: JIRA Service Desk Request Attachments by ID or Key

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Attachments by ID or Key

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case,

please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,attachment`

Insert JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,attachment`

Update JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,attachment`

Delete JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,attachment`

Field Selection Method: NotRequired

Base Path: `attachments.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestAttachmentsByIdOrKey. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
issueIdOrKey	string	☑		The ID or key to be retrieved.

Table Function Columns

The columns of the table function `RequestAttachmentsByIdOrKey` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
author_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this

Name	Data Type	Label	Required	Documentation
				value as an error case if you need to access more information about the user.
author_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
author_active	boolean		☐	Indicates whether the user is active.
author_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
author_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
author_key	string		☐	This property is no longer available and will be removed from the documentation soon
author_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
author_name	string		☐	This property is no longer available and will be removed from the documentation soon.
author_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
created	datetime		☐	Date the attachment was added.
filename	string		☐	Filename of the item attached.
mimeType	string		☐	MIME type of the attachment.
request_actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addComment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_removeParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on which the request was created.
request_currentStatus_status	string		☐	Name of the status condition.

Name	Data Type	Label	Required	Documentation
request_currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on which the status was attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
request_reporter_active	boolean		☐	Indicates whether the user is active.
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and will be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.

Name	Data Type	Label	Required	Documentation
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.
size	int64		☐	Size of the attachment in bytes.

4.1.9 RequestByIdOrKey: JIRA Service Desk Request by ID or Key

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request by ID or Key

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action`

Insert JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action`

Update JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action`

Delete JIRA Service Desk API URL: `/request/{issueIdOrKey}?`
`expand=requestType,serviceDesk,action`

Field Selection Method: NotRequired

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestByIdOrKey. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
issueIdOrKey	string	☑		The ID or key to be retrieved.

Table Function Columns

The columns of the table function `RequestByIdOrKey` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
actions_addComment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
actions_addParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
actions_removeParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
createdDate	datetime		☐	Date on which the request was created.
currentStatus_status	string		☐	Name of the status condition.
currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
currentStatus_statusDate	datetime		☐	Date on which the status was attained.

Name	Data Type	Label	Required	Documentation
issueId	string		☐	ID of the request, as the peer issue ID.
issueKey	string		☐	Key of the request, as the peer issue key.
reporter_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
reporter_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
reporter_active	boolean		☐	Indicates whether the user is active.
reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
reporter_key	string		☐	This property is no longer available and will be removed from the documentation soon
reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
reporter_name	string		☐	This property is no longer available and will be removed from the documentation soon.
reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
requestType_description	string		☐	Description of the request type.
requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.

Name	Data Type	Label	Required	Documentation
requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
requestType_helpText	string		☐	Help text for the request type.
requestType_icon_id	string		☐	ID of the request type icon.
requestType_id	string		☐	ID for the request type.
requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
requestType_name	string		☐	Short name for the request type.
requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
requestTypeId	string		☐	ID of the request type for the request.
serviceDesk_id	string		☐	ID of the service desk.
serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
serviceDesk_projectName	string		☐	Name of the project and service desk.
serviceDeskId	string		☐	ID of the service desk the request belongs to.

4.1.10 RequestCommentAttachments: JIRA Service Desk Request Comment Attachments

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Comment Attachments

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,comment,attachment`

Insert JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,comment,attachment`

Update JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,comment,attachment`

Delete JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,comment,attachment`

Field Selection Method: NotRequired

Base Path: `values[*].comments.value[*].attachments.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestCommentAttachments. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
approvalStatus	string	☐		Filters results to customer requests based on their approval status:MY_PENDING_APPROVAL returns customer requests pending the user's approval.MY_HISTORY_APPROVAL returns customer requests where the user was an approver.Note: Valid only when used with requestOwnership = APPROVER.
organizationId	int32	☐		Filters customer requests that belong to a specific organization (note that the user must be a member of that organization). Note: Valid only when used with requestOwnership=ORGANIZATION.
requestOwnership	string	☐		Filters customer requests using the following values: OWNED_REQUESTS returns customer requests where the user is the creator. PARTICIPATED_REQUESTS returns customer requests where the user is a participant. ORGANIZATION returns customer requests for an organization of which the user is a member when used in conjunction with organizationId. ALL_ORGANIZATIONS returns customer requests that belong to all organizations of which the user is a member. APPROVER returns customer requests where the user is an approver. Can be used in conjunction with approvalStatus to filter pending or complete approvals.

Name	Data Type	Required	Default Value	Documentation
requestStatus	string	☐		Filters customer requests where the request is closed, open, or either of the two where: CLOSED_REQUESTS returns customer requests that are closed. OPEN_REQUESTS returns customer requests that are open. ALL_REQUESTS returns all customer requests.
requestTypeId	int32	☐		Filters customer requests by request type. Note that the serviceDeskId must be specified for the service desk in which the request type belongs.
searchTerm	string	☐		Filters customer requests where the request summary matches the searchTerm. Wildcards can be used in the searchTerm parameter.
serviceDeskId	int32	☐		Filters customer requests by service desk.

Table Function Columns

The columns of the table function `RequestCommentAttachments` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
author_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
author_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
author_active	boolean		☐	Indicates whether the user is active.
author_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.

Name	Data Type	Label	Required	Documentation
author_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
author_key	string		☐	This property is no longer available and will be removed from the documentation soon
author_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
author_name	string		☐	This property is no longer available and will be removed from the documentation soon.
author_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
comment_author_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
comment_author_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
comment_author_active	boolean		☐	Indicates whether the user is active.
comment_author_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
comment_author_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
comment_author_key	string		☐	This property is no longer available and will be removed from the documentation soon
comment_author_locale	string		☐	The locale of the user. Depending on the user's privacy

Name	Data Type	Label	Required	Documentation
				setting, this may be returned as null.
comment_author_name	string		☐	This property is no longer available and w ill be removed from the documentation soon.
comment_author_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
comment_body	string		☐	Content of the comment.
comment_created	datetime		☐	Date the attachment w as added.
comment_id	string		☐	ID of the comment.
comment_public	boolean		☐	Indicates w hether the comment is public (true) or private/internal (false).
comment_renderedBody_html	string		☐	HTML representation.
created	datetime		☐	Date the attachment w as added.
filename	string		☐	Filename of the item attached.
mimeType	string		☐	MIME type of the attachment.
request_actions_addAttachment_a llow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
request_actions_addComment_allo w ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
request_actions_addParticipant_all ow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
request_actions_removeParticipant _allow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on w hich the request w as created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCate gory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on w hich the status w as attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, w hich uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID w ith value unknow n is returned w hen there is

Name	Data Type	Label	Required	Documentation
				missing data for a user, w hich only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the follow ing values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
request_reporter_active	boolean		☐	Indicates w hether the user is active.
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and w ill be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and w ill be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canA ddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canR aiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDesk Id	string		☐	ID of the service desk the request type belongs to.

Name	Data Type	Label	Required	Documentation
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.
size	int64		☐	Size of the attachment in bytes.

4.1.11 RequestCommentAttachmentsByIdOrKey: JIRA Service Desk Request Comment Attachments by ID or Key

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Comment Attachments by ID or Key

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request/{issueIdOrKey}? expand=requestType,serviceDesk,action,comment,attachment`

Insert JIRA Service Desk API URL: `/request/{issueIdOrKey}? expand=requestType,serviceDesk,action,comment,attachment`

Update JIRA Service Desk API URL: `/request/{issueIdOrKey}? expand=requestType,serviceDesk,action,comment,attachment`

Delete JIRA Service Desk API URL: `/request/{issueIdOrKey}? expand=requestType,serviceDesk,action,comment,attachment`

Field Selection Method: NotRequired

Base Path: `comments.value[*].attachments.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestCommentAttachmentsByIdOrKey. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four

parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
issueIdOrKey	string	☑		The ID or key to be retrieved.

Table Function Columns

The columns of the table function `RequestCommentAttachmentsByIdOrKey` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
author_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
author_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
author_active	boolean		☐	Indicates whether the user is active.
author_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
author_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
author_key	string		☐	This property is no longer available and will be removed from the documentation soon
author_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.

Name	Data Type	Label	Required	Documentation
author_name	string		☐	This property is no longer available and will be removed from the documentation soon.
author_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
comment_author_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
comment_author_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
comment_author_active	boolean		☐	Indicates whether the user is active.
comment_author_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
comment_author_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
comment_author_key	string		☐	This property is no longer available and will be removed from the documentation soon
comment_author_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
comment_author_name	string		☐	This property is no longer available and will be removed from the documentation soon.
comment_author_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
comment_body	string		☐	Content of the comment.
comment_created	datetime		☐	Date the attachment was added.

Name	Data Type	Label	Required	Documentation
comment_id	string		☐	ID of the comment.
comment_public	boolean		☐	Indicates whether the comment is public (true) or private/internal (false).
comment_renderedBody_html	string		☐	HTML representation.
created	datetime		☐	Date the attachment was added.
filename	string		☐	Filename of the item attached.
mimeType	string		☐	MIME type of the attachment.
request_actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addComment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_removeParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on which the request was created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on which the status was attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk

Name	Data Type	Label	Required	Documentation
				account representing an external service desk
request_reporter_active	boolean		☐	Indicates whether the user is active.
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and will be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.

Name	Data Type	Label	Required	Documentation
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.
size	int64		☐	Size of the attachment in bytes.

4.1.12 RequestComments: JIRA Service Desk Request Comments

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Comments

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request? expand=requestType,serviceDesk,action,comment,comment.renderedBody`

Insert JIRA Service Desk API URL: `/request? expand=requestType,serviceDesk,action,comment,comment.renderedBody`

Update JIRA Service Desk API URL: `/request? expand=requestType,serviceDesk,action,comment,comment.renderedBody`

Delete JIRA Service Desk API URL: `/request? expand=requestType,serviceDesk,action,comment,comment.renderedBody`

Field Selection Method: NotRequired

Base Path: `values[*].comments.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestComments. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
approvalStatus	string	☐		Filters results to customer requests based on their approval status:MY_PENDING_APPROVA

Name	Data Type	Required	Default Value	Documentation
				L returns customer requests pending the user's approval.MY_HISTORY_APPRO VAL returns customer requests where the user was an approver.Note: Valid only when used with requestOwnership = APPROVER.
organizationId	int32	☐		Filters customer requests that belong to a specific organization (note that the user must be a member of that organization). Note: Valid only when used with requestOwnership=ORGANIZATI ON.
requestOwnership	string	☐		Filters customer requests using the following values: OWNED_REQUESTS returns customer requests where the user is the creator. PARTICIPATED_REQUESTS returns customer requests where the user is a participant. ORGANIZATION returns customer requests for an organization of which the user is a member when used in conjunction with organizationId. ALL_ORGANIZATIONS returns customer requests that belong to all organizations of which the user is a member. APPROVER returns customer requests where the user is an approver. Can be used in conjunction with approvalStatus to filter pending or complete approvals.
requestStatus	string	☐		Filters customer requests where the request is closed, open, or either of the two where: CLOSED_REQUESTS returns customer requests that are closed. OPEN_REQUESTS returns customer requests that are open. ALL_REQUESTS returns all customer requests.
requestTypeId	int32	☐		Filters customer requests by request type. Note that the serviceDeskId must be specified for the service desk in which the request type belongs.
searchTerm	string	☐		Filters customer requests where the request summary matches the searchTerm. Wildcards can be used in the searchTerm parameter.
serviceDeskId	int32	☐		Filters customer requests by service desk.

Table Function Columns

The columns of the table function `RequestComments` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
author_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
author_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
author_active	boolean		☐	Indicates whether the user is active.
author_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
author_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
author_key	string		☐	This property is no longer available and will be removed from the documentation soon
author_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
author_name	string		☐	This property is no longer available and will be removed from the documentation soon.
author_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
body	string		☐	Content of the comment.
created	datetime		☐	Date the attachment was added.
id	string		☐	ID of the comment.

Name	Data Type	Label	Required	Documentation
public	boolean		☐	Indicates whether the comment is public (true) or private/internal (false).
renderedBody_html	string		☐	HTML representation.
request_actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addComment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_removeParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on which the request was created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on which the status was attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
request_reporter_active	boolean		☐	Indicates whether the user is active.

Name	Data Type	Label	Required	Documentation
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and will be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.

4.1.13 RequestCommentsByIdOrKey: JIRA Service Desk Request Comments by ID or Key

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Comments by ID or Key

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request/{issueIdOrKey}?` `expand=requestType,serviceDesk,action,comment,comment.renderedBody`

Insert JIRA Service Desk API URL: `/request/{issueIdOrKey}?` `expand=requestType,serviceDesk,action,comment,comment.renderedBody`

Update JIRA Service Desk API URL: `/request/{issueIdOrKey}?` `expand=requestType,serviceDesk,action,comment,comment.renderedBody`

Delete JIRA Service Desk API URL: `/request/{issueIdOrKey}?` `expand=requestType,serviceDesk,action,comment,comment.renderedBody`

Field Selection Method: NotRequired

Base Path: `comments.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestCommentsByIdOrKey. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
issueIdOrKey	string	☑		The ID or key to be retrieved.

Table Function Columns

The columns of the table function `RequestCommentsByIdOrKey` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
author_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
author_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
author_active	boolean		☐	Indicates whether the user is active.
author_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
author_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
author_key	string		☐	This property is no longer available and will be removed from the documentation soon
author_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
author_name	string		☐	This property is no longer available and will be removed from the documentation soon.
author_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
body	string		☐	Content of the comment.
created	datetime		☐	Date the attachment was added.
id	string		☐	ID of the comment.
public	boolean		☐	Indicates whether the comment is public (true) or private/internal (false).
renderedBody_html	string		☐	HTML representation.
request_actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not

Name	Data Type	Label	Required	Documentation
				(false).
request_actions_addComment_allow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
request_actions_addParticipant_all ow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
request_actions_removeParticipant _allow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on w hich the request w as created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCate gory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on w hich the status w as attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, w hich uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID w ith value unknow n is returned w hen there is missing data for a user, w hich only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the follow ing values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
request_reporter_active	boolean		☐	Indicates w hether the user is active.
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.

Name	Data Type	Label	Required	Documentation
request_reporter_key	string		☐	This property is no longer available and w ill be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and w ill be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.

4.1.14 RequestParticipants: JIRA Service Desk Request Participants

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Participants

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request? expand=requestType,serviceDesk,action,participant`

Insert JIRA Service Desk API URL: `/request? expand=requestType,serviceDesk,action,participant`

Update JIRA Service Desk API URL: `/request? expand=requestType,serviceDesk,action,participant`

Delete JIRA Service Desk API URL: `/request? expand=requestType,serviceDesk,action,participant`

Field Selection Method: NotRequired

Base Path: `values[*].participants.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestParticipants. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
approvalStatus	string	☐		Filters results to customer requests based on their approval status:MY_PENDING_APPROVAL returns customer requests pending the user's approval.MY_HISTORY_APPROVAL returns customer requests where the user was an approver.Note: Valid only when used with requestOwnership = APPROVER.
organizationId	int32	☐		Filters customer requests that belong to a specific organization (note that the user must be a member of that organization). Note: Valid only when used with requestOwnership=ORGANIZATION.

Name	Data Type	Required	Default Value	Documentation
requestOwnership	string	☐		Filters customer requests using the following values: OWNED_REQUESTS returns customer requests where the user is the creator. PARTICIPATED_REQUESTS returns customer requests where the user is a participant. ORGANIZATION returns customer requests for an organization of which the user is a member when used in conjunction with organizationId. ALL_ORGANIZATIONS returns customer requests that belong to all organizations of which the user is a member. APPROVER returns customer requests where the user is an approver. Can be used in conjunction with approvalStatus to filter pending or complete approvals.
requestStatus	string	☐		Filters customer requests where the request is closed, open, or either of the two where: CLOSED_REQUESTS returns customer requests that are closed. OPEN_REQUESTS returns customer requests that are open. ALL_REQUESTS returns all customer requests.
requestTypeId	int32	☐		Filters customer requests by request type. Note that the serviceDeskId must be specified for the service desk in which the request type belongs.
searchTerm	string	☐		Filters customer requests where the request summary matches the searchTerm. Wildcards can be used in the searchTerm parameter.
serviceDeskId	int32	☐		Filters customer requests by service desk.

Table Function Columns

The columns of the table function `RequestParticipants` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is

Name	Data Type	Label	Required	Documentation
				missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
active	boolean		☐	Indicates whether the user is active.
displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
key	string		☐	This property is no longer available and will be removed from the documentation soon
locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addComment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_removeParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on which the request was created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.

Name	Data Type	Label	Required	Documentation
request_currentStatus_statusDate	datetime		☐	Date on which the status was attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
request_reporter_active	boolean		☐	Indicates whether the user is active.
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and will be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or

Name	Data Type	Label	Required	Documentation
				not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.
timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.

4.1.15 RequestParticipantsByIdOrKey: JIRA Service Desk Request Participants by ID or Key

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Participants by ID or Key

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,participant`

Insert JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,participant`

Update JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,participant`

Delete JIRA Service Desk API URL: `/request/{issueIdOrKey}?`
`expand=requestType,serviceDesk,action,participant`

Field Selection Method: NotRequired

Base Path: `participants.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestParticipantsByIdOrKey. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
issueIdOrKey	string	☑		The ID or key to be retrieved.

Table Function Columns

The columns of the table function `RequestParticipantsByIdOrKey` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
accountId	string		☐	The account ID of the user, w hich uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID w ith value unknow n is returned w hen there is missing data for a user, w hich only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
accountType	string		☐	The user account type. Can take the follow ing values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk

Name	Data Type	Label	Required	Documentation
active	boolean		☐	Indicates whether the user is active.
displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
key	string		☐	This property is no longer available and will be removed from the documentation soon
locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addComment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_removeParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on which the request was created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on which the status was attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users.

Name	Data Type	Label	Required	Documentation
				We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
request_reporter_active	boolean		☐	Indicates whether the user is active.
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and will be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.

Name	Data Type	Label	Required	Documentation
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.
timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.

4.1.16 Requests: JIRA Service Desk Requests

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Requests

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request?`
`expand=requestType,serviceDesk,action`

Insert JIRA Service Desk API URL: `/request?`
`expand=requestType,serviceDesk,action`

Update JIRA Service Desk API URL: `/request?`
`expand=requestType,serviceDesk,action`

Delete JIRA Service Desk API URL: `/request?`
`expand=requestType,serviceDesk,action`

Field Selection Method: NotRequired

Base Path: `values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function Requests. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
approvalStatus	string	☐		Filters results to customer requests based on their approval status:MY_PENDING_APPROVAL returns customer requests pending the user's approval.MY_HISTORY_APPROVAL returns customer requests where the user was an approver.Note: Valid only when used with requestOwnership = APPROVER.
organizationId	int32	☐		Filters customer requests that belong to a specific organization (note that the user must be a member of that organization). Note: Valid only when used with requestOwnership=ORGANIZATION.
requestOwnership	string	☐		Filters customer requests using the following values: OWNED_REQUESTS returns customer requests where the user is the creator. PARTICIPATED_REQUESTS returns customer requests where the user is a participant. ORGANIZATION returns customer requests for an organization of which the user is a member when used in conjunction with organizationId. ALL_ORGANIZATIONS returns customer requests that belong to all organizations of which the user is a member. APPROVER returns customer requests where the user is an approver. Can be used in conjunction with approvalStatus to filter pending or complete approvals.
requestStatus	string	☐		Filters customer requests where the request is closed, open, or either of the two where: CLOSED_REQUESTS returns customer requests that are closed. OPEN_REQUESTS returns customer requests that are open. ALL_REQUESTS returns all customer requests.

Name	Data Type	Required	Default Value	Documentation
requestTypeId	int32	☐		Filters customer requests by request type. Note that the serviceDeskId must be specified for the service desk in w hich the request type belongs.
searchTerm	string	☐		Filters customer requests w here the request summary matches the searchTerm. Wildcards can be used in the searchTerm parameter.
serviceDeskId	int32	☐		Filters customer requests by service desk.

Table Function Columns

The columns of the table function `Requests` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
actions_addAttachment_allow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
actions_addComment_allow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
actions_addParticipant_allow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
actions_removeParticipant_allow ed	boolean		☐	Indicates w hether the user can undertake the action (true) or not (false).
createdDate	datetime		☐	Date on w hich the request w as created.
currentStatus_status	string		☐	Name of the status condition.
currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
currentStatus_statusDate	datetime		☐	Date on w hich the status w as attained.
issueId	string		☐	ID of the request, as the peer issue ID.
issueKey	string		☐	Key of the request, as the peer issue key.
reporter_accountId	string		☐	The account ID of the user, w hich uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID w ith value unknow n is returned w hen there is missing data for a user, w hich

Name	Data Type	Label	Required	Documentation
				only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
reporter_accountType	string		☐	The user account type. Can take the follow ing values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
reporter_active	boolean		☐	Indicates w hether the user is active.
reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
reporter_key	string		☐	This property is no longer available and w ill be removed from the documentation soon
reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
reporter_name	string		☐	This property is no longer available and w ill be removed from the documentation soon.
reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
requestType_description	string		☐	Description of the request type.
requestType_fields_canAddReque stParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
requestType_fields_canRaiseOnB ehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
requestType_helpText	string		☐	Help text for the request type.
requestType_icon_id	string		☐	ID of the request type icon.
requestType_id	string		☐	ID for the request type.
requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
requestType_name	string		☐	Short name for the request type.
requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.

Name	Data Type	Label	Required	Documentation
requestTypeId	string		☐	ID of the request type for the request.
serviceDesk_id	string		☐	ID of the service desk.
serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
serviceDesk_projectName	string		☐	Name of the project and service desk.
serviceDeskId	string		☐	ID of the service desk the request belongs to.

4.1.17 RequestSLAs: JIRA Service Desk Request Participants

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Participants

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,sla`

Insert JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,sla`

Update JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,sla`

Delete JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,sla`

Field Selection Method: NotRequired

Base Path: `values[*].sla.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestSLAs. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
approvalStatus	string	☐		Filters results to customer requests based on their approval status:MY_PENDING_APPROVAL returns customer requests pending the user's approval.MY_HISTORY_APPROVAL returns customer requests where the user was an approver.Note: Valid only when used with requestOwnership = APPROVER.
organizationId	int32	☐		Filters customer requests that belong to a specific organization (note that the user must be a member of that organization). Note: Valid only when used with requestOwnership=ORGANIZATION.
requestOwnership	string	☐		Filters customer requests using the following values: OWNED_REQUESTS returns customer requests where the user is the creator. PARTICIPATED_REQUESTS returns customer requests where the user is a participant. ORGANIZATION returns customer requests for an organization of which the user is a member when used in conjunction with organizationId. ALL_ORGANIZATIONS returns customer requests that belong to all organizations of which the user is a member. APPROVER returns customer requests where the user is an approver. Can be used in conjunction with approvalStatus to filter pending or complete approvals.
requestStatus	string	☐		Filters customer requests where the request is closed, open, or either of the two where: CLOSED_REQUESTS returns customer requests that are closed. OPEN_REQUESTS returns customer requests that are open. ALL_REQUESTS returns all customer requests.
requestTypeId	int32	☐		Filters customer requests by request type. Note that the serviceDeskId must be specified for the service desk in which the request type belongs.

Name	Data Type	Required	Default Value	Documentation
searchTerm	string	☐		Filters customer requests where the request summary matches the searchTerm. Wildcards can be used in the searchTerm parameter.
serviceDeskId	int32	☐		Filters customer requests by service desk.

Table Function Columns

The columns of the table function `RequestSLAs` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
id	string		☐	ID of the Service Level Agreement (SLA).
name	string		☐	Description of the SLA.
ongoingCycle_breached	boolean		☐	Indicates whether the SLA has been breached (true) or not (false).
ongoingCycle_breachTime	string		☐	Time and date at which the SLA cycle would have breached its limit.
ongoingCycle_elapsedTime_friendly	string		☐	Duration in a user-friendly text format.
ongoingCycle_elapsedTime_millis	int64		☐	Duration in milliseconds.
ongoingCycle_goalDuration_friendly	string		☐	Duration in a user-friendly text format.
ongoingCycle_goalDuration_millis	int64		☐	Duration in milliseconds.
ongoingCycle_paused	boolean		☐	Indicates whether the SLA is paused (true) or not (false).
ongoingCycle_remainingTime_friendly	string		☐	Duration in a user-friendly text format.
ongoingCycle_remainingTime_millis	int64		☐	Duration in milliseconds.
ongoingCycle_startTime	datetime		☐	Time and date at which the SLA cycle started.
ongoingCycle_withinCalendarHours	boolean		☐	Indicates whether the SLA it timed during calendared working hours only (true) or not (false).
request_actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addComment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_removeParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not

Name	Data Type	Label	Required	Documentation
				(false).
request_createdDate	datetime		☐	Date on w hich the request w as created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on w hich the status w as attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, w hich uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID w ith value unknow n is returned w hen there is missing data for a user, w hich only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the follow ing values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
request_reporter_active	boolean		☐	Indicates w hether the user is active.
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and w ill be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and w ill be removed

Name	Data Type	Label	Required	Documentation
				from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.

4.1.18 RequestSLAsByIdOrKey: JIRA Service Desk Request Participants by ID or Key

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Participants by ID or Key

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,sla`

Insert JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,sla`

Update JIRA Service Desk API URL: `/request/{issueIdOrKey}?`
`expand=requestType,serviceDesk,action,sla`

Delete JIRA Service Desk API URL: `/request/{issueIdOrKey}?`
`expand=requestType,serviceDesk,action,sla`

Field Selection Method: NotRequired

Base Path: `sla.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestSLAsByIdOrKey. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
issueIdOrKey	string	☑		The ID or key to be retrieved.

Table Function Columns

The columns of the table function `RequestSLAsByIdOrKey` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
id	string		☐	ID of the Service Level Agreement (SLA).
name	string		☐	Description of the SLA.
ongoingCycle_breached	boolean		☐	Indicates whether the SLA has been breached (true) or not (false).
ongoingCycle_breachTime	string		☐	Time and date at which the SLA cycle would have breached its limit.
ongoingCycle_elapsedTime_friendly	string		☐	Duration in a user-friendly text format.
ongoingCycle_elapsedTime_millis	int64		☐	Duration in milliseconds.
ongoingCycle_goalDuration_friendly	string		☐	Duration in a user-friendly text format.
ongoingCycle_goalDuration_millis	int64		☐	Duration in milliseconds.

Name	Data Type	Label	Required	Documentation
ongoingCycle_paused	boolean		☐	Indicates whether the SLA is paused (true) or not (false).
ongoingCycle_remainingTime_friendly	string		☐	Duration in a user-friendly text format.
ongoingCycle_remainingTime_millis	int64		☐	Duration in milliseconds.
ongoingCycle_startTime	datetime		☐	Time and date at which the SLA cycle started.
ongoingCycle_withinCalendarHours	boolean		☐	Indicates whether the SLA it timed during calendared working hours only (true) or not (false).
request_actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addComment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_removeParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on which the request was created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on which the status was attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems,

Name	Data Type	Label	Required	Documentation
				customer Jira Service Desk account representing an external service desk
request_reporter_active	boolean		☐	Indicates whether the user is active.
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and will be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.

Name	Data Type	Label	Required	Documentation
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.

4.1.19 RequestStatusTransitions: JIRA Service Desk Request Status Transitions

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Status Transitions

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,status`

Insert JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,status`

Update JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,status`

Delete JIRA Service Desk API URL: `/request?expand=requestType,serviceDesk,action,status`

Field Selection Method: NotRequired

Base Path: `values[*].status.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestStatusTransitions. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
approvalStatus	string	☐		Filters results to customer requests based on their approval status:MY_PENDING_APPROVAL returns customer requests

Name	Data Type	Required	Default Value	Documentation
				pending the user's approval.MY_HISTORY_APPROVAL returns customer requests where the user was an approver.Note: Valid only when used with requestOwnership = APPROVER.
organizationId	int32	☐		Filters customer requests that belong to a specific organization (note that the user must be a member of that organization). Note: Valid only when used with requestOwnership=ORGANIZATION.
requestOwnership	string	☐		Filters customer requests using the following values: OWNED_REQUESTS returns customer requests where the user is the creator. PARTICIPATED_REQUESTS returns customer requests where the user is a participant. ORGANIZATION returns customer requests for an organization of which the user is a member when used in conjunction with organizationId. ALL_ORGANIZATIONS returns customer requests that belong to all organizations of which the user is a member. APPROVER returns customer requests where the user is an approver. Can be used in conjunction with approvalStatus to filter pending or complete approvals.
requestStatus	string	☐		Filters customer requests where the request is closed, open, or either of the two where: CLOSED_REQUESTS returns customer requests that are closed. OPEN_REQUESTS returns customer requests that are open. ALL_REQUESTS returns all customer requests.
requestTypeId	int32	☐		Filters customer requests by request type. Note that the serviceDeskId must be specified for the service desk in which the request type belongs.
searchTerm	string	☐		Filters customer requests where the request summary matches the searchTerm. Wildcards can be used in the searchTerm parameter.
serviceDeskId	int32	☐		Filters customer requests by service desk.

Table Function Columns

The columns of the table function `RequestStatusTransitions` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
request_actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addComment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_removeParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on which the request was created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on which the status was attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
request_reporter_active	boolean		☐	Indicates whether the user is active.

Name	Data Type	Label	Required	Documentation
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and will be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.
status	string		☐	Name of the status condition.
statusCategory	string		☐	Status category the status belongs to.Valid values:

Name	Data Type	Label	Required	Documentation
				UNDEFINED, NEW, INDETERMINATE, DONE.
statusDate	datetime		☐	Date on which the status was attained.

4.1.20 RequestStatusTransitionsByIdOrKey: JIRA Service Desk Request Status Transitions by ID or Key

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Status Transitions by ID or Key

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,status`

Insert JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,status`

Update JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,status`

Delete JIRA Service Desk API URL: `/request/{issueIdOrKey}?expand=requestType,serviceDesk,action,status`

Field Selection Method: NotRequired

Base Path: `status.values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function RequestStatusTransitionsByIdOrKey. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
issueIdOrKey	string	☑		The ID or key to be retrieved.

Table Function Columns

The columns of the table function `RequestStatusTransitionsByIdOrKey` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
request_actions_addAttachment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addComment_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_addParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_actions_removeParticipant_allowed	boolean		☐	Indicates whether the user can undertake the action (true) or not (false).
request_createdDate	datetime		☐	Date on which the request was created.
request_currentStatus_status	string		☐	Name of the status condition.
request_currentStatus_statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
request_currentStatus_statusDate	datetime		☐	Date on which the status was attained.
request_issueId	string		☐	ID of the request, as the peer issue ID.
request_issueKey	string		☐	Key of the request, as the peer issue key.
request_reporter_accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
request_reporter_accountType	string		☐	The user account type. Can take the following values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk

Name	Data Type	Label	Required	Documentation
request_reporter_active	boolean		☐	Indicates whether the user is active.
request_reporter_displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
request_reporter_emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_key	string		☐	This property is no longer available and will be removed from the documentation soon
request_reporter_locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
request_reporter_name	string		☐	This property is no longer available and will be removed from the documentation soon.
request_reporter_timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.
request_requestType_description	string		☐	Description of the request type.
request_requestType_fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
request_requestType_fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
request_requestType_helpText	string		☐	Help text for the request type.
request_requestType_icon_id	string		☐	ID of the request type icon.
request_requestType_id	string		☐	ID for the request type.
request_requestType_issueTypeId	string		☐	ID of the issue type the request type is based upon.
request_requestType_name	string		☐	Short name for the request type.
request_requestType_serviceDeskId	string		☐	ID of the service desk the request type belongs to.
request_requestTypeId	string		☐	ID of the request type for the request.
request_serviceDesk_id	string		☐	ID of the service desk.
request_serviceDesk_projectId	string		☐	ID of the peer project for the service desk.
request_serviceDesk_projectKey	string		☐	Key of the peer project of the service desk.
request_serviceDesk_projectName	string		☐	Name of the project and service desk.
request_serviceDeskId	string		☐	ID of the service desk the request belongs to.
status	string		☐	Name of the status condition.

Name	Data Type	Label	Required	Documentation
statusCategory	string		☐	Status category the status belongs to.Valid values: UNDEFINED, NEW, INDETERMINATE, DONE.
statusDate	datetime		☐	Date on which the status was attained.

4.1.21 RequestTypes: JIRA Service Desk Request Types

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Request Types

This is a read-only table. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/requesttype`

Insert JIRA Service Desk API URL: `/requesttype`

Update JIRA Service Desk API URL: `/requesttype`

Delete JIRA Service Desk API URL: `/requesttype`

Field Selection Method: NotRequired

Base Path: `values[*]`

Table Columns

The columns of the table `RequestTypes` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
description	string		☐	Description of the request type.
fields_canAddRequestParticipants	boolean		☐	Flag indicating if participants can be added to a request (true) or not.
fields_canRaiseOnBehalfOf	boolean		☐	Flag indicating if a request can be raised on behalf of another user (true) or not.
helpText	string		☐	Help text for the request type.
icon_id	string		☐	ID of the request type icon.
id	string		☐	ID for the request type.
issueTypeId	string		☐	ID of the issue type the request type is based upon.
name	string		☐	Short name for the request type.

Name	Data Type	Label	Required	Documentation
serviceDeskId	string		☐	ID of the service desk the request type belongs to.

4.1.22 ServiceDesks: JIRA Service Desk Service Desks

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Service Desks

This is a read-only table. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/servicedesk`

Insert JIRA Service Desk API URL: `/servicedesk`

Update JIRA Service Desk API URL: `/servicedesk`

Delete JIRA Service Desk API URL: `/servicedesk`

Field Selection Method: NotRequired

Base Path: `values[*]`

Table Columns

The columns of the table `ServiceDesks` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
id	string		☐	ID of the service desk.
projectId	string		☐	ID of the peer project for the service desk.
projectKey	string		☐	Key of the peer project of the service desk.
projectName	string		☐	Name of the project and service desk.

4.1.23 UsersByOrganization: JIRA Service Desk Users by Organization

Catalog: JIRA

Schema: Service

Primary Keys: id

Label: Users by Organization

This is a read-only table function. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case,

please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

Select JIRA Service Desk API URL: `/organization/{organizationId}/user`

Insert JIRA Service Desk API URL: `/organization/{organizationId}/user`

Update JIRA Service Desk API URL: `/organization/{organizationId}/user`

Delete JIRA Service Desk API URL: `/organization/{organizationId}/user`

Field Selection Method: NotRequired

Base Path: `values[*]`

Parameters of Table Function

The following parameters can be used to control the behaviour of the table function UsersByOrganization. A value must be provided at all times for required parameters, but optional parameters in general do not need to have a value and the execution will default to a pre-defined behaviour. Values can be specified by position and by name. In both cases, all parameters not specified will be treated using their default values.

Value specification by position is done by listing all values from the first to the last needed value. For example with `select * from table(value1, value2, value3)` on a table with four parameters will use the default value for the fourth parameter and the specified values for the first three.

Value specification by name is done by listing all values that require a value. For example with `select * from table(name1 => value1, name3 => value3)` on the same table will use the default values for the second and fourth parameters and the specified values for the first and third.

Name	Data Type	Required	Default Value	Documentation
organizationId	int32	☑		

Table Function Columns

The columns of the table function `UsersByOrganization` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID with value unknown is returned when there is missing data for a user, which only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
accountType	string		☐	The user account type. Can take the following values: atlassian

Name	Data Type	Label	Required	Documentation
				regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
active	boolean		☐	Indicates whether the user is active.
displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
key	string		☐	This property is no longer available and will be removed from the documentation soon
locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
name	string		☐	This property is no longer available and will be removed from the documentation soon.
timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.

5 Schema: Views
5.1 Views
5.1.1 OrganizationUsers

Catalog: JIRA

Schema: Views

This is a read-only view. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

View Columns

The columns of the view `OrganizationUsers` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
accountId	string		☐	The account ID of the user, which uniquely identifies the user across all Atlassian

Name	Data Type	Label	Required	Documentation
				products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID w ith value unknow n is returned w hen there is missing data for a user, w hich only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
accountType	string		☐	The user account type. Can take the follow ing values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
active	boolean		☐	Indicates w hether the user is active.
displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
key	string		☐	This property is no longer available and w ill be removed from the documentation soon
locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
name	string		☐	This property is no longer available and w ill be removed from the documentation soon.
ogn_id	string		☐	A unique system generated ID for the organization.
ogn_name	string		☐	Name of the organization.
timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.

5.1.2 ServiceDeskCustomers

Catalog: JIRA

Schema: Views

This is a read-only view. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please

use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

View Columns

The columns of the view `ServiceDeskCustomers` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
accountId	string		☐	The account ID of the user, w hich uniquely identifies the user across all Atlassian products. For example, 5b10ac8d82e05b22cc7d4ef5. Required in requests. An account ID w ith value unknow n is returned w hen there is missing data for a user, w hich only happens for deleted users. We recommend you treat this value as an error case if you need to access more information about the user.
accountType	string		☐	The user account type. Can take the follow ing values: atlassian regular Atlassian user account, app system account used for Connect applications and OAuth to represent external systems, customer Jira Service Desk account representing an external service desk
active	boolean		☐	Indicates w hether the user is active.
displayName	string		☐	The display name of the user. Depending on the user's privacy setting, this may return an alternative value.
emailAddress	string		☐	The email address of the user. Depending on the user's privacy setting, this may be returned as null.
key	string		☐	This property is no longer available and w ill be removed from the documentation soon
locale	string		☐	The locale of the user. Depending on the user's privacy setting, this may be returned as null.
name	string		☐	This property is no longer available and w ill be removed from the documentation soon.
sdk_id	string		☐	ID of the service desk.
sdk_projectId	string		☐	ID of the peer project for the service desk.
sdk_projectKey	string		☐	Key of the peer project of the service desk.

Name	Data Type	Label	Required	Documentation
sdk_projectName	string		☐	Name of the project and service desk.
timeZone	string		☐	The time zone specified in the user's profile. Depending on the user's privacy setting, this may be returned as null.

5.1.3 ServiceDeskQueues

Catalog: JIRA

Schema: Views

This is a read-only view. The JIRA Service Desk API may not support changing the data or the Invantive SQL driver for JIRA Service Desk does not cover it. In the latter case, please use the table `NativePlatformScalarRequests` to upload data to the JIRA Service Desk API.

View Columns

The columns of the view `ServiceDeskQueues` are shown below. Each column has an SQL data type.

Name	Data Type	Label	Required	Documentation
id	string		☐	ID for the queue.
issueCount	int64		☐	The count of customer requests in the queue.
jql	string		☐	JQL query that filters reqeusts for the queue.
name	string		☐	Short name for the queue.
sdk_id	string		☐	ID of the service desk.
sdk_projectId	string		☐	ID of the peer project for the service desk.
sdk_projectKey	string		☐	Key of the peer project of the service desk.
sdk_projectName	string		☐	Name of the project and service desk.

Index

- E -

- F -

Fail on Error   15
FAIL_ON_ERROR   15
fields_canAddRequestParticipants   82
fields_canRaiseOnBehalfOf   82
filename   24, 28, 35, 41
force-case-sensitive-identifiers   2
forced-casing-identifiers   2

- H -

helpText   82
highlight   17
HTTP Disk Cache Maximum Age (sec)   15
HTTP Memory Cache Maximum Age (sec)   15
HTTP Method   15
HTTP Status Code   15
HTTP_DISK_CACHE_MAX_AGE_SEC   15
HTTP_DISK_CACHE_SAVE   15
HTTP_DISK_CACHE_USE   15
HTTP_MEMORY_CACHE_MAX_AGE_SEC   15
HTTP_MEMORY_CACHE_SAVE   15
HTTP_MEMORY_CACHE_USE   15
HTTP_METHOD   15
HTTP_STATUS_CODE   15
http-disk-cache-compression-level   2
http-disk-cache-directory   2
http-disk-cache-ignore-write-errors   2
http-disk-cache-max-age-sec   2
http-get-timeout-max-ms   2
http-get-timeout-ms   2
http-memory-cache-compression-level   2
http-memory-cache-max-age-sec   2
http-post-timeout-max-ms   2
http-post-timeout-ms   2

- I -

icon_id   82
ignore-http-400-errors   2
ignore-http-401-errors   2
ignore-http-402-errors   2
ignore-http-403-errors   2
ignore-http-404-errors   2
ignore-http-422-errors   2
ignore-http-429-errors   2
ignore-http-500-errors   2
ignore-http-502-errors   2

ignore-http-503-errors   2
includeCount   22
Information   21
invalid-json-on-get-max-tries   2
invalid-json-on-get-sleep-initial-ms   2
invalid-json-on-get-sleep-max-ms   2
invalid-json-on-get-sleep-multiplicator   2
invalid-json-on-post-max-tries   2
invalid-json-on-post-sleep-initial-ms   2
invalid-json-on-post-sleep-max-ms   2
invalid-json-on-post-sleep-multiplicator   2
invantive-sql-compress-sparse-arrays   2
invantive-sql-correct-invalid-date   2
invantive-sql-forward-filters-to-data-containers   2
invantive-sql-share-byte-arrays   2
invantive-sql-share-strings   2
invantive-sql-shuffle-fetch-results-data-containers   2
invantive-use-cache   2
isLicensedForUse   21
issueCount   22, 88
issueId   32, 63
issueIdOrKey   28, 32, 41, 51, 59, 71, 79
issueKey   32, 63
issueTypeId   82

- J -

JIRA Service Desk   1, 15, 17, 18, 19, 21, 22, 24, 28, 32, 35, 41, 46, 51, 54, 59, 63, 67, 71, 75, 79, 82, 83, 85, 86, 88
JiraServiceDesk   1
join-set-points-per-request   2
jql   22, 88

- K -

key   19, 54, 59, 83, 85, 86

- L -

limit-partition-calls-left   2
locale   19, 54, 59, 83, 85, 86
log-native-calls-to-disk-max-events   2
log-native-calls-to-disk-max-seconds   2
log-native-calls-to-disk-on-error   2
log-native-calls-to-disk-on-success   2
log-native-calls-to-trace   2

- T -

- U -

- V -