行为树使用说明

# 更新日志

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本号 | 更新内容 | 操作人员 |
| 2017/5/3 | V1.0 | 文档建立 | 吕鑫 |

目录

[1. 更新日志 1](#_Toc19868)

[2. 文档目的 3](#_Toc32140)

[3. 编辑器介绍 3](#_Toc16009)

[3.1. 编辑器位置 3](#_Toc18007)

[3.2. 工作区 3](#_Toc22280)

[3.3. 分区 3](#_Toc21334)

[4. 基础操作 4](#_Toc7989)

[4.1. 新建文件夹 4](#_Toc21869)

[4.2. 新建行为树 4](#_Toc4282)

[4.3. 保存编辑器数据 4](#_Toc681)

[4.4. 导出数据 4](#_Toc18508)

[4.5. 上传编辑器操作 5](#_Toc20543)

[4.6. 上传导出数据 6](#_Toc27713)

[5. 节点介绍 6](#_Toc350)

[5.1. 任务节点 6](#_Toc14475)

[5.2. 序列 7](#_Toc22951)

[5.3. 循环 7](#_Toc25567)

[5.4. 空操作 7](#_Toc30789)

[5.5. 选择 7](#_Toc28375)

[5.6. 条件 7](#_Toc18691)

[5.7. 动作 8](#_Toc13639)

[6. 功能介绍 8](#_Toc8913)

[6.1. 主树与子树 8](#_Toc6947)

[6.1.1. 主树 8](#_Toc10124)

[6.1.2. 子树 8](#_Toc14757)

[6.1.3. 关联操作 8](#_Toc22289)

[6.2. 触发条件（子树） 9](#_Toc22164)

[6.2.1. 副本相关 9](#_Toc6115)

[6.2.2. 怪物相关 11](#_Toc32479)

[6.3. 动作 11](#_Toc18914)

[6.3.1. 副本相关 11](#_Toc9955)

[7. 注意事项 15](#_Toc23533)

[7.1. 编辑时 15](#_Toc1562)

[7.1.1. 主树名称 15](#_Toc25444)

[7.1.2. 循环任务勾选 15](#_Toc20740)

[7.1.3. 节点 16](#_Toc27873)

[7.1.4. 设置节点 16](#_Toc19329)

[7.1.5. 预制物体 16](#_Toc14250)

[7.2. 导出时 17](#_Toc16133)

[7.2.1. 特别注意 17](#_Toc22943)

[7.2.2. 导出数据 17](#_Toc31553)

[7.3. 上传时 17](#_Toc30646)

[7.3.1. AI生效 17](#_Toc32056)

# 文档目的

介绍编辑器的使用方法，描述各个常用节点的含义和使用方法。

# 编辑器介绍

使用编辑器前的准备工作：

1. 导出编辑器所在文件夹
2. 打开工作区
3. 选定分区

## 编辑器位置

导出svn的tools文件夹

编辑器位置：svn://172.16.200.154/M1Tools/trunk/AIEdtior/BTDesigner/BehaviacDesigner.exe

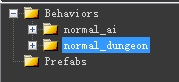
## 工作区

打开【BehaviacDesigner】编辑器后，打开工作区。



工作区位置：M1Tools\AIEdtior\BTDesignerSpace\_AgentProxy\AgentProxy.workspace.xml

## 分区



**副本:** normal\_dungeon（所有副本相关的行为树都放在该文件夹下）

**怪物AI:** normal\_ai（所有怪物AI的行为树都放在该文件夹下）

# 基础操作

## 新建文件夹



每个单位的行为树单元拥有自己的文件夹，在m1data编辑器中将单位和行为树单元关联时需要填写文件夹名称。

## 新建行为树

每个单位文件夹下保存着该单位用到的所有行为树。

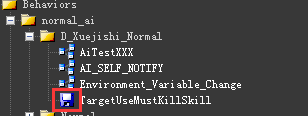
新增功能时，需要新建行为树进行编辑，新建行为树时需要编辑行为树的的类型：副本使用DungeonNormal，竞技场使用DungeonJJC1v1，怪物使用Monster。

快捷命令：ctrl+n

## 保存编辑器数据

编辑器需要保存两份数据，一份是编辑器操作数据，另一份是导出数据（供程序使用）

当进行编辑时，编辑器数据发生变化，编辑器会有保存提示，保存后提示消失。

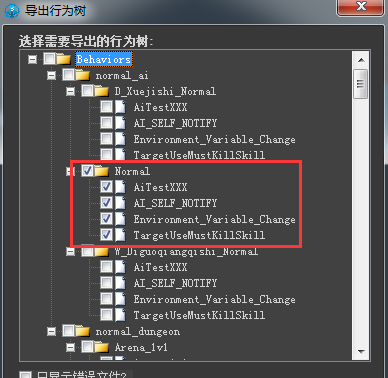


保存快捷命令：ctrl+s

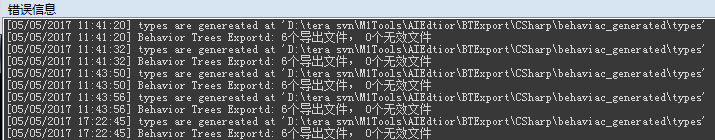
## 导出数据

编辑器数据编写完成后，需要将对应的数据进行导出，供服务器使用。

导出时需要勾选自己修改的行为树，勾选后进行导出。注：导出格式为Xml



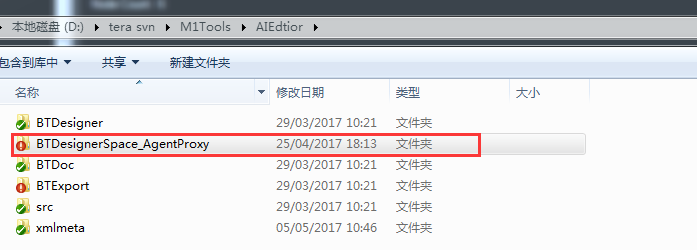
导出快捷公式：ctrl+shift+t

Log：导出后会在【错误信息】中显示导出log，如报错则导出有问题。

## 上传编辑器操作

保存编辑器数据后，如果数据跟本地数据不同则存放编辑器数据的文件夹将会变红，需要上传编辑器操作数据。

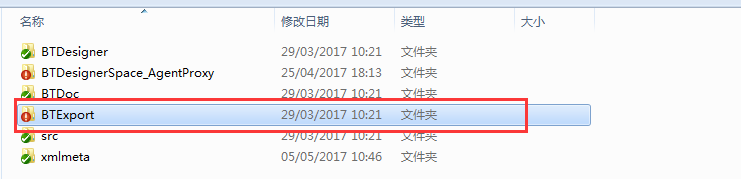
编辑器操作数据位置：



## 上传导出数据

导出数据后，如果导出的数据跟本地数据不符则文件夹会变红，需要上传导出数据。

导出数据文件夹位置：



注：导出时需要在该文件夹下看自己导出是否成功。如你修改了某棵子树并导出，则该文件夹的该子树将会有冲突。通过这个操作来确定自己的导出是否成功。

# 节点介绍

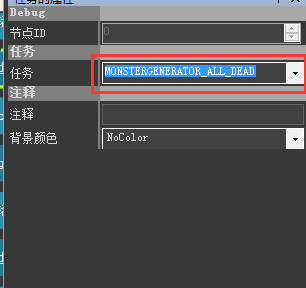
在使用编辑器的过程中，使用到的节点只包括【任务】【序列】【循环】【空操作】【选择】

【动作】【条件】

## 任务节点

位置：【组合节点】-【任务节点】-【任务】

作用：控制子树的触发条件（子树的触发条件是基于行为树的类型，类型为monster的行为树的触发条件就与dungeonnormal的触发条件不同），注：为了方便维护，子树的名称要与任务名称一致。



## 序列

位置：【组合节点】-【序列节点】-【序列】

定义：一种动作的执行方式，该节点以给定的顺序依次执行其子节点，直到所有子节点成功返回，该节点也返回成功。只要其中某个子节点失败，那么该节点也失败。

作用1：按顺序执行动作

作用2：按不同的条件执行该条件打成后的动作集合

## 循环

位置：【装饰器】-【循环】

作用：创建主树时，用于关联子树使用。

## 空操作

位置：【动作】-【空操作】

作用：创建主树时，位于循环之后的动作。

## 选择

位置：【组合】-【选择器】-【选择】

作用：某些触发条件需要根据条件不同产生不同的效果，此时需要用选择节点。

## 条件

位置：【条件】-【叶节点】-【条件】

作用：判断某些参数是否满足

## 动作

位置：【动作】-【动作】

作用：按动作的类型执行对应的功能

# 功能介绍

## 主树与子树

每个行为树单元（文件下）都需要拥有主树与子树。

### 主树

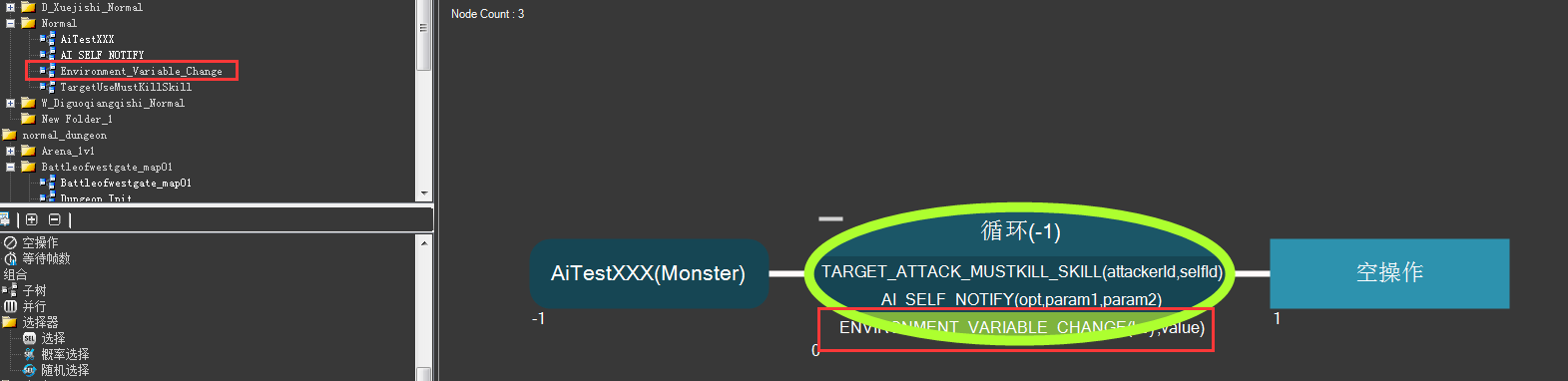
将所有子树关联到一起

### 子树

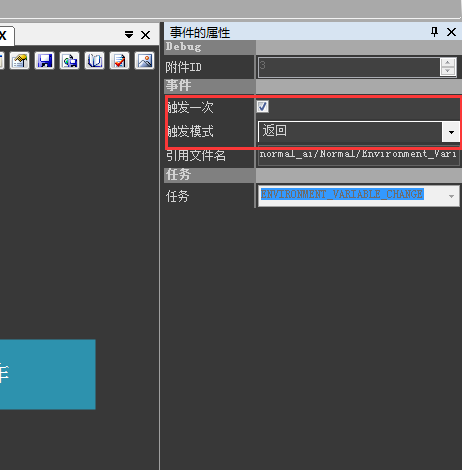
单元的不同节点的触发条件。如：怪物的待机阶段，战斗阶段；副本的玩家进入区域触发等。

### 关联操作

1. 创建主树行为树，创建后需要修改主树名称（副本分区下主树名称需要与文件夹名称一致）。
2. 创建后增加【装饰品】中的【循环】节点，在【循环】节点后增加【空操作】
3. 拖拽子树至【循环】节点中



1. 编辑子树的事件属性，勾选【触发一次】修改触发模式为【返回】



## 触发条件（子树）

### 副本相关

行为树类型为DungeonNormal的任务分类

#### Dungeon\_Init

副本初始化，一般在这个阶段给副本中的元素赋予相应属性

#### DungeoN\_prepare

副本准备阶段，这个阶段是指如竞技场人物进入几秒不能攻击的时间段。

#### DUNGEON\_START

副本开始阶段，是指副本正式开始

#### Dungeon\_reward

副本结算阶段

#### Role\_enter\_map

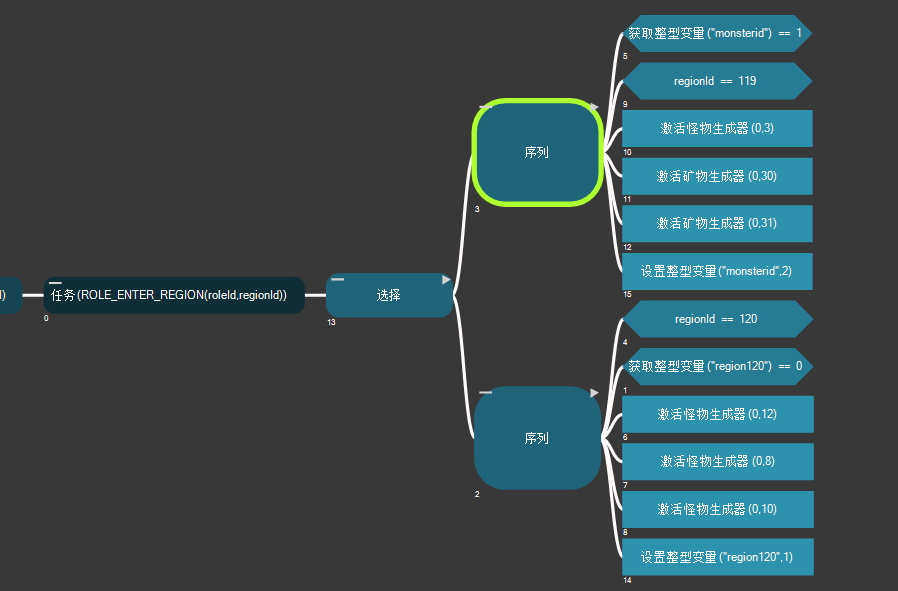
角色进入地图，暂无用

#### Role\_leave\_map

角色退出地图，暂无用

#### Role\_enter\_region

角色进入区域，注意因为同一个场景需要有多个场景，所以该子树的父节点需要使用【选择】节点。

例：

具体逻辑：执行【条件】节点满足的序列，并按照序列从上到下排序的执行动作。

#### Role\_leave\_region

角色离开区域

#### Dead\_monster

怪物死亡，也要使用【选择】节点和【条件】节点

#### Dead\_role

主角死亡

#### Role\_enter\_dungeon

角色进入副本，注意没进入一个角色执行一次。

#### Role\_leave\_dungeon

角色离开副本

#### Dungeon\_timer

副本计时器，用于实时检测副本中怪物某型属性如血量，暂无用

#### Monster\_enter\_dungeon

怪物进入副本，暂无用

#### Role\_safe\_resurrent

主角复活次数达到上限后如何处理，目前直接关联一个动作【离开副本】。

#### Role\_gather

采集矿物，需要使用【选择】和【条件】节点

#### Monstergenerator\_all\_dead

怪物集合死亡，使用该功能前需要在初始化时将怪物生成器的怪物提前编号组。

#### Cg\_eND

Cg结束

#### Entity\_enter\_region

实例进入区域

#### Obstacle\_disappear

实例消失

### 怪物相关

宇萌整理

## 动作

### 副本相关

#### NPC交谈

Npc与场景中的某个特殊的roleid的角色交谈，其他玩家看不见

#### NPC交谈所有人

场景中的所有人

#### State

暂无用

#### 播放cg

给场景中指定玩家或者全部玩家播放cg

#### 单人掉落

指定玩家执行掉落

#### 到达某个时间点删除某个单位

某个生成器生成单位指定时候后移除该单位

#### 地图内传送

传送到地图内指定的目标点

#### 执行所有人掉落

如题

#### Log\_string/log\_string\_float/log\_string\_int

打log播放指定字符，测试功能使用

#### 副本刷怪warring

刷怪警告，测试功能用

#### 改变npc ai

改变所有模板为填写id的npc的ai

#### 改变monster ai

改变所有模板为填写id的monster的ai

#### 改变生成器中实例的ai

改变生成器单元中所有单位的ai

#### 给怪物增加BUFF

根据副本参与人数给怪物增加buff，在副本初始化时新增一个序列配根据人数配5个buff。

#### 给角色增加buff

根据副本参与人数给人物增加buff，在副本初始化时新增一个序列配根据人数配5个buff。

#### 获取怪物血量百分比

获取某一id的怪物的血量百分比

#### 获取环境变量

获取某一场景变量

#### 激活npc/矿物/怪物生成器

激活特定生成器中的实例

#### 激活所有npc/矿物/怪物生成器

激活场景中所有npc/矿物/怪物生成器

#### 将生成器npc/怪物变成巡逻状态

将某个生成器id的中npc/怪物变为巡逻状态，并按照填写的路线id进行巡逻

#### 将生成中的怪物变为被动怪/主动怪

将生成器id中的怪物变成被动怪/主动怪

#### 将生成器中的怪物变成可攻击可选中/不可攻击不可选中

将生成器中的怪物变成可攻击可选中/不可攻击不可选中

#### 将生成器中的npc变为召唤物

用于跟随玩家

#### 离开副本

将玩家踢出副本

#### 离开副本进入巨石要塞

裂隙玩法专属动作

#### 删除entity

移除实例

#### 设置副本准备时间/持续时间/结算时间

设置各个阶段的时间，在副本init时设置

#### 设置生成器实例消失时间

设置某个生成器中所有实例消失的时间，到时间移除。

#### 生成/移除障碍

生成/移除指定id的障碍

#### 是否通过任务激活

用于【条件】节点，通过玩家身上的任务进行判断。

#### 跳转结算

跳转结算界面，执行结算逻辑

#### 无收益杀死副本地图所有怪物

服务器杀死副本所有怪物

#### 增加副本目标（与NPC交谈/增加采集目标/增加杀怪目标/增加进入区域）

如题

#### 增加怪物死亡逻辑组

将某几个生成器中的怪物变成组。

#### 设置整型变量

用于某些特定情况，增加一些变量控制行为树。

例子：如某个区域只有第一次进入会刷新怪物，则在进入前设置整型变量和数量，进入后获取该整型变量并再次设置该整形变量的数量，让其不满足进入条件。

#### 获取整型变量

如上。

# 注意事项

## 编辑时

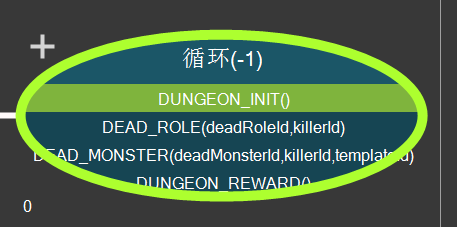
### 主树名称

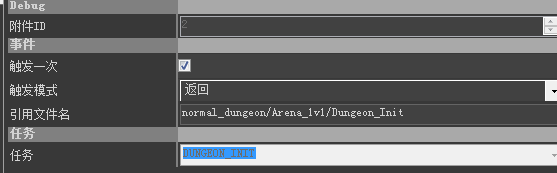
副本主树和文件夹名相同，怪物AI不要求



### 循环任务勾选

任务需要勾选，触发一次，和返回





### 节点

不要是用复杂的父节点。有且只有，序列，选择，条件。



### 设置节点

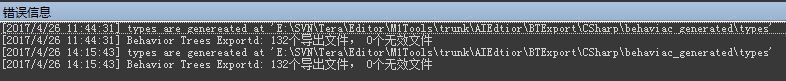
不要手动设置父节点的返回状态。

### 预制物体

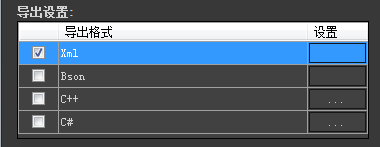
不要使用预制物

## 导出时

### 特别注意

导出时，请使用全不到处。并且尽量导出两次，防止编辑器导出不成功。即使导出提示导出成功后，依然再次导出，防止导出失败。目前编辑器有概率导出不成功。导出两次效果。

### 导出数据



## 上传时

### AI生效

如果是本地，直接copy到 服务器对应bin/debug/Behaviac目录对应文件夹下。

如果是内服：需要上传到svn，然后重启服务器。内服上传时，请保证本地测试通过。