Model3D api说明文档

属性说明

columns

描述：列名和列定义相关的信息键值对集合

Default：{}

（注：

1. Model3D实例中有个dataSource内部属性管理实例关联的行数据（对应用开发不可见），model提供setDataSource方法和实例属性mode配置dataSource；

2. 数据源和列信息相互独立，两者一起构成Model3D实例的结构。）

mode

描述：由于描述dataSource(记录数据源) 数据来源特征。

枚举类型：可选值包括 ’Remote’/’Local’

Remote: 本地数据源中不完全包含所有数据，只是和远程数据源保持一个映射关系，必要的时候自动去远程获取。

Local ： 数据全部来自于客户端，数据源不和服务端交互。

Default：’Remote

readOnly

描述：Model是否可编辑，用于说明是否能对数据源中的数据进行修改（不约束columns的变化；

Default：true

editMode

描述：编辑模式（editable为true时才有效），用于控制对应视图(grid)的编辑模式。

枚举类型：可选值包括 ‘CellEditor’（后续添加’RowEditor’等其他编辑模式）

Default：’CellEditor’ (单元格编辑模式)

pageSize

描述：分页大小（-1表示不分页，此时视图不会显示分页条，如果不为pageSiz>0,则视图会根据model的pager属性，在对应viewModel中找有相应class的元素，并实例化为分页条）。

Default：50

pager:'.pager'//默认分页条视图为viewmodel下的 .pager元素,

描述：model对应视图的分页条对应的class

Default：’ pager'

pageIndex

描述：显示的页索引（从0开始）

Default：0

showCheckBox

描述：对应的grid是否显示复选框列。

Default：true

showRowNo

描述：对应的grid是否显示行号.

Default：true

autoWrap

描述：对应的grid是否自动换行显示信息

Default：true

Commands

描述：命令列信息设置，object类型，包含3个属性。

cmds：命令描述对象数组；（每个命令描述对象包含name,text,click,className属性）

pos：说明命令列的显示位置，目前支持在尾部和首部两种显示方式；可选参数，默认‘tail’（尾部显示）。

width:列宽；可选；

isVisible:是否可见，默认true

示例：commands:{

cmds:[{name:’add’,text:’添加’,className:’g-cmd-add’},

{name:’copy’,text:’复制’,click:function(rowIndex){} }]

Width:200,

pos:’header’,

isVisible:false

}

注：Model3D内置了一些命令例如（add，delete）内置命令无需指定click的处理过程。

/\*\*注释里面的属性基本不会需要程序控制，主要用于和视图信息同步

multiSort

描述：对应的grid是否支持多列排序

Default：true

sortFields

描述：排序信息数组元素为一个长度为2的数组，且这个数组的第一个元素为列字段名，第二个元素的值为1或 -1，分别表示升序、降序。

例如 [[‘field1’,1],[‘field2’,-1]] 表示 order by field1 asc ,field2 desc

Default：[]

\*\*/

frozenField

描述：固定列边界对应的字段名，即这个字段前面的字段固定显示

Default：undefined

上面这些属性在创建Model3D实例时可通过构造函数的实参进行配置，

/\*请忽略注释的部分

也可以在实例化后在页面初始化时init\_extend方法中应用服务端的数据前通过set方法逐个设置，或通过setData方法批量设置。

Model3D实例设置好columns属性后，不能在init\_extend

方法外调用Model3D实例的set/get方法直接访问model3D实例属性，而要通过Model3D提供的其他方法对属性进行访问操作。（下面会说明Model3D实例方法集），把Model3D的实例的set/get方法视为内部管理方法，不接受外部调用（为了使用方便，允许在对应页面的init\_extend中使用）

\*/

不支持set/get方法访问Model3D的实例属性，仅提供setData方法批量设置属性，通过提供的其他方法获取相关的属性。（如果发现现有的api不足以实现某些应用或不方便实现，可给我发邮件lizhchc@yonyou.com，到时通过增加相关的方法供调用，避免应用开发写太多重复的逻辑。由于考虑的应用场景不够全面，很可能会缺失一些必要的方法/事件，到时逐步完善）

//使用下列方法设置Model3D的属性和具体行/列/单元格属性

setStata(prop,value)

getState(prop)

setRowState(rowIndex,prop,value)

getRowState(rowIndex,prop)

setColumnState(field,prop,value)

getColumnState(field,prop)

setCellState(rowIndex,field,prop,value)

getCellState(rowIndex,field,prop)

应用开发可使用的方法

setData(data)

功能：批量设置实例属性

Params:

@data:Object

setColumns(columns,fieldNames)

功能：重新设置columns属性

Params:

@ columns:Object

@ fieldNames：可选参数，补充说明列的排序；默认按columns中列的定义次序显示。

setReadOnly(readOnly)

功能：重新表格是否只读，非只读状态，即为可编辑状态，这是不支持分组显示，不支持排序；只读状态下不支持编辑调用。

Params:

@ readOnly:Boolean

getReadOnly()

showColumns/hideColumns(fields)

功能：显示/隐藏指定的列

Params:

@ fields: Array[String]

注：如果只对一列操作时，可直接传递列名。另外命令列的field为$commands

setAutoWrap(autoWrap)

功能：自动换行属性控制

Params:

@ autoWrap: Boolean

getAutoWrap()

setDataSource方法

功能：设置数据源

根据mode属性值的不同，setDataSource时传的参数也不同

当mode值为Local时，调用方式

setDataSource(rows); // rows为一个记录数组

当mode值为Remote时，调用方式

setDataSource (proxy,method,queryParams,pageData,callback)

Params:

@ proxy: Object类型，服务代理对象

@ method:String 服务代理对象

@ queryParams: Object类型， 服务使用的参数，例如多个查询条件

@ pageData: Object类型，一页数据对应的对象，例如

{

currentPageData:[{},{}],

totalCount:20,

pageCount:2,

pageSize:10, pageIndex:0}

@ callback: Function类型，回调方法，显示对应页数据后的回调

pageData,callback都为可选参数，如果为指定pageData，会自动使用查询服务pageServer和queryParams自动去服务器请求一页数据，如果有设置callback数据渲染完后会处理回调。

getTotalCount()

功能：数据源数据记录数（所有用户可查看的数据的数量，不包括用户编辑过程中删除的，包括添加的）

getRow (rowIndex,keepStates)

功能：获取当前页数据中rowIndex对应位置的数据的副本。

Params:

@ rowIndex: Number（从0开始），行索引行

@ keepStates: Boolean, 如果为true，返回的完整的row数据，字段的值不一定是一个值，可能是包括字段状态的对象。

例如，返回

{

name:{‘value’:’aaa’,readOnly:true},

Code:’12341asfasdf’

}

getCurrentPageRows(keepStates) //alias:getRows

功能：获取当前页数据的副本。

Params:

@ rowIndex: Number（从0开始），行索引行

@ keepStates: Boolean 作用同getRow方法的keepStates参数

/\*焦点行控制暂未支持

setFocusedIndex(rowIndex)

功能：把当前页的rowIndex设置为焦点行。

Params:

@ rowIndex: Number（从0开始），行索引行

getFocusedIndex()

功能：获取焦点行索引。

getFocusedRow()

功能：获取焦点行数据副本。

\*/

select(indexs)

功能：选择indexs数组中值表示的索引对应的行。

Params:

@ indexs: Array

类似地有

unselect(indexs)

selectAll()

unselectAll()

getSelectedRows()

功能：选择所有选中的行(支持多页)。

getPageSelectedIndexs()

功能：获取当前显示的行数据中选中的行的索引。

getPageSelectedRows()

功能：获取当前页选中的记录数据

registerFieldEditor(name,Editor)

功能：注册当前实例可使用的编辑器

Params:

@ name: String,编辑器使用时引用的名称

@ Editor: Editor类型（实现约定的接口）

编辑api

getCellValue(rowIndex, fieldName)

功能：获取指定位置指定字段的值。

setCellValue (rowIndex, fieldName, value)

功能：设置指定位置指定字段的值。

insertRow(rowIndex,rowData)

功能：在指定位置插入一行数据。

appendRow(rowData)

功能：在尾部追加一行数据。

updateRow (rowIndex, modifyData,quiet)

功能：更新指定行的数据。

Params:

@ quiet:如果设置true，update过程中不会触发相关的before/after事件；

deleteRows(indexs)

功能：删除若干行。

Params:

@ indexs: Array ，要删除的行的索引

注:也可以 已deleteRows(1,2,3)的形式调用删除2，3，4行

getPkName()

功能：获取主键列字段名。

getTsName()

功能：获取时间戳列字段名。

getPageInfo()

功能：获取分页信息。当model中pageSize选项>0时，分页显示数据，方法返回一个包含 pageIndex,pageSize,totalCount属性的对象；不分页时返回一个表示数据总行数的数值。

事件管理(model通用方法，可使用这些方法管理自定义事件)

on(eventName,fn,context)

un(eventName,fn)

hasEvent(eventName)

execute(eventName)：触发对应的事件

fireEvent(eventName):触发对应的before事件和after事件

发布的事件

clickRow、dblclickRow 单机行，双击行

对应的事件参数格式示例{index:2,row:{name:’aaa’,code:’adfas’}}

BeforeCellEditing

描述：单元格编辑前触发（用于编辑前校验）

Args: {index:2,row:{name:’aaa’,code:’adfas’},field:’name’}

beforeInsertRow

描述：插入一行数据前触发

Args: {index:2,row:{name:’aaa’,code:’adfas’},field:’name’}

afterInsertRow

描述：插入一行数据后触发

Args:null

beforeDeleteRows

描述：插入行数据前触发

Args: rows（一个行数据数组）

beforeCellChange

描述：修改行数据中的字段值前触发

Args: { index:1, field: ‘name’, value: ‘’, oldValue: ‘bbb’}

afterCellChange（和beforeCellChange类似）

附加功能：

1.可编辑控制：

readOnly控制model3d或对应列的数据是否可修改，alwaysReadOnly控制model3d或列的readOnly属性是否可变，如果为true则一直readOnly状态，readOnly状态不能通过setState方法修改。这两个属性都是作用于model中的数据的；列的editable属性控制单元格对应的试图是否可编辑。只读状态下，不能进入编辑状态。

有时候需要根据数据，进行界面的可编辑性控制，model3d中的checks属性内的editable子属性提供定义可编辑控制机制checks.editable属性的值为一个判断函数，函数的参数分别为行索引和字段名，方法返回false时，对应的单元格数据不可进入编辑态。

下面是一个例子：

{

columns:{...},

checks:{

editable:function(rowIndex,field){

if(rowIndex==1)return false;

}

}

}

例子中定义的策略可以使grid中第二行不可编辑。

实现说明：当用户点击单元格将进入编辑状态之前，会根据相关的设置判断是否可编辑。先看Model3d的对应的列的设置，如果没有返回false，在判断checks.editable属性定义可编辑策略，如果没返回false则进入编辑态。

2.单元格编辑态控制

通过checks的customEditor属性定义的方法控制编辑时使用的编辑器信息。

示例：

{

columns:{...},

checks:{

customEditor: function (rowIndex, field) {

if (field === 'crowno') return {

ctrlType: 'Refer',

refCode: "code",

refId: "600015",

refKey: "pk\_org",

refName: "name"

}

}

}

}

}

customEditor方法可以返回一个配置控件的对象。其中ctrlType指定控件类型，其它的为控件实例化使用的参数信息。上面就定义了一个参照编辑器。如果没有返回一个合法的对象，就根据列设置使用默认的编辑器。（目前支持的编辑器类型有 TextBox、NumberBox、ComboBox、DateBox、DataTimeBox、CheckBox、Refer、Password，后续可以根据需要添加，也可以考虑提供类型注册机制）

（注：customEditor中的this并不是Model3D）