

# FBI tags in TA3D

Roland Brochard

## **Abstract**

This document has been adapted from OTA's FBI document from the Total Annihilation Design Guide

Tags can be of several types:

- integers
- decimal numbers
- booleans (1/0 or true/false)
- strings

When an integer is expect and you provide a decimal number, only the integer part is read.

Angles are expressed in OTA units:

radians → OTA angle	$\times \frac{65536}{2\pi}$
degrees → OTA angle	$\times \frac{65536}{360}$

By default booleans are false if not defined.

## 1 Acceleration

The unit acceleration, can be a decimal number. Expressed in *mapblocs/OTAticks*,  $1mapbloc = 16pixels$ ,  $10OTAticks = 1s$ .

## 2 ActivateWhenBuilt

It's a boolean. When set to true, the unit is ACTIVATED (or ON) when built.

## 3 altfromsealevel

It's an integer. Altitude from sea. For flying units.

## 4 amphibious

It's a boolean. Is the unit amphibious (can go under water, & on land).

## 5 antiweapons

It's a boolean. Set to true if your unit can shoot at weapons of other units (eg. Missile defence systems).

## 6 attackrunlength

It's an integer. For planes. The distance they must travel before being able to drop their bombs/shoot their missiles.

## 7 BMcode

It's a boolean. false for buildings, true for mobile units.

## 8 **BadTargetCategory**

A list of The type of target that is not good for that low accuracy at this type of unit. In general. The value specified identifies all units that have that value specified in their own "Category" settings. BadTargetCategory=VTOL;

## 9 **BrakeRate**

Like acceleration but when the unit wants to slow down. In *mapblocs/OTAtick*,  $1mapbloc = 16pixels$ ,  $10OTAticks = 1s$ .

## 10 **BuildAngle**

It's an integer. It is the maximum angle between the unit's heading and its build target when building. Setting this to 0 means when your unit is building something it'll be facing it (but don't do it, you should at least let a small error tolerance).

## 11 **BuildCostEnergy**

It's an integer. The cost in energy of building this unit. The amount of energy that will be consumed during the building of this unit.

## 12 **BuildCostMetal**

It's an integer. The cost in metal of building this unit. The amount of metal that will be consumed during the building of this unit.

## 13 **BuildTime**

It's an integer. The time it takes to build this unit. On average, 10000 is medium time, things like fusion plants are more like 100000, and krogoth like 300000. If a builder has a WorkerTime flag set to 1, this correspond to the number of seconds it takes to build the unit (See WorkerTime).

## 14 **Builddistance**

This is a decimal number. The distance at which a unit can build if it is a builder, in pixels.

## 15 **Builder**

It's a boolean. True if this unit can build other units.

## 16 **canattack**

It's a boolean. True if this unit can attack.

## **17 CanCapture**

It's a boolean. True if this unit can capture other units.

## **18 CanDgun**

It's a boolean. True if the unit can DGUN (ie. has a gun like the commander)

## **19 Canfly**

It's a boolean. True for flying units.

## **20 canguard**

It's a boolean. True if this unit can guard/assist other units.

## **21 canhover**

It's a boolean. True if this unit can hover.

## **22 canload**

It's a boolean. True if this unit can transport(or load) other units.

## **23 canmove**

It's a boolean. True if this unit can move.

## **24 canpatrol**

It's a boolean. True if this unit can patrol.

## **25 CanReclamate**

It's a boolean. True if this unit can reclaim things.

## **26 canstop**

It's a boolean. True if this unit can stop.

## **27 cantbetransported**

It's a boolean. True if this is a unit transports cannot load eg. Ships, krogoth

## 28 Category

It's a string (a space separated list). A category class. Look at some of the FBI files in totala1.hpi for examples. Essentially, this places the unit in several named categories that can be named in other variables such as "wpri\_badTargetCategory" described below, and in the AI Text files as well.

For instance:

```
Category=ARM TANK LEVEL2 CONSTR NOWEAPON NOTAIR NOT-  
SUB CTRL_B;
```

Some categories have been internally optimized in the engine:

- kamikaze
- notair
- notsub
- jam
- commander
- weapon
- level3

## 29 CloakCost

It's an integer. The energy cost of this unit remaining cloaked when standing still. It is expressed in *energyunits/s*.

## 30 CloakCostMoving

It's an integer. The energy cost of this unit remaining cloaking while moving. It is expressed in *energyunits/s*.

## 31 Commander

This is a boolean. True if the unit is a commander.

## 32 Corpse

This is a string. The name of the corpse of this unit (Defined in the TDF file in the corpses directory).

(12/29/98) From Draconious: The name of the feature the unit turns into when the unit dies. If the unit is a feature (Dragons Teeth) the unit instantly turns into this feature, when it is finished being built. (this is not the 3d model file of the feature, the feature entry tells the 3d model file to use in the feautres OBJECT variable).

## 33 cruisealt

This is an integer. The altitude at which the unit flies.

## 34 DamageModifier

This is a decimal number. This is the part of weapon damages the unit will take. By default it is set to 1.0 (full damage), if set to 0.0 the unit is indestructible, if set to 0.5 it only takes half the damages.

## 35 DefaultMissionType

This is a string. The default orders of this unit for missions. Available missions for this tag are:

- standby (this is the default)
- vtol\_standby
- guard\_nomove
- standby\_mine

## 36 Description

This is a string. The description of the unit seen at the bottom of the screen when you hold the cursor over it in the game.

For instance:

Description=Very Heavy Assault Tank;

## 37 Designation

This is a string. The designation. This is unimportant and can have virtually any value (there may be a limit to the number of character it will except)

## 38 digger

This is a boolean. Does this unit have pieces which extend under ground. I think thats what it means. Things like the pop-up heavy cannon have this = 1.

NB: this is not used by TA3D but OTA uses it.

## 39 EnergyMake

This is an integer. The energy this unit produces when ACTIVE. It can be negative.

## 40 EnergyStorage

This is an integer. The amount added to the maximum amount of energy you can store while the unit is alive.

## 41 EnergyUse

This is an integer. The amount of energy the unit uses while it is ACTIVE. This can be negative.

## 42 ExplodeAs

This is a string. The explosion name of the explosion that will happen if the unit explodes. See totala1.hpi for other examples.

It's something like:

```
ExplodeAs=LARGE_BUILDINGEX;
```

## 43 ExtractsMetal

It's an integer. The rate at which the unit extracts metal. (Yes, the MOHOmine only extracts at like 0.01). This is a factor used in a more complex equation which takes the unit footprint surface into account when computing the metal production rate.

## 44 firestandorders

This is an integer (0, 1 or 2). This is the default value of the firestandorders button of the unit in its build/orders menu

## 45 Floater

This is a boolean. Does this unit float on water.

## 46 FootprintX

It's an integer. The X axis footprint. Should correlate with the unit model's actual size. The footprint helps control how close units can get to one another. This is expressed in map blocs (1 bloc is equivalent to 16 pixels)

## 47 FootprintZ

It's an integer. As with FootprintX, but the Z axis (ie. up and down on the screen while playing TA). Should correlate with the unit model's actual size.

## 48 FrenchDescription

As description, but in French.

## 49 FrenchName

As Name, but in French.

## 50 GermanDescription

As description, but in German.

## 51 GermanName

As Name, but in german.

## 52 HealTime

It's an integer. The time (in seconds) it take for this unit to heal itself. Typically used by commanders.

## 53 HideDamage

It's a boolean. Whether or not to show the amount of damage done to this unit to the opponent(s). Commanders have this value set.

## 54 HoverAttack

It's a boolean. Will this attack while remaining in the same spot, even though it is an aircraft. Like the brawler.

## 55 ImmuneToParalyzer

It's a boolean. Is this unit immune to being paralyzed.

## 56 init\_cloaked

It's a boolean. When built this unit is cloaked.

## 57 IsAirBase

It's a boolean. Is this unit an airbase. Only the air repair pad and aircraft carriers have this as 1. This must be true for repair pads, otherwise they won't work.

## 58 IsFeature

It's a boolean. Whan true, this unit turns into a feature when built like the dragons teeth. Features do not appear on radar.

## 59 istargetingupgrade

It's a boolean. Does this unit upgrade the targeting so that units shoot at things on radar like at things seen by that unit.

WARNING: this is not used by TA3D yet.

## 60 ItalianDescription

As description, but in Italian.



## **61 ItalianName**

As Name, but in Italian.

## **62 JapanesDescription**

As description, but in Japanese.

## **63 JapaneseName**

As Name, but in Japanese.

## **64 kamikaze**

This is a boolean. Does this unit kill itself to attack its target eg. The roach/invader.

## **65 kamikazedistance**

This is an integer. How far from the target must the unit be (in pixels) to do a kamikaze attack.

## **66 MakesMetal**

This is a decimal number. Does this unit make metal. eg. Metal Maker NS. This is also the amount of metal the unit actually "creates" (not extract!). Can be negative.

## **67 maneuverleashlength**

This is an integer. How far from the course (in pixels) that the unit is patrolling / moving on will it vary if it is attacked / distracted. Also used by planes when repairs are needed if there is a repair pad that is closer than maneuverleashlength.

## **68 MaxDamage**

This is an integer. Amount of damage unit can take before dying (i.e "Hit Points"). Kbots have about 600, Tanks have about 2000, Planes have about 500, Commander has 3000.

## **69 MaxSlope**

This is an integer. What is the maximum slope this unit can go on. In TA3D it is used as the maximum difference of altitude between two map blocs the unit can cross, I am not sure OTA uses it the same way. From 0 to 255.

## 70 MaxVelocity

This is a decimal number. The maximum speed of this unit. Commander is 1.07, Core Storm is 1.25, Arm Hawk is 12. Expressed in *mapblocs/s*

## 71 MaxWaterDepth

It's an integer. What is maximum depth of water this unit can go in. default is 0. (can be anything but only 0-255 makes sense).

## 72 MetalMake

Same as MakesMetal, both can be set and they are cumulative.

## 73 MetalStorage

It's an integer. The amount added to your maximum metal storage while this unit is alive.

## 74 mincloakdistance

It's an integer. The distance (in pixels) the unit must have on every side to remain cloaked. If another unit moves into this area, the unit uncloaks.

## 75 MinWaterDepth

It's an integer. The minimum depth of water this unit can be in. See MaxWaterDepth.

## 76 MobileStandOrders

0, 1 or 2. Basically, I think, can this unit hold position of all mobile units, only the transports (all of them) have this equal to 0. default : 1

## 77 MoveRate1

Planes seem to use this. For example, all planes except the air transports have values of 8, the transports have values of 1.

WARNING: not used by TA3D

## 78 MovementClass

This is a string. How the unit moves. See totala1.hpi for examples. Movement classes are typically described in TDF files in the gamedata directory. (The moveinfo.tdf file is a good guess). It is used as a shortcut way of setting certain parameters (such as those listed here) by grouping them and giving them a name. Settings defined that way have priority over ones defined in the FBI file.

## 79 Name

It's a string. The name you see when you move the cursor over the unit in the game. This is the name that appears on the bottom of the screen when you select a unit.

## 80 NoAutoFire

It's a boolean. Double negative. True means this unit will not automatically fire at other units. False means it will.

## 81 NoChaseCategory

This is a string. Category the unit will not chase. The value specified identifies all units that have that value specified in their own "Category" settings.

## 82 norestrict

It's a boolean. Is there no restriction on this unit (presumably for the AI. (12/29/98) From Draconious: If the unit appears in the Unit Restrictions menu, the way the commander does not appear in the list.

WARNING: not used by TA3D.

## 83 NoShadow

It's a boolean. Does this unit not have a shadow (most water units don't).

NB: since TA3D is an engine with less limitations than OTA's engine, this is ignored.

## 84 Objectname

It's a string. Name of 3D model file for this unit. This is the name of your 3DO/3DM/S3O/3DS/OBJ/... file without extension. You'll do that later, for now leave it the same as Unitname (your unit's "short name").

## 85 onoffable

This is a boolean. Can this unit be turned on and off. On is ACTIVE, OFF is INACTIVE.

## 86 RadarDistance

This is an integer. Distance (in pixels) that shown to you on radar by your unit.

## 87 RadarDistanceJam

This is an integer. Distance (in pixels) that is jammed by unit.

## 88 Scale

This is a decimal number. The size modifier from 3D model file into the game. default : 1.0, better leave this to 1.0 or leave it not set and make a properly scaled model (lighting errors will probably occur with values different from 1.0).

## 89 SelfDestructAs

It's a string. The explosion that happens when a unit self desctructs. See totala1.hpi for examples.

Usually looks like:

```
SelfDestructAs=SMALL_UNIT;
```

## 90 selfdestructcountdown

It's an integer. The number (in seconds) the self-destruct count down starts at. default : 5. (from 0 to 255).

## 91 ShootMe

It's a boolean. Does this unit broadcast itself as a target? If this is false, then enemy units will not choose to attack this unit. default : false.

NB: that's the way units don't automatically attack non offensive buildings.

## 92 ShowPlayerName

It's a boolean. Show the players name as a description (like the commander).

NB: TA3D doesn't use it yet.

## 93 Side

This is a string. Which side does the unit belong to ARM, or CORE. The side your unit will be for. See gamedata/sidedata.tdf for complete description of sides.

## 94 SightDistance

It's an integer. The distance (in pixels) from the unit that everything is shown in. ie. Distance it gets rid of Fog of War. Beware that huge values can slow the game down on big maps with lots of units and fog of war enabled.

## 95 SonarDistance

It's an integer. The distance (in pixels) given to you as sonar, on the mini map.

## 96 SonarDistanceJam

It's an integer. This is how far (in pixels) the unit jams sonar on the mini map.

## 97 SoundCategory

It's a string. The "sound category" the unit uses. Sound categories are typically described in TDF files in the gamedata directory. (The sounds.tdf file is a good place to start). It is used to describe a group of sounds to associate with a unit, such as what sounds does it make when it starts, stops, arrives at a destination, is activated, is deactivated, etc.

## 98 SpanishDescription

As description, in Spanish.

## 99 SpanishName

As Name, but in Spanish.

## 100 StandingFireOrder

This is the initial fire order the unit starts with. 0 = Hold Fire; 1 = Return Fire; 2 = Fire at Will. (default: 1)

## 101 StandingMoveOrder

This is the initial movement order the unit starts with. 0 = Hold Position; 1 = Move; 2 = Roam. (default: 1)

## 102 Stealth

It's a boolean. Is this unit invisible on sonar and radar.

## 103 SteeringMode

This is an integer. The way in which the unit turns. See totala1.hpi for options.

WARNING: not used by TA3D.

## 104 TEDClass

This is a string. Important... Define what type of unit it is. It also defines the icon that is displayed in tactical zoom mode. Can be any of:

- water
- ship
- energy

- vtol
- kbot
- plant
- tank
- special
- fort
- metal
- cnstr
- commander

## 105 teleporter

It's a boolean. Is this unit a teleporter. The two galatic gates have this = 1.

NB: I don't know any use of this tag, TA3D just ignores it for now.

## 106 TidalGenerator

This is a boolean. Is the unit a Tidal generator?

## 107 TransMaxUnits

It's an integer. Maximum number of units that can be transported.

## 108 transportcapacity

It's an integer. Like TransMaxUnits. The number of units that can be transported.

## 109 TurnRate

it's an integer. How quickly the unit turns in *OTAangleunits/s*.

## 110 UnitName

It's a string. The name of the files used for the unit. This is a unit ID. This is the (internal) name of your unit, (what I refer to as the "short name" and the name your files you decided on. e.g.. your file in this case would be called coralpha.fbi.

## 111 UnitNumber

It's an integer. The ID# for the unit. TA3D doesn't use it but it's important in OTA that this ID be unique.

## 112 Upright

It's a boolean. Is the unit upright? For kbots - this keeps the unit upright when climbing hills. For a flying unit laying down on the ground whenever it lands (air unit), setting upright=1 can solve the problem.

## 113 Version

TA3D doesn't use it.

## 114 WaterLine

It's a decimal number. How high up the 3d model to put the water line. For ships.

## 115 Weapon1

It's a string. The unit's primary weapon. The name specified is the name of a weapon described in any of the number of TDF files found in the weapon subdirectories.

## 116 Weapon2

It's a string. The unit's secondary weapon. The name specified is the name of a weapon described in any of the number of TDF files found in the weapon subdirectories.

## 117 Weapon3

It's a string. The unit's third weapon. The name specified is the name of a weapon described in any of the number of TDF files found in the weapon subdirectories.

## 118 Weapon[n]

*(without '[' and ']')*

It's a string. The unit's nth weapon. The name specified is the name of a weapon described in any of the number of TDF files found in the weapon subdirectories. n is an integer  $i=1$ . For n  $i=4$ , all Weapon[p] must be filled for p  $i$  n in order to load it (if there is a hole, it'll stop there).

## 119 WindGenerator

It's an integer. The amount of energy generated by the wind by the unit.  
default : 0

## 120 WorkerTime

It's an integer. How quickly the unit nanolathes. See BuildTime.

## 121 wpri\_badTargetCategory

It's a string. A bad target (lower chance of hitting) for the primary weapon. The value specified identifies all units that have that value specified in their own "Category" settings.

## 122 wsec\_badTargetCategory

It's a string. A bad target for the secondary weapon.

## 123 wthi\_badTargetCategory

It's a string. A bad target for the third weapon.

## 124 w[n]\_badTargetCategory

*(without '[' and ']')*

It's a string. A bad target for the nth weapon.

## 125 YardMap

It's a string. Defines in more detail the "footprint" of a construction yard (or any stationary building). As an example, the ARM Adv. Air Plant data at the right has 6 groups of 7 values defining an X-Z coordinate "footprint" for the construction yard. The following values are available:

- "f" indicates the space is occupied by a feature (e.g. Dragons teeth)
- "o" indicates occupied points on ground where a unit may not pass. By default, all values are "o".
- "O" - Occupied when building is open - never used.
- "c" defines the "hole" (typically in construction plant units) where the units are built and where they enter or exit. closed yard - "Openyard/Closeyard" on land.
- "C" - closed yard - "Openyard/Closeyard" on water.
- "w" seems to indicate those parts of a ship yard that are above water
- "g" (or "G") indicates that the space can operate over a geothermal vent. Otherwise, operates just like an "O". Created specifically for the geothermal plant.
- "y" - standard yard - no footprint on land
- "Y" - standard water yard - no footprint on water. "Y" is simply a location that will never have a footprint, no matter what.

As you know, footprints are square, but your units aren't always. This allows you to shape the footprint a little. (i.e. ARMLAB, the corners are always "footprintless")

If the unit doesn't use "OpenYard"/"CloseYard" you can simply state "YardMap=o" no matter the size of the footprint. You could also do the opposite by stating "YardMap=c", and the whole footprint will appear and disappear no matter the footprint size.

An example:



```
// For ARM Adv. Air Plant
// FootprintX=7;
// FootprintZ=6;
YardMap=0000000 0000000 0000000 0000000 0000000 0000000;
// Thus, the "South" end
// of the plant is open
// while the other sides
// are blocked.
```

Spaces are ignored, they are here for readability.

## 126 ZBuffer

It's a boolean. Of course TA3D uses a depth buffer to render everything so this is useless for us.