**工具脚本词典**

本文将介绍原版FA2及FA2SP HDM Edition新增脚本函数的参数及使用效果。本作品采用[知识共享署名-非商业性使用-相同方式共享 4.0 国际许可协议](http://creativecommons.org/licenses/by-nc-sa/4.0/)进行许可，转载请注明作者。

作者：Handama

最后修订：2025.2.25

1. **原版函数词典**

Execute：是否执行，如果该变量为true、yes，或数值不为零，则函数执行，否则不执行。Execute变量是可选的，若不填，则始终执行。

除了FA2内置的一些全局变量外，用户自定义的变量并不一定需要添加%%。变量的工作原理是将参数中出现的等于变量名的文本直接替换为变量值。下文中的变量若带%%，则表示该变量的值会被函数改变。

原版函数的更多逻辑细节，可以参考源代码[UserScriptsDlg.cpp](https://github.com/electronicarts/CNC_TS_and_RA2_Mission_Editor/blob/6abf0f557469baea73079c6bf6550709e2e3584e/MissionEditor/UserScriptsDlg.cpp)。

1. **全局变量**

%Width% 地图宽度

%Height% 地图高度

%IsoSize% 地图宽度+地图高度

%WaypointCount% 路径点数量

%UnitCount% 车辆数量

%InfantryCount% 步兵数量

%StructureCount% 建筑数量

%AircraftCount% 飞行器数量

%TerrainCount% 地形对象数量

%Theater% 当前地图类型

%PlayerCount% 单人地图返回1，多人地图返回0~7路径点的计数

%HousesCount% 所属方数量，如果没有[Houses]小节，则返回国家数量

%CountriesCount% 国家数量

%DeleteAllowed% 是否开启AllowDelete，修改这个变量的值并不能修改AllowDelete状态

%AddAllowed% 是否开启AllowAdd，修改这个变量的值并不能修改AllowAdd状态

%SafeMode% 是否开启SafeMode，修改这个变量的值并不能修改SafeMode状态

1. **变量计算类**

Add("%N1%", "N2" , "Execute"); 将%N1%的值加上N2

Multi("%N1%", "N2" , "Execute"); 将%N1%的值乘以N2

Substract("%N1%", "N2" , "Execute"); 将%N1%的值减去N2

Divide("%N1%", "N2" , "Execute"); 将%N1%的值除以N2，N2等于零会报错

Mod("%N1%", "N2" , "Execute"); 将%N1%的值除以N2求余，N2等于零会报错

Is("N1", "Comparator", "N2", "%Result%", "Execute"); 将数字N1与N2进行比较，布尔值结果1或0存储至%Result%。Comparator的可用值有：<、<=、=、>=、>、!=。注意等于是=而不是==

Or("%Result%", "V1" , "V2"… , "VN"); 若V1、V2……VN中有任意一个变量为true、yes，或数值不为零，%Result%为1，否则为0。本函数没有Execute

And("%Result%", "V1" , "V2"… , "VN"); 若V1、V2……VN中有任意一个变量为false、no，或数值为零，%Result%为0，否则为1。本函数没有Execute

Not("%Result%", "Execute"); 若%Result%为true、yes，或数值不为零，%Result%为0，否则为1

SetVariable("%Variable%", "Value" , "Execute"); 将%Variable%的值设置为Value

GetRandom("Result", "Execute"); 随机一个0~32767之间的数，并保存至Result

1. **辅助功能类**

Message("Content", "Title", "Execute"); 弹出一个对话框，内容为Content，标题为Title

SetSafeMode("Enabled", "Message", "Execute"); 如果Enabled为false或no，禁用SafeMode，并弹出对话框进行确认，显示Message的内容。如果Enabled为其他值，则启用SafeMode（默认为启用）

RequiresMP("Execute"); 执行至此时，如果地图不是多人地图，则终止脚本

RequiresSP("Execute"); 执行至此时，如果地图不是单人地图，则终止脚本

Cancel("Execute"); 如果执行，则终止脚本

Print("Message", "Execute"); 将Message打印至工具脚本窗口的报告中

JumpTo("Jump" , "Execute"); 跳转至**:**Jump**:**行（目标行在变量的两端加上冒号）

SetAutoUpdate("Enabled" , "Execute"); 如果Enabled为true、yes或数值不为零，启用AutoUpdate，否则禁用（默认为启用），禁用时，脚本完全执行结束后才会刷新地图显示

AllowAdd("Description" , "Execute"); 会弹出对话框询问是否启用AllowAdd，Description为对话框中的文本（默认为禁用）

AllowDelete("Description" , "Execute"); 会弹出对话框询问是否启用AllowDelete，Description为对话框中的文本（默认为禁用）

1. **文本编辑类**

LowerCase("%Text%", "Execute"); 将%Text%的所有字母变为小写

UpperCase("%Text%", "Execute"); 将%Text%的所有字母变为大写

GetFreeWaypoint("%ID%", "Execute"); 获取最小的可用路径点ID，并保存至%ID%

Insert("%Text%", "InsertText", "Pos", "Execute"); 将InsertText插入到%Text%的第Pos个字符后

Length("%Result%", "Text" , "Execute"); 将Text的字符数量保存至%Result%

Trim("%Text%" , "Execute"); 删除%Text%左右两侧的空格

GetChar("%Result%", "Text", "Pos", "Execute"); 获取Text的第Pos个字符，并保存至%Result%

Replace("%Text%", "OriText", "NewText", "Execute"); 将%Text%中所有OriText替换为NewText

Remove("%Text%", "Pos", "Length", "Execute"); 删除%Text%自Pos起长度为Length的字符

GetParam("%Result%", "Text", "Pos", "Execute"); 将Text基于","拆分，获取第Pos部分，并保存至%Result%中。Pos是从0开始的

SetParam("%Text%", "Pos", "Value", "Execute"); 将%Text%基于","拆分，将第Pos部分替换为Value后，再合并起来，并保存至%Text%中。Pos是从0开始的

GetParamCount("%Result%", "Text" , "Execute"); 将Text基于","拆分，统计拆分的数量，并保存至%Result%中

1. **添加/删除对象类**

AddTrigger("%TriggerID%", "TriggerValue", "ActionValue" , "EventValue", "AddTag" , "Execute"); 添加一个触发，会自动获取可用ID，并将ID输出至%TriggerID%。如果AddTag为false或no，则不添加标签。三个Value为对应INI的值，如"Americans,<none>,Reveal Map Debug Trigger,0,1,1,1,0"，需要启用AllowAdd

AddTag("%TagID%", "TagValue", "Execute"); 添加一个Tag，会自动获取可用ID，并将ID输出至% TagID %，需要启用AllowAdd

AddAITrigger("%AITriggerID%", "AITriggerValue", "Execute"); 添加一个AI触发，会自动获取可用ID，并将ID输出至%AITriggerID%，需要启用AllowAdd

AddInfantry("Value" , "Execute"); 添加一个步兵，Value为对应INI的值，会自动寻找目标单元格可用的subcell位置，不会覆盖已有，**不需要**启用AllowAdd

AddVehicle("Value" , "Execute"); 添加一个车辆，Value为对应INI的值，不会覆盖已有，**不需要**启用AllowAdd

AddAircraft("Value" , "Execute"); 添加一个飞行器，Value为对应INI的值，不会覆盖已有，**不需要**启用AllowAdd

AddStructure("Value" , "Execute"); 添加一个建筑，Value为对应INI的值，不会覆盖已有，**不需要**启用AllowAdd

AddTerrain("TerrainType", "Y", "X" , "Execute"); 在(X, Y)添加一个类型为TerrainType的地形对象，不会覆盖已有的，需要启用AllowAdd

AddSmudge("SmudgeType", "Y", "X" , "Execute"); 在(X, Y)添加一个类型为SmudgeType的污染，不会覆盖已有的，**不需要**启用AllowAdd

DeleteInfantry("Pos" , "Execute"); 删除第Pos个步兵（一般情况下等于ID），需要启用AllowDelete

DeleteAircraft("Pos" , "Execute"); 删除第Pos个飞行器（一般情况下等于ID），需要启用AllowDelete

DeleteStructure("Pos" , "Execute"); 删除第Pos个建筑（一般情况下等于ID），需要启用AllowDelete

DeleteVehicle("Pos" , "Execute"); 删除第Pos个车辆（一般情况下等于ID），需要启用AllowDelete

DeleteTerrain("Pos" , "Execute"); 删除第Pos个地形对象（一般情况下等于ID），需要启用AllowDelete

IsInfantryDeleted("%Result%", "Pos" , "Execute"); 检查第Pos个步兵是否已删除，将布尔值结果1或0保存至%Result%

IsTerrainDeleted("%Result%", "Pos" , "Execute"); 检查第Pos个地形对象是否已删除，将布尔值结果1或0保存至%Result%

SetWaypoint("Waypoint", "Y" , "X" , "Execute"); 将路径点Waypoint移动到(X, Y)，注意参数先填Y再填X，需要禁用SafeMode

Resize("Left", "Top", "Width", "Height", "Execute"); 修改地图大小。Left为旧地图距离新地图左侧的距离，Top为旧地图距离新地图上侧的距离，Width和Height为新的宽度和高度

1. **获取属性类**

GetWaypointPos("ID", "%Y%", "%X%", "Execute"); 获取ID路径点的坐标，并保存至%X%, %Y%，注意参数中先Y后X

GetInfantry("%Result%", "Pos" , "Execute"); 获取第Pos个步兵（一般情况下等于ID）的INI值，并保存至%Result%

GetAircraft("%Result%", "Pos" , "Execute"); 获取第Pos个步兵（一般情况下等于ID）的INI值，并保存至%Result%

GetStructure("%Result%", "Pos" , "Execute"); 获取第Pos个步兵（一般情况下等于ID）的INI值，并保存至%Result%

GetVehicle("%Result%", "Pos" , "Execute"); 获取第Pos个步兵（一般情况下等于ID）的INI值，并保存至%Result%

GetHouse("%Result%", "Pos" , "Execute"); 获取第Pos个所属方的ID，并保存至%Result%。如果没有[Houses]小节，则读取国家小节

GetCountry("%Result%", "Pos" , "Execute"); 获取第Pos个国家的ID，并保存至%Result%

1. **INI编辑类**

SetIniKey("Section", "Key", "Value", "Execute"); 将地图Section小节的Key设置为Value，需要禁用SafeMode

GetIniKey("%Result%", "Section", "Key" , "Execute", ["LoadFrom"]); 读取地图Section小节Key的值，并保存至%Result%。在**FA2SP HDM Edition**中，为这个函数新增了第五个可选参数LoadFrom，可以自定义读取的INI文件，0=FAData，1=Rules，2=Rules+Map（也就是将地图INI合并至Rules的结果），3=Map，4=Art，5=Sound，6=Theme，7=AI+Map，8=EVA, 9=Theater

1. **对话框输入类**

Ask("%Result%", "Content", "Title", "Execute"); 弹出一个确认对话框，将选择结果以布尔值1或0存储至%Result%。对话框内容为Content，标题为Title

AskContinue("Message", "Execute"); 弹出一个确认继续对话框，内容为Message

UInputGetInteger("%Variable%", "Description", "Min", "Max", "Execute"); 弹出一个对话框，要求用户输入一个数字，并将数字保存至%Variable%。Description为对话框中的描述文本，Min和Max为取值范围，均为闭区间

UInputGetString("%Variable%", "Description", "Execute"); 弹出一个对话框，要求用户输入字符串，并将字符串保存至%Variable%。Description为对话框中的描述文本

UInputGetHouse("%Result%", "Description" , "Execute"); 弹出对话框，在对话框中选择一个所属方，并将所属方ID保存至%Result%。Description为对话框中的描述文本。多人地图中效果等同于UInputGetCountry

UInputGetCountry("%Result%", "Description" , "Execute"); 弹出对话框，在对话框中选择一个国家，并将国家ID保存至%Result%。Description为对话框中的描述文本

UInputGetTrigger("%Result%", "Description" , "Execute"); 弹出对话框，在对话框中选择一个触发，并将触发ID保存至%Result%。Description为对话框中的描述文本

UInputGetTag("%Result%", "Description" , "Execute"); 弹出对话框，在对话框中选择一个标签，并将标签ID保存至%Result%。Description为对话框中的描述文本

1. **FA2SP HDM Edition新增函数词典**

RequireFA2SPHE(); 输出“本脚本需要FA2SP HDM Edition才能运行，不要将其拷贝到其他版本中使用。”

GetIniKey("%Result%", "Section", "Key" , "Execute", ["LoadFrom"]); 读取地图Section小节Key的值，并保存至%Result%。在**FA2SP HDM Edition**中，为这个函数新增了第五个可选参数LoadFrom，可以自定义读取的INI文件，0=FAData，1=Rules，2=Rules+Map（也就是将地图INI合并至Rules的结果），3=Map，4=Art，5=Sound，6=Theme，7=AI+Map，8=EVA, 9=Theater

GetAvailableIndex("%ID%" , "Execute"); 获取一个可用的8位ID，并保存至%ID%中

GetAvailableKey("%Result%", "Section", "Execute"); 在地图INI的Section小节中获取第一个可用键，并保存至%Result%中

KeyExists("%Result%", "Section", "Key", "Execute", ["LoadFrom"]); 判断地图INI是否存在Section小节Key键，将布尔值结果1或0存储至%Result%。LoadFrom为自定义读取的INI文件，参数同上

SectionExists("%Result%", "Section", "Execute", ["LoadFrom"]); 判断地图INI是否存在Section小节，将布尔值结果1或0存储至%Result%。LoadFrom为自定义读取的INI文件，参数同上

GetKeyAt("%Result%", "Section", "Pos", "Execute", ["LoadFrom"]); 读取地图INI指定小节第Pos项的键，并将结果保存至%Result%。LoadFrom仅有三个选项，0=Map，1=Rules，2=Rules+Map

GetValueAt("%Result%", "Section", "Pos", "Execute", ["LoadFrom"]); 读取地图INI指定小节第Pos项的值，并将结果保存至%Result%。LoadFrom仅有三个选项，0=Map，1=Rules，2=Rules+Map

RandomBetween("%Result%", "Start", "End", "Execute"); 在[Start, End)范围内随机获取一个整数，并保存至%Result%中

GetTheater("%Result%", "Execute"); 将地图类型索引保存至%Result%中。0=TEMPERATE，1=SNOW，2=URBAN，3=NEWURBAN，4=LUNAR，5=DESERT

DeleteSmudge("Pos", "Execute"); 删除第Pos个污染（一般情况下等于ID），**不需要**启用AllowDelete

SetOverlay("Y", "X", "Overlay", "Execute"); 在(X, Y)单元格设置覆盖图=Overlay（255=清除覆盖图）

SetOverlayData ("Y", "X", "OverlayData", "Execute"); 在(X, Y)单元格设置覆盖图数据= OverlayData

PlaceTileAt("Y", "X", "TileIndex", ["SubTileIndex"], ["AltImage"], "Execute"); 在(X, Y)放置索引为TileIndex的地形，并且可以指定SubTileIndex（默认0）和AltImage（默认随机）。可以使用属性查看功能查看各地形的索引与子索引

IsCoordInMap("%Result%", "Y", "X", "Execute"); 判断(X, Y)坐标是否在地图内，将布尔值结果1或0存储至%Result%

GetCellInfo("%Result%", "Y", "X", "InfoType", "Execute"); 获取(X, Y)单元格的特定信息，并将结果保存至%Result%。InfoType决定信息类型，有效值如下：0=地形索引，1=地形子索引，2=地形AltImage，3=高度，4=地表类型，5=斜坡类型，6=允许放置建筑，7=地形组索引，8=1或4号子单元格步兵ID（-1=无），9=2号子单元格步兵ID，10=3号子单元格步兵ID，11=飞行器ID，12=建筑ID，13=车辆ID，14=地形对象ID，15=污染ID，16=路径点ID，17=覆盖图，18=覆盖图数据，19=是否被多选，20=是否被隐藏

SetHeightAt("Y", "X", "Height", ["Absolute"], "Execute"); 设置(X, Y)单元格的高度。Absolute若为true、yes，或数值不为零，将单元格直接设置为Height，否则在原有高度的基础上加Height，默认为true

StringEquals("%Result%", "StringA", "StringB", "Execute"); 比较字符串StringA和StringB是否相等，将布尔值结果1或0存储至%Result%

UInputSelect("%Result%", "Description" , "Section", "Execute", ["LoadFrom"] , ["ReadKey"], ["ValueAsName"], ["StrictOrder"]); 弹出对话框，在对话框中选择某个INI小节的内容，并将选择项保存至%Result%。Description为对话框中的描述文本。Section为读取的INI小节，LoadFrom为自定义读取的INI文件，参数同上。ReadKey若为true、yes，或数值不为零，则读取键而不是值，默认为false。ValueAsName若为true、yes，或数值不为零，则加载键的值作为选项名称，需要ReadKey=true，默认为false。StrictOrder若为true、yes，或数值不为零，则将加载的INI小节视为注册表，使用自0开始的严格排序，需要ReadKey=true，默认为false

SetVariableInPool("Key", "Value" , "Execute"); 向内置变量池中添加一个变量，名称为Key，值为Value。变量池是一个std::map<CString, CString>，可以存储无限多个变量，如果重复设置相同的Key，新值会替换旧值。变量池会在脚本运行结束时被清空

GetVariableInPool("%Result%", "Key" , "Execute"); 从内置变量池中读取一个变量，名称为Key，并将值保存至%Result%中

WaypointToString("%Result%", "Waypoint" , "Execute"); 将数字形式的Waypoint转换为字母形式，并保存至%Result%中

StringToWaypoint("%Result%", "Waypoint" , "Execute"); 将字母形式的Waypoint转换为数字形式，并保存至%Result%中

HideTileAt("Y", "X", "Mode", "Execute"); 隐藏或显示(X, Y)单元格。Mode为执行模式，0=隐藏，1=显示，2=切换状态

HideTileSet("TileSet", "Mode", "Execute"); 隐藏或显示TileSet地形组。Mode为执行模式，0=隐藏，1=显示，2=切换状态

HideTileIndex("TileIndex", "Mode", "Execute", ["SubTileIndex"]); 隐藏或显示索引为TileIndex的地形。Mode为执行模式，0=隐藏，1=显示，2=切换状态。SubTileIndex为地形子索引，默认为-1（即全部）

MultiSelectTileAt("Y", "X", "Mode", "Execute"); 将(X, Y)单元格加入或移除多选单元格。Mode为执行模式，0=加入，1=移除，2=切换状态

MultiSelectTileSet("TileSet", "Mode", "Execute"); 将TileSet地形组加入或移除多选单元格。Mode为执行模式，0=加入，1=移除，2=切换状态

MultiSelectTileIndex("TileIndex", "Mode", "Execute", ["SubTileIndex"]); 将索引为TileIndex的地形加入或移除多选单元格。Mode为执行模式，0=加入，1=移除，2=切换状态。SubTileIndex为地形子索引，默认为-1（即全部）