

Plugin Scripting Guide

ESAI v4.2

Eric Dobbs

bf2eai@gmail.com

Operator Naming Conventions

ESAI uses alternative Compare operators in most condition definitions. Here is a list of these operators and the corresponding comparisons:

"_"	<i>Difference</i>
"<_"	<i>DifferenceSmaller</i>
">_"	<i>DifferenceGreater</i>
"/"	<i>Quotient</i>
"</"	<i>QuotientSmaller</i>
">/"	<i>QuotientGreater</i>

The quotation marks in the list above are literal - if you use these operators in your scripts, don't forget to include them.

ESAI also names all prerequisites after the strategies they are tied to. It is suggested that plugin authors follow this rule too, as it enhances code readability. For example, the core strategy *basicAttack1* is activated by a prerequisite named *basicAttackPrereq*.

Definition of a Plugin

A plugin is a portion of AI code that creates new conditions, prerequisites, or strategies. Plugins are always loaded after *esaicore.ai*. Plugins are separate entities from Strategy Sets. Plugins create the building blocks of new strategy, while Strategy Sets assign strategies to each team.

Basic Concepts

A plugin can make use of any condition or prerequisite that has already been defined. This includes both the definitions in *esaicore.ai*, as well as content that has been defined by other plugins.

Load order matters. In other words, if you write a plugin that depends on code from a preexisting plugin, you must make sure you load your new plugin **after** you load the plugin your code depends on.

Plugin Rules

The cardinal rule for any plugin that creates new conditions: **after condition creation is finished, you must call the function *aiStrategy.CompleteConditions***. Failure to include this line will likely crash BF2.¹

Every name defined in a new plugin must be a unique name not already defined by the mod's default strategies or by the ESAI package. If you do not follow this rule, ESAI will be unstable and may crash BF2.

Examples

The simplest plugin is one that creates a new strategy based on the ESAI stock prerequisites. Suppose you want to define a new strategy to replace *basicAttack3*. Your plugin file would look similar to this:

```
aiStrategy.createStrategy basicAttack3user
aiStrategy.Agression 1
aiStrategy.NumberOfAttacks 3
aiStrategy.NumberOfDefences 0
aiStrategy.setPrerequisite basicAttackPrereq
```

The first line creates a strategy with a unique name – in this example I just added the word 'user' to the end of the new strategy's name. The last line sets the strategy's prerequisite to *basicAttackPrereq*, which is already defined for you in the ESAI core.

More complex plugins can be written to create new conditions, prerequisites, and strategies. Remember that all these names must be unique. I will use the ESAI default plugin *grabNeutrals.ai* to illustrate the process.

¹ I do not know with 100% certainty that this **must** be done, but until I adopted this rule ESAI was often subject to crash issues. The code is in the package now, the crashes are long gone, so I strongly suggest obeying the rule.

```

rem ~~~~ Check if [number of Neutral SAs] - [number of Owned CPs] >= 2 ~~~~~~
rem :
aiStrategy.createHeterogeneousCondition neutralsAvailable Crisp ">-" Friendly
NumberOfNeutralAreas ControlPoint
aiStrategy.TargetValue 2.0
aiStrategy.setConditionStrength Required

```

aiStrategy.CompleteConditions

```

aiStrategy.createPrerequisite grabNeutralsPrereq
aiStrategy.addCondition neutralsAvailable 50.0

aiStrategy.createStrategy grabNeutrals
aiStrategy.Agression 1
aiStrategy.NumberOfAttacks 6
aiStrategy.NumberOfDefences 0
aiStrategy.setStrategicObjectsModifier ControlPoint 30.0 Neutral
aiStrategy.setStrategicObjectsModifier ControlPoint 4.0 Hostile
aiStrategy.setStrategicObjectsModifier ControlPoint 0.8 Owned
aiStrategy.setPrerequisite grabNeutralsPrereq

```

First, the plugin creates a single condition: *neutralsAvailable*

Second, the plugin calls *aiStrategy.CompleteConditions* , as no other conditions will be defined.

Next, the plugin creates a single prerequisite: *grabNeutralsPrereq*. Notice the weight given to the condition inside this prerequisite, it is needed to for the strategy to activate at reasonably predictable times. This is the reason why:

The ESAI core conditions already cover every possible situation on the battlefield. The condition *neutralsAvailable* can be “true”² at the same time as a condition from the ESAI core. The hefty weight assigned in the prerequisite ensures that 99% of the time, the SAI will choose this strategy instead of one of the ESAI core strategies.

² Conditions in Battlefield strategy scripts are not Boolean. The word *true* here is a simplification.

ESAI Default Conditions

If you load the ESAI core and default plugins, these are the available conditions for use in a user written plugin:

- enemyGainingStrength
- strongAsEnemy
- enemyLosingStrength
- enemyHasNoFlags
- friendlyHasNoFlags
- neutralsAvailable

ESAI Default Prerequisites

If you load the ESAI core and default plugins, these are the available prerequisites for use in a user written plugin:

- basicAttackPrereq
- basicAttackHarderPrereq
- basicAttackHardestPrereq
- counterAttackPrereq
- counterAttackHarderPrereq
- counterAttackHardestPrereq
- pressAttackPrereq
- pressAttackHarderPrereq
- pressAttackHardestPrereq
- grabNeutralsPrereq
- defendMapPrereq
- lastStandPrereq

ESAI User Conditions

ESAI also comes with a library of conditions ready for use in your plugins. These conditions are defined in the file `\ESAI\Plugin\user\userConditions.ai`. If you run this plugin, you have the following conditions available for use in your prerequisites:³

- noNeutralsAvailable
- allAreasSafe
- enemyNearBase
- groundTacticalAdvantage
- groundTacticalDisadvantage
- enemyTicketAdvantage
- friendlyTicketAdvantage
- friendlyHasBase

All extra conditions listed above come in 3 strength variants- Required, Advisory, and AdvisoryPositive. The naming convention is ***conditionName_conditionStrength***. conditionStrength is either R, A, or AP. Example: enemyNearBase with strength Required is named “enemyNearBase_R”.

Putting It All Together

All strategies, be they ESAI defaults or user written, are assigned to one or more Strategy Sets. These sets are defined in the files residing in `\ESAI\Strategy\default\` or `\ESAI\Strategy\user\`.

The strategy set file is the last item you run in your map file. It is responsible for assigning strategies to both teams. If you create a custom strategy then you must also create a new strategy set that includes it. An example is provided below:

```
rem --My Example Strategy Set--

rem Team 1
ai.addSAIStrategy 1 basicAttack2
ai.addSAIStrategy 1 pressAttack2
ai.addSAIStrategy 1 counterAttack3

rem Team2
ai.addSAIStrategy 2 basicAttack2
ai.addSAIStrategy 2 pressAttack2
ai.addSAIStrategy 2 counterAttack3
```

³ Refer to the code comments in *userConditions.ai* for more details on use of these conditions.

The strategy set in the example is very basic, providing only enough strategies to **ensure that there is always a valid strategy** available to the SAI. When you assign strategies to sets, you should make sure you do it in such a way as to fulfill this requirement.

To cover all possible situations on the battlefield, you need only assign each team one strategy from each of the main categories: *basicAttack**, *pressAttack**, and *counterAttack**. After that, customize the new strategy set to your liking.

Your last step is to write a map file to load the components of your script, to wit: The ESAI core, all required plugin files, and finally your strategy set file. I will close with an example map file.⁴

```
1  rem·ESAI·Strategy-·smallMap·with·focusedCounterAttack,·pushForward+
2
3  rem·==·load·Enhanced·SAI·core·==
4  run·/mods/bf2/ESAI/Core/esaicore.ai
5
6  rem·==·load·Enhanced·SAI·default·plugins·==
7  run·/mods/bf2/ESAI/Plugin/default/grabNeutrals.ai
8  run·/mods/bf2/ESAI/Plugin/default/endGame.ai
9
10 rem·==·load·user·plugins·==
11 run·/mods/bf2/ESAI/Plugin/user/userConditions.ai
12 run·/mods/bf2/ESAI/Plugin/user/focusedCounterAttack.ai
13 run·/mods/bf2/ESAI/Plugin/user/pushForward.ai
14
15 rem·==·load·strategy·==
16 run·/mods/bf2/ESAI/Strategy/user/smallFCAPF.ai
```

⁴ If you still need help learning to write strategy scripts based on ESAI, feel free to make inquiries via the project email address. I will make every effort to assist you.