银企互联提单处理手册

**开发：吴高盛**

**部门：EAS往来与收付系统管理部**

**日期：2016-2-17**

**1、问题描述：**

银行账户余额查询-过滤银行账户时，没有银行账户可以选择

**提单号：**R20160225-0339

**提单类型：**应用类型

**解决方案：**

select \* from t\_be\_accountbankbalance a,t\_bd\_accountbanks b,t\_bd\_currency c

where a.fcompanyid in (select fid from t\_org\_baseunit where fname\_l2 = ‘公司名称’) and a.faccountbankid = b.fid and a.fcurrencyid = c.fid and b.fisClosed = 0 and b.fisSetBankInterface = 1

如上面语句没查出数据，查如下语句

select \* from t\_be\_accountbankbalance a where a.fcompanyid in (select fid from t\_org\_baseunit where fname\_l2 = ‘公司名称’)

发现该公司没有交易记录，没有余额信息，因此在该过滤界面中选不到 银行账户 的信息

**2、问题描述：**银企互联-付款时提示上送账户名称和合约登记不符

**提单号：**R20160228-0017

**提单类型：**应用类型

**解决方案：**

在银行账户中修改银企账户名称信息

**3、问题描述：**银行付款单打回时提示该银行付款单已存在

**提单类型：**已出补丁解决

**解决方案：**

打如下补丁 出纳补丁：PT099400、资金补丁：PT099318

**4、问题描述：**行名行号在线导入的时候报java.lang.NullPointerException

**提单类型：**已出补丁解决

**解决方案：**

打资金补丁：PT097037

**5、问题描述：**提交一批银行付款单时，有一笔失败，其他好多笔都提交失败,具体错误信息如下：

com.kingdee.eas.fm.be.BEException: 银企未匹配收款地区:南昌，请填写收款地区的行政规划标准名称再尝试。如仍出现该报错，请联系金蝶银企平台研发部门

at com.kingdee.eas.fm.be.app.service.BankPayUtil.pay(BankPayUtil.java:110)

at com.kingdee.eas.fm.be.app.bankpay.BankPayingBillPayUtil.batchBankPay(BankPayingBillPayUtil.java:487)

at com.kingdee.eas.fm.be.app.bankpay.BankPayingBillPayUtil.doBatchPay(BankPayingBillPayUtil.java:576)

at com.kingdee.eas.fm.be.app.bankpay.BankPayingBillPayUtil.doBatchMergePay(BankPayingBillPayUtil.java:616)

at com.kingdee.eas.fm.be.app.bankpay.BankPayingBillPayUtil.doBatchPayByType(BankPayingBillPayUtil.java:656)

at com.kingdee.eas.fm.be.app.bankpay.BankPayingBillPayUtil.batchBankPay(BankPayingBillPayUtil.java:735)

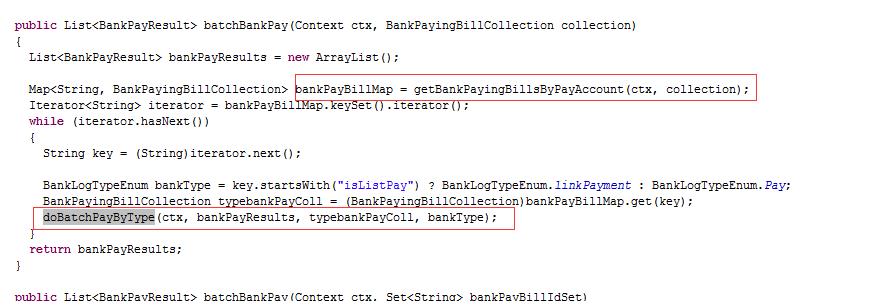
at com.kingdee.eas.fm.be.app.bankpay.BankPayingBillPayUtil.batchBankPay(BankPayingBillPayUtil.java:761)

at com.kingdee.eas.fm.be.app.BankFacadeControllerBean.\_batchSubmit(BankFacadeControllerBean.java:3408)

**提单类型：**应用类

**解决方案：**

是这样的，我们选中一批提交付款时，先会按 付款账户 进行分组，一组一组(一组可能包含好几笔) 提交，这次你提交的一批中，其中有一笔的收款地区填错了，这个是在 银企平台 校验的，因此银企平台在这组提交中未给消息体response中body添加任何信息，而将异常写入response中的exception中，因此我们判断这组都失败，因为根据银企平台返回的信息，我们无法判断到底是哪笔出问题了





**6、问题描述：**行名行号在线导入的时候报java.lang.NullPointerException

**提单类型：**已出补丁解决

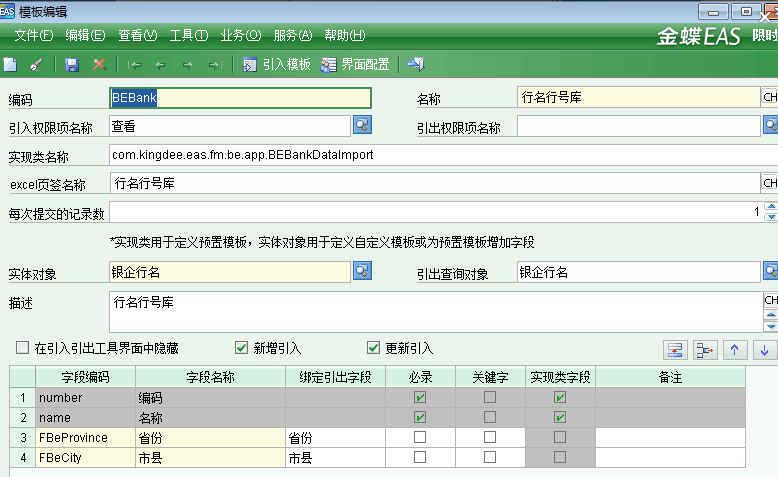
**解决方案：**

打资金补丁：PT097037

**7、问题描述：**银企互联 导出 行名行号 导出的模板是缺列的

**提单类型：**20160318-0039

**解决方案：**



**8、问题描述：**银企互联 银行账户余额查询出现重复

**提单类型：**R20160328-0126

**解决方案：**

我们联机查询余额的时候，会从银企平台下载数据，放到内存中；程序会已相同条件查询本地数据库里面的数据，如果有数据的话，会将其删掉，在将下载的数据保存到数据库中，这里面的删除操作，保存操作的事务属性都是依赖于获取余额这个方法的事务，我们在开始获取余额的时候会开一个新事务，删除和保存都是用的这同一个事务，当两个客户端在同时联机查询的时候，当查询到的余额量多的时候，可能时间会有些长，两个客户端的操作在数据库几乎同时提交，可能就会造成两条重复的数据，这种可能性的概率一般是很滴的！

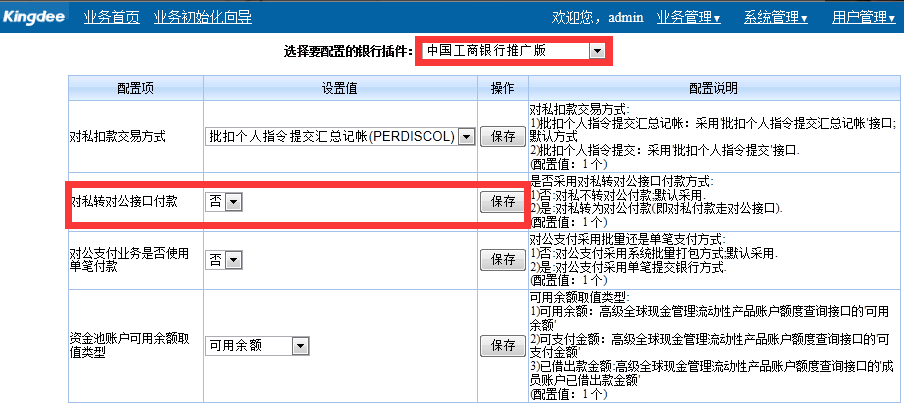
可以在不和余额的后台事务设置的那个时间点一起联机查询余额 或者 出现重复后 在一次联机查询即可，或者删掉重复的！

如此问题仍有疑问，请及时与我联系QQ：2272372029 ，请不要随意退单，谢谢合作！

**9、问题描述：**付款单对私付款报错，银行返回信息为，企业付款帐号未注册企业财务室

**提单类型：**R20160329-0213

**解决方案：**需要在银企平台设置一下，将对私付款的接口该成对私付款走对公通道就可以了，具体的请咨询蔡瑾钊



将否该成是

**10、问题描述：**上划单拉交易明细的时候，提示有些字段为null

**提单类型：** R20160325-0283

**解决方案：**

如果是上划单或者是下拨单拉银行交易明细，会检查 这笔交易明细的对方银行账户，对方单位，交易时间 ，贷款金额，付款金额，id是否为null，而不是为0，如果为null的话，就会报上面的提示；这是检查的第一步，第二部是根据对方银行账户的编码，获取对方银行账户这个对象，有且仅有一个，如果为0或者大于1都会提示相应的错误提示。第三步检查对方银行账户的开户单位，如果为null，也提示有问题，第4步检查对方银行账户的母帐户（subAccount）是否为null ，如果为null的话也会有问题。

**11、问题描述：**招商查询余额或者前置机时，银行前置机会报错

**提单类型：** R20160330-0046

**解决方案：**

1. 从管理的角度进行控制
2. 客户想实现 如下需求

第一个用户发起查询业务后，第二此查询系统控制在60s后允许发起第二次查询余额或者交易操作，目前不支持

**12、问题描述：**上划单拉交易明细时中断，具体日志信息如下：

Caused by: java.lang.IllegalArgumentException: The argument 'id' is invalid. 'null' length:4

at com.kingdee.bos.util.BOSUuid.read(BOSUuid.java:178)

at com.kingdee.bos.dao.ormapping.impl.ImplUtils.convToBOSUuid(ImplUtils.java:120)

at com.kingdee.bos.dao.ormapping.impl.RTFilterParameter.<init>(RTFilterParameter.java:42)

at com.kingdee.bos.dao.ormapping.impl.EntityAccess.innerAddFilter(EntityAccess.java:823)

at com.kingdee.bos.dao.ormapping.impl.EntityAccess.processAddFilter(EntityAccess.java:763)

at com.kingdee.bos.dao.ormapping.impl.EntityAccess.addFilter(EntityAccess.java:695)

at com.kingdee.bos.dao.ormapping.impl.EntityAccess.addFilter(EntityAccess.java:680)

at com.kingdee.bos.dao.ormapping.impl.EntityAccess.initialize(EntityAccess.java:144)

at com.kingdee.bos.dao.ormapping.impl.EntityAccess.getInstance(EntityAccess.java:275)

at com.kingdee.bos.dao.ormapping.impl.ObjectHandler.getEntityAccess(ObjectHandler.java:48)

at com.kingdee.bos.dao.ormapping.impl.ObjectReader.select(ObjectReader.java:139)

at com.kingdee.bos.dao.ormapping.ORMappingDAO.innerGetCollection(ORMappingDAO.java:785)

at com.kingdee.bos.dao.ormapping.ORMappingDAO.getCollection(ORMappingDAO.java:683)

at com.kingdee.bos.framework.ejb.AbstractEntityControllerBean.innerGetCollection(AbstractEntityControllerBean.java:580)

at com.kingdee.bos.framework.ejb.AbstractEntityControllerBean.innerGetCollection(AbstractEntityControllerBean.java:596)

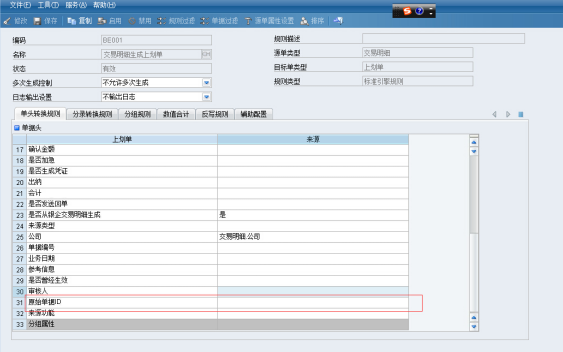
at com.kingdee.eas.framework.app.AbstractCoreBaseControllerBean.\_getCollection(AbstractCoreBaseControllerBean.java:362)

at com.kingdee.eas.framework.app.AbstractObjectBaseControllerBean.\_getCollection(AbstractObjectBaseControllerBean.java:187)

**提单类型：** R20160325-0283

**解决方案：**

像这种问题，大多数是传参数传错了，经分析发现是botp配置错误，客户是在 交易明细转换为上划单 这个botp规则里面，在单据头转换规则里，原始单据ID 对应了交易明细ID，而标准产品的8.1这里是空的，在分录转换规则里，上游单据id客户里填的是交易明细原始id,而标准产品是 公式结果=交易明细ID;按照8.1标准的botp规则配置来即可。



8.1标准界面 1



8.2标准界面2

**11、问题描述：**银行付款单交易失败,银行返回信息：未能获取账户开户行的联行号

**提单类型：** R201604070-0269

**解决方案：**

701版本中，在银企互联设置里面有个行名行号标签，里面的编码指的是联行号，银行返回的是 收款银行 的联行号，该联行号可以在行名行号里面设置 对应的是编码这个值，要么 收款银行的名称完全正确，要么 联行号正确，这样的话收款信息才会匹配正确，维护好后，在付款单的收款银行中在选到对应的收款银行信息即可。

**11、问题描述：** 新增开户地区报错



**提单类型：** R20160401-0254

**解决方案：**

Eas中银企互联是最新版本，银企平台是eas600,该异常是提供服务的那一端没有资源，eas705是要和银企平台705对应的

**12、问题描述：** 二次开发开了了一个类似代发单的单据提交银企付款，现在银行交易明细的 摘要 字段取字源单据的 用途 字段，现想从 分录的 备注 字段取值，有没有比较好的方法

**问题分析：**经与需求和银企平台的同事谢金林确认知，银行账户交易明细的摘要字段description对应的是银企平台交易字段的摘要哦字段explation, 付款时detail.setDesc字段对应的是摘要字段expplation,而付款时的用途字段detail.setUse(取固定值)

**解决方法：**在beServiceHelper的如下方法中

com.kingdee.eas.fm.be.app.service.beServiceHelper.parseAgencySalaryDetail(Context, BankAgentPayBillInfo)

用

detail.setDesc(entryInfo.getRemark()) 取代

detail.setDesc(info.getUsage()) 方法

13、**问题描述：**在出纳提交银企的时候报异常，异常信息如下：

at com.kingdee.bos.rpc.impl.ThreadPool2$Worker.run(ThreadPool2.java:140)

Caused by: java.lang.NoSuchMethodError: com/kingdee/eas/fm/be/BankPayingBillInfo.setExpectTransTime(Ljava/sql/Timestamp;)V

at com.kingdee.eas.fi.cas.PaymentBill2BankPayBillBuilder.getBankPayBillCollection(PaymentBill2BankPayBillBuilder.java:119)

at com.kingdee.eas.fi.cas.PaymentBill2BankPayBillBuilder.getBankPayBillInfo(PaymentBill2BankPayBillBuilder.java:61)

at com.kingdee.eas.fm.be.app.BankPayingBillControllerBean.\_commit2BankPayBill(BankPayingBillControllerBean.java:240)

at com.kingdee.eas.fm.be.app.AbstractBankPayingBillControllerBean.commit2BankPayBill(AbstractBankPayingBillControllerBean.java:297)

at sun.reflect.GeneratedMethodAccessor707.invoke(Unknown Source)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:55)

at java.lang.reflect.Method.invoke(Method.java:619)

**问题分析：**在异常是提交银企生成银行付款单的时候报错，原因是没有找到BankPayingBillInfo 对象中 setExpectTransTime(Ljava/sql/Timestamp;)中 参数为Timestamp的方法，因为资金补丁最新的话，银行付款单中 setExpectTransTime的参数是Date类型，而非Timestamp类型（中间做过补丁修改，将timestamp修改为了Date类型）

**解决方法：**把出纳和资金的补丁打到最新

14、**问题描述：**在出纳提交银企的时候报异常，异常信息如下：

Caused by: java.lang.NoSuchMethodError: com/kingdee/eas/fm/be/BankPayingBillInfo.setExpectTransTime(Ljava/util/Date;)V

at com.kingdee.eas.fi.cas.PaymentBill2BankPayBillBuilder.getBankPayBillCollection(PaymentBill2BankPayBillBuilder.java:119)

at com.kingdee.eas.fi.cas.PaymentBill2BankPayBillBuilder.getBankPayBillInfo(PaymentBill2BankPayBillBuilder.java:61)

at com.kingdee.eas.fm.be.app.BankPayingBillControllerBean.\_commit2BankPayBill(BankPayingBillControllerBean.java:240)

at com.kingdee.eas.fm.be.app.AbstractBankPayingBillControllerBean.commit2BankPayBill(AbstractBankPayingBillControllerBean.java:297)

**问题分析：**在异常是提交银企生成银行付款单的时候报错，原因是没有找到BankPayingBillInfo 对象中 setExpectTransTime(Ljava/util/Date)中 参数为Date的方法，因为出纳补丁最新的话，银行付款单中 setExpectTransTime的参数是Timestamp类型，而非Date类型（中间做过补丁修改，将timestamp修改为了Date类型）

**解决方法：**把出纳和资金的补丁打到最新

15、**问题描述：**银企互联--行名行号节点，进行在线导入银行行号等相关信息，但是在导入时提示“请选择银行”，但是银行信息为空白，无法选择

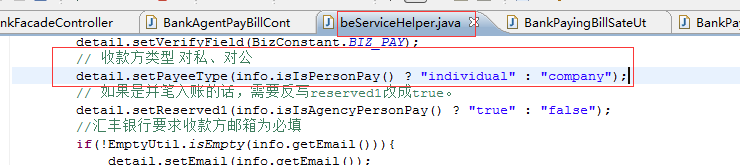
**问题分析：**当我们点击在线导入的银行时，后台会去银企平台同步BankCode，服务端方法为BankFacadeControllerBean中的\_synBeBank方法

**解决方法：**去银企平台维护相关银行信息，在银企平台的web版界面中也有行名行号，如果银企平台的相关银行信息维护好后是可以在那个页面中(银企平台web行名行号)中选择到对应的银行的

16、**问题描述：**怎么知道付款是对公支付还是对私支付？

**问题分析：**业务单据像付款单或者结算单或者代发单有个结算方式，比如结算方式为对私，则传到银行付款单这边

**解决方法：**



17、**问题描述：**修改参数BE001后，银行付款单上的审核和反审核按钮仍是灰色的？

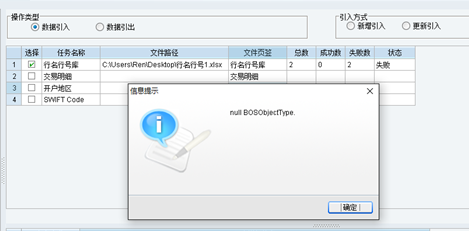
**问题分析：**TM018要设置为 银行付款单，而不能设置为 业务单据，BE001 设置需要审核！

**提单类型：** R20160422-0060

**解决方法：** 看银行付款单BankPayingBillList 的onload的方法即可知道。

18、**问题描述：**在银企行名行号中导入的时候出现错误，错误信息如下:

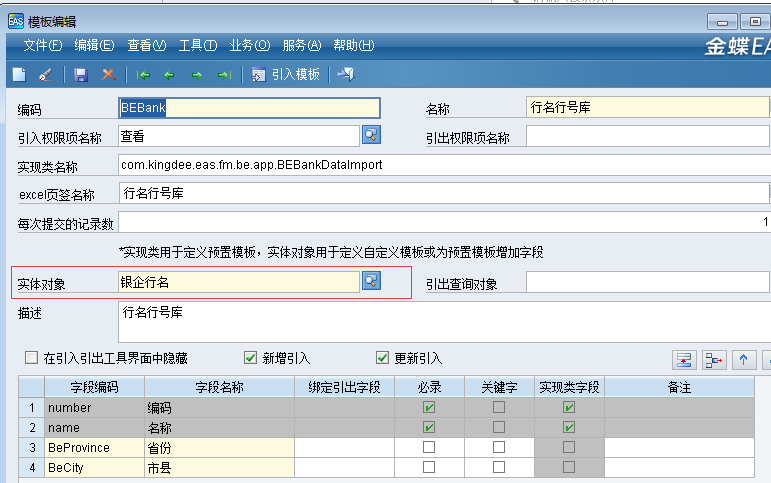
Null BOSObjectType 如下图



**问题分析：**在引入引出模板中选择行名行号库，点击修改，发现实体对象 中并没有选择

**提单类型：** R20160429-0063

**解决方法：** 如下图



19、**问题描述：**开户地区有什么作用？

**提单类型：**R20160510-0255

**解决方法：**

开户地区是指商业银行对其分支机构的地区编码；

通过设置各银行的开户地区编码和名称，可以满足同行付款时提交银行交易明细报文格式需要；

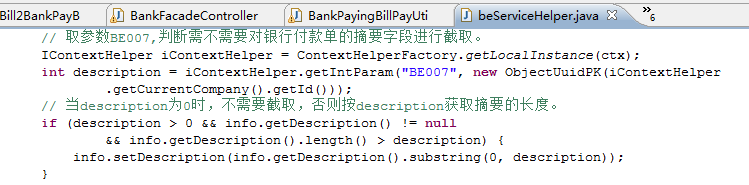
一般情况下，同行付款时对于收款银行，需要录入收款地区（即开户地区）或收款方省/市这两个基础资料中的一个，至于用户想录哪个字段，由银企互联下的参数BE009控制隐藏或者显示

20、**问题描述：**付款单摘要字段大于6个字之后，提交银企后，回单只有6个字

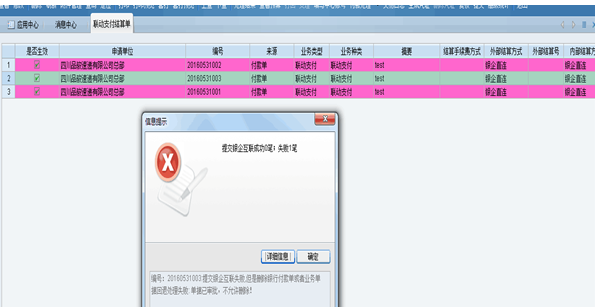
**提单类型：**R20160517-0285

**解决方法：**

这个地方是由一个参数控制的，为BE007,当BE007为0时，不进行控制，当BE007为6时，在提交银企拼报文的时候就会截断6个字，这个可以从报文中看出，代码如下：

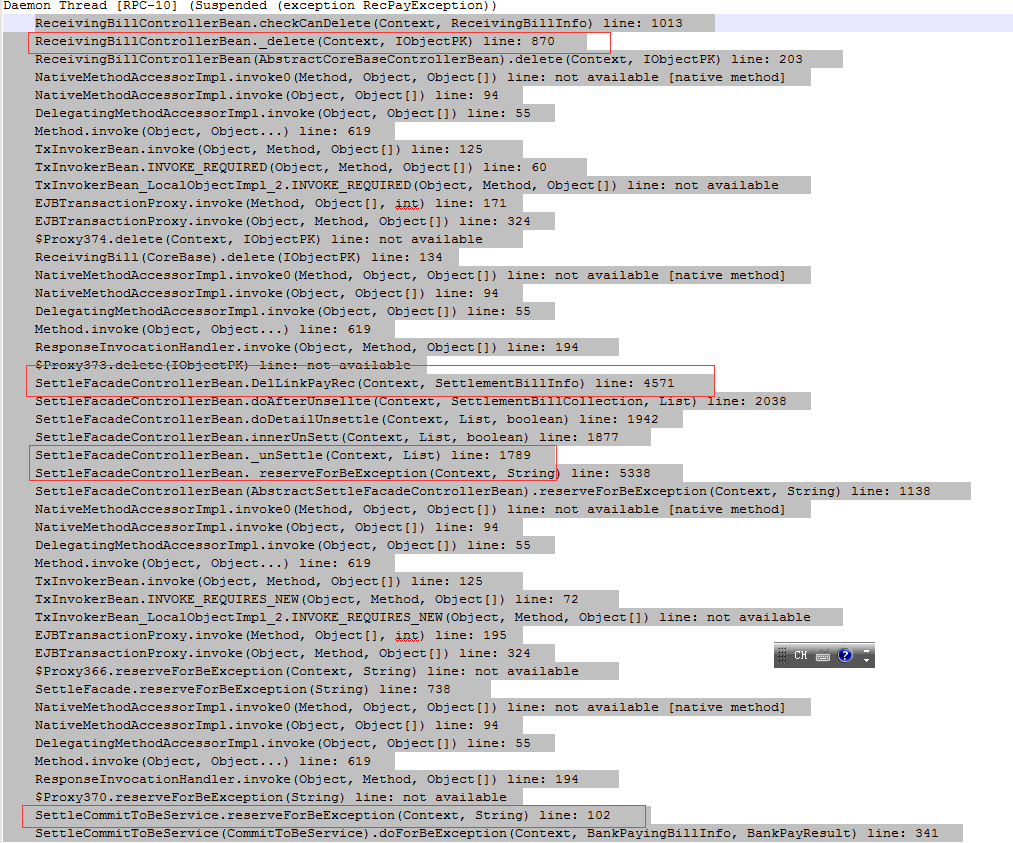


21、**问题描述：**结算单提交银企互联失败，但是删除银行付款单或者业务单据回退处理失败，单据已审批，不允许删除。



**提单类型：**R20160531-0376

**问题分析：**通过和现场连调发现，报错代码如下



和客户沟通发现，客户设置了结算单在受理生成收款单的时候，将状态直接设置成了已审批状态，导致在删除的时候不能删除，所以导致反写失败，整个事务回滚，导致银行付款单等操作回退。



注意：结算单botp转换规则中，单据状态这可以设置，之前客户是设置成已审批状态而非已提交状态，导致出现了反写失败的情况。

22、**问题描述：**交易明细上划与下拨的取数逻辑？

**提单号：**R20160618-0015

**解决方法：**

取数逻辑如下

if(!EmptyUtil.isEmpty(detail.getTransType()){

transDetailInfo.setIsTransDown(detail.getTransType().equalsIgnoreCase("autotransdown")?true:false);

transDetailInfo.setIsTransUp(detail.getTransType().equalsIgnoreCase("autotransUp")?true:false);

}

23、**问题描述：**生成银行付款单或者在银行付款单打回这些如何测试?

**提单号：**R20160701-0289

**解决方法：**

打回的测试方法：

1、在测试环境里面将银企互联服务设置的ip和端口不填，之后保存

2、将参数TM018(这个参数是在资金结算下)单据类型改为银行付款单，而不是业务单据，之后做一笔付款单提交银企，此时下查该付款单，应该可以生成银行付款单，并且银行付款单的单据状态为准备提交，之后打回，看是否能成功打回。

3、在测试环境将付款单删掉。

24、**问题描述：**提交银行付款单提示单据已锁定？

**提单号：**R20160628-0132

**解决方法：**

在 企业建模-安全管理-网络互斥 里面将银行账户或者是银行付款单的锁都清掉

25、**问题描述：**客户在银企互联点击测试，提示连接不通，但是在eas里面和银企平台web版上都能查询余额

**提单号：**无

**解决方法：**

是否更新了银企平台的补丁，如果没有，请更新银企平台最新补丁

PT104628

R20160705-0288

26、**问题描述：**eas付款单上是收款方省，收款方市是上海市，到前置机的报文则变成了福州省，福州市

**提单号：**R20160705-0288

**解决方法：**

27、**问题描述：**7.5版本提交银企和付款报错，报错信息如下

Caused by: java.lang.NoSuchMethodError: com/kingdee/eas/fi/cas/PaymentBillInfo.isIsRefundmentPay()Z

at com.kingdee.eas.fi.cas.app.service.PayPayService.doService(PayPayService.java:254)

at com.kingdee.eas.fi.cas.app.PaymentBillControllerBean.\_payForBook(PaymentBillControllerBean.java:523)

at com.kingdee.eas.fi.cas.app.PaymentBillControllerBean.\_payForBook(PaymentBillControllerBean.java:4410)

at com.kingdee.eas.fi.cas.app.AbstractPaymentBillControllerBean.payForBook(AbstractPaymentBillControllerBean.java:787)

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

**提单号：**R20160701-0308

**解决方法：**

现场分析知，AbstractPaymentBillInfo加载的是lib/industry的类，请联系房地产的同事.像这种PaymentBillInfo缺字段的，请分析其AbstractXXX的类。

**28、问题描述：**在收款单或者结算单联机下载交易明细时，发现交易时间的时间大于今天，这笔交易明细实际上是不存在的。

**提单号：**R20160705-0078

**解决方法：**

经现场分析，通过select \* from t\_be\_transdetail where ftranpackageid = ‘detailid’ 数据库中有两条，其中一条是正确的，一条是错误的。蒙牛已出现过3，4次了。

第一笔交易明细刚下载的时候是正常的，重复下载几次，这笔的最后一次下载时，交易时间转换或者计算错误，因此这笔交易明细和本地数据库中对应的那笔不是完全相同的，是相似的，因此会更新操作。因此这笔的数据就错误了，在进行一次下载时，此时时间并没有转换或者计算错，因此这笔下载下来的找不到本地对应的交易明细作为新增的一笔保存下来，所以下游单据在下载的时候就会看到两条。

**29、问题描述：**如何修改或者重设银企互联密码设置

**提单号：**无

**解决方法：**

Delete from t\_be\_submitpassword

**30、问题描述：**银行付款单失败，银行返回信息：企业签名时间晚于工行服务器时间，最后同步失败的银行返回原因：没有符合条件的记录

**提单号：**R20160711-0252

**解决方法：**

银企平台的时间不对，调整银企平台的时间。

**31、问题描述：**银行付款单失败，银行返回信息：企业签名时间晚于工行服务器时间，最后同步失败的银行返回原因：没有符合条件的记录

**提单号：**R20160711-0252

**解决方法：**

银企平台的时间不对，调整银企平台的时间。

**32、问题描述：**银行代发单的一条单据中如果分录超过100行，不能同步成功，但是客户端显示同步成功。

**提单号：**R20160714-0067

**解决方法：**

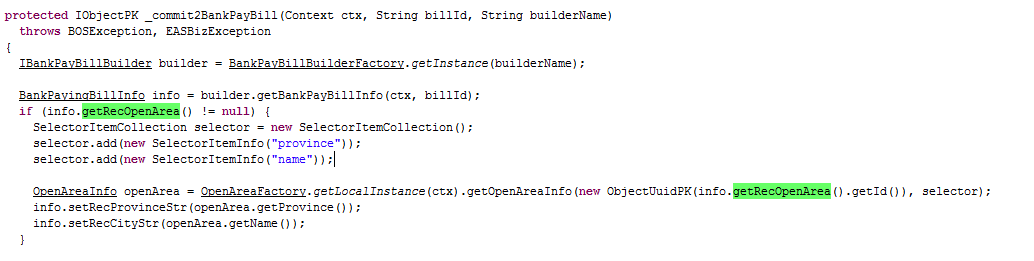
通过查看EAS703版本代码的SVN提交日志发现，该版本对于这个问题已经打了补丁，补丁号为：PT076243，处理办法：让客户将该补丁打上，并测试。

**33、问题描述：**提交银企互联后，报没有“收方省份”

**提单号：**R20160708-0261

**解决方法：**

该问题是eas600的问题，付款地区里面选择了收款地区（开户地区）但是收款地区里面没有维护省份字段导致。



**34、问题描述：**提交银企时提示付款金额不能大于未付金额

**提单号：**R20160727-0421

**解决方法：**该提示信息是费用报销部门的，这个与周艳姐沟通得知**。**

**35、问题描述：**银行付款单打回很慢，需要2,3分钟

**提单号：**R20160802-0074

**解决方法：**客户eas为703版本.经分析性能日志知，是在查询select \* from t\_cas\_journal where fbillid in (‘’)时间较长，并且在一个action中有执行几次，因此很慢，通过分析t\_cas\_journal表里面的数据知，客户目前有100多万（1000000）条日记账，并且fbillid中是没有索引的，通过分析该语句的执行计划走的是全表扫描，因此时间较长。因此通过在该字段里面加索引后，走索引范围扫描时间大大缩减，问题解决。

**查看一个表里面有哪些索引**

select \* from ksql\_indexes where table\_name = 'T\_CAS\_JOURNAL'

**通过索引名称查询该索引是建立在哪个表里面的哪个字段中**

select \* from user\_ind\_columns where index\_name IN ('IX\_CAS\_JOUR','IX\_CAS\_JOUR\_ACCTBK','IX\_CAS\_JL\_MORG','PK\_CAS\_JOURNAL')（oracle在查询分析器中需要通过方言的方式来执行）

**在该字段添加索引**

if not exists (select \* from ksql\_indexes where KSQL\_INDNAME='IX\_CAS\_JOUR\_BILLID') CREATE INDEX IX\_CAS\_JOUR\_BILLID ON T\_CAS\_JOURNAL(FBILLID)

**oracle索引的知识**

http://blog.csdn.net/warden2010/article/details/11757309

**36、问题描述：** 银企互联设置提示”获取配置文件信息出错，请设置配置文件格式是否正确”

**提单号：**R20160728-0304

**解决方法：**该客户的现场环境是eas701,该提示信息是银企互联业务抛出来的，真正的异常并未打印到日志中，即被吃掉了。现场的情况是银企互联下与银企平台所有交互的功能都不能用，查余额和交易明细时报错，报错信息如下：

**faultString: org.xml.sax.SAXException: Invalid element in com.kingdee.bos.ebgateway.common.BankAcnt - bankVersion**

**faultActor:**

**faultNode:**

**faultDetail:**

**{http://xml.apache.org/axis/}hostname:jtyhserver**

**org.xml.sax.SAXException: Invalid element in com.kingdee.bos.ebgateway.common.BankAcnt - bankVersion**

**at org.apache.axis.message.SOAPFaultBuilder.createFault(SOAPFaultBuilder.java:222)**

**at org.apache.axis.message.SOAPFaultBuilder.endElement(SOAPFaultBuilder.java:129)**

**at org.apache.axis.encoding.DeserializationContext.endElement(DeserializationContext.java:1087)**

**at org.apache.xerces.parsers.AbstractSAXParser.endElement(Unknown Source)**

**at org.apache.xerces.impl.XMLNSDocumentScannerImpl.scanEndElement(Unknown Source)**

**at** org.apache.xerces.impl.XMLDocumentFragmentScannerImpl$FragmentContentDispatcher.dispatch(Unknown Source)

at org.apache.xerces.impl.XMLDocumentFragmentScannerImpl.scanDocument(Unknown Source)

at org.apache.xerces.parsers.XML11Configuration.parse(Unknown Source)

at org.apache.xerces.parsers.XML11Configuration.parse(Unknown Source)

at org.apache.xerces.parsers.XMLParser.parse(Unknown Source)

at org.apache.xerces.parsers.AbstractSAXParser.parse(Unknown Source)

at org.apache.xerces.jaxp.SAXParserImpl.parse(Unknown Source)

at org.apache.axis.encoding.DeserializationContext.parse(DeserializationContext.java:227)

at org.apache.axis.SOAPPart.getAsSOAPEnvelope(SOAPPart.java:696)

at org.apache.axis.Message.getSOAPEnvelope(Message.java:435)

at org.apache.axis.handlers.soap.MustUnderstandChecker.invoke(MustUnderstandChecker.java:62)

at org.apache.axis.client.AxisClient.invoke(AxisClient.java:206)

at org.apache.axis.client.Call.invokeEngine(Call.java:2784)

at org.apache.axis.client.Call.invoke(Call.java:2767)

at org.apache.axis.client.Call.invoke(Call.java:2443)

at org.apache.axis.client.Call.invoke(Call.java:2366)

at org.apache.axis.client.Call.invoke(Call.java:1812)

at com.kingdee.eas.fm.be.app.BankServiceUtil.getBalance(BankServiceUtil.java:103)

at com.kingdee.eas.fm.be.app.BankFacadeControllerBean.\_getBalance(BankFacadeControllerBean.java:238)

at com.kingdee.eas.fm.be.app.BankFacadeControllerBean.\_getBalancenew(BankFacadeControllerBean.java:379)

at com.kingdee.eas.fm.be.app.AbstractBankFacadeControllerBean.getBalancenew(AbstractBankFacadeControllerBean.java:120)

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:79)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)

at java.lang.reflect.Method.invoke(Method.java:618)

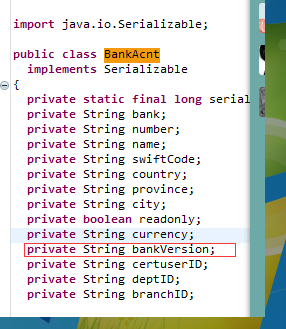
通过查看eas中的BankAcnt对象和wsdl，发现wsdl里面并没有bankVersion这个字段的描述，通过银企平台同事分析银企平台补丁分析，是银企平台补丁打的有问题导致。

<http://192.168.1.223:5283/ebgateway/services/BankService?wsdl>



现场的wsdl

而eas标准则是有的。



现场的代码。

**37、问题描述：** 金蝶提交银企报错，相关信息如下

编号：102SZQS-FKD-201608-0355:select data error.Too many items were returned in a SELECT list.. SQLCODE=-840, SQLSTATE=54004, DRIVER=3.68.61

Caused exception message is: An error occurred during implicit system action type "2". Information returned for the error includes SQLCODE "-840",

SQLSTATE "54004" and message tokens "".. SQLCODE=-727, SQLSTATE=56098, DRIVER=3.68.61

**提单号：**R20160811-0015

**问题分析：**从db2提供的错误码知，是select的时候查询的列太多，而金蝶最近添加了税务的相关私包，更新了脚本，在company表中添加了十几个字段，由于没找到提交银企的日志(金蝶产生的日志飞快)，但发现出纳生成凭证报同样类型的错误，最后发现提交银企和生成凭证都会走那同一段代码，可知是同一个问题导致，最后出私包，在选择列的时候只选择需要用到的相关列即可，生成凭证的具体异常如下

错误信息：select data error.Too many items were returned in a SELECT list.. SQLCODE=-840, SQLSTATE=54004, DRIVER=3.68.61

Caused exception message is: An error occurred during implicit system action type "2". Information returned for the error includes SQLCODE "-840", SQLSTATE "54004" and message tokens "".. SQLCODE=-727, SQLSTATE=56098, DRIVER=3.68.61 [Client -- String Serialize]

错误详细堆栈信息：com.kingdee.bos.dao.ormapping.SQLAccessException: select data error.Too many items were returned in a SELECT list.. SQLCODE=-840, SQLSTATE=54004, DRIVER=3.68.61

Caused exception message is: An error occurred during implicit system action type "2". Information returned for the error includes SQLCODE "-840", SQLSTATE "54004" and message tokens "".. SQLCODE=-727, SQLSTATE=56098, DRIVER=3.68.61 [Client -- String Serialize]

at com.kingdee.bos.dao.ormapping.ORMappingDAO.innerGetCollection(ORMappingDAO.java:822)

at com.kingdee.bos.dao.ormapping.ORMappingDAO.getCollection(ORMappingDAO.java:683)

at com.kingdee.bos.framework.ejb.AbstractEntityControllerBean.innerGetCollection(AbstractEntityControllerBean.java:580)

at com.kingdee.eas.framework.app.AbstractCoreBaseControllerBean.\_getCollection(AbstractCoreBaseControllerBean.java:339)

………(此处省略了一些代码)

com.kingdee.eas.fi.cas.app.AbstractPaymentBillControllerBean.\_getCollection(AbstractPaymentBillControllerBean.java:177)

at com.kingdee.eas.fi.cas.app.AbstractPaymentBillControllerBean.getPaymentBillCollection(AbstractPaymentBillControllerBean.java:163)

……

at com.kingdee.bos.transaction.EJBTxFacade.TxInvokerBean.invoke(TxInvokerBean.java:125)

at com.kingdee.bos.transaction.EJBTxFacade.TxInvokerBean.INVOKE\_SUPPORTS(TxInvokerBean.java:64)

at com.kingdee.bos.transaction.EJBTxFacade.TxInvokerBean\_LocalObjectImpl\_2.INVOKE\_SUPPORTS(Unknown Source)

at com.kingdee.bos.transaction.EJBTransactionProxy.invoke(EJBTransactionProxy.java:179)

at com.kingdee.bos.transaction.EJBTransactionProxy.invoke(EJBTransactionProxy.java:324)

at com.sun.proxy.$Proxy621.getPaymentBillCollection(Unknown Source)

at com.kingdee.eas.fi.cas.PaymentBill.getPaymentBillCollection(PaymentBill.java:115)

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:60)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:37)

at java.lang.reflect.Method.invoke(Method.java:611)

at com.kingdee.bos.ResponseInvocationHandler.invoke(ResponseInvocationHandler.java:194)

**at com.sun.proxy.$Proxy620.getPaymentBillCollection(Unknown Source)**

**at com.kingdee.eas.fi.cas.RecPayHelper.getPayColl(RecPayHelper.java:1519)**

**at com.kingdee.eas.fi.cas.RecPayHelper.getPayValidMapByAction(RecPayHelper.java:1658)**

at com.kingdee.eas.fi.cas.app.PaymentBillControllerBean.\_generateVoucher(PaymentBillControllerBean.java:2100)

at com.kingdee.eas.framework.app.AbstractCoreBillBaseControllerBean.generateVoucher(AbstractCoreBillBaseControllerBean.java:634)

at com.kingdee.eas.framework.app.CoreBillBaseControllerBean.\_generateVoucher(CoreBillBaseControllerBean.java:919)

at com.kingdee.eas.framework.app.AbstractCoreBillBaseControllerBean.generateVoucher(AbstractCoreBillBaseControllerBean.java:498)

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:60)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:37)

at java.lang.reflect.Method.invoke(Method.java:611)

at com.kingdee.bos.transaction.EJBTxFacade.TxInvokerBean.invoke(TxInvokerBean.java:125)

at com.kingdee.bos.transaction.EJBTxFacade.TxInvokerBean.INVOKE\_REQUIRED(TxInvokerBean.java:60)

at com.kingdee.bos.transaction.EJBTxFacade.TxInvokerBean\_LocalObjectImpl\_2.INVOKE\_REQUIRED(Unknown Source)

at com.kingdee.bos.transaction.EJBTransactionProxy.invoke(EJBTransactionProxy.java:171)

at com.kingdee.bos.transaction.EJBTransactionProxy.invoke(EJBTransactionProxy.java:324)

at com.sun.proxy.$Proxy621.generateVoucher(Unknown Source)

at com.kingdee.eas.framework.CoreBillBase.generateVoucher(CoreBillBase.java:337)

at rpc\_generate.\_PROXY\_com\_1\_kingdee\_1\_eas\_1\_fi\_1\_cas\_1\_IPaymentBill.pi113(Unknown Source)

at rpc\_generate.\_PROXY\_com\_1\_kingdee\_1\_eas\_1\_fi\_1\_cas\_1\_IPaymentBill.processInvoke(Unknown Source)

Caused By: com.ibm.db2.jcc.am.SqlException

com.ibm.db2.jcc.am.SqlException: An error occurred during implicit system action type "2". Information returned for the error includes SQLCODE "-840", SQLSTATE "54004" and message tokens "".. SQLCODE=-727, SQLSTATE=56098, DRIVER=3.68.61

at com.ibm.db2.jcc.am.gd.a(gd.java:752)

………(此处省略)

at com.ibm.db2.jcc.am.vo.b(vo.java:4385)

at com.ibm.db2.jcc.am.vo.hc(vo.java:746)

at com.ibm.db2.jcc.am.vo.executeQuery(vo.java:715)

at com.kingdee.bos.sql.shell.KDPreparedStatement.executeQuery(KDPreparedStatement.java:307)

at com.apusic.jdbc.adapter.PreparedStatementHandle.executeQuery(PreparedStatementHandle.java:

**解决方法：**

赖工在RecPayHelper中做了修改，私包解决了

**38、问题描述：** 银行账户余额查询当日余额消失了

操作步骤：先联机查询当日余额可以查到，查历史余额和当日余额后会提示不支持历史接口，此时在联机或者脱机查询当日的余额时查不到数据

**提单号：**李周俊邮件的内容

**问题分析：** 分析代码知这是AccountBankBalanceUI中定义的成员变量导致，它本意是用来做联机查询的优化的，就是联机查询时如果该银行账户联查失败了，在联机查的时候会过滤掉这个银行账户的联机查询，但是代码写的不好导致会影响脱机也过滤掉了，此处没有这样优化的必要，已出补丁解决，具体修改方法请见fm/be的svn记录



**39、问题描述：** 提交银企的时候提示’付款金额不能大于未付金额’

**提单号：** 无

**问题分析：** 此处是费用报销单的校验逻辑，异常堆栈信息如下：

[Thread-414 15:10:42]第01个业务单据 : FKD-201607-139679 生成银行付款单失败。

com.kingdee.eas.cp.bc.BizCollException: 付款金额不能大于未付金额

at com.kingdee.eas.cp.bc.app.reverse.payment.BillReverseHandler.payHandler(BillReverseHandler.java:997)

at com.kingdee.eas.cp.bc.app.reverse.payment.BillReverseHandler.paymentSingleHandler(BillReverseHandler.java:172)

at com.kingdee.eas.cp.bc.app.reverse.payment.BillReverseHandler.paymentHandler(BillReverseHandler.java:99)

at com.kingdee.eas.cp.bc.BizPayBillReBack.dispose(BizPayBillReBack.java:101)

at com.kingdee.eas.fi.cas.app.service.handler.ServiceHandler.dispose(ServiceHandler.java:783)

at com.kingdee.eas.fi.cas.app.service.PayPayService.doService(PayPayService.java:263)

at com.kingdee.eas.fi.cas.app.PaymentBillControllerBean.\_payForBook(PaymentBillControllerBean.java:624)

at com.kingdee.eas.fi.cas.app.PaymentBillControllerBean.\_payForBook(PaymentBillControllerBean.java:4436)

at com.kingdee.eas.fi.cas.app.AbstractPaymentBillControllerBean.payForBook(AbstractPaymentBillControllerBean.java:786)

at sun.reflect.GeneratedMethodAccessor1554.invoke(Unknown Source)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)

at java.lang.reflect.Method.invoke(Method.java:597)

at com.kingdee.bos.transaction.EJBTxFacade.TxInvokerBean.invoke(TxInvokerBean.java:125)

at com.kingdee.bos.transaction.EJBTxFacade.TxInvokerBean.INVOKE\_REQUIRED(TxInvokerBean.java:60)

at com.kingdee.bos.transaction.EJBTxFacade.TxInvokerBean\_LocalObjectImpl\_2.INVOKE\_REQUIRED(Unknown Source)

at com.kingdee.bos.transaction.EJBTransactionProxy.invoke(EJBTransactionProxy.java:171)

at com.kingdee.bos.transaction.EJBTransactionProxy.invoke(EJBTransactionProxy.java:324)

at $Proxy527.payForBook(Unknown Source)

at com.kingdee.eas.fi.cas.PaymentBill.payForBook(PaymentBill.java:504)

at sun.reflect.GeneratedMethodAccessor1553.invoke(Unknown Source)