### .........刘鸿洋交接文档..........

# Xmis部分

## 财务流水账

查询财务流水账，过滤条件分为按部门查询和模糊查询。其中模糊查询可根据合同名称和项目编码模糊查询数据。

### 文件路径

前端：

webRoot/financy/financyRunningAccountListPanel.js

webRoot/financy/financyRunningAccountList.js(布局)

webRoot/financy/financyRunningAccountList.jsp(页面)

后台：

Action : com.bitc.jk.xmis.web.action.FinancyAction.java

Service : com.bitc.jk.xmis.service.FinancyService.java

serviceImpl : com.bitc.jk.xmis.service.impl.FinancyServiceImpl.java

DAO :com.bitc.jk.xmis.dao.FinancyDAO.java

DAOImpl : com.bitc.jk.xmis.dao.Impl.FinancyDAOImpl.java

SQL : com.bitc.jk.xmis.util.sql.SQLScript\_Financy.java

### 说明

1. 财务来款流水账

Store(url= /financyMain.do?method=getFinancyRunningAccountList&searchType=1)🡪 getFinancyRunningAccountList(action)🡪 getFinancyIncomeRunningAccountList(service)🡪 getFinancyIncomeRunningAccountList(dao)🡪 SQLScript\_Financy.getFinancyIncomeRunningAccountList(sql)

1. 财务支出流水账

Store(url= /financyMain.do?method=getFinancyRunningAccountList&searchType=2)🡪 getFinancyRunningAccountList(action)🡪 getFinancyIncomeRunningAccountList(service)🡪getFinancyOutpayRunningAccountList(dao)🡪SQLScript\_Financy.getFinancyOutpayRunningAccountList(sql)

## 合同综合业务查询合并

以原有的业务查询为主，将综合查询中的高级查询整合到业务查询中。并且在不同情况下查询出不同列，多于列不显示。后台代码仅将原综合查询的高级查询中的收付款情况拆分为：收付款情况、发票情况、预计来款。

思路：将所有的列存放在数组中。收入合同为：allIncomeColumn，支出合同为：allOutColumn。针对每一种查询，从中抽去部分列。由于用户所在的事业部对数据的查询需求不同，因而可分为：关心业务数据（监控事业部）和不关心业务数据两种。以收入合同为例，将查询按不同情况分为：

1. 普通查询，即原业务查询。因是否关心业务数据分成2个数组。
2. 高级查询，新签订合同。因是否关心业务数据分成2个数组。
3. 高级查询，质保期到期。因是否关心业务数据分成2个数组。
4. 高级查询，收付款情况。因只有关心业务数据的部门有此数据，因此为1个数组。
5. 高级查询，发票情况。因只有关心业务数据的部门有此数据，因此为1个数组。
6. 高级查询，预计来款。因只有关心业务数据的部门有此数据，因此为1个数组。
7. 高级查询，财务到款情况。因是否关心业务数据分成2个数组。

因此收入合同定义了11个数组，数组的每一个元素为数字，表示待选列在allIncomeColumn中的下标。通过WebRoot/contract/contractList.js文件中的getColumns函数得到列模型，查询时通过改变gridpanel的store和colModel来展现最终结果。原store的url与原来相同。

高级查询各种情况在后台的分类代码如下：

文件：com.bitc.jk.xmis.service.impl.ContractServiceImpl.java

新签订合同：Type\_Contract\_Integration\_New

预计来款：Type\_Contract\_Integration\_EstimateContractAmt

发票情况：Type\_Contract\_Integration\_Invoice

质保到期：Type\_Contract\_Integration\_Warranty

收付款情况：Type\_Contract\_Integration\_Performed

财务到款情况：Type\_Contract\_Integration\_Finance

### 文件路径

前端：

WebRoot/contract/contractList.js

WebRoot/contract/contractList.jsp

后台：

Action : com.bitc.jk.xmis.web.action.ContractAction.java

Service : com.bitc.jk.xmis.service.ContractService.java

serviceImpl : com.bitc.jk.xmis.service.Impl.ContractServiceImpl.java

DAO : com.bitc.jk.xmis.dao.ContractDAO.java

DAOImpl : com.bitc.jk.xmis.dao.Impl.ContractDAOImpl.java

SQL : com.bitc.jk.xmis.util.SQLScript.java

## 到款辅助查询

功能描述参考文档‘到款辅助查询.docx’。

### 文件路径

前端：

WebRoot/contract/paymentCondition.js（付款计划）

WebRoot/contract/paymentCondition.jsp

WebRoot/project/estimatedPaymentPanel.js（手动设置预计来款）

WebRoot/project/projectInfo.js

WebRoot/project/projectInfo.jsp

WebRoot/contract/incomeAssistSearchPanel.js（到款辅助查询）

WebRoot/contract/incomeAssistSearch.js（到款辅助查询页面布局）

WebRoot/contract/incomeAssistSearch.jsp

WebRoot/contract/confirmPayment.js（确认付款）

WebRoot/contract/contractList.js

WebRoot/contract/contractList.jsp

----contractList.js中和到款有关的三列:按进度应收账款，按进度已开票未收款，按进度未开票应收款

后台：

Action: com/bitc/jk/xmis/web/action/ContractAction.java

Service : com.bitc.jk.xmis.service.ContractService.java

serviceImpl : com.bitc.jk.xmis.service.Impl.ContractServiceImpl.java

DAO : com.bitc.jk.xmis.dao.ContractDAO.java

DAOImpl : com.bitc.jk.xmis.dao.Impl.ContractDAOImpl.java

SQL : com.bitc.jk.xmis.util.SQLScript.java

### 说明

1. 付款计划的查询(WebRoot/contract/paymentCondition.js文件)

paymentStore(js)🡪getPaymentConditionList(action)🡪getPaymentConditionList(service)🡪getPaymentConditionList(dao)🡪 SQLScript.getPaymentConditionByContract(sql)

-----自己添的

项目的付款计划查询(WebRoot/contract/estimatedPaymentPanel.js文件)

com.bitc.xmis.ProjectPaymentConditionGrid (js)-> getPaymentConditionListByProjectID (action)->

getPaymentConditionListByProjectID (service)-> getPaymentConditionList (dao)->SQLSCript. getPaymentConditionByContract (sql)

--比较合同的付款计划，只是在service层多了一步:

int contractID = this.getContractIDByProjectID(projectID);

项目的收款计划中比合同的收款计划少一列"预计开票日期",代码已经有了，如果要加上就把注释的放开即可。

1. 付款计划的新建及修改(WebRoot/contract/paymentCondition.js文件中url为：/contractMain.do?method=savePaymentCondition1)

Ext.Ajax.request(js)🡪 savePaymentCondition1(action,先前的版为savePaymentCondition)🡪 doSavePaymentCondition1(service)🡪 doSavePaymentCondition(dao)🡪 SQLScript.savePaymentCondition(sql)

1. 付款计划的删除(WebRoot/contract/paymentCondition.js文件中url为：/contractMain.do?method=delPaymentCondition)

paymentForm.getForm().submit(js)🡪 delPaymentCondition(action)🡪 doDelPaymentCondition(service)🡪 doDelPaymentCondition(dao)🡪 SQLScript.deletePaymentCondition(sql)

1. 手动设置预计来款的查询(WebRoot/project/estimatedPaymentPanel.js中的com.bitc.xmis.ProjectEstimatedPaymentGrid)

Store(js)🡪 getProjectEstimatedPaymentList(action)🡪 getProjectEstimatedPaymentList(service)🡪 getProjectEstimatedPaymentList(dao)🡪 SQLScript.getProjectEstimatedPaymentList(sql)

1. 手动设置预计来款的新建(WebRoot/project/estimatedPaymentPanel.js文件中的url为：/contractMain.do?method=saveEstimatedInfo)

Ext.Ajax.request(js)🡪 saveEstimatedInfo(action)🡪 doSaveEstimatedInfo(service)🡪 doSaveEstimatedInfo(dao)🡪 SQLScript.doSaveEstimatedInfo(sql)

1. 手动设置预计来款的修改(WebRoot/project/estimatedPaymentPanel.js文件中的url为：/contractMain.do?method=saveEstimatedInfo)

Ext.Ajax.request(js)🡪 saveEstimatedInfo(action)🡪 doSaveEstimatedInfo(service)🡪doUpdateEstimatedInfo(dao)🡪 SQLScript.doUpdateEstimatedInfo(sql)

1. 手动设置预计来款的删除(WebRoot/project/estimatedPaymentPanel.js文件中的url为：/contractMain.do?method=deleteEstimatedInfo)

Ext.Ajax.request(js)🡪 deleteEstimatedInfo(action)🡪 doDeleteEstimatedInfo(service)🡪 doDeleteEstimatedInfo(dao)🡪 SQLScript.doDeleteEstimatedInfo(sql)

1. 确认付款操作(WebRoot/contract/confirmPayment.js)

payForm.getForm().submit(js)🡪 savePaymentDetailQuickly(action)🡪 doSavePaymentDetailQuickly(service)🡪 doSavePaymentDetail(service)🡪 doSavePaymentDetail(dao)🡪 SQLScript.saveChargedDetail(sql)

flag = **this**.doSavePaymentDetail(pay, userDNA);

1. 名词解释

A = **按进度预计来款合计** ：目节点的实际进入日期小于等于当前日期的所有节点的预计来款金额总和。

B = **已开票总额**

C = **财务来款**

**按进度应收账款**

按进度应收账款 = A-C，若A<C，按进度应收账款 = 0

**按进度未开票应收款**

1. 当A≤B，按进度未开票应收款 = 0
2. 当A>B

* 当B>C，按进度未开票应收款 = A-B
* 当B<C，按进度未开票应收款 = A-C
* 当B=C，按进度未开票应收款 = A-B(或C)

**按进度已开票未收款**

1. 当A≤B，按进度已开票未收款 = A-C。若A<C，按进度已开票未收款 = 0
2. 当A>B

* 当B≤C，按进度已开票未收款= 0
* 当B>C，按进度已开票未收款 = B-C

1. 数据库变更

xmis.con\_收款条件表增加列‘预计开票日期’datetime

xmis.con\_收款条件表增加列‘是否已到款’bit

1. 功能说明，参考‘到款辅助查询.docx’

## 人员登录统计

### 文件路径

前端：

WebRoot/developmentProgress/accounting.js

WebRoot/developmentProgress/PersonMonthAccounting.js

WebRoot/developmentProgress/PersonWeekAccounting.js

WebRoot/developmentProgress/accounting.jsp

后台：

Action:com/bitc/jk/xmis/web/action/pay.java

Service: com/bitc/jk/xmis/service/DevelopProgressService.java

serviceImpl: com/bitc/jk/xmis/service/impl/DevelopProgressServiceImpl.java

dao: com/bitc/jk/xmis/dao/DevelopProgressDAO.java

daoImpl: com/bitc/jk/xmis/dao/impl/DevelopProgressDAOImpl.java

### 说明

用到了Ext.grid.PivotGrid、Ext.grid.PivotAxis组件。并重写了PivotAxis的以下方法：

getTuples：修改该方法中对数据排序的逻辑。其中为该组件增加属性：realSortColumn，若realSortColumn的值不为NULL或undefined，则以指定的列排序。如下：

**new** com.bitc.xmis.accounting.DeptAxis({

dataIndex: 'deptName',

realSortColumn :'deptPx',

direction :'asc',

width : 135

})

由上所示，该轴的数据为'deptName'，排序为'deptPx'，若不指定realSortColumn，则按照dataIndex排序。

具体代码如下，修改部分文字底色为红色。

getTuples: **function**() {

**var** newStore = **new** Ext.data.Store({});

newStore.data = **this**.store.data.clone();

newStore.fields = **this**.store.fields;

**var** sorters = [],

dimensions = **this**.dimensions,

length = dimensions.length,

i;

**for** (i = 0; i < length; i++) {

sorters.push({

field : dimensions[i].realSortColumn == **undefined** ? dimensions[i].dataIndex : dimensions[i].realSortColumn,

direction: dimensions[i].direction || 'ASC'

});

}

newStore.sort(sorters);

**var** records = newStore.data.items,

hashes = [],

tuples = [],

recData, hash, info, data, key;

length = records.length;

**for** (i = 0; i < length; i++) {

info = **this**.getRecordInfo(records[i]);

data = info.data;

hash = "";

**for** (key **in** data) {

hash += data[key] + '---';

}

**if** (hashes.indexOf(hash) == -1) {

hashes.push(hash);

tuples.push(info);

}

}

newStore.destroy();

**return** tuples;

}

buildHeaders：修改原组件显示top轴和left轴的标题显示及增加链接。修改部分文字底色为红色。

buildHeaders :**function**() {

**var** tuples = **this**.getTuples(),

rowCount = tuples.length,

dimensions = **this**.dimensions,

dimension,

colCount = dimensions.length,

headers = [],

tuple, rows, currentHeader, previousHeader, span, start, isLast, changed, i, j;

**for** (i = 0; i < colCount; i++) {

dimension = dimensions[i];

rows = [];

span = 0;

start = 0;

**for** (j = 0; j < rowCount; j++) {

tuple = tuples[j];

isLast = j == (rowCount - 1);

currentHeader = tuple.data[dimension.dataIndex];

changed = previousHeader != **undefined**&& previousHeader != currentHeader;

**if** (i > 0 && j > 0) {

changed = changed || tuple.data[dimensions[i-1].dataIndex] != tuples[j-1].data[dimensions[i-1].dataIndex];

}

**if** (changed) {

**var** t = previousHeader;

**if**(/^\d{4}-\d{2}$/.test(previousHeader)){

t = '<a href=\"javascript:;\" onclick=\"javascript:reloadDeptMonthWin(\''+previousHeader+'\');\">' + parseInt(previousHeader.substring(5,8)) + '月</a>';

}**elseif**(/^\d{4}$/.test(previousHeader)){

t = previousHeader + '年';

}**else**{

t = '<a href=\"javascript:;\" onclick=\"javascript:showPersonMonthWin(\''+previousHeader+'\',\'dept\');\">' + previousHeader + '</a>';

}

rows.push({

header: t,

span : span,

start : start

});

start += span;

span = 0;

}

**if** (isLast) {

**var** t = currentHeader;

**if**(currentHeader == '小计')

t = '<a href=\"javascript:;\" onclick=\"javascript:reloadDeptMonthWin(\''+currentHeader+'\');\">' + currentHeader + '</a>';

**if**(currentHeader == '合计')

t = '<a href=\"javascript:;\" onclick=\"javascript:showPersonMonthWin(\''+currentHeader+'\',\'comp\');\"><font color=\"red\">' + currentHeader + '</font></a>';

rows.push({

header: t,

span : span + 1,

start : start

});

start += span;

span = 0;

}

previousHeader = currentHeader;

span++;

}

headers.push({

items: rows,

width: dimension.width || **this**.defaultHeaderWidth

});

previousHeader = **undefined**;

}

**return** headers;

}

# 院综合查询

## 项目进出信息查询表



### 相关文件

存储过程：zhcx\_searchProjectInfo

Js文件：/report/src/main/webapp/report/js/projectBaseInfo.js

报表文件：/report/src/main/webapp/report/projectBaseInfo.rptdesign

### 说明

主表： 预算项目档案表dbo.sky\_jczl\_item

相关表： 经费类别表（经费属性）dbo.sky\_jczl\_jflb

部门名称表 dbo.department

人员名称表 dbo.hr\_hi\_person

预算项目课题对应试图 dbo.v\_sky\_jczl\_itemkt

预算课题档案表 dbo.sky\_jczl\_kt

预算项目附件表 dbo.sky\_jczl\_item\_fj

计算信息：

允许赤字数：该项目所辖课题允许赤字数总和

## 课题基础信息查询表



### 相关文件

存储过程：zhcx\_searchIssueInfo

Js文件：/report/src/main/webapp/report/js/IssueBaseInfo.js

报表文件：/report/src/main/webapp/report/IssueBaseInfo.rptdesign

### 说明

主表： 预算项目档案表dbo.sky\_jczl\_kt

相关表： 经费类别表（经费属性）dbo.sky\_jczl\_jflb

部门名称表 dbo.department

人员名称表 dbo.hr\_hi\_person

预算项目课题对应试图 dbo.v\_sky\_jczl\_itemkt

预算项目档案表 dbo.sky\_jczl\_item

预算课题附件表 dbo.sky\_jczl\_kt\_fj

计算信息：

允许赤字数：该项目所辖课题允许赤字数总和

## 课题外协业务费情况表



### 相关文件

存储过程：zhcx\_ktCooperativeUnit

Js文件：/report/src/main/webapp/report/js/ktCooperativeUnit.js

报表文件：/report/src/main/webapp/report/ktCooperativeUnit.rptdesign

### 说明

外协单位表：dbo.sky\_jczl\_xzdw

|  |  |
| --- | --- |
| Itemcode | 课题号，即课题表的ktcode |
| csupcode | 外协单位编码 |
| xzmoney | 外协业务费金额 |

凭证表：dbo.sky\_gl\_accvouch

|  |  |
| --- | --- |
| cDefine12 | 外协单位编码 |
| citem\_id | U8课题号 |

单位表：dbo.Vendor

|  |  |
| --- | --- |
| cVenCode | 单位编码 |
| cVenName | 单位名称 |

已收外协业务费：dbo.sky\_gl\_accvouch表自定义字段12 + u8课题编号求 sum（md）

SELECTcDefine12,citem\_id,SUM(md)AStotalMD

INTO#ktyfk

FROMdbo.sky\_gl\_accvouch

WHEREcDefine12ISNOTNULL

GROUPBYcDefine12,citem\_id

尚未支付外协费：外协业务费-已收外协业务费(xzmoney- totalMD)

## 项目外协业务费情况表



### 相关文件

存储过程：zhcx\_projectCooperativeUnit

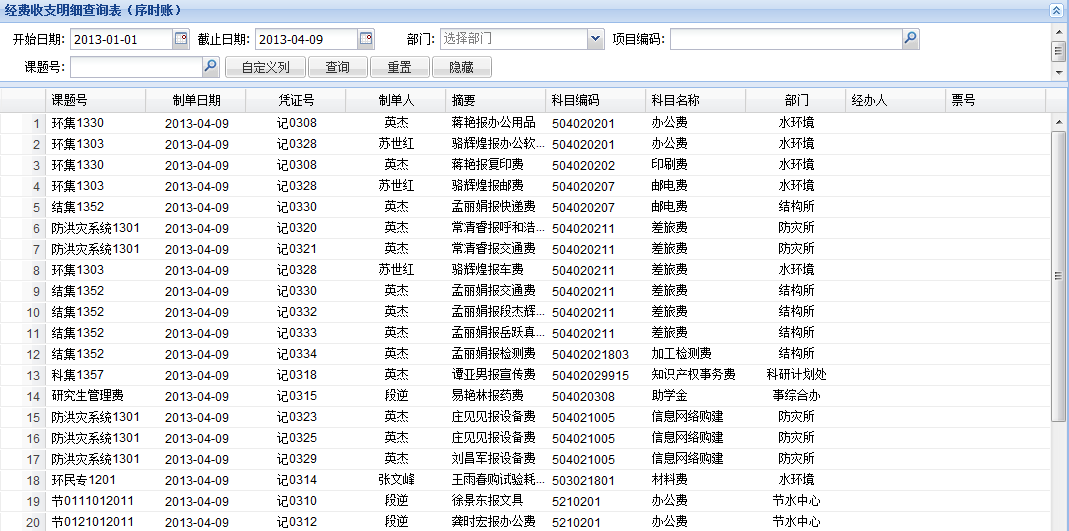
Js文件：/report/src/main/webapp/report/js/projectCooperativeUnit.js

报表文件：/report/src/main/webapp/report/projectCooperativeUnit.rptdesign

### 说明

类似[课题外协业务费情况表](#_课题外协业务费情况表)，得到课题的信息后，对项目再求sum。

## 经费收支序时帐



### 相关文件

存储过程: zhcx\_fundExpensesInfo

Js文件：/report/src/main/webapp/report/js/fundExpensesInfo.js

报表文件：/report/src/main/webapp/report/js/fundExpensesInfo\_ext.js

### 说明

相关表

课题基本信息 dbo.sky\_jczl\_kt

凭证表 dbo.GL\_accvouch

项目课题对照表dbo.v\_sky\_jczl\_itemkt

部门表 dbo.Department

人员表 dbo.hr\_hi\_person

会计科目表 dbo.code

## 预算科目与会计科目对照表（水资源项目、民口水专项）



### 相关文件

存储过程：zhcx\_budgetWithAccountingMappingInfo

Js文件：/report/src/main/webapp/report/js/budgetWithAccountingMappingInfo.js

报表文件：/report/src/main/webapp/report/budgetWithAccountingMappingInfo.rptdesign

### 说明

相关表：

经费类别表（经费属性）dbo.sky\_jczl\_jflb

经费类别对应的预算科目体系dbo.sky\_jczl\_jflbs

预算科目对应的会计科目dbo.sky\_jczl\_jflbs\_detail

会计科目定义表dbo.code

其他：

水资源项目和民口水专项为具体的经费类别，选择即可。

## 房屋租赁合同信息表



### 相关文件

存储过程：zhcx\_rentalHousing

Js文件：/report/src/main/webapp/report/js/rentalHousing.js

报表文件：/report/src/main/webapp/report/rentalHousing.rptdesign

### 说明

相关表：

租赁合同基本信息表：dbo.sky\_ht

合同预警设置表：dbo.sky\_ht\_yjset

部门表：dbo.department

预警信息：

距离合同到期90天之内的合同，预警信息为剩余天数。

距离合同到期大于90天的合同，若该合同在下次付款预警日期内，则提示天数及金额。

## 已开发票未到款情况统计表



### 相关文件

存储过程：zhcx\_invoiceUnIncome

Js文件：/report/src/main/webapp/report/js/invoiceUnIncome.js

报表文件：/report/src/main/webapp/report/invoiceUnIncome.rptdesign

### 说明

相关表：

发票登记表：dbo.sky\_fp

凭证表：dbo.sky\_gl\_accvouch

部门表：dbo.department

人员表：dbo.hr\_hi\_person

其他：

凭证表中的发票号：cDefine2

## 固定资产查询（在用、已报废）



### 相关文件

存储过程：zhcx\_fixedAssetsOnline

Js文件：/report/src/main/webapp/report/js/fixedAssetsOnline.js

报表文件：/report/src/main/webapp/report/fixedAssetsOnline.rptdesign

### 说明

相关表：

固定资产表：ufdata\_403\_xxxx.fa\_cards

固定资产自定义项目表：ufdata\_403\_xxxx.fa\_ItemsManual

固定资产使用状况定义表：ufdata\_403\_xxxx.fa\_Status

固定资产类别表：ufdata\_403\_xxxx.fa\_AssetTypes

部门表：ufdata\_403\_xxxx.Department

资金来源：fa\_ItemsManual.man195s where ufdata\_403\_xxx.fa\_cards.cCardNum =

ufdata\_403\_xxx.fa\_ItemsManual.cCardNum

保管人: fa\_ItemsManual.man179s where ufdata\_403\_xxx.fa\_cards.cCardNum =

ufdata\_403\_xxx.fa\_ItemsManual.cCardNum

使用状态：fa\_Status.sName where ufdata\_403\_xxx.fa\_cards.sStatusID = ufdata\_403\_xxx.fa\_Status.sID

数量：缺省为1

使用部门：sDeptNames（sDeptNum，关联表Department.cDepCode，本字段设计为可多部门，用逗号分隔，目前尚无此情况。bMultiDept 是否多部门使用）

## 借款序时帐



### 相关文件

存储过程：zhcx\_loanAccount

Js文件：/report/src/main/webapp/report/js/loanAccount.js

报表文件：/report/src/main/webapp/report/loanAccount.rptdesign

### 说明

相关表：

凭证表：dbo.sky\_gl\_accvouch

课题表：dbo.sky\_jczl\_kt

人员表：dbo.hr\_hi\_person

会计科目表：dbo.code

部门表：dbo.department

## 借款情况表



概述：查询某人在某课题上的借款情况。

### 相关文件

存储过程：zhcx\_loanInfo

Js文件：/report/src/main/webapp/report/js/loanInfo.js

报表文件：/report/src/main/webapp/report/loanInfo.rptdesign

### 说明

凭证表：dbo.sky\_gl\_accvouch

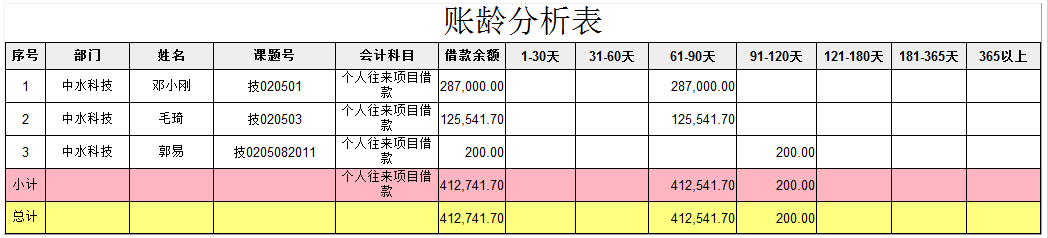
课题表：dbo.sky\_jczl\_kt

人员表：dbo.hr\_hi\_person

会计科目表：dbo.code

部门表：dbo.department

## 账龄分析表



概述：计算某部门的某人在某课题及会计科目上的借款总额及各时段内的借款情况。

### 相关文件

存储过程：zhcx\_aginAnalysis

Js文件：/report/src/main/webapp/report/js/aginAnalysis.js

报表文件：/report/src/main/webapp/report/aginAnalysis.rptdesign

### 说明

相关表：

凭证表：dbo.sky\_gl\_accvouch

课题表：dbo.sky\_jczl\_kt

人员表：dbo.hr\_hi\_person

会计科目表：dbo.code

部门表：dbo.department

借款账龄定义：

例如，2013年1月5日，某人A借款10万，票号20130105xxx。截止2013年1月25日，还款额为0，借款余额为10万，账龄25-5=20天，则账龄分析描述为：20天账龄的10万借款；如果在1月10日，还款2万，借款余额为10-2=8万，分析描述为：20天账龄的8万借款。

借款发生额：md

借款还款额：mc

借款余额：md-mc

还款原则是：按票号（cn\_id）冲减

## 其他

### 经费类别下拉选择树

组件相关文件：/report/src/main/webapp/extjs/js/ext-plugins.js

组件名称：Ext.form.TreeField

参数：详见代码

实例相关文件：/report/src/main/webapp/extjs/js/ext-extra.js

实例名称：combos.JflbTreeCombo

实例用法：

1. 显示所有经费类别，所有节点都可选

**var** jflb = **new** combos.JflbTreeCombo();

1. 显示所有经费类别，非子节点不可选

**var** jflb = **new** combos.JflbTreeCombo({

selectParent :**false**

});

1. 显示某经费下的经费类别，其中‘001’为经费类别编码

**var** jflb = **new** combos.JflbTreeCombo({

treeRootConfig : {

id :'001',

text : '经费类别',

draggable :**false**

}

});

后台：

相关文件：

/report/src/main/java/com/bitc/report/mvc/IssueAction.java

/report/src/main/java/com/bitc/report/service/IssueService.java

/report/src/main/java/com/bitc/report/dao/IssueDAO.java

访问的方法：

Action:getFundType

Service:getFundType(String bookcode,String jflbcode)

DAO：getFundType(BookInfo bookInfo,String jflbcode)

### 项目课题选择组件

组件相关文件：/report/src/main/webapp/extjs/js/ext-plugins.js

组建名称：Ext.ux.form.ChooseField

项目配置文件：/report/src/main/webapp/common/choose/Project.json.jsp

课题配置文件：/report/src/main/webapp/common/choose/Issue.json.jsp

项目、课题配置文件说明：

该组件可多选，与XMIS中相比配置文件中增加属性‘selectColumns’，该属性配置选择框中右侧‘以选择的内容’Grid的列模型，其他属性与XMIS中一致。

项目选择使用方法：

**new** Ext.ux.form.ChooseField({

fieldLabel : '项目编码',

name :'prjCode',

width : 250,

selectedGridTitle : '已选择的项目',

editable :**true**,

winConfig : {

modal :**true**,

width : 765,

title : '选择对应项目编号'

},

baseParamFun :**function**() {

**return** {

limit : 20,

bookCode : \_\_bookCode,

deptCode : depts.getValue()

};

},

paramFun :**function**() {

**return** {

start : 0

};

},

chooseUrl : \_\_path + '/common/choose/Project.json.jsp',

setValueFun :**function**(o) {

**if** (Ext.isEmpty(o)) {

Ext.Msg.alert('提示', '您还没有选择，请选择');

form.getForm().findField('prjCode').onTrigger1Click();

form.getForm().findField('pCode').setValue(**null**);

} **else** {

**var** xcode = [];

**var** xname = [];

**for** (**var** i = 0; i < o.length; i++) {

xcode.push(o[i].data['id']);

xname.push(o[i].data['dname']);

} form.getForm().findField('pCode').setValue(xcode.join(','));

form.getForm().findField('prjCode').setValue(xname.join(','));

}

},

clearValueFun :**function**() {

form.getForm().findField('prjCode').setValue(**null**);

form.getForm().findField('pCode').setValue(**null**);

}

})

其中，form.getForm().findField('prjCode')组件本身，form.getForm().findField('pCode')为隐藏属性。

课题选择使用方法：

**new** Ext.ux.form.ChooseField({

fieldLabel : '课题号',

name :'ktsearch',

width : 250,

selectedGridTitle : '已选择的课题',

editable :**true**,

winConfig : {

modal :**true**,

width : 765,

title : '选择对应合同课题编号'

},

baseParamFun :**function**() {

**return** {

limit : 20,

bookCode : \_\_bookCode,

deptCode : depts.getValue(),

ktcode : form.getForm().findField('ktsearch').getValue()

};

},

paramFun :**function**() {

**return** {

start : 0

};

},

chooseUrl : \_\_path+ '/common/choose/Issue.json.jsp',

setValueFun :**function**(o) {

**if** (Ext.isEmpty(o)) {

Ext.Msg.alert('提示','您还没有选择，请选择');

}**else**{

**var** kcode = [];

**var** kname = [];

**for**(**var** i = 0 ; i < o.length ; i++ ){

kcode.push(o[i].data['id']);

kname.push(o[i].data['dname']);

}

form.getForm().findField('ktsearch').setValue(kname.join(','));

form.getForm().findField('ktcode').setValue(kcode.join(','));

}

},

clearValueFun :**function**() { form.getForm().findField('ktsearch').setValue(**null**); form.getForm().findField('ktcode').setValue(**null**);

}

})

### 文件下载

相关文件：

Action: /report/src/main/java/com/bitc/report/mvc/FileDownload.java

Service: /report/src/main/java/com/bitc/report/service/FileDownloadService.java

Dao: /report/src/main/java/com/bitc/report/dao/FileDownloadDAO.java

调用方法：

Action层的downloadFile(String attachmentId,String fileSourceType,String d)

attachmentId：附件ID

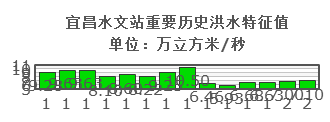
fileSourceType：附件类型，“1”为项目附件，其他为课题附件

d：账套

# 三峡综合展示

### 宜昌水文站重要历史洪水特征值

#### 界面



#### 访问方法

首页🡪枢纽运行🡪防洪补水(页面右侧)

#### 文件位置

页面：hmbj/cjfh.jsp

Js文件：hmbj/js/FusionChartsHelper.js

#### 相关函数说明

chartHelper.writeBar3函数

chartHelper.writeBar3=function(title,colors,names,values,width,height,target,ymin,ymax,links)

title：标题（String）

colors：每组数据显示的颜色（array[String]）

names：每组数据显示的名称（array[String]）

values：数据（array[float]）

width：图标的宽度(float)

height：图表的高度(float)

target：该报表显示的目标位置的ID（String）

ymin：y轴显示的最小值(float)

ymax：y轴显示的最大值(float)

links：图表穿透，为报表每个值存放链接的数组（array[String]）

#### 其他说明

使用类库：FusionCharts

使用图表：MSColumn2D.swf

数据：静态数据，存放在数组中

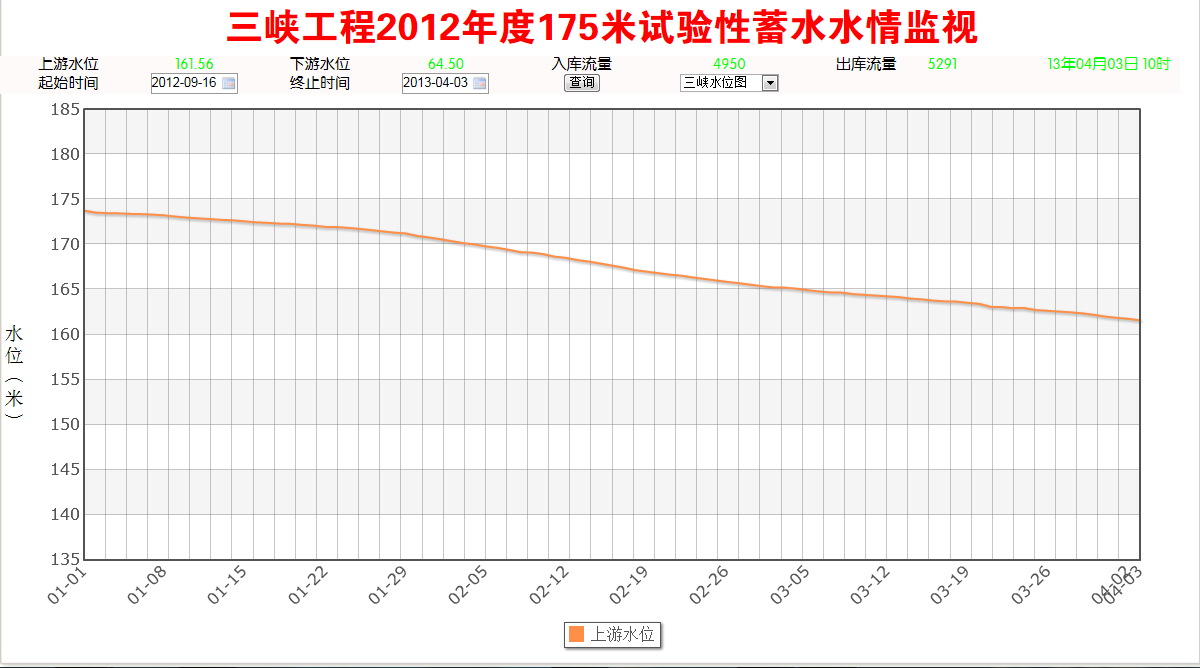
显示位置：target="grpah31"，id为"grpah31"的div中

FusionCharts相关文档：<http://docs.fusioncharts.com/charts/>

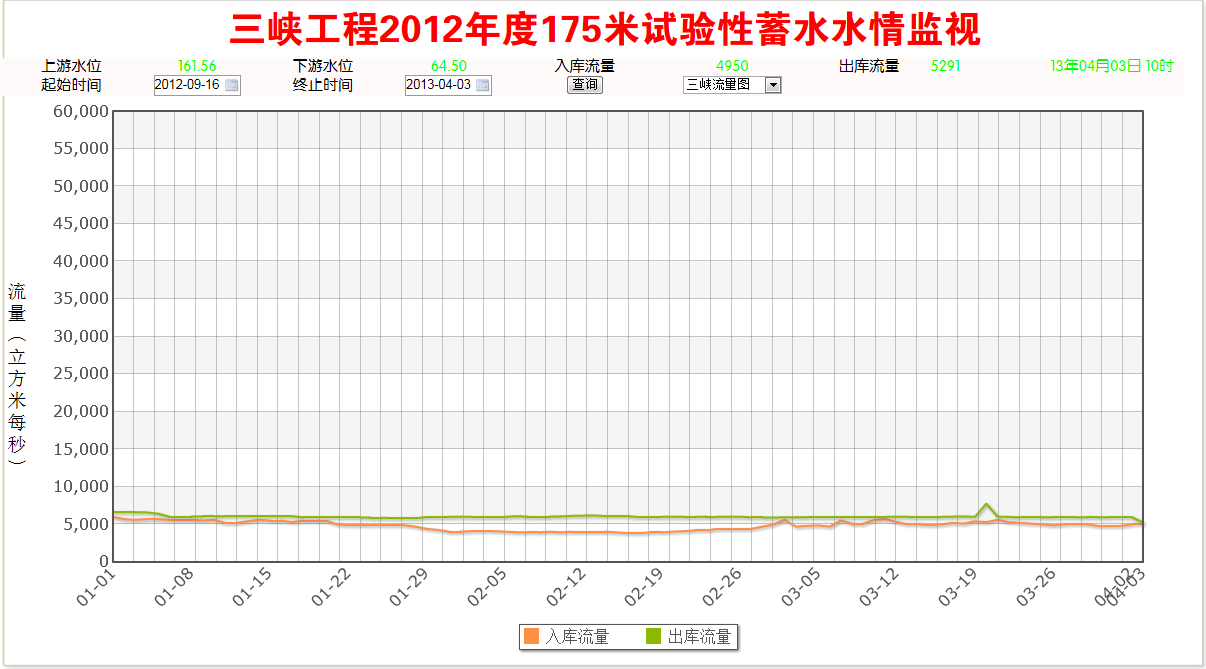
### 三峡水位图、三峡流量图、向家坝水位图、向家坝流量图

#### 页面（MSLine）

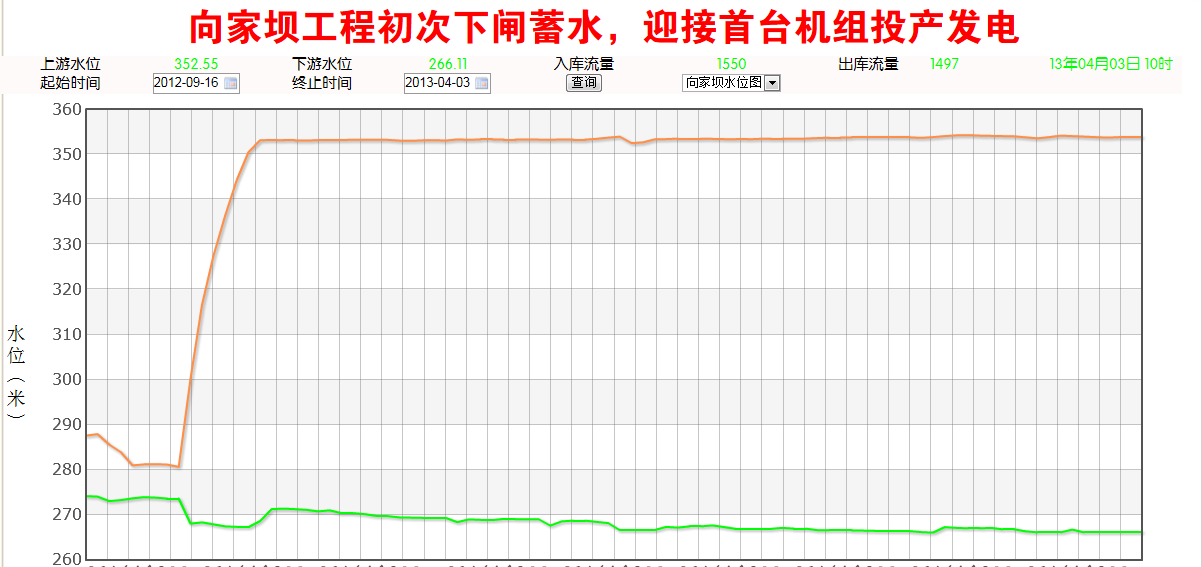
三峡水位图



三峡流量图



向家坝水位图

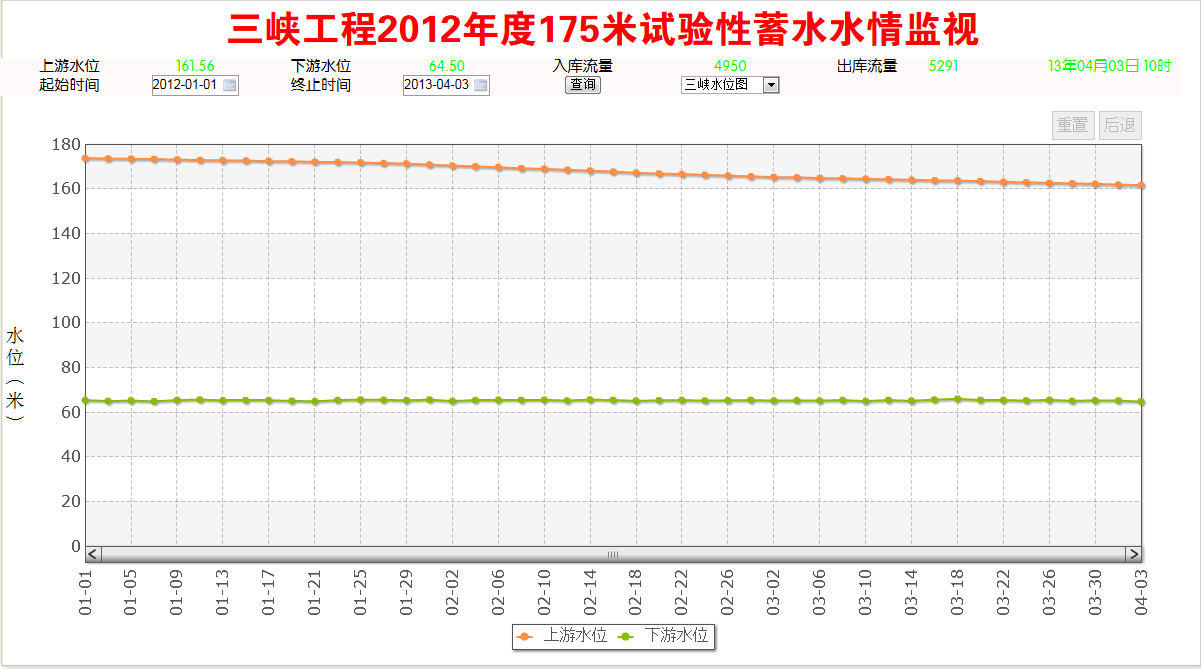


向家坝流量图

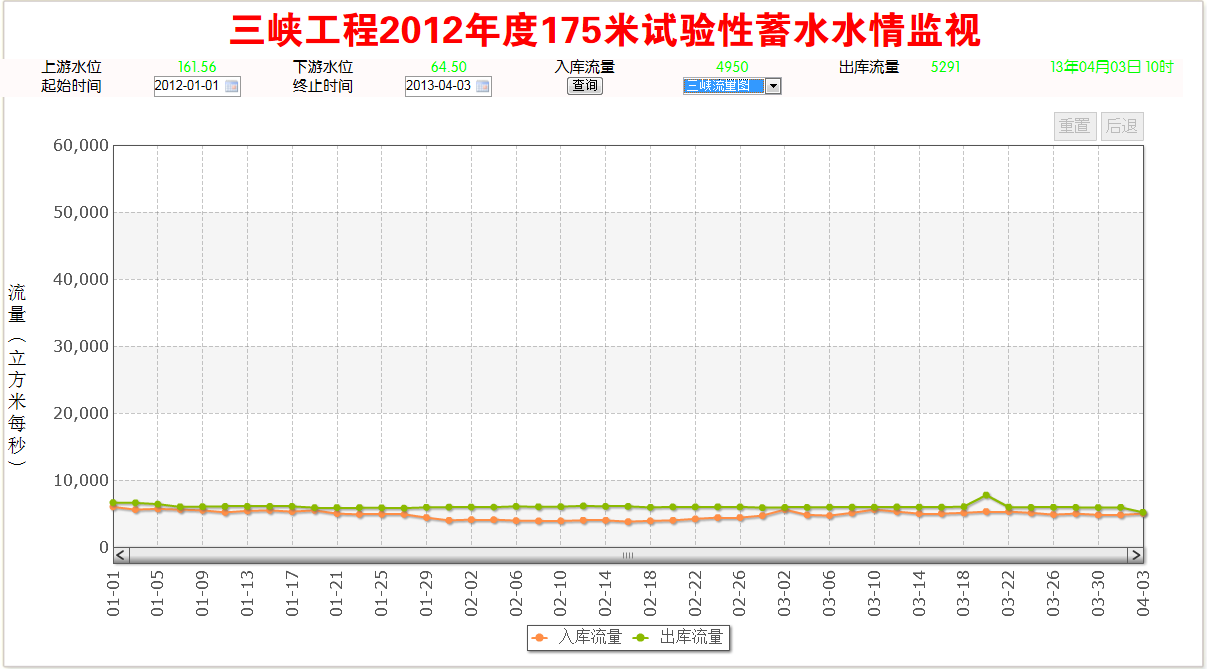


#### 页面（ZoomLine）

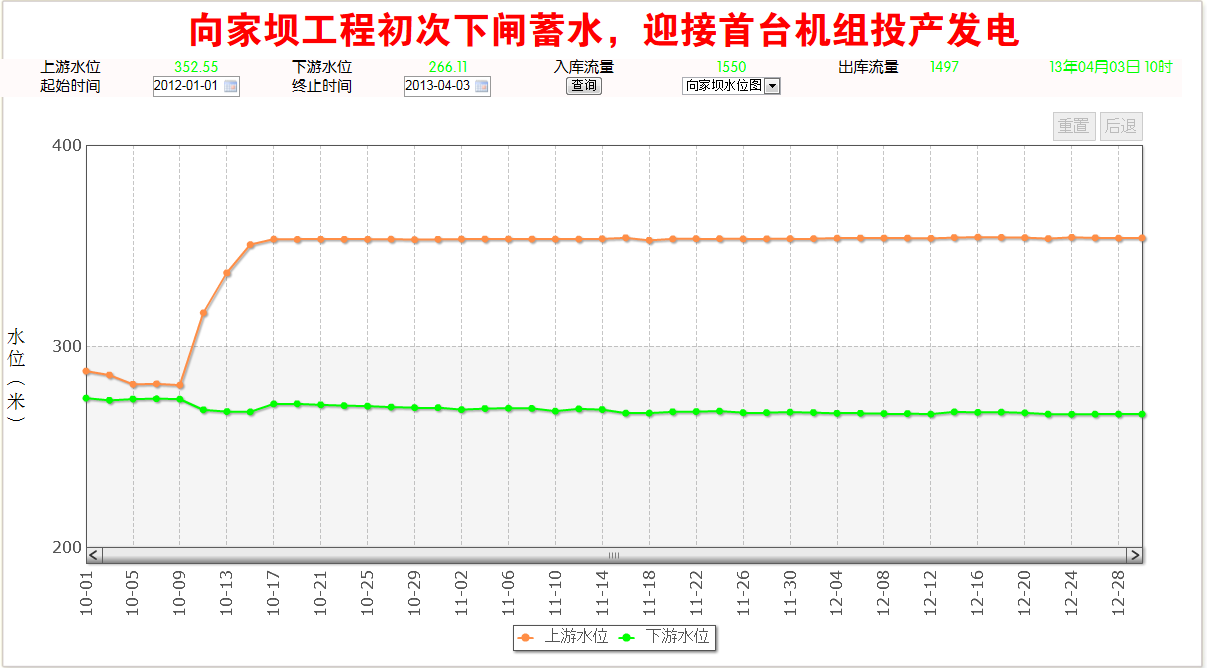
三峡水位图



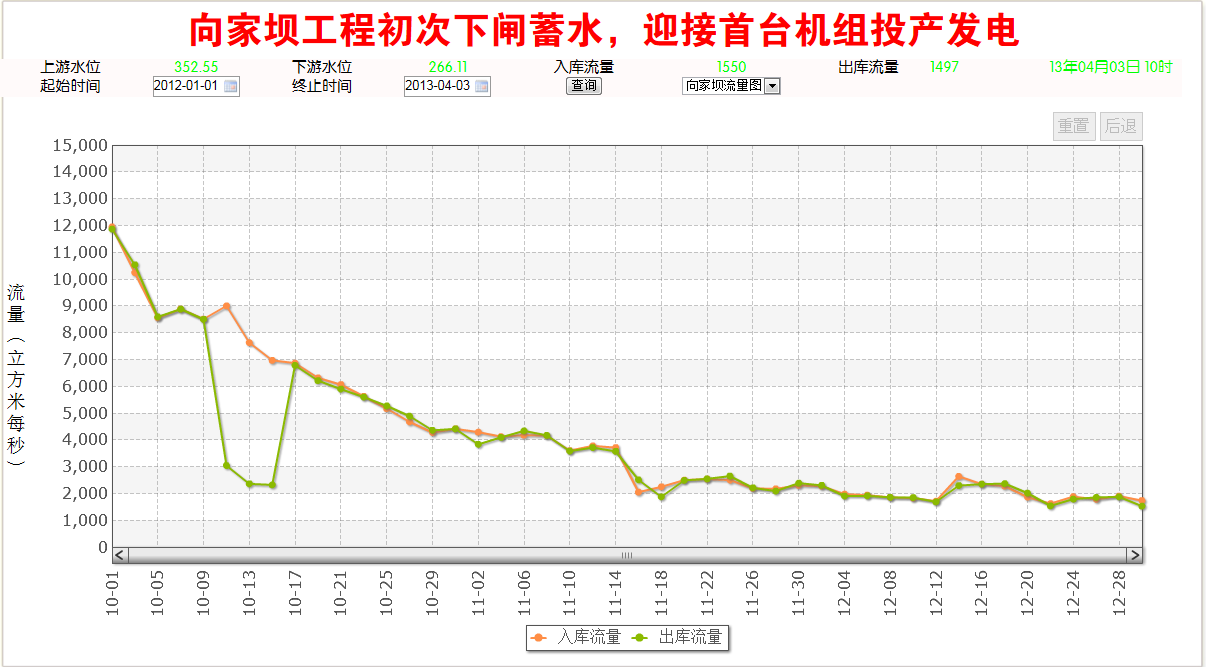
三峡流量图



向家坝水位图



向家坝流量图



#### 访问方法

<http://10.66.12.213:7001/hmbj/b_half.html>

[http://10.66.12.213:7001/hmbj/ b\_half-zoom.html](http://10.66.12.213:7001/hmbj/%20b_half-zoom.html)

#### 文件位置

JQ类库：hmbj/jquery.js

FusionCharts类库：hmbj/b-half/FusionCharts.js

日历组件：hmbj/b-half/WdatePicker.js

图表生成函数：hmbj/b-half/FusionChartsHelper.js

三峡实时数据JSP：hmbj/sanxiashishishuju.jsp

三峡水位流量数据：hmbj/b\_half/server.js

向家坝实时数据JSP：hmbj/xiangjiabashishishuju.jsp

向家坝水位流量数据JSP：hmbj/xiangjiaba.jsp

页面：（MSLine）hmbj/b\_half.html

页面（ZoomLine）：hmbj/b\_half-zoom.html

#### 相关函数说明

1. 三峡水位流量数据(server.js，由于原程序无JAVA代码，因此个参数只能写死)

$$=function(method,param,flag)

Method = “getWaters”

Param = “{year:2013}”

Flag = false

1. 异步访问数据(b\_half.html，JQ的API：<http://overapi.com/jquery/>)

jQuery.ajax({

url: "sanxiashishishuju.jsp?random=" + Math.random(),

type: "get",

cache:false,

dateType: 'json',

success: function(data){

}

})

url:加载数据的地址，其中三峡实时数据为“sanxiashishishuju.jsp”，向家坝实时数据为“xiangjiabashishishuju.jsp”，向家坝水位流量数据为“xiangjiaba.jsp”。参数“random”为随机数，防止读缓存。

Type:请求的类型，有“get”和“post”两种。

Cache:是否读取浏览器缓存。

dateType:返回结果的格式。(xml, json, script, or html)

success：请求成功后的回调函数。其中第一次参数data为返回的数据。在查询实时数据时：data[0].updata返回上游水位，data[1].downdata返回下游水位，data[2].indata返回入库流量，data[3].outdata返回出库流量。查询向家坝水位流量时：data[i].speed1为入库流量，data[i].speed2为出库流量，data[i].level1为上游水位，data[i].level2为下游水位。

注：本处使用jQuery.ajax而不是常用的$.ajax是因为原系统的JS脚本中定义过名为$的函数，为确保不影响原系统，因而改为jQuery.ajax。

1. Search = function(b\_half.html，无参数)

用途：查询前设置默认的开始日期及结束日期。

1. changeData = function(type,startTime,endTime)（b\_half.html）

type:查询的类型，可用值（1，2，3，4）。

1：三峡流量图

2：向家坝流量图

3：三峡水位图

4：向家坝水位图

startTime：开始日期

endTime：结束日期

1. chartHelper.writeLines99=function(title,colors,lines,names,arrays,width,height,target,charDiv,ymin,ymax,chartRightMargin)(FusionChartsHelper.js)

title：图表的标题

colors：曲线颜色，如["FF8E46","00FF00"]

lines：曲线名称，如["上游水位","下游水位"]

names：曲线上每一点数据的名称

arrays：数据，如[[123,456,789],[987,654,321]]

width:图表宽度

height：图表高度

target：图表显示位置

charDiv：图表自身对象

ymin：y轴最小值

ymax：y轴最大值

chartRightMargin：图表右边距

1. chartHelper.writeZoomLine=function(title,colors,lines,names,arrays,width,height,target,charDiv,ymin,ymax,chartRightMargin)(FusionChartsHelper.js)

title：图表的标题

colors：曲线颜色，如["FF8E46","00FF00"]

lines：曲线名称，如["上游水位","下游水位"]

names：曲线上每一点数据的名称

arrays：数据，如[[123,456,789],[987,654,321]]

width:图表宽度

height：图表高度

target：图表显示位置

charDiv：图表自身对象

ymin：y轴最小值

ymax：y轴最大值

chartRightMargin：图表右边距

1. 其他使用到的JQ（详情参考<http://overapi.com/jquery/>）

jQuery('#ID')：根据ID选择html页面中的元素。

jQuery('.class')：根据CLASS选择html的元素。

jQuery('#chartTitle').append("三峡工程2012年度175米试验性蓄水水情监视")：在元素中追加文字内容，本例是在ID为“chartTitle”的元素中增加文字“三峡工程2012年度175米试验性蓄水水情监视”。

jQuery('#chartTitle').empty()：清空元素中的内容。可以是文字，也可能是其他标签。

#### 其他说明

使用类库：

FusionCharts

JQuery

使用图表：MSLine.swf;ZoomLine.swf.swf

FusionCharts相关文档：<http://docs.fusioncharts.com/charts/>

### Flash读取xml

#### Xml定义

Pages:包含所有的flash文件。

Page:每个flash文件对应一个page,’name’属性：flash文件在AS脚本中的名字。

Text:每个动态文本框一个text标签.在一个page标签中的text标签的name属性唯一。

Property:一个文本框的属性。

1. content:文本内容，其中@h为换行，@t为要替换文字的占位符
2. font\_type:字体颜色
3. font\_algin:段落对齐方式
4. font\_color:文本颜色
5. height:文本框高度
6. width:文本框宽度
7. x\_position:文本框左上角在图层的x轴坐标
8. y\_position:文本框左上角在图层的y轴坐标
9. font\_leading:文本行间距
10. font\_letterSpacing:文本字间距
11. font\_bold:是否为粗体
12. font\_italic:是否为斜体
13. font\_leftMargin:段落左边距
14. font\_rightMargin:段落右边距
15. font\_indent:首行缩进（磅值）
16. border:文本框是否显示
17. font\_change:修改文本的方式，主要与颜色有关，为‘specified’时，修改的‘change’属性中配置的文字变色，颜色为‘font\_color’配置的颜色，‘all’时，文本全部变色，‘none’时，文本不变色。
18. change:该项中的子属性‘item’为文本中需要替换的文字，每个@t占位符替换一个‘item’中的文字

脚本代码：略

#### actionScript2.0源码

System.useCodepage = true;

var xml:XML = new XML();

xml.ignoreWhite = true;

//文字不可选

snjs\_leftContent.selectable = false;

snjs\_leftContent.html = true;

var format:TextFormat = new TextFormat();

xml.onLoad = function(success) {

if (success) {

//所有的pages

pages = xml.firstChild.childNodes;

for(i=0;i<pages.length;i++){

if("xiluodu\_jianjie\_page" == pages[i].attributes.name){

texts = pages[i].childNodes;

for(j=0;j<texts.length;j++){

if("snjs\_leftContent" == texts[j].attributes.name){

pros = texts[j].childNodes;

//for循环遍历每一个property 查看changeType

var changeType = "none";

for(z = 0;z<pros.length;z++){

if("font\_change" == pros[z].attributes.name){

if(undefined != pros[z].firstChild.nodeValue){

changeType = pros[z].firstChild.nodeValue;

}

break;

}

}

var color = null;

if("specified" == changeType){

for(z = 0;z<pros.length;z++){

if("font\_color" == pros[z].attributes.name){

var tmp = pros[z].firstChild.nodeValue.split("0x");

color = "#" + tmp[1];

break;

}

}

}

//for循环遍历每一个property

for(z = 0;z<pros.length;z++){

//处理内容

if("content" == pros[z].attributes.name){

var content = pros[z].firstChild.nodeValue;

var result\_h = "";

//替换换行符

if(content.indexOf("@h") > 0){

var array = content.split("@h");

for(x=0;x<array.length;x++){

if(x!=array.length -1){

result\_h = result\_h + array[x] + "<br>";

}else{

result\_h = result\_h + array[x];

}

}

}else{

result\_h = content;

}

//替换文字

var result\_h\_t = "";

if(result\_h.indexOf("@t") >= 0){

var array = result\_h.split("@t");

for(y = 0;y<pros.length;y++){

if("change" == pros[y].attributes.name){

for(x=0;x<array.length-1;x++){

var tmp = pros[y].childNodes[x].firstChild.nodeValue;

if("specified" == changeType){

tmp = "<font color=\""+color+"\">" + tmp + "</font>";

}

result\_h\_t = result\_h\_t + array[x] + tmp;

}

result\_h\_t = result\_h\_t + array[array.length-1];

}

}

}else{

result\_h\_t = result\_h;

}

snjs\_leftContent.htmlText = result\_h\_t;

continue;

}

//end 处理内容

//处理字体

if("font\_type" == pros[z].attributes.name){

format.font = pros[z].firstChild.nodeValue;

continue;

}

//end 处理字体

//处理对齐方式

if("font\_align" == pros[z].attributes.name){

format.align = pros[z].firstChild.nodeValue;

continue;

}

//end 处理对齐方式

//处理文字大小

if("font\_size" == pros[z].attributes.name){

format.size = Number(pros[z].firstChild.nodeValue);

continue;

}

//end 处理文字大小

//处理文字颜色

if(changeType == "all"){

if("font\_color" == pros[z].attributes.name){

format.color = pros[z].firstChild.nodeValue;

continue;

}

}

//end 处理文字颜色

//处理文本框高度

if("height" == pros[z].attributes.name){

if(int(pros[z].firstChild.nodeValue)>0){

snjs\_leftContent.\_height = Number(pros[z].firstChild.nodeValue);

}

continue;

}

//end 处理文本框高度

//处理文本框宽度

if("width" == pros[z].attributes.name){

if(int(pros[z].firstChild.nodeValue)>0){

snjs\_leftContent.\_width = Number(pros[z].firstChild.nodeValue);

}

continue;

}

//end 处理文本框宽度

//处理文本框横坐标位置

if("x\_position" == pros[z].attributes.name){

snjs\_leftContent.\_x = Number(pros[z].firstChild.nodeValue);

continue;

}

//end 处理文本框横坐标位置

//处理文本框纵坐标位置

if("y\_position" == pros[z].attributes.name){

snjs\_leftContent.\_y = Number(pros[z].firstChild.nodeValue);

continue;

}

//end 处理文本框纵坐标位置

//处理文本行间距

if("font\_leading" == pros[z].attributes.name){

if(int(pros[z].firstChild.nodeValue)>0){

format.leading = Number(pros[z].firstChild.nodeValue);

}

continue;

}

//end 处理处理文本行间距

//处理文本字间距

if("font\_letterSpacing" == pros[z].attributes.name){

if(int(pros[z].firstChild.nodeValue)>0){

format.letterSpacing = Number(pros[z].firstChild.nodeValue);

}

continue;

}

//end 处理处理文本字间距

//处理文本行是否为粗体

if("font\_bold" == pros[z].attributes.name){

if("true" == pros[z].firstChild.nodeValue){

format.bold = true;

}

continue;

}

//end 处理文本行是否为斜体

//处理文本是否为粗体

if("font\_italic" == pros[z].attributes.name){

if("true" == pros[z].firstChild.nodeValue){

format.italic = true;

}

continue;

}

//end 处理文本是否为斜体

//处理文本的左边距

if("font\_leftMargin" == pros[z].attributes.name){

if(int(pros[z].firstChild.nodeValue)>0){

format.leftMargin = Number(pros[z].firstChild.nodeValue);

}

continue;

}

//end 处理文本的右边距

if("font\_rightMargin" == pros[z].attributes.name){

if(int(pros[z].firstChild.nodeValue)>0){

format.rightMargin = Number(pros[z].firstChild.nodeValue);

}

continue;

}

//end 处理文本的右边距

//end 处理文本的首行缩进

if("font\_indent" == pros[z].attributes.name){

if(int(pros[z].firstChild.nodeValue)>0){

format.indent = Number(pros[z].firstChild.nodeValue);

}

continue;

}

//end 处理文本的首行缩进

//处理文本框是否显示

if("border" == pros[z].attributes.name){

if("true" == pros[z].firstChild.nodeValue){

snjs\_leftContent.border = true;

}

continue;

}

//end 处理文本框是否显示

}

} //end if

}

snjs\_leftContent.setTextFormat(format);

break;

}

}

}

};

xml.load("properties.xml");

#### .......合同新建状态，合同列表文档-------代美佳------2013.8.20..........

#### 合同新建状态

1.说明

contractObj:newflag

前台参数：newcontractflag

数据库: xmis.con\_合同 新建标志

新建标志==1,新合同 新建标志 ==2 ,旧合同.首页合同状态列表均为新合同

2.文件路径：

前台： WebRoot/resource/js/ContractState.js

WebRoot/extPortal/bitcMain.js

后台：（主要控制在合同保存时）

com.bitc.jk.xmis.web.action.contractAction.java--method:saveContractInfoNew

3.代码说明

(1).收入合同：

新创建合同，newflag=1

如果合同的新建标志位==1，判断当前使用人（登录人）是否属于该合同的乙方部门，如果等于那么newflag ==2

**int** newflag = 0;

**if** (request.getParameter("newcontractflag") == **null**

|| request.getParameter("newcontractflag") == "") {

newflag = 1;

}

**if** (Type.*GetInt*(request.getParameter("newcontractflag")) == 1) {

**if** (contractService.isEmpBelongDept(

Type.*GetInt*(request.getParameter("bDept\_id")),

userObj.getUserID())) {

newflag = 2;

} **else** {

newflag = 1;

}

}

**if** (Type.*GetInt*(request.getParameter("newcontractflag")) == 2) {

newflag = 2;

}

(2).支出合同

新创建合同，newflag ==1

合同金额分配结束（在创建新合同时，默认是分配结束的），当前登录人属于该合同的事业部，或者当前登录人有维护该事业部合同的权限(xmis.con\_人员部门特殊情况对照表)，此时newflag==2

**int** newflag = 0;

**if** (request.getParameter("newcontractflag") == **null**

|| request.getParameter("newcontractflag") == "") { newflag = 1;

} **else** {

newflag = Type.*GetInt*(request

.getParameter("newcontractflag"));

**if** (Type.*GetInt*(request.getParameter("newcontractflag")) == 0) {

**if** (Type.*GetDouble*(contractService.getAmtByConID(

con.getContractID(), 2)) == Type.*GetDouble*(con

.getContractAmt())) {

newflag = 1;

}

**if** (Type.*GetDouble*(contractService.getAmtByConID(

con.getContractID(), 2)) == Type.*GetDouble*(con

.getContractAmt())

&& (contractService.isEmpBelongDept(Type

.*GetInt*(request.getParameter("aDept\_id")),

userObj.getUserID()) || contractService

.isEmpDeptSpecial(Type.*GetInt*(request

.getParameter("aDept\_id")),

userObj.getUserID()))) {

newflag = 2;

}

}

**if** (Type.*GetInt*(request.getParameter("newcontractflag")) == 1) {

**if** (contractService.isEmpBelongDept(

Type.*GetInt*(request.getParameter("aDept\_id")),

userObj.getUserID())

|| contractService.isEmpDeptSpecial(Type

.*GetInt*(request.getParameter("aDept\_id")),

userObj.getUserID())) {

newflag = 2;

}

}

}

#### 合同列表整合

1.说明

合同列表整合是把原有的合同业务查询和合同综合查询合并。以业务查询为主，将综合查询中的高级查询整合至业务查询中，并设定在不同查询条件、不同权限下显示不同的列。

并将原有高级查询中的收付款情况拆分为：收付款情况、发票情况、预计来款。

2.文件路径：

前台：

WebRoot/contract/contractList.js contractList.jsp

后台：

com.bitc.jk.xmis.web.action.ContractAction.java

com.bitc.jk.xmis.service.ContractService.java

com.bitc.jk.xmis.service.impl.ContractServiceImpl.java

com.bitc.jk.xmis.dao.ContractDAO.java

com.bitc.jk.xmis.dao.impl.ContractDAOImpl.java

SQL:

com.bitc.jk.xmis.util.SQLScript.java

3.权限设定：

(1).deptLimitsFlag : 是否关心业务数据，用户所在部门访问业务数据的权限

后台：LoginServiceImpl.java--method:doUserValidete

OrganizationDAOImpl.java--method:getDeptOfUserLimit

(2).decisionFlag : 查看支出合同金额的权限 emp.hasDecisionRole()

(3).contractOpType : 对合同的操作类型 contractOpType==5时有编辑权限

合同列表中显示当前登录人可查看的所有合同（包括只读和可编辑），在进入指定合同时，再判断当前登录人对此合同的contractOpType权限。没有权限的将toolbar隐藏，只能查看，不可编辑。

对于合同列表中可通过财务对账差异、未收合同款登记、合同欠款等连接快速查看的合同款收付登记、付款计划、付款推迟原因登记等信息窗口也要判断当前登录人针对合同是否有contractOpType权限，没有权限的将toolbar隐藏，只有只读权限，不可编辑。

3.查询条件说明：

(1).收入合同：

基本查询和高级查询均可和公共查询条件进行组合查询。

公共查询条件：合同来源查询、部门查询(事业部)、执行部门查询、模糊查询，其中执行部 门查询当部门查询选择的事业部存在二级部门时可选。

基本查询：合同编号、合同简称、签订日期、项目编码、甲方公司、质保期止、收款条件金 额、项目负责人、乙方部门、乙方代表、执行部门

高级查询 ：签订合同、质保期到期、收付款情况、发票情况、预计来款、财务到款情况， 其中收付款情况、发票情况、预计来款是业务数据，有deptLimitsFlag权限控制

(2).支出合同：

基本查询和高级查询均可和公共查询条件进行组合查询

公共查询条件：部门查询、模糊查询

基本查询：合同编号、合同简称、签订日期、项目编码、乙方公司、质保期止、收款条件金 额、项目负责人、甲方部门、甲方代表

高级查询： 签订日期、质保期到期、收付款情况、发票情况，其中收付款情况、发票情况 是业务数据，有deptLimitsFlag权限控制。

#### .......合同列表查询条件说明文档-------代美佳------2013.9.5..........

合同查询列表将基本查询、高级查询合并为组合查询，不再分页显示，放在一个面板上。

变动部分：

1.收入合同的基本查询条件去掉签订日期、质保期止、乙方部门、执行部门，项目负责人更名为项目经理；支出合同去掉签订日期、质保期止、收款条件金额、甲方部门、项目编码，将项目负责人更名为合同负责人。这些均为数据库操作。

2.组合查询的基本查询条件可以和高级查询条件组合，为且关系。

3.收入、支出均在组合查询面板上加一个条件重置按钮，此按钮功能为重置组合查询面板上的条件（公共查询条件不重置），并将当前查询状态置于基本查询。

其他说明：

1.组合查询右侧的签订的合同、质保期到期、收付款情况、发票情况、预计来款、财务到款情况（高级查询）为互斥关系。组合查询左侧chebox列的条件（基本查询）可以组合，为且关系。

2.所有公共查询条件（即：收入合同包括合同来源查询、部门查询(事业部)、执行部门查询、模糊查询，支出合同包括部门查询、模糊查询）在变更时触发load，在load之前需要确定当前store，即判断当前处于基本查询或高级查询。

3.在组合查询（包括基本查询、高级查询）查询前需要加入合同来源、部门、执行部门、模糊查询参数。