TIBCO ActiveSpaces® Enterprise Edition

Release Notes

Version 4.9.0 | August 2023

Contents

About This Product

The TIBCO ActiveSpaces® Enterprise Edition software is a distributed in-memory data grid product. Some features of ActiveSpaces® include the use of familiar database concepts, high I/O capacity, and network scalability.

Product Editions

ActiveSpaces is available in two editions: Community Edition and Enterprise Edition.

	Community Edition	Enterprise Edition
Ideal for	Getting started with ActiveSpaces for implementing application projects, including proof of concept projects, for testing, and for deploying applications in a production environment. Production deployments running up to 5 nodes (a total of the copyset nodes or proxies in your data grid) For more information, see Terms used in Community and Enterprise Editions.	All application development projects, and for deploying and managing applications in the production environment of an enterprise. Production deployments with more than 5 nodes (a total of the copyset nodes or proxies in your data grid) For more information, see Terms used in Community and Enterprise Editions.
Features	All features of the Enterprise Edition except enterprise monitoring using dashboards.	Includes all the features presented in this documentation set.
Limitations	Run up to 5 nodes (a total of the copyset nodes or proxies in your data grid). Although the community license limits the number of production instances, you can easily upgrade to the enterprise edition as	No limitations on a total of the copyset nodes or proxies in your data grid.

	your use of ActiveSpaces expands.	
Cost	Free	Paid
Compatibility	Compatible with both the enterprise and community editions of TIBCO FTL®	Depends on the enterprise edition of TIBCO FTL for monitoring and management of data grid components and secure communication.
TIBCO Support	No access to TIBCO Support	Access to TIBCO Support

Terms used in Community and Enterprise Editions

- Node - a copyset node or proxy where each copyset node or proxy is an operating system process with a unique process ID.
- Process ID - For the purposes of the definition of Node, Process ID means a standard computer industry term that uniquely identifies each operating system process.
- Copyset - For the purposes of the definition of Node, “Copyset” means a logical grouping of nodes such that a portion of the data is shared uniformly by all the nodes that form a copyset.

Terminology Used to Address the TIBCO FTL Realm

With TIBCO FTL 6.1 or later, ActiveSpaces uses the realm service capabilities or processes of the TIBCO FTL server. The following changes are made to the terminology to generically address the components of TIBCO FTL 5.x and TIBCO FTL 6.x:

The Term Used in the Document	The Equivalent Component in TIBCO FTL 5.4.1	The Equivalent Component in TIBCO FTL 6.1 or Later
Realm service	Realm server	Realm service running on the TIBCO FTL server
Realm service URL	Realm server URL	TIBCO FTL server URL
Backup realm service	Backup realm server	TIBCO FTL server that is a member of a cluster of three or more TIBCO FTL servers
Primary Realm	Primary Realm Server and its Backup Realm Server	A cluster of primary TIBCO FTL servers that provide realm services for the data grid.
Satellite Realm	Satellite Realm Server and its Backup Realm Server	A cluster of satellite TIBCO FTL servers that are connected to a cluster of primary TIBCO FTL servers.

New Features

The following features have been added in this release of TIBCO ActiveSpaces® Enterprise Edition:

- **SQL UPDATE Statements**

  You can modify the columns of the rows in a table using SQL UPDATE statements. The SQL UPDATE statement modifies the rows in a table that satisfy the WHERE clause in the statement. For large updates, TIBCO recommends using the parameters in the WHERE clause to update the rows in batches.

- **Regular Expression Support**

  Two new SQL functions `regexp()` and `regexp_extract()` are added for matching and extracting the regular expressions in the SQL queries.

- **New SQL String Function Support**

  The `json_extract(X,P1,P2,...)` function extracts and returns one or more values from the well-formed JSON string X. If only a single path P1 is provided, then the datatype of the result depends on the value extracted from the JSON. JSON numeric values are returned as Long or Double datatype as appropriate. The JSON boolean values true or false are returned as Long values 1 or 0. JSON strings, arrays, or objects are returned as Strings.

- **Transaction check**

  Nodes involved in transactions now periodically check if any stale transactions must be cleaned up by requesting the status from their transaction leader.

- **Connect to proxy by using the FTL server**

  Simplified grid configuration when clients outside of a Kubernetes cluster must connect to proxies inside the cluster. Now, if the grid is created with the `default_client_proxy_transport` grid configuration option set to auto, clients can connect to the grid through the TIBCO FTL® Server. This removes the need to configure the LoadBalancer or NodePort services to expose the proxies directly. There is also no need to configure external hosts or ports for each of the proxies.

- **TIBCO ActiveSpaces® Enterprise Edition Grid Activity monitoring dashboard**

  The Grafana grid activity dashboard now displays the SQL DELETE and SQL UPDATE

statistics. New panels in the dashboard display the number of active SQL statements and the number of active ExecuteUpdate requests.

- **New tibdg commands**

  Two new commands have been added to the tibdg command-line tool:

  - `tibdg table purge` completely removes all the tables that have been deleted and have had all their rows deleted from all the nodes in all the copysets.
  - `tibdg index purge` completely removes all the indexes that have been deleted and have had all their data deleted from all the nodes in all the copysets.

  For these commands to purge anything, all the nodes in all the copysets must be running.

Changes in Functionality

The following functionality has been changed in this release of TIBCO ActiveSpaces® Enterprise Edition:

SELECT statements using LIKE operator

SELECT statements that use the LIKE operator in a WHERE clause, must now avoid using the wildcard characters '%' and '_' as the LIKE operator's ESCAPE character.

Previously a secondary index was used if one was compatible with the WHERE clause, but now a full table scan is done.

Grafana dashboards

- All the Grafana dashboards have been updated to work with the Messaging Monitor for TIBCO FTL®. The dashboards do not work out-of-the-box with FTL monitoring for versions earlier than TIBCO FTL 6.10.0. If you are not using Messaging Monitor for TIBCO FTL and are still using FTL monitoring, you must create a data source with the new name "TIBCO Messaging - FTL". Use the following steps to add it to FTL monitoring:
    - Ensure that you have started TIBCO FTL monitoring.
    - Copy TIB_FTL_INSTALL_DIR/monitoring/conf/influxdb/datasource.json file from your FTL installation directory to a temporary directory.
    - Edit the datasource.json file and find the line with "name": "FTL".
    - Change the name from "FTL" to "TIBCO Messaging - FTL". Save the file.
    - Run TIBDG_INSTALL_DIR/bin/urly POST http://<grafana_username>:<grafana_password>@<grafana_host>:<grafana_port>/api/datasources datasource.json command referencing the urly executable in your TIBCO ActiveSpaces® Enterprise Edition installation. For example, /opt/tibco/as/4.9/bin/urly POST http://admin:admin@localhost:3000/api/datasources datasource.json.
    - Verify the data source was added by:

    - Opening the Grafana web interface.
    - Selecting the **Configuration** icon.
    - Selecting Data Sources and verifying that the "TIBCO Messaging - FTL" data source is listed.
    - If the data source is not listed, restart TIBCO FTL monitoring.
- Some of the panels within the Grafana dashboards are updated to indicate requests instead of operations. Requests are the units being measured to display statistics for Statements, Queries, Iterators, Delete, Update, or ExecuteUpdate requests.
- The Concurrent Queries panel of the Grafana Grid Activity dashboard has been updated to show how many of the concurrent queries are globally consistent queries.
- Added new panels to the Grafana Proxy Activity dashboard. The panels allow you to see the number of active SQL statements by statement type and the number of active ExecuteUpdate requests by request type.

  SQL statement types: DELETE, INSERT, SELECT, UPDATE.

  ExecuteUpdate request types: DELETE, INSERT, UPDATE.
- The SQL Insert Operations Rate panel of the Grafana Node Activity dashboard is changed to SQL Operations Rate. This panel now includes individual graphs of the number of rows by request type for INSERT, DELETE, and UPDATE requests.
- Added a new panel to the Grafana Node Activity dashboard which displays SQL DELETE and UPDATE request rates.
- A new panel has been added to the Node Activity dashboard. The panel displays the number of transactions and globally consistent queries waiting to be processed.

Global Transactions Cancellation

If a globally consistent operation waits on the global queue longer than the standard wait based on the `client_req_timeout` setting, the operation is not performed. Not performing the operation means that a timeout error is returned to the caller, which in turn reduces the resource utilization on the `tibdgnode` and the coordination within the grid.

Deprecated and Removed Features

No features have been deprecated or removed as of this version of ActiveSpaces.

Migration and Compatibility

This version of ActiveSpaces is backward compatible with ActiveSpaces 3.x. However, ActiveSpaces 3.0 and later is not backward compatible with ActiveSpaces 2.x.

For information about upgrading from an earlier version of ActiveSpaces, see "Upgrading from an Earlier Version" in *TIBCO ActiveSpaces Installation*.

This release requires TIBCO FTL® 6.8.1 or later. For information about migrating from TIBCO FTL 5.x to TIBCO FTL 6.x, see the "Migration and Compatibility" section in *TIBCO FTL® Release Notes* and "Upgrade Migration to a New Release" section in *TIBCO FTL® Administration*.

Note: For the upgrade to be successful, the latest version of ActiveSpaces must be used to run `tibdg grid rebuild` on the existing data grid before upgrading the ActiveSpaces components.

Any new features listed in Changes in Functionality will be available after installing this version of Active Spaces.

Closed Issues

The following issues have been fixed in this release of TIBCO ActiveSpaces® Enterprise Edition.

Key	Summary
FFY-4927	Client requests sometimes timeout when more than one proxy fails in a short time. Now the client correctly tracks the currently bound proxy and retries the requests rather than returning a timeout error to the application.
FFY-4918	A tibdgkeeper process may crash if there is an error handling a realm update.
FFY-4897	Creating a table through SQL with a negative default_ttl does not result in an error.
FFY-4894	For some versions of TIBCO FTL®, tibdgConnection_Close may leak memory.
FFY-4869	The tibdgnode now defers cleanup of in-memory data redistribution records until the on-disk state can be updated. This prevents starting a new redistribution operation while there is an on-disk state for a previous operation.
FFY-4851	When adding a copyset, migrating rows to the new copyset may time out due to an incorrect interaction with a checkpoint. Now, there is no timeout error in this scenario.
FFY-4843	A primary grid no longer gets stuck mirroring the same checkpoint to a mirror grid in scenarios where a copyset experiences a failover to a new primary node.
FFY-4841	Earlier, there was no indication from the client library that it had lost communication with its proxy. Now the client

Key	Summary
	library logs warnings when the client unbinds from its proxy.
FFY-4835	A small memory leak in the data redistribution code is addressed now.
FFY-4831	The tibdgnode process cleans up transaction records that exist at some copysets but are missing at other copysets. It prevents a repeated "Got liveness check" log message.
FFY-4825	The tibdgnode process now correctly handles transaction liveness checks that arrive during startup, before reconciling with a copyset secondary.
FFY-4823	The SQL fetch and update timeouts are not used when they are specified in the JDBC URL.
FFY-4822	Earlier, there was no way for an application to determine that a checkpoint had been mirrored to all required components in a mirror grid. The tibdg proxy status command has been updated to include the last checkpoint and the last successful checkpoint from a proxy.
FFY-4811	The proxy on the Microsoft Windows platform could crash when parsing certain invalid SQL queries.
FFY-4807	When taking a checkpoint, the tibdgnode process no longer tries to take locks on tables that have been dropped.
FFY-4803	Client programs executing SQL CREATE TABLE commands sometimes display an error that a workspace timeout is not valid.
FFY-4802	Earlier, the clean-up processing that tibdgnodes perform after a transaction has committed involved all the copysets. Now, only those nodes affected by the transaction are involved.
FFY-4801	In certain failure scenarios, a node fails to cleanup a

Key	Summary
	canceled transaction at all copysets. This can lead to timeouts and other unexpected behavior for future operations that access the table.
FFY-4791	The data redistribution process now retries failed writes to system tables. This addresses cases where the data redistribution process could leave extra records in the system table. This also addresses incorrect attempts to restart data redistribution after it had already completed.
FFY-4790	When a disk write fails at a newly added copyset during data redistribution, the redistribution process is canceled and marked to retry later.
FFY-4782	Previously an error was returned when the tibdg proxy shed command was used to shed a client that was no longer connected to that proxy. Now, the tool reports that 0 clients have been shed.
FFY-4781	Earlier, retries were given too much time to complete. They now respect the client request timeout correctly.
FFY-4779	Transacted operations that arrive after a transaction is timed out or canceled lead to a timeout error instead of the expected transaction not found error.
FFY-4767	If the client enables the TIBDG_CONNECTION_PROPERTY_BOOLEAN_IMMUTABLE_ROWS property, then reading the values from a row object that is returned by a ResultSet object may read from the freed memory.
FFY-4755	The tibdgnode process terminates unexpectedly in certain cases when attempting to determine the liveness of stale transactions.
FFY-4735	If you are in the ftl-admin role, you can create and destroy tables with the tibdg command-line tool, but you do not have

Key	Summary
	access through the client API. Now, ftl-admin users have full access to all the tables regardless of the tool or API being used.
FFY-4732	Retrieving the ColumnType in Java from a Row, TableMetadata, or ResultSetMetadata object creates more string objects than necessary.
FFY-4731	The mirroring process handles changes in mirroring strategy and configuration more smoothly than in the previous releases. Failovers are less impactful on the overall mirroring process. Cases where a node can continue with a previous mirroring operation incorrectly are also addressed. This previously resulted in incorrect or missing rows in the mirror checkpoint. Additionally, the participants in the mirroring process avoid using incremental mirroring if we expect that generating the diff between the two checkpoints would be inefficient. This causes nodes to fall back to using bulk mirroring when there are a large number of checkpoints to be spanned.
FFY-4727	The number of checkpoint journals used for incremental mirroring is now limited. If too many journals are required to do efficient incremental mirroring, bulk mirroring is used instead.
FFY-4722	The tibdgnode process correctly handles incremental mirroring operations larger than 1M rows where previously data may have failed to mirror.
FFY-4715	Iterators created with global consistency sometimes remain in a node following a copyset leader failure. This results in the consistent query limit being exceeded for longer than is necessary.
FFY-4687	A node upgraded from a version earlier than 4.0 is unable to run a necessary one-time integrity scan.

Key	Summary
FFY-4680	A proxy server that is not restarted following a gridset event change could return invalid responses to clients.
FFY-4676	When the admin attempts to remove a grid from a gridset, no error is returned if it is not a member of the gridset. No error is returned if the grid does not exist.
FFY-4675	When a grid that has checkpoints but no schema is added as a mirror grid, it can fail to mirror some checkpoints from the primary grid.
FFY-4672	A primary tibdgnode could send incomplete information to a secondary tibdgnode during catchup if writes were in progress.
FFY-4670	A grid that was previously a part of a gridset and was then added to a new gridset, fails to take checkpoints.
FFY-4629	When redistribution on a table occurs at the same time as the SQL DELETE command, the node may exit unexpectedly.
FFY-4627	The tibdgnode process correctly tracks the checkpoints that were previously marked as unavailable. This eliminates some cases where a *Got error response with MIRRORING COMPLETE (Can't open checkpoint.)* warning is displayed.
FFY-4625	A tibdgnode in a primary grid reclaims journals early if there were two or more mirror grids in the gridset.
FFY-4624	Earlier, the entries in the checkpoint leader's system table were not always getting updated from "in progress" to "success". Now the entries have their status recorded correctly.
FFY-4623	A tibdgnode leaves the journal and mirroring data in an incorrect state after completing tibdg grid redistribute to a newly added copyset.

Key	Summary
FFY-4618	A tibdgnode in a primary grid can incorrectly handle closing a checkpoint after finishing mirroring.
FFY-4617	The tibdg checkpoint list returns SUCCESS in some cases for the checkpoint status instead of the correct status (such as MIRRORING).
FFY-4614	The proxy is not correctly processing replies from a node for an INSERT statement that has been timed out by the client.
FFY-4607	Incremental mirroring can be used in situations where it was previously not possible. Earlier, after changing the primary grid in a gridset, bulk mirroring was required for the first checkpoint mirrored.
FFY-4606	The tibdg status command displays the node as BUSY status due to excessive background scanning during mirroring operations.
FFY-4603	A tibdgnode incorrectly handles replicated writes during a timeout or epoch change in the copyset, which results in a failure.
FFY-4591	Rows that are mirrored to a mirror grid with a short TTL behave incorrectly. This causes unexpected results in the checkpoint at the mirror grid.
FFY-4581	The string returned by tibdgEventTypeList_ToString needs to be deallocated by the same allocator that created it. You cannot do this through the public API. A new method, tibdgEventTypeList_DestroyString, has been added to address this issue.
FFY-4564	The tibdgnode processes should not take checkpoints if they are not caught up to the primary. Previously a dead secondary could incorrectly take a checkpoint that did not agree with the primary.

Key	Summary
FFY-4561	A tibdgnode fails to detect and repair incorrect journal and mirroring information on the disk.
FFY-4546	Checkpoint journals of tibdgnodes are closed by retrying when disk write errors are encountered. This could otherwise lead to incorrect journal and mirroring information on the disk. This could also lead to long background scanning of the metadata.
FFY-4545	A grid that was previously a part of a gridset and then added to a new gridset fails to take new checkpoints.
FFY-4541	A tibdgnode sometimes has an incorrect active requests count after some operations have timed out.
FFY-4538	Executing tibdg grid modify without any config options does not result in an error.
FFY-4517	A tibdgnode exits unexpectedly in some cases if gridset connection information is modified about the grids in the gridset.
FFY-4514	The tibdg grid create command could fail with error messages that contain corrupted values.
FFY-4485	API documentation on the Row object no longer has incorrect information about attempting to set a column value to null.
FFY-4469	Repeatedly creating and deleting tables takes progressively longer and longer. This has been addressed by creating the tibdg table purge command to remove all unused tables from the grid permanently. Typically tables become purgeable 1-2 minutes after they are deleted. Use the tibdg table purge command after 50-100 tables are dropped to ensure that the creation and deletion of tables does not slow down too much.

Key	Summary
FFY-4452	The node can occasionally exit uncleanly if it gets disconnected from the FTL server while it is sending a response to a tibdg command that is received when the node is shutting down.
FFY-4438	The proxy does not respond to changes in copyset leadership as expected. This results in client operations on statements taking longer than necessary to time out.
FFY-4291	The proxy continues to process an INSERT statement after the client has timed out.
FFY-3222	In rare circumstances, a grid redistribution completes irrespective of not having all the data redistributed.
FFY-3174	The error code returned by functions when a table is not found is changed from TIB_INVALID_VALUE to TIB_NOT_FOUND.

Known Issues

The following issues exist in this release of TIBCO ActiveSpaces® Enterprise Edition.

Key	Summary
FFY-3493	**Summary:** Using tibdgRow_ToString does not display all columns when the SELECT list contains duplicate column names. **Workaround:** In the SELECT list, instead of column names, use the unique label names. For example, SELECT lastname, firstname, lastname AS ln2 FROM mytable.
FFY-2942	**Summary:** Queries involving GROUP BY do not work with expressions. The following limitations apply when using GROUP BY: • SQL CASE expressions cannot be used in select lists when aggregation is used. • SQL CASE expressions cannot contain aggregation functions. • Nested functions and nested expressions in select lists cannot be used when aggregation is used. **Workaround:** None
FFY-2711	**Summary:** The SQL INSERT parameter values must exactly match the type defined for the column in the table. No attempt is made to cast from the data type of the parameter to the data type of the column in the table. **Workaround:** Set the parameter value into the statement by using the type that exactly matches the column type of the

Key	Summary
	table.
FFY-2140	**Summary:** Removing a mirror grid from a gridset, either by using the tibdg gridset remove command or by using the tibdg gridset delete command, results in unexpected behavior if the -p or --makePrimary flags are not used. As a result, data grids cannot be added back after they are removed from the gridsets. **Workaround:** Use the -p or --makePrimary flag when performing an operation to remove a data grid from a gridset. Using these options transitions the mirror grid into a standalone grid. As a result, the data grid removed cannot be added to a gridset as a mirror grid.
FFY-1654	**Summary:** When creating a table using SQL, the SQL data types are mapped to their underlying FTL types. When the metadata for the table is returned, the original data types are not preserved and you can only see the following mapping: long => BIGINT string => VARCHAR double => DOUBLE datetime => TIMESTAMP opaque => VARBINARY **Workaround:** For information about how SQL data types are mapped to ActiveSpaces data types, see the "SQL Data Type Mapping" section in *TIBCO ActiveSpaces® Enterprise Edition Administration* .
FFY-1569	**Summary:** In an iterator, calling tibdgRow_ToString() on a row or executing a query such as SELECT * FROM <table> does not always return columns in the order defined in the table.

Key	Summary
	Workaround: Retrieve the columns individually by name and in the order you want.
FFY-1474	**Summary:** Occasionally, using special characters for identifiers is out-of-sync between the tibdg tool and the SQL DDL commands. In some scenarios, you can create a table by using the TABLE CREATE command, but not drop the table by using the tibdg tool or by using the SQL DROP TABLE command. **Workaround:** Create simple table names. The tables names must use lowercase characters, but no special characters or embedded spaces.

TIBCO Documentation and Support Services

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the TIBCO Product Documentation website, mainly in HTML and PDF formats.

The TIBCO Product Documentation website is updated frequently and is more current than any other documentation included with the product.

Product-Specific Documentation

The following documentation for TIBCO ActiveSpaces® Enterprise Edition is available on the TIBCO ActiveSpaces® Enterprise Edition Product Documentation page:

- TIBCO ActiveSpaces® Enterprise Edition *Release Notes*
- TIBCO ActiveSpaces® Enterprise Edition *Installation*
- TIBCO ActiveSpaces® Enterprise Edition *Concepts*
- TIBCO ActiveSpaces® Enterprise Edition *Administration*
- TIBCO ActiveSpaces® Enterprise Edition *API Reference*
- TIBCO ActiveSpaces® Enterprise Edition *Security Guidelines*
- TIBCO ActiveSpaces® Enterprise Edition *ActiveSpaces4-Sizing-Guide*

How to Contact TIBCO Support

Get an overview of TIBCO Support. You can contact TIBCO Support in the following ways:

- For accessing the Support Knowledge Base and getting personalized content about products you are interested in, visit the TIBCO Support website.
- For creating a Support case, you must have a valid maintenance or support contract with TIBCO. You also need a user name and password to log in to TIBCO Support website. If you do not have a user name, you can request one by clicking **Register** on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the TIBCO Ideas Portal. For a free registration, go to TIBCO Community.

Legal and Third-Party Notices

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the “readme” file for the availability of a specific version of Cloud SG software on a specific operating system platform.

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

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. CLOUD SG MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S), THE PROGRAM(S), AND/OR THE SERVICES DESCRIBED IN THIS DOCUMENT AT ANY TIME WITHOUT NOTICE.

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

This and other products of Cloud SG may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at https://www.tibco.com/patents.