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	$\times \frac{65536}{360}$

0.0.1 Acceleration

The unit acceleration, can be a decimal number. Expressed in $pixel/s^2$.

0.0.2 ActivateWhenBuilt

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

0.0.3 altfromsealevel

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

0.0.4 amphibious

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

0.0.5 antiweapons

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

0.0.6 attackrunlength

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

0.0.7 BMcode

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

0.0.8 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;

0.0.9 BrakeRate

Like acceleration but when the unit wants to slow down.

0.0.10 BuildAngle

No idea. Here's a possibility: Random angle to build unit on, example is the solar panel some times facing 180 degrees from the solar panel next to it. BuildAngle=32768;

BuildCostEnergy The cost in energy of building this unit. The amount of energy that will be consumed during the building of this unit. I think value ¡ 0 generate errors, but I've never tried. BuildCostEnergy=985; BuildCostMetal The cost in metal of building this unit. The amount of metal that will be consumed during the building of this unit. Again, I think values ¡ 0 produce errors. BuildCostMetal=0; BuildTime The time is takes to build this unit. On average, 10000 is medium time, things like fusion plants are more like 100000, and krogoth like 300000 BuildTime=9894; Builddistance The distance at which a unit can build if it is a builder. Builddistance=80; Builder Can this unit build other units. Builder=1; canattack Can the unit attack. canattack=1; CanCapture Can this unit capture other units. CanCapture=1; CanDgun Can the unit DGUN (ie. has a gun like the commander) candgun=1; Canfly Can this unit fly. Canfly=1; canguard Can this unit guard. canguard=1; canhover Can this unit hover canhover=1; canload Can this unit load other units. canload=1; canmove Can this unit move. canmove=1; canpatrol Can this unit patrol. canpatrol=1; CanReclamate Can this unit reclaim things. CanReclamate=1; canstop Can this unit stop. canstop=1; cantbetransported Is this a unit that transports cannot load eg. Ships, krogoth cantbetransported=1; Category A category class. Look at some of the FBI files in totala1.hpi for examples. Essentially, this places the unit in several named catagories that can be named in other variables such as "wpri_badTargetCategory" described below, and in the AI Text files as well. Category=ARM TANK LEVEL2 CONSTR NOWEAPON NOTAIR NOTSUB CTRL_B; CloakCost The energy cost of this unit remaining cloaked when standing still. CloakCost=7; CloakCostMoving The energy cost of this unit remaining cloaking while moving. CloakCostMoving=800; Commander Is this unit the (a?) commander. Commander=1; Copyright The copyright of this unit. Must be as shown to the right, or won't work. (Most people copy this line as a comment, and put their own copyright infomation in the commented-out version. Copyright=Copyright 1997 Humongous Entertainment. All rights reserved.; Corpse 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 3do file of the feature, the feature entry tells the 3do file to use in the feautres OBJECT variable). Corpse=fortification_core; cruisealt The altitude at which the unit flys cruisealt=90; DamageModifier The rate at which this unit heals itself. DamageModifier=0.7; DefaultMissionType The default orders of this unit for missions. DefaultMissionType=VTOL_standby; Description The description of the unit seen at the bottom of the screen when you hold the cursor over it in the game. Description=Very Heavy Assault Tank; Designation The designation. This is unimportant and can have virtually any value (there may be a limit to the number of character it will except) Designation=WB-S; digger 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. digger=1; Downloadable Is this a unit which has been downloaded off the internet.

If you want to be able to use the unit, put 1.

(12/28/98) Note from The Raptor: Not true. I've never included this variable in all of my units, and they work just fine. It's just one that we started seeing in the Cavedog weekly units, so assumed that's what it meant. It really seems to do nothing. Downloadable=1; EnergyMake The energy this unit produces when ACTIVE. EnergyMake=9; EnergyStorage The amount added to the maximum amount of energy you can store while the unit is alive. EnergyStorage=80; EnergyUse The amount of energy the unit uses while it is ACTIVE. EnergyUse=9; ExplodeAs The explosion ID of the explosion that will happen if the unit explodes. See totala1.hpi for other examples. ExplodeAs=LARGE_BUILDINGEX; ExtractsMetal The rate at which the unit extracts metal. (Yes, the MOHOmine only extracts at like 0.01) ExtractsMetal=0.003; firestandorders Can this unit give firestandorders. You know as much as me. firestandorders=1; Floater Does this unit float on water. Floater=1; FootprintX 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. FootprintX=9; FootprintZ 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. FootprintZ=9; FrenchDescription As description, but in French. FrenchDescription=Serpent des mers; FrenchName As Name, but in French FrenchName=Zeus; GermanDescription As description, but in German GermanDescription=Überschwerer Gefechtspanzer; GermanName As Name, but in german GermanName=A.K.; HealTime The time it take for this unit to heal itself. (Typically used by commanders) HealTime=27; HideDamage Whether or not to show the amount of damage done to this unit to the opponent(s). Commanders have this value set. HideDamage=1; HoverAttack Will this attack while remaining in the same spot, even though it is an aircraft. Like the brawler. HoverAttack=1; ImmuneToParalyzer Is this unit immune to being paralyzed. ImmuneToParalyzer=1; init_cloaked When built this unit is cloaked. init_cloaked=1; IsAirBase Is this unit an airbase. Only the air repair pad and aircraft carriers have this as 1 IsAirBase=1; IsFeature Is this unit a feature. eg. The dragons teeth. Features do not appear on radar. IsFeature=1; istargetingupgrade Does this unit upgrade the targeting so that units shoot at things on radar like at things seen by that unit. istargetingupgrade=1; ItalianDescription As description, but in Italian ItalianDescription=Veicolo veloce da attacco; ItalianName As Name, but in Italian ItalianName=Zipper; JapanesDescription As description, but in Japanese JapaneseDescription=XXXX; JapaneseName As Name, but in Japanese JapaneseName=Jeffysan; kamikaze Does this unit kill itself to attack its target eg. The roach/invader kamikaze=1; kamikazedistance How far from the target must the unit be to do a kamikaze attack kamikazedistance=80; MakesMetal Does this unit make metal. eg. Metal Maker NS MakesMetal=1; maneuverleashlength How far from the course that the unit is patrolling / moving on will it vary if it is attacked / distracted. maneuverleashlength=640; MaxDamage 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 MaxDamage=992; MaxSlope What is the maximum slope this unit can go on. MaxSlope=8; MaxVelocity What is the maximum speed of this unit. Commander is 1.07, Core Storm is 1.25, Arm Hawk is 12 MaxVelocity=9; MaxWaterDepth What is maximum depth of water this unit can go in MaxWaterDepth=35; MetalMake The amount of metal generated by construction units MetalMake=1; MetalStorage The amound added to

4

your maximum metal storage while this unit is alive MetalStorage=900; mincloakdistance The distance the unit must have on every side to remain cloaked. If another unit moves into this area, the unit uncloaks. mincloakdistance=90; MinWaterDepth The minimum depth of water this unit can be in MinWaterDepth=8; MobileStandOrders Basically, I think, can this unit hold position of all mobile units, only the transports (all of them) have this equal to 0. MobileStandOrders=1; MoveRate1 Planes seem to use this. For example, all planes except the air transports have values of 8, the transports have values of 1. MoveRate1=1; MovementClass 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. MovementClass=TANKSH3; Name 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. Name=Zipper; NoAutoFire Double negative. 1 means this unit will not automatically fire at other units. 0 means it will. NoAutoFire=1; NoChaseCategory Category the unit will not chase. The value specified identifies all units that have that value specified in their own "Category" settings. NoChaseCategory=VTOL; norestrict Is there no resistriction on this unit (presumably for the AI. (12/29/98) From Draconious: If the unit apears in the Unit Restrictions menu, the way the commander does not apear in the list. norestrict=1; NoShadow Does this unit not have a shadow (most water units don't) NoShadow=1; Objectname Name of 3Dobject file for this unit. This is the name of your 3dO file. You'll do that later, for now leave it the same as Unitname (your unit's "short name"). Objectname=CORWIN; onoffable Can this unit be turned on and off. On is ACTIVE, OFF is INACTIVE onoffable=1; Ovradjust No idea. Some units just have this Ovradjust=1; PigLatinDescription As description but in Pig Latin. PigLatinDescription=Astfay attackay Outingscay Ehiclevay; PigLatinName As Name but in Pig Latin. PigLatinName=Effyjay; PitchScale PitchScale=1; RadarDistance Distance that shown to you on radar by your unit RadarDistance=700; RadarDistanceJam Distance that is jammed by unit. RadarDistanceJam=730; Scale The size modifier from 3DO file into the game. I think. I don't actually know. Scale=1; SelfDestructAs The explosion that happens when a unit self desctructs. See totala1.hpi for examples. SelfDestructAs=SMALL_UNIT; selfdestructcountdown The number the self-destruct count down starts at. selfdestructcountdown=2; ShootMe Does this unit broadcast itself as a target? (eg. Dragons's Teeth has this as 0, all other units as 1) If this is 0, then enemy units will not choose to attack this unit regardless of their orders. ShootMe=1; ShowPlayerName Show the players name as a description (like the commander) ShowPlayerName=1; Side Which side does the unit belong to ARM, or CORE. The side your unit will be for. Side=CORE; SightDistance The distance from the unit that everything is shown in. ie. Distance it gets rid of Fog of War. Unfortunately, this value seems to have a limit of around 400. SightDistance=90; SonarDistance The distance given to you as sonar, on the mini map. SonarDistance=650; SonarDistanceJam This is how far the unit jams sonar on the mini map. SonarDistanceJam=650; sortbias sortbias=0; SoundCategory 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 as-

sociate with a unit, such as what sounds does it make when it starts, stops, arrives at a destination, is activated, is deactivated, etc. Sound-Category=VIPE; SpanishDescription As description, in Spanish. SpanishDescription=Vehículo todoterreno de ataque; SpanishName As Name, but in Spanish SpanishName=Zipper; StandingFireOrder This is the initial fire order the unit starts with. 0 = Hold Fire; 1 = Return Fire; 2 = Fire at Will. StandingFireOrder=0; StandingMoveOrder this is the initial movement order the unit starts with. 0 = Hold Position; 1 = Move; 2 = Roam StandingMoveOrder=1; Stealth Is this unit inivisible on sonar and radar. Stealth=1; SteeringMode The way in which the unit turns. See totala1.hpi for options. SteeringMode=2; TEDClass Important...I think. Define what type of unit it is. TEDClass=WATER; teleporter Is this unit a teleporter. The two galatic gates have this = 1. teleporter=1; ThreeD Is the unit 3D. ThreeD=1; TidalGenerator Is the unit a Tidal generator? TidalGenerator=1; TransMaxUnits Maximum number of units that can be transported. TransMaxUnits=1; transportcapacity Like TransMaxUnits I guess. The number of units that can be transported. transportcapacity=5; transportsize How big the transport is. I don't quite understand. transportsize=3; TurnRate How quickly the unit turns. TurnRate=999; UnitName The name of the files used for the unit? This is a unit ID. This is the 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. UnitName=CORWIN; UnitNumber The ID# for the unit. If this is the same as some other unit, your unit may not work. Unique unit ID. VERY IMPORTANT. All IDs from 0-about 400 have been used. Use other ones ¿ 10000 is the best. The more digits, the more unlikely you'll have a ID conflict with someone else. UnitNumber=9; Upright Is the unit upright? For kbots - this keeps the unit upgright when climbing hills. In my case for a flying unit, my unit "layed down" on the ground whenever it landed (air unit). By setting upright=1, it solved my problem. Upright=1; Version The version of ta the unit will work under. Do not be fooled. This is very important. It is version of TA under which the unit will run (beta 2 or whatever) Version=1; WaterLine How high up the 3d model to put the water line. For ships. WaterLine=43; Weapon1 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. Weapon1=crblmssl; Weapon2 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. Weapon2=coramph_weapon2; Weapon3 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. Weapon3=CORSEAP_WEAPON3; WindGenerator The amount of energy generated by the wind by the unit. WindGenerator=30; WorkerTime How quickly the unit nanolathes. WorkerTime=80; wpri_badTargetCategory 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. wpri_badTargetCategory=VTOL; wsec_badTargetCategory A bad target for the secondry weapon. wsec_badTargetCategory=VTOL; YardMap 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 oc-

cupied 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. // For ARM Adv. Air Plant // FootprintX=7; // FootprintZ=6; YardMap=ooooooo ooooooo occccco occccco occccco occccco; // Thus, the "South" end // of the plant is open // while the other sides // are blocked. ZBuffer Does the unit have a z buffer. (All units have this = 1) ZBuffer=1;