

概述

2015年9月16日 17:50

Pbuilder(project builder)是一个快速搭建项目提高工作效率的工具，它的原理是提取数据库的表结构信息生成哪些仅仅需要表结构信息就能生成的文件，如JavaBean，mybatis/ibatis配置文件，甚至是通用的DAO，Service，Controller，Javascript，JSP文件。重复而乏味的工作交给工具完成，人则解决灵活、多变、复杂的问题。

多语言环境

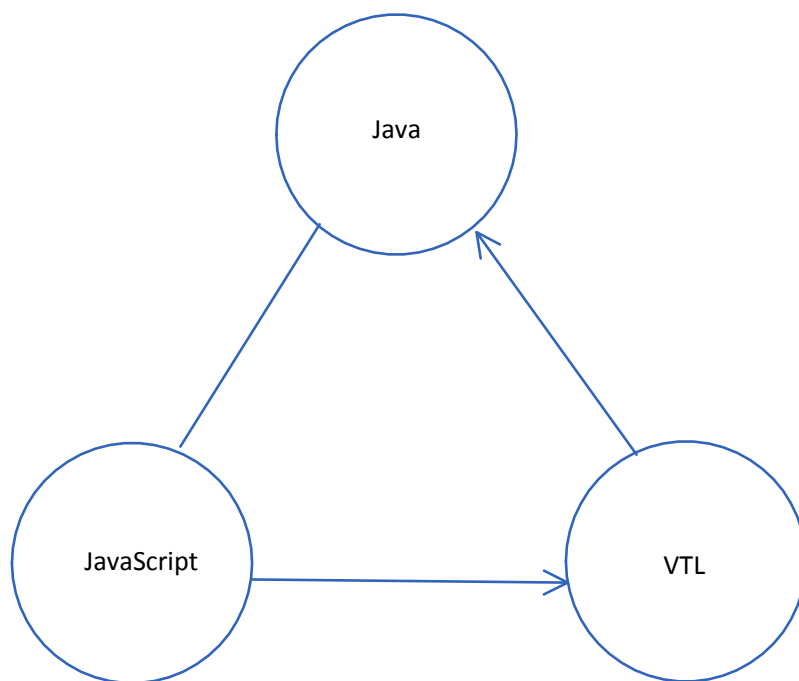
2015年9月16日 17:51

Pbuilder5.0之前的版本:



Pbuilder5.0新增了JavaScript支持:

1. Java和Javascript之间允许相互调用
2. 可以在Velocity Template Language里调用Javascript



快速开始

2015年9月16日 18:37

现在有一个部门表 PA_DEPT

Name	Type
▶ ID	NUMBER(28)
PDEPT_ID	NUMBER(28)
DEPT_NAME	VARCHAR2(255)
SORT_ID	NUMBER(28)
VERSION	NUMBER(28)
DEPT_TYPE	NUMBER(28)
STATUS	NUMBER(12)
TYPE	VARCHAR2(255)
CREATOR_ID	NUMBER(12)
CREATOR_NAME	VARCHAR2(255)
CREATE_TIME	DATE
UPDATE_TIME	DATE

目标：提取表信息生成JavaBean

1. 建立模板包，配置文件与模板文件

```
└─ templates
   └─ first
      ├── ${bean.name}.java.vm
      └── template.vm.properties
```

其中templates文件夹是pbuilder默认存放模板包的文件夹

模板包：first

配置文件：template.vm.properties

```
1 pbuilder.template.enable=true
2
3 solution=oracle
4 jdbc.url=jdbc:oracle:thin:@192.168.88.249:1521:afocus
5 jdbc.username=pauser
6 jdbc.password=pa123
7
8 task.tables=PA_DEPT
```

模板文件：\${bean.name}.java.vm

```
1 public class ${bean.name}{
2   #foreach($field in $bean)
3     private ${field.type} ${field.name};
4   #end
5   #foreach($field in $bean)
6     public ${field.type} ${field.getMethod}() {
7       return ${field.name};
8     }
9     public void ${field.setMethod}(${field.type} ${field.name}) {
10      this.${field.name} = ${field.name};
11    }
12  #end
13 }
```

2. 运行 com.afocus.pbuilder.Application

3.运行结果

└─ generate
 └─ first
 └─ PaDept.java

PaDept.java

```
1 public class PaDept{  
2     private Long id;  
3     private Long pdeptId;  
4     private String deptName;  
5     private Long sortId;  
6     private Long version;  
7     private Long deptType;  
8     private Long status;  
9     private String type;  
10    private Long creatorId;  
11    private String creatorName;  
12    private Date createTime;  
13    private Date updateTime;  
14    public Long getId() {  
15        return id;  
16    }  
17    public void setId(Long id) {  
18        this.id = id;  
19    }  
20    public Long getPdeptId() {  
21        return pdeptId;  
22    }  
23    public void setPdeptId(Long pdeptId) {
```

成功生成！你或许会不太满意，因为生成的是PaDept.java。如果你想要生成Dept.java，那么就是Javascript做贡献的时候了。

在模板包的根目录下添加一个文件：script.vm.js

```
1 on("transformer", function(event) {  
2     if (event.to instanceof Bean) {  
3         if (event.from instanceof UserTables) {  
4             event.to.name = StringUtils  
5                 .toCamelCase(event.from.tableName, 1, true);  
6         }  
7     }  
8 });
```

重新运行，生成结果如下：

└─ generate
 └─ first
 └─ Dept.java

实际上我们使用Javascript替换了pbuilder的默认数据转换规则。先不要着急与理解以上代码的含义，下文会进行详细解释。

数据结构

2015年9月16日 19:57

上层数据结构：

Column

name	String	列名
type	String	列类型
comment	String	列注释
alias	String	列别名
isNullable	Boolean	是否可空
tableName	String	所在表名
precision	Long	数字类型字段的数据长度
scale	Long	数字类型字段的小数长度
jdbcType	String	对应的mybatis jdbcType
original	Object	原生列对象的引用

Table

name	String	表名
type	String	类型：TABLE表示表，VIEW表示视图
comment	String	表注释
alias	String	表别名
schema	String	数据库名/表空间名
original	Object	原生表对象的引用

同时Table继承了List<Column>

Field

name	String	字段名,如：creatorId
type	String	字段类型，如：Integer
comment	String	字段注释
getMethod	String	getMethod方法名
setMethod	String	setMethod方法名
fullType	String	类型全名，如：java.util.Date
jdbcType	String	对应的mybatis jdbcType
original	Object	原生列对象的引用

Bean

name	String	JavaBean名称，如：CommunityUsers
comment	String	JavaBean注释
alias	String	JavaBean别名，如：communityUsers
original	Object	原生表对象的引用

同时Bean继承了List<Field>

原生数据结构:

1.Mysql

Columns

tableCatalog	String	
tableSchema	String	
tableName	String	
columnName	String	
ordinalPosition	Long	
columnDefault	String	
isNullable	String	
dataType	String	
characterMaximumLength	Long	
characterOctetLength	Long	
numericPrecision	Long	
numericScale	Long	
datetimePrecision	Long	
characterSetName	String	
collationName	String	
columnType	String	
columnKey	String	
extra	String	
privileges	String	
columnComment	String	

Tables

tableCatalog	String	
tableSchema	String	
tableName	String	
tableType	String	
engine	String	
version	Long	
rowFormat	String	
tableRows	Long	
avgRowLength	Long	
dataLength	Long	
maxDataLength	Long	
indexLength	Long	
dataFree	Long	
autoIncrement	Long	
createTime	Date	
updateTime	Date	
checkTime	Date	
tableCollation	String	

checksum	Long	
createOptions	String	
tableComment	String	
columnsList	List<Columns>	

2.Oracle

UserTabColumns

tableName	String	Table, view or cluster name
columnName	String	Column name
dataType	String	Datatype of the column
dataTypeMod	String	Datatype modifier of the column
dataTypeOwner	String	Owner of the datatype of the column
dataLength	Integer	Length of the column in bytes
dataPrecision	Integer	Length: decimal digits (NUMBER) or binary digits (FLOAT)
dataScale	Integer	Digits to right of decimal point in a number
nullable	String	Does column allow NULL values?
columnId	Integer	Sequence number of the column as created
defaultLength	Integer	Length of default value for the column
dataDefault	String	Default value for the column
numDistinct	Integer	The number of distinct values in the column
lowValue	String	The low value in the column
highValue	String	The high value in the column
density	Double	The density of the column
numNulls	Integer	The number of nulls in the column
numBuckets	Integer	The number of buckets in histogram for the column
lastAnalyzed	Date	The date of the most recent time this column was analyzed
sampleSize	Integer	The sample size used in analyzing this column
characterSetName	String	Character set name
charColDeclLength	Integer	Declaration length of character type column
globalStats	String	Are the statistics calculated without merging underlying partitions?
userStats	String	Were the statistics entered directly by the user?
avgColLen	Integer	The average length of the column in bytes
charLength	Integer	The maximum length of the column in characters
charUsed	String	C is maximum length given in characters, B if in bytes
v80FmtImage	String	Is column data in 8.0 image format?
dataUpgraded	String	Has column data been upgraded to the latest type version format?
histogram	String	
comments	String	The comments of the column

UserTables

tableName	String	Name of the table
tableName	String	Name of the tablespace containing the table
clusterName	String	Name of the cluster, if any, to which the table belongs

iotName	String	Name of the index-only table, if any, to which the overflow or mapping table entry belongs
status	String	Status of the table will be UNUSABLE if a previous DROP TABLE operation failed, VALID otherwise
pctFree	Integer	Minimum percentage of free space in a block
pctUsed	Integer	Minimum percentage of used space in a block
iniTrans	Integer	Initial number of transactions
maxTrans	Integer	Maximum number of transactions
initialExtent	Integer	Size of the initial extent in bytes
nextExtent	Integer	Size of secondary extents in bytes
minExtents	Integer	Minimum number of extents allowed in the segment
maxExtents	Integer	Maximum number of extents allowed in the segment
pctIncrease	Integer	Percentage increase in extent size
freelists	Integer	Number of process freelists allocated in this segment
logging	String	Logging attribute
backedUp	String	Has table been backed up since last modification?
numRows	Integer	The number of rows in the table
blocks	Integer	The number of used blocks in the table
emptyBlocks	Integer	The number of empty (never used) blocks in the table
avgSpace	Integer	The average available free space in the table
chainCnt	Integer	The number of chained rows in the table
avgRowLen	Integer	The average row length, including row overhead
avgSpaceFreelistBlocks	Integer	The average freespace of all blocks on a freelist
numFreelistBlocks	Integer	The number of blocks on the freelist
degree	String	The number of threads per instance for scanning the table
instances	String	The number of instances across which the table is to be scanned
cache	String	Whether the table is to be cached in the buffer cache
tableLock	String	Whether table locking is enabled or disabled
sampleSize	Integer	The sample size used in analyzing this table
lastAnalyzed	Date	The date of the most recent time this table was analyzed
partitioned	String	Is this table partitioned? YES or NO
iotType	String	If index-only table, then IOT_TYPE is IOT or IOT_OVERFLOW or IOT_MAPPING else NULL
temporary	String	Can the current session only see data that it place in this object itself?
secondary	String	Is this table object created as part of icreate for domain indexes?
nested	String	Is the table a nested table?
bufferPool	String	The default buffer pool to be used for table blocks
rowMovement	String	Whether partitioned row movement is enabled or disabled
globalStats	String	Are the statistics calculated without merging underlying partitions?
userStats	String	Were the statistics entered directly by the user?

duration	String	If temporary table, then duration is sys\$session or sys\$transaction else NULL
skipCorrupt	String	Whether skip corrupt blocks is enabled or disabled
monitoring	String	Should we keep track of the amount of modification?
clusterOwner	String	Owner of the cluster, if any, to which the table belongs
dependencies	String	Should we keep track of row level dependencies?
compression	String	Whether table compression is enabled or not
dropped	String	Whether table is dropped and is in Recycle Bin
comments	String	Comments of the table
userTabColumnsList	List<UserTabColumns>	Columns of the table

UserViews

viewName	String	Name of the view
textLength	Integer	Length of the view text
text	String	View text
typeTextLength	Integer	Length of the type clause of the object view
typeText	String	Type clause of the object view
oidTextLength	Integer	Length of the WITH OBJECT OID clause of the object view
oidText	String	WITH OBJECT OID clause of the object view
viewTypeOwner	String	Owner of the type of the view if the view is a object view
viewType	String	Type of the view if the view is a object view
superviewName	String	Name of the superview, if view is a subview
comments	String	Comments of the view
userTabColumnsList	List<UserTabColumns>	Columns of the view

属性配置

2015年9月16日 20:36

凡是模板包根目录下文件后缀为“.vm.properties”的文件都会被解析为pbuilder配置文件

属性名	是否可空	默认值	备注
pbuilder.template.enable	可空	false	true表示启用，false表示未启用，扫描模板包时只会扫描一起用的模板包
solution	可空	oracle	数据来源解决方案，目前支持两种：oracle和mysql
jdbc.url	不可空		数据库地址
jdbc.username	不可空		数据库用户名
jdbc.password	不可空		数据库密码
task.tables	可空		需要使用的表名，支持同时配置多个表，可以使用逗号隔开；支持使用%_模糊匹配
pbuilder.template.export	可空	generate	目标文件夹路径，生成结果将会存入这个文件夹中
pbuilder.template.excludes	可空		需要排除在外的文件列表，使用逗号隔开
pbuilder.template.excludes.common	可空	false	是否排除普通文件，为false时会复制普通文件到目标文件夹，为true是会跳过普通文件

除此之外，还可以在配置文件中配置上下文。比如：context.project=pa，这意味着在模板中访问“project”变量的时候会得到“pa”，即：\${project}生成结果为pa。

模板脚本

2015年9月17日 18:06

凡是模板包根目录下文件后缀为“.vm.js”的文件都会被解析为pbuilder脚本文件，当模板运行时执行编写的脚本；另外，在VTL中可以使用#javascript命令嵌入Javascript。

Pbuilder使用了Nashorn Javascript方言，这种方言实现了除了Dom Api之外的所有EcmaScript5.1特性，并且添加了调用Java类和操作Java对象的功能。

操作Java对象示例：

```
var HashSet = Java.type("java.util.HashSet");
var set = new HashSet();
set.add(1);
set.add('a');
print(set);
>>[1, a]
```

事件

事件监听方法：

on(eventType, callback);

eventType：事件类型

callback：回调函数

Pbuilder采用的是同步事件，即事件触发后等待所有的监听器执行完毕后会继续往下执行。

事件类型	事件对象	备注
init	InitEvent	初始化事件，这个事件在模板的配置与上下文初始化后触发，此时上下文中没有表信息，并且Spring尚未初始化。模板编写人员可以在监听本事件后修改配置和上下文。
transformer	TransformerEvent	数据转换事件，当原生对象转换成Bean，Field，Table，Column这四种对象时会触发。模板编写人员可以监听本事件后更改相应的数据转化规则

InitEvent

properties	Properties	模板配置
velocityContext	Context	模板上下文

TransformerEvent<F,T>

from	F	来源对象，Mysql有：Tables，Columns；Oracle有：UserTabColumns，UserTables，UserViews，UserTabOrViews
to	T	转化结果，Table，Column，Bean，Field之一

内置Java类型

类型名	Java类名
Table	com.afocus.pbuilder.common.context.Table
Column	com.afocus.pbuilder.common.context.Column
Bean	com.afocus.pbuilder.common.context.Bean
Field	com.afocus.pbuilder.common.context.Field
Tables	com.afocus.pbuilder.mysql.model.Tables
Columns	com.afocus.pbuilder.mysql.model.Columns

UserTables	com.afocus.pbuilder.oracle.model.UserTables
UserViews	com.afocus.pbuilder.oracle.model.UserViews
UserTabOrViews	com.afocus.pbuilder.oracle.model.UserTabOrViews
UserTabColumns	com.afocus.pbuilder.oracle.model.UserTabColumns

再来看这一段代码

```
1= on("transformer", function(event) {
2=     if (event.to instanceof Bean) {
3=         if (event.from instanceof UserTables) {
4             event.to.name = StringUtils
5                 .toCamelCase(event.from.tableName, 1, true);
6         }
7     }
8 });
```

它的含义是：在类型UserTables的对象转换为类型为Bean的对象后，把转换后对象的name属性使用驼峰命名法命名，忽略系统前缀，首字母大写。

Velocity是一个基于Java的模板引擎（template engine）。它允许任何人仅仅使用简单的模板语言（template language）来引用Java代码定义的对象。VTL即Velocity模板语言，是Velocity Template Language的缩写。如果你从未了解过VTL，请先阅读《[Velocity User Guide](#)》，如果你已经具有VTL的基本知识或者曾经使用过VTL，那么接下来介绍几个pbuilder中独有的Velocity命令。

1. #javascript

javascript命令允许VTL中嵌入Javascript代码，使用方式为：

```
#javascript($a, $b, $c)
```

```
//javascript-field, your javascript code
```

```
#end
```

#javascript允许你传入任意个数的参数，在javascript-field中，你可以使用arg0获取第一个参数，arg1获取第二个参数，argn获取第n+1个参数；允许你操作velocity上下文，通过context.put(key, value)更新上下文；允许你调用公用的Javascript函数和当前模板脚本中定义的Javascript函数。

为方便理解，这里用一个实际例子来说明问题：

```
#javascript(1, 2)
```

```
context.put("sum", arg0 + arg1);
```

```
#end。
```

```
$sum
```

当Velocity模板引擎执行到#javascript的时候实际上执行的代码是：

```
(function(arg0, arg1, context){
```

```
context.put("sum", arg0 + arg1);
```

```
})(1, 2, $velocityContext);
```

当执行到\$sum的时候得到输出结果：3

其中\$velocityContext是代表Velocity上下文的Java对象。

** 在#javascript虽然使用脚本引擎执行javascript代码，但是Velocity模板引擎依然会对其进行语法分析，因此这些javascript代码如果不符合VTL语法将会抛出语法错误，所以在标签中应当尽量避免使用\$, #等符号。*

2. #element

VTL可以严格的控制目标代码的格式（缩进，空格，换行等），然而遇到循环时事情就变得复杂了。设想这样一种情况：有一个字符串列表，要依次输出，使用逗号隔开，并且每行5个元素，每行前面要有四个空格。即：

Input: [a, b, c, d, e, f, g, h, i, j, k, l, m]

Output:

```
a,b,c,d,e,
```

```
f,g,h,i,j,k,
```

```
l,m
```

为达到这个效果，可能会让你头大，因为要在循环中写各种判断，是不是一行的开始？是不是该换行了？是不是最后一个元素？等等。#element就是为了解决循环的格式化问题而诞生的，使用方式为：

```
#foreach($e in $list)
```

```
#element($list.size(), $lineSize, $separator, $linePrefix) [VTL-field] #end
```

```
#end
```

如要实现上面例子的效果，编写这样的代码就解决问题了：

```
#foreach($e in $list)
```

```
#element($list.size(), 5, ",", " ")$e#end
```

```
#end
```

** #element只能直接使用在#foreach中，用在其它地方无效。*

3. #date

#date : 输出当前时间

#date() : 输出当前时间

#date(\$time): 输出\$time所代表的时间

#date(\$time, \$fmt): 输出\$time所代表的时间，用\$fmt字符串格式化

#date(\$fmt): 输出当前时间，使用\$fmt字符串格式化

上下文配置

2015年10月3日 11:27

pbuilder支持自定义上下文配置，如配置系统名，模块名等

方式一 使用配置文件配置

```
context.project=PA
${project}
>>PA
```

方式二 使用脚本文件配置

```
on("init", function(e){
    e.context.put("project", "PA");
});
${project}
>>PA
```

方式三 在模板中配置

```
#set($project = "PA")
${project}
>>PA
```

```
#javascript()
    context.put("project", "PA");
#end
${project}
>>PA
```