

TaurusDB for PostgreSQL

内核介绍

文档版本 01
发布日期 2025-11-19



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

目 录

- 1 逻辑订阅故障转移（ Failover Slot ） 1
- 2 Babelfish..... 3
 - 2.1 Babelfish 简介..... 3
 - 2.2 兼容性说明..... 3
 - 2.2.1 Babelfish 支持的功能..... 3
 - 2.2.2 Babelfish 不支持的功能..... 14
 - 2.3 Babelfish 部分支持的功能示例..... 23
 - 2.4 常用 SQL 语句示例..... 34
 - 2.5 支持的插件..... 35

1 逻辑订阅故障转移（Failover Slot）

使用场景

TaurusDB for PostgreSQL的逻辑订阅故障转移（Failover Slot）功能用来将那些指定为failover slot的逻辑复制槽信息从主实例同步到备实例，当主备切换之后逻辑订阅能够继续进行，实现逻辑复制槽的故障转移。

当用户使用逻辑复制时，由于slot信息不会随着主备切换转移到新的主实例上，一旦实例发生主备切换，逻辑订阅就会断开，此时除非手动重新创建slot，否则逻辑订阅无法重新连接。TaurusDB for PostgreSQL的Failover Slot功能可以将所有的logical slot从主实例同步到备实例，避免主备切换后逻辑订阅断开。

注意事项

- 当前只支持logical slot的故障转移，physical slot暂不支持。
- 该功能由于引入新的日志类型，因此一旦使用了该功能，通过备份文件恢复到新实例时，会保留旧的slot信息，恢复到新实例后需要手动删除slot信息。

使用方法

在发布端执行如下SQL，创建一个Failover Slot。

```
SELECT * FROM pg_create_logical_replication_slot('slotname', 'pgoutput', false, false, true);
```

表 1-1 参数说明

参数	说明
slotname	logical slot的名称。
pgoutput	plugin的名称，也可以更改为其他支持的插件。
第三个参数（例如false）	该slot是否为临时slot。
第四个参数（例如false）	是否启动两阶段提交选项。
第五个参数（例如true）	该复制槽是否为Failover slot。

因此，如果要创建一个Failover Slot，在第三个参数临时slot字段必须为false，最后一个参数必须为true。如果最后一个参数省略不写，则创建的是非Failover Slot。

查看方法

在发布端执行如下SQL，查询当前数据库的复制槽信息。

```
select * from pg_get_replication_slots();
```

- 查看“failover”字段，如果值为“true”，表示该复制槽为Failover Slot。
- 如果没有“failover”字段或该字段值为“false”，说明该复制槽不是Failover Slot。

逻辑订阅完整使用示例

- 发布端创建表

```
create table tableName(id int primary key, num int);
```
- 发布端创建发布

```
create publication pubName for table tableName;
```
- 发布端创建一个Failover Slot

```
SELECT * FROM pg_create_logical_replication_slot('slotname', 'pgoutput', false, false, true);
```
- 发布端插入数据

```
insert into tableName values(1,1);  
insert into tableName values(2,2);
```
- 订阅端创建表

```
create table tableName (id int primary key, num int);
```
- 订阅端创建订阅，指定创建的Failover Slot名称

```
create subscription subName connection 'host=192.168.0.10 dbname=postgres user=root port=5432  
password=xxxxxxx' publication pubName with(copy_data=true,create_slot=false,slot_name= slotname);
```
- 订阅端查询数据，验证数据是否订阅到

```
select * from tableName;
```
- 主备切换
发布端继续插入数据，在订阅端查看，逻辑订阅不会断开。

2 Babelfish

2.1 Babelfish 简介

云数据库TaurusDB for PostgreSQL的Babelfish特性提供SQL Server语法兼容层，支持在PostgreSQL环境中无缝运行T-SQL应用程序，主要特性包括：

- T-SQL语法兼容：支持存储过程、触发器、函数等SQL Server语法元素。
- 数据类型映射：自动转换SQL Server特有数据类型（如NVARCHAR、DATETIME2）至PostgreSQL等效类型。
- 协议兼容：通过TDS端口（默认1433）提供与SQL Server原生协议兼容的连接方式。

适用场景

- SQL Server迁移上云：将本地或云上SQL Server应用迁移至华为云PostgreSQL。
- 多数据库整合：统一使用PostgreSQL管理兼容SQL Server的遗留系统。

周边服务

- Babelfish Compass：评估工具，分析SQL Server脚本的兼容性并生成适配建议。
- 数据库和应用迁移：通过华为云UGO的对象迁移功能，将源端SQL Server数据库中的存储过程、函数和视图等数据对象迁移至TaurusDB for PostgreSQL中。
- 数据复制服务：华为云DRS提供一站式迁移解决方案，支持全量+增量数据同步，在线完成数据迁移。

更多详细内容请参考[Babelfish](#)。

2.2 兼容性说明

2.2.1 Babelfish 支持的功能

本章节介绍Babelfish 4.3版本支持的T-SQL功能语法、内置函数、系统定义的@@变量、系统存储过程和SQL Server视图。

支持的 T-SQL 功能或语法

在下表中，介绍了 Babelfish 4.3版本支持的 T-SQL 功能或语法。

表 2-1 支持的 T-SQL 功能或语法

T-SQL功能或语法	4.3.0
SELECT 语句的 4 部分对象名称引用	支持
用于更改数据库所有者的 ALTER AUTHORIZATION 语法	支持
ALTER DATABASE <db_name> MODIFY NAME = <new_db_name>	支持
ALTER FUNCTION	不支持
ALTER PROCEDURE	支持
ALTER ROLE	支持
ALTER USER...WITH LOGIN	支持
CREATE FUNCTION 中的 AS 关键字	支持
AT TIME ZONE 子句	支持
Babelfish 实例作为链接服务器	支持
比较运算符 !< 和 !>	支持
CREATE INDEX ... ON partition_scheme_name (partition_column_name)	支持
在 SQL Server 视图上 CREATE，而不是触发器（DML）	支持
CREATE PARTITION FUNCTION	支持
CREATE PARTITION SCHEME	支持
CREATE ROLE	支持
CREATE TABLE ... ON partition_scheme_name (partition_column_name)	支持
CREATE TRIGGER	支持
创建唯一索引	支持
跨数据库过程执行	支持
跨数据库引用 SELECT、INSERT、UPDATE、DELETE	支持

T-SQL功能或语法	4.3.0
游标类型的参数仅用于输入参数（不用于输出）	支持
使用 bcp 客户端实用程序进行数据迁移	支持
数据类型 TIMESTAMP、ROWVERSION	支持
对存储过程和函数的调用中的 DEFAULT 关键字	支持
DBCC CHECKIDENT	支持
DROP DATABASE	支持
DROP IF EXISTS（对于 SCHEMA、DATABASE 和 USER 对象）	支持
DROP INDEX index ON schema.table	支持
DROP INDEX schema.table.index	支持
DROP PARTITION FUNCTION	支持
DROP PARTITION SCHEME	支持
DROP ROLE	支持
ENABLE/DISABLE TRIGGER	支持
FULL TEXT SEARCH	支持
使用 CONTAINS 子句进行全文搜索	支持
几何和地理空间数据类型	支持
GRANT	支持
GRANT permission [,...n] ON SCHEMA	支持
GUC babelfishpg_tds.product_version	支持
带前导点字符的标识符	支持
针对表的 INSTEAD OF 触发器	支持
针对视图的 INSTEAD OF 触发器	支持
KILL	支持
\$PARTITION.partition_function_name(partition_column_value)	支持
PIVOT	支持
REVOKE	支持
REVOKE permission [,...n] ON SCHEMA	支持
SELECT...OFFSET...FETCH 子句	支持

T-SQL功能或语法	4.3.0
SELECT FOR JSON AUTO	支持
SET BABELFISH_SHOWPLAN_ALL ON (和 OFF)	支持
SET BABELFISH_STATISTICS PROFILE ON (OFF)	支持
SET CONTEXT_INFO	支持
SET LOCK_TIMEOUT	支持
SET NO_BROWSETABLE	支持
SET rowcount	支持
SET SHOWPLAN_ALL	支持
SET STATISTICS IO	支持
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ	支持
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE	支持
SET TRANSACTION ISOLATION LEVEL 语法	支持
SSMS: 使用 Object Explorer 连接对话框进行连接	支持
SSMS: 使用导入/导出向导进行数据迁移	支持
SSMS: 部分支持 Object Explorer	支持
STDEV	支持
STDEVP	支持
具有多个 DML 操作的触发器可以引用转换表	支持
T-SQL 提示 (联接方法、索引用法、MAXDOP)	支持
带 LIKE 谓词的 T-SQL 方括号语法	支持
存储过程调用中未加引号的字符串值和默认值	支持
VAR	支持
VARP	支持

支持的 T-SQL 内置函数

在下表中，介绍了 Babelfish 4.3版本支持的 T-SQL内置函数。

表 2-2 支持的 T-SQL 内置函数

T-SQL 内置函数	4.3.0
APP_NAME	支持
ATN2	支持
CHARINDEX	支持
CHOOSE	支持
COL_LENGTH	支持
COL_NAME	支持
COLUMNS_UPDATED	支持
COLUMNPROPERTY (CharMaxLen, AllowsNull only)	支持
CONCAT_WS	支持
CONTEXT_INFO	支持
CURSOR_STATUS	支持
DATABASE_PRINCIPAL_ID	支持
DATEADD	支持
DATEDIFF	支持
DATEDIFF_BIG	支持
DATEFROMPARTS	支持
DATENAME	支持
DATEPART	支持
DATETIMEFROMPARTS	支持
DATETIME2FROMPARTS	支持
DATETIMEOFFSETFROMPARTS	支持
DATETRUNC	支持
DATE_BUCKET	支持
EOMONTH	支持
EXECUTE AS CALLER	支持
fn_listextendedproperty	支持

T-SQL 内置函数	4.3.0
FOR JSON	支持
FULLTEXTSERVICEPROPERTY	支持
HAS_DBACCESS	支持
HAS_PERMS_BY_NAME	支持
HOST_NAME	支持
HOST_ID	支持
IDENTITY	支持
IS_MEMBER	支持
IS_ROLEMEMBER	支持
IS_SRVROLEMEMBER	支持
ISJSON	支持
JSON_MODIFY	支持
JSON_QUERY	支持
JSON_VALUE	支持
NEXT VALUE FOR	支持
OBJECT_DEFINITION	支持
OBJECT_SCHEMA_NAME	支持
OPENJSON	支持
OPENQUERY	支持
ORIGINAL_LOGIN	支持
PARSENAME	支持
PATINDEX	支持
ROWCOUNT_BIG	支持
SCHEMA_NAME	支持
SESSION_CONTEXT	支持
SESSION_USER	支持
SID_BINARY (始终返回 NULL)	支持
SMALLDATETIMEFROMPARTS	支持
SQUARE	支持
STR	支持

T-SQL 内置函数	4.3.0
STRING_AGG	支持
STRING_SPLIT	支持
SUSER_SID	支持
SUSER_SNAME	支持
SWITCHOFFSET	支持
SYSTEM_USER	支持
TIMEFROMPARTS	支持
TODATETIMEOFFSET	支持
TO_CHAR	支持
TRIGGER_NESTLEVEL（仅限不带参数）	支持
TRY_CONVERT	支持
TYPE_ID	支持
TYPE_NAME	支持
UPDATE	支持

支持的 T-SQL 系统定义的@@变量

在下表中，介绍了 Babelfish 4.3版本支持的 T-SQL 系统定义的@@变量。

 **注意**

@@version 返回的数据格式与 SQL Server 默认输出的格式存在差异。若您的应用程序逻辑依赖于 @@version 的特定格式，可能会导致预期之外的行为。

表 2-3 支持的 T-SQL 系统定义的@@变量

T-SQL 系统定义的 @@ 变量	4.3.0
@@CURSOR_ROWS	支持
@@DATEFIRST	支持
@@DBTS	支持
@@ERROR	支持
@@ERROR=213	支持
@@FETCH_STATUS	支持

T-SQL 系统定义的 @@ 变量	4.3.0
@@IDENTITY	支持
@@LANGUAGE	支持
@@LOCK_TIMEOUT	支持
@@MAX_CONNECTIONS	支持
@@MAX_PRECISION	支持
@@MICROSOFTVERSION	支持
@@NESTLEVEL	支持
@@PROCID	支持
@@ROWCOUNT	支持
@@SERVERNAME	支持
@@SERVICENAME	支持
@@SPID	支持
@@TRANCOUNT	支持
@@VERSION	支持

支持的 T-SQL 系统存储过程

在下表中，介绍了 Babelfish 4.3版本支持的 T-SQL 系统存储过程。

表 2-4 支持的 T-SQL 系统存储过程

T-SQL 系统存储过程	4.3.0
sp_addextendedproperty	支持
sp_addlinkedserver	支持
sp_addlinkedsrvlogin	支持
sp_addrole	支持
sp_addrolemember	支持
sp_babelfish_volatility	支持
sp_column_privileges	支持
sp_columns	支持
sp_columns_100	支持
sp_columns_managed	支持

T-SQL 系统存储过程	4.3.0
sp_cursor	支持
sp_cursor_list	支持
sp_cursorclose	支持
sp_cursorexecute	支持
sp_cursorfetch	支持
sp_cursoropen	支持
sp_cursoroption	支持
sp_cursorprepare	支持
sp_cursorprepexec	支持
sp_cursorunprepare	支持
sp_databases	支持
sp_datatype_info	支持
sp_datatype_info_100	支持
sp_describe_cursor	支持
sp_describe_first_result_set	支持
sp_describe_undeclared_parameters	支持
sp_dropextendedproperty	支持
sp_droplinkedserver	支持
sp_droprole	支持
sp_droprolemember	支持
sp_dropserver	支持
sp_enum_oledb_providers	支持
sp_execute	支持
sp_execute_postgresql(CREATE, ALTER, DROP)	支持
sp_executesql	支持
sp_fkeys	支持
sp_getapplock	支持
sp_helpdb	支持
sp_helpdbfixedrole	支持

T-SQL 系统存储过程	4.3.0
sp_helplinkedsvlogin	支持
sp_helprole	支持
sp_helprolemember	支持
sp_helpsrvrolemember	支持
sp_helpuser	支持
sp_linkedservers	支持
sp_oledb_ro_usrname	支持
sp_pkeys	支持
sp_prefix	支持
sp_prepare	支持
sp_procedure_params_100_managed	支持
sp_releaseapplock	支持
sp_rename	支持
sp_renamedb	支持
sp_rename (适用于 object_type 'DATABASE')	支持
sp_reset_connection	不支持
sp_serverOption (connect_timeout 选项)	支持
sp_set_session_context	支持
sp_special_columns	支持
sp_sproc_columns	支持
sp_sproc_columns_100	支持
sp_statistics	支持
sp_statistics_100	支持
sp_stored_procedures	支持
sp_table_privileges	支持
sp_tablecollations_100	支持
sp_tables	支持
sp_testlinkedserver	支持
sp_unprepare	支持

T-SQL 系统存储过程	4.3.0
sp_updateextendedproperty	支持
sp_who	支持
xp_qv	支持

支持的 SQL Server 视图

在下表中，介绍了 Babelfish 4.3版本支持的 SQL Server 视图。

表 2-5 支持的 SQL Server 视图

Babelfish 支持的 SQL Server 视图	4.3.0
information_schema.key_column_usag e	支持
information_schema.routines	支持
information_schema.schemata	支持
information_schema.sequences	支持
sys.all_columns	支持
sys.all_objects	支持
sys.all_parameters	支持
sys.all_sql_modules	支持
sys.all_views	支持
sys.columns	支持
sys.configurations	支持
sys.data_spaces	支持
sys.database_files	支持
sys.database_mirroring	支持
sys.database_principals	支持
sys.database_role_members	支持
sys.databases	支持
sys.destination_data_spaces	支持
sys.dm_exec_connections	支持
sys.dm_exec_sessions	支持

Babelfish 支持的 SQL Server 视图	4.3.0
sys.dm_hadr_database_replica_states	支持
sys.dm_os_host_info	支持
sys.dm_os_sys_info	不支持
sys.endpoints	支持
sys.extended_properties	支持
sys.indexes	支持
sys.partitions	支持
sys.partition_functions	支持
sys.partition_parameters	支持
sys.partition_range_values	支持
sys.partition_schemes	支持
sys.schemas	支持
sys.server_principals	支持
sys.server_role_members	支持
sys.sql_modules	支持
sys.sysconfigures	支持
sys.syscurconfigs	支持
sys.syslogins	支持
sys.sysprocesses	支持
sys.sysusers	支持
sys.table_types	支持
sys.tables	支持
sys.types	支持
sys.xml_schema_collections	支持
syslanguages	支持
sysobjects.crdate	支持

2.2.2 Babelfish 不支持的功能

当前不支持的功能

在下表中，介绍了Babelfish当前不支持的某些功能的信息。

表 2-6 不支持的功能或语法

功能或语法	描述
程序集模块和 SQL 公共语言运行时 (CLR) 例程	不支持与组装模块和 CLR 例程相关的功能。
列属性	不支持 ROWGUIDCOL、SPARSE、FILESTREAM 和 MASKED。
包含的数据库	不支持在数据库级别而不是在服务器级别验证了登录名的包含数据库。
跨数据库 DDL	尚不支持执行跨多个数据库引用或操作对象的 DDL 语句。
游标（可更新）	不支持可更新的游标。
游标（全局）	不支持 GLOBAL 游标。
游标（获取行为）	不支持以下游标获取行为：FETCH PRIOR、FIRST、LAST、ABSOLUTE 和 RELATIVE
游标类型的输出参数	输出参数不支持游标类型的变量和参数（引发错误）。
游标选项	SCROLL、KEYSET、DYNAMIC、FAST_FORWARD、SCROLL_LOCKS、OPTIMISTIC、TYPE_WARNING 和 FOR UPDATE
数据加密	不支持数据加密。
数据层应用程序（DAC）	不支持使用 DAC 程序包（.dacpac）或 DAC 备份（.bacpac）文件进行数据层应用程序（DAC）导入或导出操作。
DBCC 命令	不支持 Microsoft SQL Server 数据库控制台命令 (DBCC)；支持 DBCC CHECKIDENT。
DROP IF EXISTS	USER 和 SCHEMA 对象不支持此语法。对象 TABLE、VIEW、PROCEDURE、FUNCTION 和 DATABASE 都支持此语法。
加密	内置函数和语句不支持加密。
ENCRYPT_CLIENT_CERT 连接	不支持客户端证书连接。
EXECUTE AS 语句	不支持此语句。
EXECUTE AS SELF 子句	函数、过程或触发器不支持此子句。
EXECUTE AS USER 子句	函数、过程或触发器不支持此子句。
引用数据库名称的外键约束	不支持引用数据库名称的外键约束。
FORMAT	不支持用户定义的类型。

功能或语法	描述
具有大于 100 个参数的函数声明	包含 100 个以上参数的函数声明不受支持。
外部定义的函数	不支持 SQL CLR 函数等外部函数。
全局临时表（名称以 ## 开头的表）	不支持全局临时表。
图表功能	不支持所有 SQL 图表功能。
一般扩展存储过程	某些系统存储过程可为各种维护活动提供从 SQL Server 实例到外部程序的接口，但此处不支持此类系统存储过程，包括 xp_cmdshell 和其他系统存储过程。更多信息请参阅 General Extended stored procedures 。
具有多个前导 @ 字符的标识符（变量或参数）	不支持以多个前导 @ 开头的标识符。
包含 @ 或]] 字符的标识符、表或列名	不支持包含 @ 符号或方括号的表或列名。
内联索引	不支持内联索引。
调用名称在变量中的过程	不支持使用变量作为过程名称。
具体化视图	不支持具体化视图。
NOT FOR REPLICATION 子句	接受并忽略此语法。
ODBC 转义函数	不支持 ODBC 转义函数。
将 DEFAULT 作为参数值的过程调用	不支持将 DEFAULT 作为参数值的过程调用；支持将 DEFAULT 作为参数值的函数调用。
包含 100 个以上参数的过程声明	不支持包含 100 个以上参数的声明。
外部定义的过程	不支持 SQL CLR 过程等外部定义的过程。
过程版本控制	不支持过程版本控制。
过程 WITH RECOMPILE	不支持 WITH RECOMPILE（与 DECLARE 和 EXECUTE 语句结合使用时）。
远程对象引用	不支持针对 Babelfish 链接服务器执行存储过程。由四部分组成的对象名称仅用于读取，不适用于修改远程表。UPDATE 可以在 FROM 子句中引用远程表，而无需对其进行修改。
行级别安全性	不支持具有 CREATE SECURITY POLICY 和内联表值函数的行级安全性。
服务代理功能	不支持服务代理功能。

功能或语法	描述
SESSIONPROPERTY	不支持的属性：ANSI_NULLS、ANSI_PADDING、ANSI_WARNINGS、ARITHABORT、CONCAT_NULL_YIELDS_NULL 和 NUMERIC_ROUNDABORT
SET LANGUAGE	此语句不支持 english 或 us_english 之外的任何值。
SP_CONFIGURE	不支持此系统存储过程。
SQL 关键字 SPARSE	接受并忽略关键字 SPARSE。
表值构造函数语句（FROM 子句）	不支持的语法适用于使用 FROM 子句构造的派生表。
临时表	不支持临时表。
临时过程不会自动删除	不支持此函数。
外部定义的触发器	这些触发器不受支持，包括 SQL 公共语言运行时 (CLR)。
无 SCHEMABINDING 子句	不支持在不使用 SCHEMABINDING 的情况下创建视图，但创建视图时似乎指定了 WITH SCHEMABINDING。在创建函数、过程和触发器时使用 SCHEMABINDING 会被无提示忽略。

不支持的设置

不支持以下设置：

- SET ANSI_NULL_DFLT_OFF ON
- SET ANSI_NULL_DFLT_ON OFF
- SET ANSI_PADDING OFF
- SET ANSI_WARNINGS OFF
- SET ARITHABORT OFF
- SET ARITHIGNORE ON
- SET CURSOR_CLOSE_ON_COMMIT ON
- SET NUMERIC_ROUNDABORT ON
- SET PARSEONLY ON（命令未按预期工作）
- SET FMTONLY ON（仅能阻止 SELECT 语句的执行，而不会影响其他语句的运行，与预期行为不符）

不支持的命令

不支持以下命令的某些功能：

- ADD SIGNATURE
- ALTER DATABASE、ALTER DATABASE SET

- BACKUP/RESTORE DATABASE/LOG
- BACPAC 和 DACPAC FILES RESTORE
- CREATE、ALTER、DROP AUTHORIZATION。数据库对象支持 ALTER AUTHORIZATION。
- CREATE、ALTER、DROP AVAILABILITY GROUP
- CREATE、ALTER、DROP BROKER PRIORITY
- CREATE、ALTER、DROP COLUMN ENCRYPTION KEY
- CREATE、ALTER、DROP DATABASE ENCRYPTION KEY
- CREATE、ALTER、DROP、BACKUP CERTIFICATE
- CREATE AGGREGATE
- CREATE CONTRACT
- CHECKPOINT

不支持的列名称或属性

不支持以下列名称：

- \$IDENTITY
- \$ROWGUID
- IDENTITYCOL

不支持的数据类型

不支持数据类型：HIERARCHYID

不支持的对象类型

不支持以下对象类型：

- COLUMN MASTER KEY
- CREATE、ALTER EXTERNAL DATA SOURCE
- CREATE、ALTER、DROP DATABASE AUDIT SPECIFICATION
- CREATE、ALTER、DROP EXTERNAL LIBRARY
- CREATE、ALTER、DROP SERVER AUDIT
- CREATE、ALTER、DROP SERVER AUDIT SPECIFICATION
- CREATE、ALTER、DROP、OPEN/CLOSE SYMMETRIC KEY
- CREATE、DROP DEFAULT
- CREDENTIAL
- CRYPTOGRAPHIC PROVIDER
- DIAGNOSTIC SESSION
- 索引视图
- SERVICE MASTER KEY
- SYNONYM

不支持的函数

不支持以下内置函数：

表 2-7 不支持函数

函数类型	具体函数
聚合函数	• APPROX_COUNT_DISTINCT • CHECKSUM_AGG • GROUPING_ID • 使用 WITHIN GROUP 子句的 STRING_AGG
加密函数	• CERTENCODED 函数 • CERTID 函数 • CERTPROPERTY 函数
安全函数	• CERTPRIVATEKEY • LOGINPROPERTY
语句、运算符、其他函数	• EVENTDATA 函数 • GET_TRANSMISSION_STATUS • OPENXML

函数类型	具体函数
元数据函数	• COLUMNPROPERTY • TYPEPROPERTY • SERVERPROPERTY 函数 — 不支持以下属性： - BuildClrVersion - ComparisonStyle - ComputerNamePhysicalNetBIOS - HadrManagerStatus - InstanceDefaultDataPath - InstanceDefaultLogPath - IsClustered - IsHadrEnabled - LCID - NumLicenses - ProcessID - ProductBuild - ProductBuildType - ProductUpdateReference - ResourceLastUpdateDateTime - ResourceVersion - ServerName - SqlCharSet - SqlCharSetName - SqlSortOrder - SqlSortOrderName - FilestreamShareName - FilestreamConfiguredLevel - FilestreamEffectiveLevel

不支持的语法

不支持以下语法：

- ALTER DATABASE
- ALTER DATABASE SCOPED CONFIGURATION
- ALTER DATABASE SCOPED CREDENTIAL
- ALTER DATABASE SET HADR
- ALTER INDEX
- ALTER PARTITION FUNCTION

- ALTER PARTITION SCHEME
- ALTER SCHEMA
- ALTER SERVER CONFIGURATION
- ALTER SERVICE、BACKUP/RESTORE SERVICE MASTER KEY 子句
- ALTER VIEW
- BEGIN CONVERSATION TIMER
- BEGIN DISTRIBUTED TRANSACTION
- BEGIN DIALOG CONVERSATION
- BULK INSERT
- CREATE COLUMNSTORE INDEX
- CREATE EXTERNAL FILE FORMAT
- CREATE EXTERNAL TABLE
- CREATE、ALTER、DROP APPLICATION ROLE
- CREATE、ALTER、DROP ASSEMBLY
- CREATE、ALTER、DROP ASYMMETRIC KEY
- CREATE, ALTER, DROP CREDENTIAL
- CREATE, ALTER, DROP CRYPTOGRAPHIC PROVIDER
- CREATE, ALTER, DROP ENDPOINT
- CREATE、ALTER、DROP EVENT SESSION
- CREATE, ALTER, DROP EXTERNAL LANGUAGE
- CREATE、ALTER、DROP EXTERNAL RESOURCE POOL
- CREATE、ALTER、DROP FULLTEXT CATALOG
- CREATE、ALTER、DROP FULLTEXT INDEX
- CREATE、ALTER、DROP FULLTEXT STOPLIST
- CREATE, ALTER, DROP MESSAGE TYPE
- CREATE、ALTER、DROP、OPEN/CLOSE、BACKUP/RESTORE MASTER KEY
- CREATE、ALTER、DROP QUEUE
- CREATE、ALTER、DROP RESOURCE GOVERNOR
- CREATE, ALTER, DROP RESOURCE POOL
- CREATE、ALTER、DROP ROUTE
- CREATE, ALTER, DROP SEARCH PROPERTY LIST
- CREATE, ALTER, DROP SECURITY POLICY
- CREATE, ALTER, DROP SELECTIVE XML INDEX clause
- CREATE、ALTER、DROP SERVICE
- CREATE, ALTER, DROP SPATIAL INDEX
- CREATE, ALTER, DROP TYPE
- CREATE, ALTER, DROP XML INDEX
- CREATE, ALTER, DROP XML SCHEMA COLLECTION

- CREATE/DROP RULE
- CREATE, DROP WORKLOAD CLASSIFIER
- CREATE、ALTER、DROP WORKLOAD GROUP
- ALTER TRIGGER
- CREATE TABLE... GRANT 子句
- CREATE TABLE... IDENTITY 子句
- CREATE USER — 不支持此语法。PostgreSQL语句CREATE USER不会创建与SQL Server 的 CREATE USER语法等效的用户。
- DENY
- END, MOVE CONVERSATION
- EXECUTE with AS LOGIN or AT option
- GET CONVERSATION GROUP
- GROUP BY ALL clause
- GROUP BY CUBE clause
- GROUP BY ROLLUP clause
- INSERT... DEFAULT VALUES
- MERGE
- READTEXT
- REVERT
- UNPIVOT
- SELECT TOP x PERCENT WHERE x <> 100
- SELECT TOP... WITH TIES
- SELECT... FOR BROWSE
- SELECT... FOR XML AUTO
- SELECT... FOR XML EXPLICIT
- SELECT... FOR XML PATH
- SEND
- SET DATEFORMAT
- SET DEADLOCK_PRIORITY
- SET FMONLY
- SET FORCEPLAN
- SET NUMERIC_ROUNDABORT ON
- SET OFFSETS
- SET REMOTE_PROC_TRANSACTIONS
- SET SHOWPLAN_TEXT
- SET SHOWPLAN_XML
- SET STATISTICS
- SET STATISTICS PROFILE
- SET STATISTICS TIME

- SET STATISTICS XML
- SHUTDOWN statement
- UPDATE STATISTICS
- UPDATETEXT
- Using EXECUTE to call a SQL function
- VIEW... CHECK OPTION clause
- VIEW... VIEW_METADATA clause
- WAITFOR DELAY
- WAITFOR TIME
- WAITFOR, RECEIVE
- WITH XMLNAMESPACES construct
- WRITETEXT
- XPATH expressions

2.3 Babelfish 部分支持的功能示例

Babelfish 的每个迭代版本都在逐步扩展对 T-SQL 功能的支持范围，以提升兼容性。但当前版本仍存在部分未支持功能及行为差异。下文详细列出了 Babelfish 与 T-SQL 之间的功能差异，并提供相应解决方案和使用指南。

自 Babelfish 1.2.0 版本起，下列功能目前为有限支持状态：

视图

在Babelfish中不存在如下SQL Server系统视图，详情如[表2-8](#)。

表 2-8 Babelfish 中不存在的 SQL Server 系统视图

视图名称	功能描述
sys.parameters	包含存储过程、函数等对象的参数信息。
sys.periods	存储时态表（历史表）的时间周期定义信息。
sys.service_queues	提供Service Broker服务队列的配置信息。
sys.sql_dependencies	记录数据库对象之间的依赖关系。
sys.stats_columns	显示统计信息关联的列信息。
sys.masked_columns	包含动态数据掩码应用的列信息。
sys.memory_optimized_tables_internal_attributes	内存优化表的内部属性信息。
sys.numbered_procedure_parameters	包含编号存储过程的参数信息。

视图名称	功能描述
sys.allocation_units	显示数据库中分配单元(存储结构)的信息。
sys.event_notifications	包含事件通知配置信息。
sys.extended_procedures	提供扩展存储过程信息。
sys.function_order_columns	包含CLR表值函数中顺序列的信息。
sys.trigger_event_types	列出触发器可以响应的事件类型。

无限值（Infinity Value）

Babelfish 支持 Infinity，SQL Server不支持。

ROWVERSION/TIMESTAMP 赋值行为差异

在 SQL Server 中，每插入（INSERT）或更新（UPDATE）一行均会分配唯一的 ROWVERSION/TIMESTAMP 值，而Babelfish在同一语句中修改的所有行将分配相同的 ROWVERSION/TIMESTAMP值。

例如，当执行影响多行的 UPDATE 或 INSERT SELECT操作时：

- SQL Server 会为每一行的 ROWVERSION/TIMESTAMP 列生成不同值。
- Babelfish (PostgreSQL) 会为所有受影响行赋予相同值。

上述细微差异通常不会对应用程序有负面影响。

对 JSON 数据类型提供有限兼容支持

Babelfish不直接支持SQL Server原生JSON数据类型。而是存储为NVARCHAR、VARCHAR或TEXT数据类型。更多详细信息请参考[JSON data in SQL Server](#)和[JSON data type](#)。

以下JSON函数是Babelfish支持的函数，可以操作JSON：

- JSON_VALUE
- JSON_QUERY
- OPENJSON
- JSON_MODIFY
- ISJSON

标识符差异

T-SQL中定义的索引名 ix1在表t1上，Babelfish会将其内部重命名为类似 ix1t1a5111d2a1767bc43a700e9f2162be019 的形式（ix1t1 + 哈希值）。

- T-SQL透明性：用户仍使用原始标识符编写查询，Babelfish自动映射到内部名称。

- PostgreSQL可见性：在PostgreSQL端（如通过psql查看），实际对象名是包含哈希值的名称。

Babelfish通过哈希算法动态调整标识符，既兼容了SQL Server的长名称和宽松规则，又满足了PostgreSQL的严格限制。这种处理对T-SQL用户无感知，但需注意在PostgreSQL中看到的对象名会包含哈希后缀。更多详细信息参考[Babelfish官方文档标识符描述](#)。

Babelfish中标识符差异对系统视图的影响：

- 若对象名称≤ 64个字符，通过SQL命令访问该对象无影响（use、select等SQL命令），且在sys.databases、sys.indexes、sys.table等系统视图中存储的是对象原始名称。
- 若对象名称>64个字符，通过SQL命令访问该对象无影响（use、select等SQL命令），但在sys.databases、sys.indexes、sys.tables等系统视图中存储的是对象经过转换后的名称"部分前缀+hash值"，且通过简单的字符串处理无法还原对象原始名称。

标识符差异的影响场景：

- DRS在以Babelfish为目标端的迁移链路中，迁移开始前会对目标Babelfish实例进行预检查。预检查会根据sys.*系统视图查询对象是否已存在，如果源端SQL Server存在超过64个字符的数据库对象，则预检查会存在问题。
- 用户的业务访问sys.*系统视图时，对于超过64字符的对象，也会存在查找无结果的问题。

系统视图表差异

如下视图存在差异：

sys.hash_indexes

- 列缺失：is_ignored_in_optimization 列在 Babelfish 中不存在。

<No Available>	<No Available>	<No Available>	is_ignored_in_optimization	104	bit		no	no
----------------	----------------	----------------	----------------------------	-----	-----	--	----	----

- 数据类型匹配：name 列的数据类型一致（因为sysname 是 nvarchar 的别名，不存在不兼容），sys.triggers，sys.procedures，sys.check_constraints同此。

sys.hash_indexes								
BABELFISH			MS-SQL			MATCH		
name	system_type_id	system_type_name	name	system_type_id	system_type_name	Name	Type	
object_id	23	int	object_id	56	int	yes	yes	
name	17173	sysname	name	231	nvarchar	yes	no	

sys.index_columns

Babelfish缺失column_store_order_ordinal列：

<No Available>	<No Available>	<No Available>	column_store_order_ordinal	48	tinyint		no	no
----------------	----------------	----------------	----------------------------	----	---------	--	----	----

sys.system_objects 和 sys.all_objects

- sysname：nvarchar 的别名。

- type：在mssql和babelfish中都存在。

sys.objects

Babelfish 的sys.objects也有type列, babelfish的“type”列的类型是char(2)，与mssql匹配，**sys.sequences** 同此；
sysname是nvarchar的别名，所以不是兼容性问题。

sys.schemas

<No Available>	<No Available>	<No Available>	schema_id	56	int		no	no
<No Available>	<No Available>	<No Available>	principal_id	56	int		no	no

Babelfish也有schema_id和principal_id列，它们的类型是“oid”，是一个4字节的int数据类型，所以不存在兼容性问题，详细信息参考[oid类型官方说明](#)

sys.sql_modules

Babelfish中缺失inline_type 和 is_inlineable 列。

<No Available>	<No Available>	<No Available>	inline_type	104	bit		no	no
<No Available>	<No Available>	<No Available>	is_inlineable	104	bit		no	no

sys.computed_columns

sysname是nvarchar的别名，不存在兼容性问题。

另一个不同点是：MSSQL 使用 tinyint，而 Babelfish 使用 int。由于 int 类型比 tinyint 大，因此这一点上，应用程序从 MSSQL 迁移到 Babelfish不存在不兼容。

sys.computed_columns							
BABELFISH			MS-SQL			MATCH	
name	system_type_id	system_type_name	name	system_type_id	system_type_name	Name	Type
object_id	23	int	object_id	56	int	yes	yes
name	17173	sysname	name	231	nvarchar	yes	no
column_id	23	int	column_id	56	int	yes	yes
system_type_id	23	int	system_type_id	48	tinyint	yes	no
user_type_id	23	int	user_type_id	56	int	yes	yes

在[Microsoft 的文档](#)中，给出了 tinyint 和 int 的限制，如下所示：

Datentyp	Range	Bereichsausdruck	Storage
bigint	-9,223,372,036,854,775,808 bis 9,223,372,036,854,775,807	-2^{63} bis $2^{63}-1$	8 Bytes
int	-2,147,483,648 bis 2,147,483,647	-2^{31} bis $2^{31}-1$	4 Bytes
smallint	-32.768 bis 32.767	-2^{15} bis $2^{15}-1$	2 Bytes
tinyint	0 bis 255	2^0-1 bis 2^8-1	1 Byte

sys.table_types和 sys.types 同此。

另外，Babelfish一侧，**sys.computed_columns** 缺少以下列：

Babelfish	MSSQL	TYPE_NAME	NOT_MATCH
?	is_data_deletion_filter_column	104 bit	0
?	ledger_view_column_type	56 int	0
?	ledger_view_column_type_desc	231 nvarchar	0
?	is_dropped_ledger_column	104 bit	0

sys.sysdatabases

差异如表

BABELFISH - MSSQL

name	system_type_id	system_type_name	name	system_type_id	system_type_name
name	25	text	name	231	nvarchar

'TEXT' 变量比 'sysname'（128字符）大，因此不存在不兼容。

sys.sysprocesses

- spid、blocked、dbid 列在 MS-SQL 中定义为 smallint，在 BABELFISH 中定义为 int。
- 在 Babelfish 的中未定义“uid”列的类型 ID，根据 Postgres 文档，该类型当前实现为无符号的四字节整数 oid；
也未定义“context_info”列的类型 ID，根据 Postgres 文档，数据类型允许存储二进制字符串 bytea
- “loginname”（ms-sql）和“loginame”（babelfish），babelfish一侧缺少 'n' 字符。
- babelfish缺少“page_resource”列。

影响：不影响功能使用。

sys.key_constraints

sysname 是 nvarchar（128）的别名。

Babelfish缺失is_enforced列，在 SQL Server 中，此列告诉我们是否强制执行了 PRIMARY KEY 或 UNIQUE 约束，即 SQL Server 在数据修改期间是否主动检查唯一性，在Babelfish中不支持该参数。

sys.default_constraints

sysname 是 nvarchar (128) 的别名; “modified_date” 与 “modify_date” 等同。

modified_date	17456	datetime	modify_date	61	datetime
---------------	-------	----------	-------------	----	----------

sys.stats

sysname 是 nvarchar (128) 的别名， babelfish中缺少 “auto_drop” 列。

sys.indexes

Babelfish缺少以下几列：

列名 (Column)	用途 (Purpose)	适用对象 (Applies To)
compression_delay	延迟压缩行组的时间（列存储索引）	列存储索引 (Columnstore indexes)
suppress_dup_key_messages	抑制重复键错误的详细信息	唯一索引 (Unique indexes)
optimize_for_sequential_key	优化递增键索引的插入性能，减少插入争用	B树索引（如自增列上的索引） (B-tree indexes)
is_ignored_in_optimization	在查询优化过程中忽略该索引	所有索引类型 (Any index)

sys.identity_columns

Babelfish缺少以下几列：

列名 (Column Name)	描述 (Description)
is_data_deletion_filter_column	标识该列是否用于筛选账本视图（ledger view）中的数据删除操作。
ledger_view_column_type	账本视图列的数字类型（用于内部存储）。
ledger_view_column_type_desc	账本视图列的描述性类型（如 GENERATED ALWAYS 或 APPEND_ONLY）。
is_dropped_ledger_column	标识该列是否原本属于账本表（ledger table），但已被删除。

列名 (Column Name)	描述 (Description)
graph_type	图的列的数字类型（用于内部存储）。
graph_type_desc	图的列的描述性类型（如 NODE 或 EDGE）。

sys.syscharsets

与之前的类似，MSSQL是smallint和tiny，babelfish是int的区别，不再赘述。

sys.assembly_modules/sys.synonyms

与之前的类似，sysname 是 nvarchar（128）的别名，不再赘述。

表元数据不匹配说明

如下视图表元数据不匹配，详情见前文，不再赘述，不匹配是由于 SQL Server 和 PostgreSQL（Babelfish 的底层引擎）之间的架构差异而导致的预期差异，只是列中的文本表示形式不同，但从功能上讲，返回的元数据是等效的，因此不会影响使用 Babelfish 的应用程序的功能正确性。

- **sys.views**
- **sys.columns**
- **sys.databases**
- **sys.foreign_keys**
- **sys.all_views**
- **sys.partitions**
- **sys.sql_expression_dependencies**
- **sys.all_columns**
- **sys.tables**

参数

Babelfish不支持以下设置：

SET ANSI_NULL_DFLT_OFF ON

SET ANSI_NULL_DFLT_ON OFF

SET ANSI_PADDING OFF

SET ANSI_WARNINGS OFF

这将导致SET ANSI_DEFAULTS命令对于这些服务器端设置无法正常工作。

以下参数禁止修改：

- `babelfishpg_tsql.allow_antlr_to_unsupported_grammar_for_testing`
- `babelfishpg_tsql.ansi_null_dflt_off`
- `babelfishpg_tsql.ansi_null_dflt_on`
- `babelfishpg_tsql.ansi_padding`
- `babelfishpg_tsql.ansi_warnings`
- `babelfishpg_tsql.arithabort`
- `babelfishpg_tsql.arithignore`
- `babelfishpg_tsql.cursor_close_on_commit`
- `babelfishpg_tsql.disable_internal_savepoint`
- `babelfishpg_tsql.language`
- `babelfishpg_tsql.numeric_roundabort`

更多详细内容请参考[Babelfish官方文档-不应修改的参数](#)

游标（cursors）

Babelfish中不支持SET CURSOR_Closed_ON_COMMIT ON。此外，Babelfish不识别在存储过程或函数中动态定义的游标。作为一种解决方法，您可以使用临时表重写此类游标。

也不支持以下游标类型和行为：

- 可更新游标（Updatable cursors）
- 全局游标（Global cursors）
- 不支持游标提取行为：FETCH PRIOR, FIRST, LAST, ABSOLUTE, 和 RELATIVE。
- 游标类型的输出参数：游标类型的变量和参数不能用作输出参数（会引发错误）。
- 光标选项：不支持以下选项：SCROLL, KEYSET, DYNAMIC, FAST_FORWARD, SCROLL_LOCKS, OPTIMISTIC, TYPE_WARNING, 和 FOR UPDATE。

此外，[Babelfish社区文档](#)指出：“当 ANSI_DEFAULTS 开启时，它会启用以下设置：ANSI_NULLS、ANSI_NULL_DFLT_ON、IMPLICIT_TRANSACTIONS、ANSI_PADDING、QUOTED_IDENTIFIER、ANSI_WARNINGS 和 CURSOR_CLOSE_ON_COMMIT。驱动程序和提供商将 CURSOR_CLOSE_ON_COMMIT 和 IMPLICIT_TRANSACTIONS 设置为 OFF（因此 CURSOR_CLOSE_ON_COMMIT 实际上在连接时处于 OFF 状态，尽管 ANSI_DEFAULTS 处于 ON 状态）”。

命令

- SET STATISTICS 和 SET STATISTICS_ALL 不支持，但是有代替的功能：
 - SET BABELFISH_STATISTICS PROFILE [ON|OFF]：对于实际查询执行计划，设置为 ON，此函数实现了 PostgreSQL 命令 EXPLAIN ANALYZE 的功能。
 - SET BABELFISH_SHOWPLAN_ALL [ON|OFF]：设置为 ON 可生成估计的查询执行计划，此函数实现 PostgreSQL 的 EXPLAIN 命令的行为，使用此命令获取给定查询的解释计划。

另外，babelfish社区文档中表示：“从版本 2.2.0 开始，您可以将 `escape_hatch_showplan_all` 参数设置为 `ignore`，以避免在 SQL Server 语法中对 SHOWPLAN_ALL 和 STATISTICS PROFILE SET 命令使用 `BABELFISH_` 前缀。”

因此，为了将其用作不带 BABELFISH 前缀的 SET **SHOWPLAN_ALL** 和 SET **STATISTICS PROFILE** 应该运行：

```
EXEC sp_babelfish_configure 'babelfishpg_tsql.escape_hatch_showplan_all', 'ignore';
```

或：

```
EXEC sp_babelfish_configure 'babelfishpg_tsql.escape_hatch_showplan_all', 'strict';
```

可以使用 sp_babelfish_configure 更改参数的设置，如下例所示：

```
execute sp_babelfish_configure 'explain_buffers', 'on';
```

- Babelfish不支持SET SHOWPLAN_XML

Babelfish可以输出查询计划，并且可以以 XML 格式输出查询计划。详细信息请[参考文档](#)了解如何查看Babelfish 查询计划。

在运行查询之前，您可以使用以下方法可视化STATISTICS PROFILE，可视化时需执行SET BABELFISH_STATISTICS PROFILE ON。

- 执行如下命令设置说明格式。

```
execute sp_babelfish_configure 'explain_format', 'xml'
```

- 执行如下命令进行可视化。

```
select set_config ( 'babelfishpg_tsql.explain_format', 'xml', false );  
SET BABELFISH_STATISTICS PROFILE ON;
```

- Babelfish 不支持SET FORCEPLAN。

- Babelfish 不支持SET SHOWPLAN_TEXT。

- 对于SET NOEXEC，Babelfish只支持"OFF" ,更多详细信息请参考[Babelfish限制社区文档](#)。

- Babelfish不支持SET TEXTSIZE功能，但有代替功能，如下：

Babelfish在会话初始化时，将TEXTSIZE参数设置为4096。如果要更改它，按如下操作：

如果将“-y”参数设置为小于“TEXTSIZE”值，则无法调整输出的文本大小，这种情况下，优先考虑-y参数。

- a. 使用sqlcmd命令行工具，执行以下命令进行连接。

```
sqlcmd -U babelfish_user -P 12345678 -No -y <textsize_value>
```

- b. 设置TEXTSIZE参数。

```
SET TEXTSIZE <textsize_value>
```

- c. 执行如下命令查询TEXTSIZE参数的值。

```
SELECT id, LEFT(name, CAST(current_setting('babelfishpg_tsql.textsize') as int )) AS name  
FROM <table_name>;  
SELECT * FROM <table_name>;
```

函数

如下函数存在差异。

DATABASE_PRINCIPAL_ID()

该函数定义了两种有效的使用模式：

- 不带参数，检索当前用户的数据库主体 ID。
- 使用指定的数据库主体名称作为字符串输入，例如（ 'db_owner' ）。

因此用户应避免传递空字符串：DATABASE_PRINCIPAL_ID('')，这是不受支持的用法。对此函数详细解释请参考[DATABASE_PRINCIPAL_ID](#)。

FULLTEXTSERVICEPROPERTY()

Babelfish中无论输入参数如何，函数总是返回0，因为当前在 Babelfish 中不支持全文搜索功能；

在 SQL Server 中，无效属性通常会返回NULL，如果安装并启用了括号内的服务，则返回1，例如FULLTEXTSERVICEPROPERTY('IsFullTextInstalled') 返回 1。

concat_ws

PostgreSQL（PG）中的一个字符串聚合函数，用于将多个字符串（或文本值）按指定的分隔符（separator）连接成一个字符串。

函数语法：

```
concat_ws(separator, str1, str2, ..., strN)
```

注意：第一个参数用作分隔符字符串，不应为 NULL，否则会返回NULL。其他参数填 NULL 无影响，会自动忽略。详情见[postgresql官方文档说明](#)，这一点上babelfish与pg保持一致。

因此为了正常使用此函数，用户应避免在第一个参数填入NULL。

HAS_PERMS_BY_NAME

SQL Server 中的一个安全相关函数，用于检查当前用户对指定数据库对象是否拥有特定权限。

函数语法：

```
HAS_PERMS_BY_NAME (  
    securable,  
    securable_class,  
    permission  
)
```

参数说明

- securable：安全对象名称。要检查权限的对象名称，可以是表、视图、存储过程等数据库对象，为 NULL 时表示当前数据库。
- securable_class：安全对象类别。
支持的对象类型：如 DATABASE, OBJECT, SCHEMA, TABLE, VIEW 等。
- permission：权限，即要检查的具体权限，如 'SELECT', 'INSERT', 'UPDATE', 'DELETE' 等。

返回值

- 1：用户拥有指定权限。
- 0：用户不拥有指定权限。
- NULL：参数无效或查询失败。

SQL server与babelfish差异

- 当对象名称为 NULL 但指定了对象类别时，SQL Server 会认为这是检查当前数据库的权限并返回 1；
- 而 Babelfish 实现则要求这种情况下两个参数必须同时为 NULL 或同时不为 NULL，否则返回 NULL。

服务器身份验证输出不匹配

因为如果身份验证方法设置为 `trust`，PostgreSQL 将允许在不验证密码的情况下登录，这是问题的根本原因。

为了确保强制实施密码身份验证，我们必须验证 `pg_hba.conf` 文件没有对有问题的用户使用 `trust` 方法。

按以下步骤操作：

1. 打开 `pg_hba.conf` 文件，该文件通常位于 PostgreSQL 数据目录（例如，`/home/postgres/taurus-postgres/pg/install/data/pg_hba.conf` 或类似目录）中。
2. 查找与特定用户或连接类型（例如，`local`、`host`）相关的条目。
3. 确保将方法字段设置为 `md5`（而不是 `trust`），这将强制进行密码检查。

```
# "local" is for Unix domain socket connections only
local    all             root                                md5
# IPv4 local connections:
host     all             root                                127.0.0.1/32        md5
# IPv6 local connections:
host     all             root                                ::1/128            md5
```

DATEADD函数输出不匹配：

运行此命令：

```
SELECT 'DATEADD - NULL Interval' AS TestCase, DATEADD(DAY, NULL, '2025-04-25') AS Result;
```

MSSQL报错：NULL 不是有效的参数，但是BABELFISH返回NULL

原因：DATEADD 函数的调用方已使用“STRICT”关键字进行声明，STRICT指示每当函数的任何参数为 `null` 时，该函数始终返回 `null`，见[参考文档](#)。

时间精度

- MS SQL：SQL Server 支持将 `datetime2` 数据类型精度精确到 7 位数字和 3 位小数数字的几分之一秒。SQL Server 将此精度舍入为 3 毫秒。这意味着，当您存储特定时间值时，毫秒可能会四舍五入，并最终与您指定的不同方式存储。

例如

```
SELECT CAST('2016-12-26 23:59:59.999' as datetime)
```

返回值：2016-12-27 00:00:00.000

- Babelfish：PostgreSQL 支持小数秒的 6 位精度，超过6位不支持。PostgreSQL 不执行 SQL Server 3 毫秒舍入。预计此行为不会产生不利影响。

详情见[babelfish官方文档](#)。

字符串类型

关于 TEXT 和 VARCHAR(MAX) 的注意事项：

- Microsoft SQL Server 已弃用 TEXT 数据类型（[参考文档](#)），建议改用 VARCHAR(MAX)。
- 在原生 MSSQL 中，VARCHAR(MAX) 在某些操作（如排序、索引）下可能受 8060 字节行限制影响，导致错误（如 Error 512）。
- Babelfish 运行于 PostgreSQL 环境，PostgreSQL 的 VARCHAR 默认无硬性长度限制（除非显式指定 VARCHAR(n)），因此：

- TEXT 类型在 Babelfish 中仍可使用，且不会因超长文本报错（与 MSSQL 行为不同）。
- VARCHAR(MAX) 在 Babelfish 中不会强制限制为 8000 字节，允许存储更大的文本（符合 PostgreSQL 特性）。
- 如需严格兼容 MSSQL 行为，请避免使用 TEXT 并注意 VARCHAR(MAX) 的行限制场景。

2.4 常用 SQL 语句示例

数据查询

- 基本查询

```
SELECT * FROM users;
```
- 条件查询

```
SELECT name, age FROM users WHERE age > 30;
```
- 排序

```
SELECT * FROM users ORDER BY create_time DESC;
```
- 分组聚合

```
SELECT age, COUNT(*) FROM users GROUP BY age;
```
- TOP 语法

```
SELECT TOP 10 * FROM users;
```

数据插入与修改

- 插入

```
INSERT INTO users (name, age) VALUES ('Alice', 25);
```
- OUTPUT 语法

```
INSERT INTO t(name) OUTPUT INSERTED.* VALUES('abc');
```
- 更新数据

```
UPDATE users SET age = age + 1 WHERE name = 'Alice';
```
- 删除数据

```
DELETE FROM users WHERE age < 18;
```

表操作

- 创建表

```
CREATE TABLE users (id INT PRIMARY KEY, name NVARCHAR(50), age INT);
```
- 删除表

```
DROP TABLE users;
```
- 添加字段

```
ALTER TABLE users ADD email NVARCHAR(100);
```
- 删除字段

```
ALTER TABLE users DROP COLUMN email;
```
- 修改字段类型

```
ALTER TABLE users ALTER COLUMN age SMALLINT;
```
- 全局临时表

```
CREATE TABLE ##tmp(x INT);
```
- 局部临时表

```
CREATE TABLE #tmp(x INT);
```

数据完整性与索引

- 添加主键
ALTER TABLE users ADD CONSTRAINT pk_users PRIMARY KEY (id);
- 创建唯一索引
CREATE UNIQUE INDEX idx_users_name ON users(name);
- 添加外键
ALTER TABLE orders ADD CONSTRAINT fk_orders_user FOREIGN KEY (user_id) REFERENCES users(id);

视图与存储过程

- 创建视图
CREATE VIEW adult_users AS SELECT * FROM users WHERE age >= 18;
- 删除视图
DROP VIEW adult_users;
- 存储过程
CREATE PROCEDURE get_adult_users AS
SELECT * FROM users WHERE age >= 18;
- 执行存储过程
EXEC get_adult_users;

数据库操作

- 查询版本
SELECT @@version;
- 数据库切换
USE testdb;

2.5 支持的插件

Babelfish支持的插件列表如表2-9所示。

表 2-9 支持的插件列表

插件列表	功能简介
address_standardizer	标准化地址数据，使其符合邮政标准格式。
address_standardizer_data_us	提供美国地址标准化所需的参考数据。
amcheck	检查 B-tree 索引的完整性，防止数据损坏。
autoinc (contrib-spi)	自动生成递增的序列值，常用于自增 ID 管理。
dblink	允许在 PostgreSQL 中跨数据库连接查询。
dict_int	全文搜索字典，支持整数类型的搜索优化。
dict_xsyn	同义词词典扩展，支持全文搜索中的同义词替换。
fuzzystrmatch	提供模糊字符串匹配功能（如 Levenshtein 距离、Soundex 等）。

插件列表	功能简介
hll	高性能 HyperLogLog 算法，用于大数据基数估算。
insert_username (contrib-spi)	自动记录插入数据的用户名。
intagg	提供整数聚合和枚举功能。
intarray	提供高效的整数数组操作函数。
ip4r	支持 IPv4 和 IPv6 地址的高效存储和查询。
isn	提供国际标准编号（如 ISBN、ISSN 等）的数据类型和校验。
jsonb_plperl	允许在 PL/Perl 中处理 JSONB 数据。
lo	管理大对象（Large Object）存储，如二进制文件。
moddatetime (contrib-spi)	自动记录表的最后修改时间。
mysql_fdw	允许 PostgreSQL 查询 MySQL 数据库（Foreign Data Wrapper）。
orafce	提供 Oracle 兼容函数，便于迁移 Oracle 数据库。
pg_buffercache	监控 PostgreSQL 共享缓冲区的使用情况。
pg_bigm	提供高效的 2-gram（二元分词）全文搜索功能，支持非拉丁语系（如中文、日文）的模糊匹配和快速检索。
pg_cron	在 PostgreSQL 内部执行定时任务（类似 crontab）。
pg_freespacemap	检查表的空闲空间映射（Free Space Map）。
pg_hint_plan	允许手动指定查询执行计划，优化复杂 SQL 查询。
pg_partman	自动分区管理，支持按时间/范围分区。
pg_prewarm	预加载表或索引到共享缓冲区，提高查询性能。
pg_repack	在线重组表，减少膨胀，无需长时间锁定。
pg_stat_statements	记录 SQL 执行统计，用于性能分析。
pg_trgm	提供基于三元组的模糊文本搜索和相似度计算。
pg_visibility	检查表的可见性映射（Visibility Map），用于优化 VACUUM。
pgAudit	提供详细的数据库审计日志，记录 SQL 操作。
pgrouting	地理空间路由计算，支持最短路径、网络分析等。
pgrowlocks	监控行级锁定情况，排查锁冲突。
pgstattuple	统计表和索引的物理存储情况，帮助优化空间使用。

插件列表	功能简介
plpgsql	PostgreSQL 默认的过程语言（PL/pgSQL）。
PostGIS	地理信息系统（GIS）扩展，支持空间数据存储和查询。
postgis_raster	PostGIS 的栅格数据处理扩展。
postgis_topology	PostGIS 的拓扑数据处理扩展。
postgis_sfcgal	为 PostGIS 提供了基于 SFCGAL 库的 3D 几何操作和高级空间分析功能。
postgres_fdw	允许 PostgreSQL 查询其他 PostgreSQL 数据库（Foreign Data Wrapper）。
SEG	提供线段（segment）数据类型，用于区间计算。
sslinfo	提供 SSL 连接信息，如客户端证书详情。
tablefunc	提供表函数（如 crosstab 交叉表分析）。
TCN	触发器变更通知，用于监听数据变更事件。
tds_fdw	允许 PostgreSQL 查询 SQL Server 数据库。
tsm_system_rows	提供基于行数的表采样方法，用于随机抽取指定行数的数据（如 TABLESAMPLE SYSTEM_ROWS(100)）。
tsm_system_time	提供基于时间的表采样方法，在指定时间内随机抽取数据（如 TABLESAMPLE SYSTEM_TIME(1000)）。
unaccent	移除文本中的变音符号（如 é → e），便于模糊搜索和标准化文本处理。
uuid-oss	生成符合标准的 UUID（通用唯一标识符），支持多种算法（如 v1、v3、v4、v5）。
pg_sql_history	记录用户执行的 SQL 语句历史，便于审计和问题排查（需配合日志配置）。
rds_pg_sql_ccl	控制并发 SQL 查询的执行，防止资源过载（如限制并发连接数）。
rds_hwdrs_ddl	通过事件触发器自动记录并管理 PostgreSQL 数据库的 DDL 操作历史，包括表/视图/索引等对象的创建、修改和删除语句。
rds_operate_privs	提供权限安全管控能力。
pg_roaringbitmap	高效压缩位图计算库，适用于大数据集的快速交集、并集运算（如用户画像分析）。
pg_stat_monitor	增强版 SQL 监控工具，提供更细粒度的查询统计（如执行计划、错误跟踪）。
passwordcheck	强制密码复杂度检查（如长度、大小写、特殊字符），提升数据库安全性。

插件列表	功能简介
pg_qualstats	收集 SQL 查询中 WHERE 子句的谓词统计，帮助优化索引设计。
pgl_ddl_deploy	自动化 DDL 变更部署，支持多节点同步（如主从架构下的表结构同步）。
test_decoding	逻辑解码插件，用于解析 WAL 日志。
auth_delay	故意延迟认证失败响应（防暴力破解），可配置延迟时间（单位：毫秒）。
pageinspect	检查数据页的底层存储结构（如 Heap/Index 页），用于高级调试和故障修复。
pg_surgery	修复损坏的表数据（如绕过约束强制修改数据）。
rum	高级全文搜索索引，支持短语搜索、邻近排序和加权查询。
xml2	提供 XML 解析和操作函数（如 XPath 查询、XML 生成），支持 XML 数据类型处理。