# HARD\_FORK\_05更新修改

# 1. 功能调整

## 1.1 新增资产市场交易功能

新增市场交易功能，用于链上不同的资产之间的交易。由资产买卖双方发起限价订单，然后由底层撮合相互匹配的订单，撮合成功则立即执行，撮合失败则可以暂存在链上，以待其他的限价订单来撮合。资产交易需要扣除接收资产一部分作为市场交易手续费，该续费一部分返还给资产的手续费池，另一部分由交易账户的注册人和引荐人分红。具体划分比例由理事会参数决定。

## 1.2 积分产生方式调整

积分产生公式不变，其中的有效余额调整，具体如下：

原有效余额 =（账户余额 + 借入用于产生积分的额度 – 借出用于产生积分的额度 – 用于见证人抵押的额度）

**effective\_balance = core\_balance + core\_leased\_in - core\_leased\_out - total\_witness\_pledge**

调整为 账户需要锁定一部分YOYO资产替代有效余额。

新有效余额 = 锁定的用于产生积分的额度

**effective\_balance = locked\_balance\_for\_feepoint**

调整之后，账户的积分累积只与锁定的YOYO以及时间有关，不再是账户所有的有效余额参与产生积分。

## 1.3 新增抵押挖矿功能

## 1.4 新增注册商、引荐人点赞分红功能

## 1.5 注册商、引荐人修改

# 2. 理事会参数调整

1. uint64\_t    min\_mining\_pledge                   = GRAPHENE\_DEFAULT\_MIN\_MINING\_PLEDGE;
2. uint32\_t    mining\_pledge\_release\_delay         = GRAPHENE\_DEFAULT\_MINING\_PLEDGE\_RELEASE\_DELAY;
3. uint32\_t    max\_pledge\_mining\_bonus\_rate        = GRAPHENE\_DEFAULT\_MAX\_PLEDGE\_MINING\_BONUS\_RATE;
4. uint32\_t    registrar\_referrer\_rate\_from\_score  = GRAPHENE\_DEFAULT\_REGISTRAR\_REFERRER\_RATE\_FROM\_SCORE;
5. uint32\_t unlocked\_balance\_release\_delay = GRAPHENE\_DEFAULT\_UNLOCKED\_BALANCE\_RELEASE\_DELAY;

## 2.1 新增参数unlocked\_balance\_release\_delay

该参数为账户调整锁定产生积分的额度的时候，待释放的额度会在经过unlocked\_balance\_release\_delay块后，将待释放额度返还给账户。该参数默认值为uint32\_t(28800\*7)，即在七天后返回待释放额度。

## 2.2 新增参数min\_mining\_pledge

## 2.3 新增参数mining\_pledge\_release\_delay

## 2.4 新增参数max\_pledge\_mining\_bonus\_rate

## 2.5 新增参数registrar\_referrer\_rate\_from\_score

# 3. 插件改动整理

## 3.1 新增插件market\_history\_plugin

# 4. 数据结构整理

## 4.1 新增limit\_order\_object

1. **class** limit\_order\_object : **public** abstract\_object<limit\_order\_object>
2. {
3. **public**:
4. **static** **const** uint8\_t space\_id = protocol\_ids;
5. **static** **const** uint8\_t type\_id  = limit\_order\_object\_type;
7. time\_point\_sec   expiration;   //订单过期时间
8. account\_uid\_type seller;       //订单出售者
9. share\_type       for\_sale;     //出售资产的数量
10. price            sell\_price;   //出售资产和接收资产的比价
11. };

更新说明：新增限价订单结构，用于存储限价订单，在资产交易市场中，匹配买卖双方，撮合资产交易订单完成。

## 4.2 新增limit\_order\_create\_operation

1. **struct** limit\_order\_create\_operation : **public** base\_operation
2. {
3. **struct** fee\_parameters\_type
4. {
5. uint64\_t fee = GRAPHENE\_BLOCKCHAIN\_PRECISION / 10;
6. uint32\_t price\_per\_kbyte = 0 \* GRAPHENE\_BLOCKCHAIN\_PRECISION;
7. uint64\_t min\_real\_fee = 0;
8. uint16\_t min\_rf\_percent = 0;
9. extensions\_type   extensions;
10. };
12. fee\_type         fee;                                     //手续费
13. account\_uid\_type seller;                                  //限价订单出售者
14. asset            amount\_to\_sell;                          //出售的资产
15. asset            min\_to\_receive;                          //最低接收的资产
17. time\_point\_sec expiration = time\_point\_sec::maximum();    //订单过期时间
19. **bool** fill\_or\_kill = **false**;                                //true：若没匹配上则取消；false：没匹配上暂存在链上
20. extensions\_type   extensions;                             //扩展项
22. };

更新说明：新增创建限价订单操作。资产市场交易由买卖双方发起限价订单，然后由底层撮合匹配的限价订单。其中，fill\_or\_kill为true表示撮合成功则执行否则取消，为false表示撮合不成功则暂存在链上，在过期时间内等待其他限价订单撮合。

## 4.3 新增limit\_order\_cancel\_operation

1. **struct** limit\_order\_cancel\_operation : **public** base\_operation
2. {
3. **struct** fee\_parameters\_type
4. {
5. uint64\_t fee = GRAPHENE\_BLOCKCHAIN\_PRECISION / 100;
6. uint32\_t price\_per\_kbyte = 0 \* GRAPHENE\_BLOCKCHAIN\_PRECISION;
7. uint64\_t min\_real\_fee = 0;
8. uint16\_t min\_rf\_percent = 0;
9. extensions\_type   extensions;
10. };
12. fee\_type             fee;                //手续费
13. limit\_order\_id\_type  order;              //要取消的订单id
14. account\_uid\_type     fee\_paying\_account; //订单的发起账户
15. extensions\_type      extensions;         //扩展项
16. };

更新说明：新增取消限价订单操作。由限价订单发起者，发起取消操作，用于取消在等待中未完成的限价订单。

## 4.4 新增market\_fee\_collect\_operation

1. **struct** market\_fee\_collect\_operation : **public** base\_operation
2. {
3. **struct** fee\_parameters\_type
4. {
5. uint64\_t fee = 1 \* GRAPHENE\_BLOCKCHAIN\_PRECISION;
6. uint32\_t price\_per\_kbyte = 0 \* GRAPHENE\_BLOCKCHAIN\_PRECISION;
7. uint64\_t min\_real\_fee = 0;
8. uint16\_t min\_rf\_percent = 0;
9. extensions\_type   extensions;
10. };
12. fee\_type          fee;           //手续费
13. account\_uid\_type  account;       //发起账户
14. asset\_aid\_type    asset\_aid;     //资产的aid
15. share\_type        amount;        //要取回的资产数量
16. extensions\_type   extensions;    //扩展项
17. };

更新说明：新增收取资产市场交易手续费的收益。资产交易者账户的注册者和引荐人可以从该账户市场交易扣除的手续费中获取部分收益。该操作用于取回市场交易手续费产生的收益。

## 4.5 新增pledge\_mining\_object

1. **class** pledge\_mining\_object : **public** graphene::db::abstract\_object<pledge\_mining\_object>
2. {
3. **public**:
4. **static** **const** uint8\_t space\_id = implementation\_ids;
5. **static** **const** uint8\_t type\_id = impl\_pledge\_mining\_object\_type;
7. account\_uid\_type     pledge\_account;                                       //抵押挖矿账户
8. account\_uid\_type     witness;                                              //抵押挖矿见证人
9. share\_type           pledge;                                               //抵押数量
11. uint32\_t             last\_bonus\_block\_num = 0;                             //上一次抵押分红处理块号
12. share\_type           releasing\_mining\_pledge;                              //待释放的抵押
13. uint32\_t             mining\_pledge\_release\_block\_number = -1;              //抵押释放块号
14. };

更新说明：新增抵押挖矿结构， 普通账户抵押给见证人，从见证人出块工资中分红。

## 4.6 新增pledge\_mining\_update\_operation

1. **struct** pledge\_mining\_update\_operation : **public** base\_operation
2. {
3. **struct** fee\_parameters\_type
4. {
5. uint64\_t fee = 1 \* GRAPHENE\_BLOCKCHAIN\_PRECISION;
6. uint64\_t min\_real\_fee = 0;
7. uint16\_t min\_rf\_percent = 0;
8. extensions\_type   extensions;
9. };
11. fee\_type            fee;               //手续费
12. account\_uid\_type    pledge\_account;    //抵押挖矿账户
13. account\_uid\_type    witness;           //抵押挖矿见证人
14. share\_type          new\_pledge;        //更新抵押量
16. extensions\_type     extensions;        //扩展项
17. };

更新说明：新增挖矿抵押更新操作，包含创建抵押挖矿，更新抵押挖矿，取消抵押挖矿 ；

限制条件：

1. 见证人开放抵押挖矿权限
2. 创建，更新抵押挖矿时抵押量必须不小于1000YOYO（理事会参数，可调整）
3. 见证人不能抵押给自己

## 4.7 新增pledge\_bonus\_collect\_operation

1. **struct** pledge\_bonus\_collect\_operation : **public** base\_operation
2. {
3. **struct** fee\_parameters\_type
4. {
5. uint64\_t fee = 1 \* GRAPHENE\_BLOCKCHAIN\_PRECISION;
6. uint64\_t min\_real\_fee = 0;
7. uint16\_t min\_rf\_percent = 0;
8. extensions\_type   extensions;
9. };
11. fee\_type          fee;           //手续费
12. account\_uid\_type  account;       //抵押分红领取账户
13. share\_type        bonus;         //领取数量
14. extensions\_type   extensions;    //扩展项
15. };

更新说明：新增领取抵押分红操作，抵押挖矿分红并不是立即分发给账户，存储在每个账户的分红池中，可通过该操作领取到账户余额中。

## 4.8 新增score\_bonus\_collect\_operation

1. **struct** score\_bonus\_collect\_operation : **public** base\_operation
2. {
3. **struct** fee\_parameters\_type
4. {
5. uint64\_t fee = 1 \* GRAPHENE\_BLOCKCHAIN\_PRECISION;
6. uint64\_t min\_real\_fee = 0;
7. uint16\_t min\_rf\_percent = 0;
8. extensions\_type   extensions;
9. };
11. fee\_type          fee;           //手续费
12. account\_uid\_type  account;       //抵押分红领取账户
13. share\_type        bonus;         //领取数量
14. extensions\_type   extensions;    //扩展项
15. };

更新说明：新增领取点赞分红操作，点赞获取收益的部分（比例由理事会参数确定） 按比例分配给引荐人与注册商，发放到各自的点赞分红池中，可通过该操作领取到账户余额中。

## 4.9 新增balance\_lock\_update\_operation

1. **struct** balance\_lock\_update\_operation : **public** base\_operation
2. {
3. **struct** fee\_parameters\_type
4. {
5. uint64\_t fee = GRAPHENE\_BLOCKCHAIN\_PRECISION / 10;
6. uint64\_t min\_real\_fee = 0;
7. uint16\_t min\_rf\_percent = 0;
8. extensions\_type   extensions;
9. };
11. fee\_type          fee;              //交易的手续费
12. account\_uid\_type  account;          //发起账户的uid
13. share\_type        new\_lock\_balance; //新的用于产生积分锁定额度
14. extensions\_type   extensions;       //扩展项
15. };

更新说明：新增balance\_lock\_update\_operation用于抵押YOYO产生积分。其中new\_lock\_balance为新的锁定额度。新的额度每次执行都是及时生效，新的额度与就额度的差值记录在accout\_statistics\_object的releasing\_locked\_feepoint中作为待释放额度。在经过理事会参数中unlocked\_balance\_release\_delay个块后，待释放额度返回账户。

## 4.10 修改account\_statistics\_object

1. **class** account\_statistics\_object : **public** graphene::db::abstract\_object<account\_statistics\_object>
2. {
3. **public**:
4. **static** **const** uint8\_t space\_id = implementation\_ids;
5. **static** **const** uint8\_t type\_id  = impl\_account\_statistics\_object\_type;
7. account\_uid\_type  owner;
9. account\_transaction\_history\_id\_type most\_recent\_op;
10. uint32\_t                            total\_ops = 0;
11. uint32\_t                            removed\_ops = 0;
13. share\_type prepaid;
14. share\_type csaf;
15. share\_type core\_balance;
16. share\_type core\_leased\_in;
17. share\_type core\_leased\_out;
19. share\_type                     average\_coins;
20. fc::time\_point\_sec             average\_coins\_last\_update;
21. fc::uint128\_t                  coin\_seconds\_earned;
22. fc::time\_point\_sec             coin\_seconds\_earned\_last\_update;
24. share\_type total\_witness\_pledge;
25. share\_type releasing\_witness\_pledge;
26. uint32\_t   witness\_pledge\_release\_block\_number = -1;
28. share\_type locked\_balance\_for\_feepoint;       //锁定的额度，用于产生积分
29. share\_type releasing\_locked\_feepoint;         //待释放的额度
30. uint32\_t   feepoint\_unlock\_block\_number = -1; //下一次返还待释放额度的块号
32. uint32\_t   last\_witness\_sequence = 0;
33. share\_type uncollected\_witness\_pay;
35. uint64\_t witness\_last\_confirmed\_block\_num = 0;
36. uint64\_t witness\_last\_aslot = 0;
37. uint64\_t witness\_total\_produced = 0;
38. uint64\_t witness\_total\_missed = 0;
39. uint64\_t witness\_last\_reported\_block\_num = 0;
40. uint64\_t witness\_total\_reported = 0;
42. share\_type total\_committee\_member\_pledge;
43. share\_type releasing\_committee\_member\_pledge;
44. uint32\_t   committee\_member\_pledge\_release\_block\_number = -1;
45. uint32\_t   last\_committee\_member\_sequence = 0;
47. **bool**     can\_vote = **true**;
48. **bool**     is\_voter = **false**;
49. uint32\_t last\_voter\_sequence = 0;
50. uint32\_t last\_platform\_sequence = 0;
52. share\_type    total\_platform\_pledge;
53. share\_type    releasing\_platform\_pledge;
54. uint32\_t      platform\_pledge\_release\_block\_number = -1;
55. post\_pid\_type last\_post\_sequence = 0;
57. custom\_vote\_vid\_type last\_custom\_vote\_sequence = 0;
58. advertising\_aid\_type last\_advertising\_sequence = 0;
59. license\_lid\_type     last\_license\_sequence = 0;
60. };

更新说明：新增了locked\_balance\_for\_feepoint锁定的额度，用于产生积分，releasing\_locked\_feepoint待释放的额度，feepoint\_unlock\_block\_number下一次返还待释放额度的块号

# 5. Database\_api 修改整理

# 6. Wallet修改整理

## 6.1 新增create\_limit\_order

1. signed\_transaction create\_limit\_order(string           seller,                   //售出资产者
2. asset\_aid\_type   sell\_asset\_symbol,        //售出资产符号
3. share\_type       sell\_amount,              //售出数量
4. asset\_aid\_type   min\_receive\_asset\_symbol, //接收资产符号
5. share\_type       min\_receive\_amount,       //最少接收数量
6. uint32\_t         expiration,               //订单过期时间
7. **bool**             fill\_or\_kill,             //true：若没匹配上则取消；false：没匹配上暂存在链上
8. **bool**             broadcast = **false**);       //是否广播

更新说明：新增创建限价订单，用于资产交易

示例：*create\_limit\_order test YOYO 10000 ABC 50000 1579862861 false true*

## 6.2 新增cancel\_limit\_order

1. signed\_transaction cancel\_limit\_order(string               seller,             //限价订单发起者
2. limit\_order\_id\_type  order\_id,           //限价订单的id
3. **bool**                 broadcast = **false**); //是否广播

更新说明：新增取消限价订单，用于取消待处理的限价订单

示例：*cancel\_limit\_order test 1.12.0 true*

## 6.3 新增collect\_market\_fee

1. signed\_transaction collect\_market\_fee(string               account,           //取回市场费用的账户
2. string               asset\_symbol,      //取回资产的符号
3. string               amount,            //取回市场费用的数量
4. **bool**                 csaf\_fee = **true**,   //使用积分付手续费。true，积分付手续费；false，余额付手续费
5. **bool**                 broadcast = **false**  //是否广播
6. );

更新说明：新增取回市场交易手续费产生的收益

示例：*collect\_market\_fee test YOYO 10000 false true*

## 6.4 新增asset\_claim\_fees

1. signed\_transaction asset\_claim\_fees(string               issuer,            //发起人
2. string               asset\_symbol,      //资产的符号
3. string               amount\_to\_claim,   //收回的资产交易市场费用
4. **bool**                 csaf\_fee = **true**,   //使用积分付手续费。true，积分付手续费；false，余额付手续费
5. **bool**                 broadcast = **false**  //是否广播
6. );

更新说明：新增资产创建人取回该资产市场交易产生的手续费

示例：*asset\_claim\_fees test YOYO 10000 false true*

## 6.5 新增update\_lock\_balance

1. signed\_transaction update\_lock\_balance(string lock\_balance\_account, //抵押发起的账户id或名字
2. string lock\_balance\_amount,  //抵押的额度
3. **bool** csaf\_fee = **true**,        //是否使用积分支付手续费
4. **bool** broadcast = **false**);     //是否广播

更新说明：新增创建或更新抵押产出积分

示例： *update\_lock\_balance test\_account1 1000 true true*