|  |  |  |  |
| --- | --- | --- | --- |
| **上周进度** | | | |
| **工作内容** | **工作描述、遇到的困难、及解决方案** | **JIRA** | **完成进度** |
| 添加创建index对指定字段长度语法的支持 | 在sql.y文件中添加相应的声明和规则并添加相应的python语义测试 | WARP-43454 | 100% |

|  |  |  |  |
| --- | --- | --- | --- |
| **上周进度** | | | |
| **工作内容** | **工作描述、遇到的困难、及解决方案** | **JIRA** | **完成进度** |
| grant select on a.\* to u1; 这样的语句在转化a为内部的database时实现错误 | 修正了在数据库名前加schema name的方法 | WARP-44337 | 90% |
| 创建local index时会将指定的索引长度字段一起识别成列名 | 新增columnDetail结构体，将索引列名和索引长度字段分开 | WARP-44714 | 90% |

|  |  |  |  |
| --- | --- | --- | --- |
| **上周进度** | | | |
| **工作内容** | **工作描述、遇到的困难、及解决方案** | **JIRA** | **完成进度** |
| grant select on a.\* to u1; 这样的语句在转化a为内部的database时实现错误 | 修正了在数据库名前加schema name的方法 | IMG_256 | 100% |
| 创建local index时会将指定的索引长度字段一起识别成列名 | 新增columnDetail结构体，将索引列名和索引长度字段分开 | IMG_257 | 100% |
| alter table add 语句不支持unique | 在sql.y中添加了相关语法支持 | IMG_258 | 100% |
| create user u1@localhost时vtgate能够通过 | 在handleDCL中添加判断，抛出异常 | IMG_259 | 100% |

|  |  |  |  |
| --- | --- | --- | --- |
| **上周进度** | | | |
| **工作内容** | **工作描述、遇到的困难、及解决方案** | **JIRA** | **完成进度** |
| use db;  drop database db // 报db not exist  create database a // 报错db not exist | mfed执行drop database db时，同时会执行drop server db，但因为db已经不存在，所以报错。  解决：对schema DDL，默认不调用use db | WARP-44347 | 100% |
| use db;  drop databas db;  select database // 报错db not exist | use db 语句给conn.Schema赋值为db,  而session.targetString = conn.Schema，session.targetString不为空，那么执行任一sql，都将默认  调用use db。  解决：每次执行完drop database db，将conn.Schema置为空。 | WARP-45023 | 100% |
| use db;  select database() // 返回kundb1\_db | 将select database()语句发到mfed上执行，将返回db | WARP-45023 | 100% |
| 未use任何数据库时：  select database(); // 返回vt\_\_mfed | 删除dbconfigs中所有用户默认指定的dbName  并禁止调用默认拼接数据库的代码 | WARP-45023 | 100% |

|  |  |  |  |
| --- | --- | --- | --- |
| **上周进度** | | | |
| **工作内容** | **工作描述、遇到的困难、及解决方案** | **JIRA** | **完成进度** |
| Python测试中，未use database时:  select database() // 返回vt\_\_mfed | 在tablet.py中将默认的数据库置为空  并修改manager的template | WARP-45023 | 90% |
| SELECT @rn := @rn+1 AS rn, t.\* FROM (SELECT @rn:=0) r, customer t; // 报错 | 将@rn识别为用户自定义变量，返回的是int(ch)。 解决: 修改条件判断，缩小限定范围，返回ID | WARP-45048 | 100% |
| USER DEFINED VARIABLES  用户自定义变量新功能添加 | 进度：  目前已支持  SET @var\_name = expr [, @var\_name = expr] ...  语句的词法语法解析，并将key-value值存入session。  下一步：参考开源vitess，看select @var\_name时，是直接从gate中session获取 还是 传到底层mysql。  改动：  1、将用户自定义变量和系统变量分开，新添token：AT\_ID、AT\_AT\_ID  用非终结符id\_or\_var替换ID，对应着ID、AT\_ID、AT\_AT\_ID。  2、ColIdent类型新添at属性 | WARP-33116 | 50% |

|  |  |  |  |
| --- | --- | --- | --- |
| **上周进度** | | | |
| **工作内容** | **工作描述、遇到的困难、及解决方案** | **JIRA** | **完成进度** |
| USER DEFINED VARIABLES（UDV）  用户自定义变量新功能添加 | 进度：  目前已支持  SET @var\_name = expr [, @var\_name = expr] ...   语句的词法语法解析，并将key-value值存入session。  expr支持Int64、Float64、varchar、bool、NULL  SELECT @v1;  vitess开源社区支持的是将UDV从session中取出后，存入bindVar中，  最后在vttablet中解析，存入rows，这样做减少了代码的重用，但是耗  费了时间，所以采取的方式是，直接从gate中获取。  改动：  1、将用户自定义变量和系统变量分开，新添token：AT\_ID、AT\_AT\_ID  用非终结符id\_or\_var替换ID，对应着ID、AT\_ID、AT\_AT\_ID。  2、ColIdent类型新添at属性  3、session中新添属性UserDefinedVariables  4、新增variableScope常量用来识别UDV  难点：  因为对ID进行了改动，并且ColIdent新添属性，所以代码改动较大。  未支持：  1、不支持expr是十六进制和二进制，即：  SET @v1 = X'41';  SET @v1 = b'1000001';  2、包括开源社区在内，不支持Prepared Statement语句，即：  SET @c = "c1";  SET @s = CONCAT("SELECT ", @c, " FROM t"); // 也不支持表达式赋值  PREPARE stmt FROM @s;  EXECUTE stmt;  DEALLOCATE PREPARE stmt; | WARP-33116 | 80% |

|  |  |  |  |
| --- | --- | --- | --- |
| **上周进度** | | | |
| **工作内容** | **工作描述、遇到的困难、及解决方案** | **JIRA** | **完成进度** |
| USER DEFINED VARIABLES（UDV）  用户自定义变量新功能添加 | 新增：  添加了expr十六进制和二进制字符的支持，目前已和vitess开源社区功能保持一致 | WARP-33116 | 100% |
| // 不支持删除外键约束  alter table child drop foreign key test  // 不支持alter添加外加约束  alter table child add foreign key fk(parent\_id) references parent(id); | 原因：sql.y中语法不完善  解决：在alter\_object\_type非终结符下添加FOREIGN规则 | WARP-45953 | 100% |
| CREATE TABLE child (    id INT, parent\_id INT,   INDEX par\_ind (parent\_id),    FOREIGN KEY (parent\_id)  // 报语法错   REFERENCES parent(id)   ON DELETE CASCADE   ) ENGINE=INNODB; | 原因：sql.y中语法考虑不周全，导致FOREIGN KEY前必须强制强制添加  CONSTRAINT [symbol]  解决：在constraint\_definition非终结符下添加constrain\_info规则  注意：当外键列有索引时将不会对外键列创建索引，而是直接添加外键约束；  当外键列没有索引时，将创建与外键列名同名的索引，再添加外键约束，并且  删除外键约束时，索引不会删除。 | WARP-45946 | 100% |
| 添加 create or replace Trigger/Procedure 语法支持 | 原因：mysql 8.0 只支持create or replace View  解决：create or replace trigger/procedure a <--->  drop if exist trigger/procedure a;  create trigger/procedure a; | WARP-46069 | 100% |
| 新增Explain explainable statement 语法支持  {EXPLAIN | DESCRIBE | DESC} ANALYZE select\_statement  {EXPLAIN | DESCRIBE | DESC} [explain\_type] {explainable\_stmt | FOR CONNECTION connection\_id}  explain\_type: { FORMAT = format\_name }  format\_name: { TRADITIONAL | JSON | TREE // 8.0 新添 }  explainable\_stmt: { // vitess只支持crud SELECT statement | TABLE statement // 8.0 新添 | DELETE statement | INSERT statement | REPLACE statement | UPDATE statement } | 原因：kundb只支持explain table（都发到mfed)，explain语句语法解析为other\_statement，  没有专门的explain语法支持。  解决：sql.y添加explain\_statement，语法树添加Explain节点，解析器添加StmtExplain选项，  在gate解析explainable\_statement, 查看Plan，如果explainable\_statement要发到\_mfed执行，  则将原sql发到\_mfed的第一个shard, 否则，发到第一个kundb1的第一个shard  暂不支持：  1、{EXPLAIN | DESCRIBE | DESC} ANALYZE select\_statement  2、{EXPLAIN | DESCRIBE | DESC} FOR CONNECTION connection\_id  3、explainable\_stmt不支持REPLACE statement、TABLE statement  4、不支持format\_name: Tree  5、Explain select\_statement 后  不支持show warnings | WARP-45242 | 80% |
| db.table.column 处理存在问题，把db当成了表名 | 原因：代码版本太老 | WARP-46024 | 100% |

|  |  |  |  |
| --- | --- | --- | --- |
| **上周进度** | | | |
| **工作内容** | **工作描述、遇到的困难、及解决方案** | **JIRA** | **完成进度** |
| 【KunDB-GXB】创建foreign key语句中的表名不正确  references 中的表明不支持db.tableName   --alter table db.t1 add constraint fkk foreign key fk(id1) references db.t2(id2); | 原因：alter\_statement只解析到foreign，之后的语法默认正确，所以在vtgate层不会  在db前添加keyspaceName，导致mysql报数据库不存在。  解决：DDL语法树中新增ForeignSpec节点，在vtgate层解析出来后，转为新的sql，再发到  mysql。 | WARP-46361 | 100% |
| 【KunDB-GXB】trigger名字对大小写敏感  create trigger Tri\_item\_insert ....  drop trigger if exists Tri\_item\_insert // 执行不报错，但是不会真正删除 | 原因：create trigger statement因为要对sql进行直接修改，所以引用的是原sql，大小写不变，  而drop trigger statement直接将语法树转为sql，语法树上的节点名都是小写，而mariadb和mysql  对triggerName的大小写敏感，故导致kundb对triggerName大小写敏感。  解决：drop trigger时不Format转小写，直接将原sql发到mysql | WARP-46416 | 100% |
| 【KunDB-GXB】create or replace 修改已存在procedure，报procedure已存在，修改失败 | 原因：采用了字符串替换的方法，风险很大  解决：新建excludeCreateOrReplace提取sql | WARP-46413 | 100% |
| 【KunDB-GXB】create procedure，名称为schema.name的格式，创建可以成功，但是DROP报存储过程不存在 | 原因：创建procedure时，procedureName带了反引号，kundb将`schema.name`都识别成procedureName  但是drop时，未带反引号，故删除的procedureName识别为name，故不存在。  解决：正确使用语法，建议创建存储过程时，如果带反引号，procedureName前不要限定schema | WARP-46415 | 100% |
| 【KunDB-GXB】外键约束不适用于sharded table | 原因：sharded table的内容分散在多个shard上，两张表的主键和外键很可能不在一个shard上，会出现找不到  主键内容的情况。  解决：在gate层进行判断，如果是sharded table，则不允许创建外键，也不允许添加外键。 | WARP-46551 | 90% |
| update不存在的行不会报错 | 原因：mysql和mariadb也不会报错 | WARP-46347 | 90% |