The Escape Velocity
Resource Bible

Last Revision:
6/27/96

WARNING: This information is not guaranteed to be 100% accurate. Use at your own risk. Has been known to cause cancer in laboratory animals. Caveat lector.

by Matt Burch
©1996 by Ambrosia Software, Inc.
All Rights Reserved Worldwide

http://www.replacementdocs.com

Part I - Game Constants

MaxShipsInSystem	36
MaxStellarObjects	1500
MaxSystems	1000
MaxShipClasses	64
MaxStellarClasses	64
JumpDistance	1000 pixels
MaxWeaponTypes	64
MaxOutfitItemTypes	128
MaxBeamsOnScreen	8
MaxDudeTypes	128
MaxGovts	128
MaxExplosionsOnScreen	16
MaxMissions	256
NumMissionBits	256
MaxCargoTypes	64
MaxPersonTypes	512
MaxShotsOnScreen	64
MaxAsteroids	10
MaxNebulae	4
MaxSimultaneousMissions	8
MaxDisasters	128
MaxFleets	128

Part II - Resource Descriptions

Note: EV's resources all start at ID number 128, but the internal storage for all data file info is zero-based. Therefore, when a field in the EV data file is said to refer to a government, stellar object, etc., it refers to it by its index number (starts at 0) unless it is specifically stated that it is referring to the ID number, which starts at 128.

Note: Some of EV's fields refer to other resource IDs or index numbers, but their values are offset by a certain amount to indicate type. For example, the mïsn resource's AvailStel field refers to the index number of a gövt resource when its value is between 10000 and 10063. In cases like this, it is necessary to add to or subtract from the field in order to force the value into the proper range: in this instance, you'd subtract 10000 to find the index number of the gövt.

Note: If a user wants to distribute a new scenario file to other users of EV, he can include his own custom graphics, sounds, or sprites in the "EV Data" file. These resources will override the ones in the "EV Graphics" and "EV Sounds" files, so you don't have to send the entire graphics and sounds files just to make changes for your scenario.

Note: Any resources in an EV plugin file automatically replace same-numbered resources in EV's main files. Additional graphics for the shipyard and outfit dialog menus (just individual menu items, not the whole menu!) are loaded from PICT ID's 5101 and up for shipyard and 6101 and up for outfit. (e.g. 5101 is the first ship type, 5105 is the fifth, etc., and the same goes for the outfit menu items) These menu items will replace the respective ones in the main menu resources, PICTs 5100 and 6100.

• The spïn resource (stored in EV Graphics)

Spin resources contain sprite info. Whenever EV needs to load a set of sprites for a particular object, it looks at that objects's spin resource, which in turn tells the game how to load the object's sprites. EV sprites are stored as paired sprite and mask PICT resources. The sprites in each PICT are arranged in a grid, which can be of any size. The spin resource tells EV what shape and size the sprites' grid is. Spin resources have the following fields:

SpritesID	ID number of the sprites' PICT resource
MasksID	ID number of the masks' PICT resource
xSize	Horizontal size of each sprite (should be a multiple of 8!)
ySize	Vertical size of each sprite
xTiles	Horizontal grid dimension
yTiles	Vertical grid dimension

Spin resources have certain reserved ID numbers, which correspond to different types of objects:

128-191	Ships
200-263	Weapons
300-363	Stellar objects
400-402	Explosions
500	Boxes

It is important to note that the ID numbers of the PICT resources are non-critical, as EV looks at the spin resources to find the sprites, and not at the actual PICT ID numbers themselves.

Note: The total number of sprites in the graphics file (used by EV to update the progress bar during startup time) is stored in 'spït' resource 128 in the EV Graphics file.

• The dësc resource

Desc resources store null-terminated text strings (descriptions) that are used by EV in a variety of places. For some desc resources, EV looks for a certain reserved ID number. Other desc resources are pointed to by fields in other resources, so their ID numbers are not necessarily fixed, and can be set to virtually anything by the scenario designer. The reserved desc ID numbers, along with the maximum length for each type, are below:

128-1628	Stellar object descriptions, shown when landed on a planet.
2000-2063	Ship class descriptions, shown in the shipyard and requisition-escort dialog.
2100-2163	Ship pilot descriptions, shown in the hire-escort dialog.
3000-3127	Outfit item descriptions, shown in ship outfitting dialog.
4000-4255	Mission descriptions, shown in mission dialog.

• The düde resource

A dude resource can be thought of as a container for ships that share certain characteristics. Any ship of a given dude class will have that dude class's AI type and governmental affiliation, and will yeild the same types of booty when boarded. In a dude resource, up to four different ship classes can be pointed to, with a probability set for each ship class. The result of all this is that, in other parts of EV's data file, you can point to a dude class and know that EV will create a ship of the proper AI type and governmental alignment, and will pick the new ship's type based on the probabilities you set in the dude resource. The dude resource's fields are:

AIType	Which type of AI to use for ships of this dude class (see below)
ShipTypes (x4)	These fields contain the ID numbers of up to four different ship classes
% Prob (x4)	These fields set the probability that a ship of this dude class will be of a certain ship type
Govt	The ID number of the dude class's government, or -1 for independent.
Booty	Flags that define what you'll get when you board a ship of this dude class. (see below)

The five different AI types are:

1 - Wimpy Trader	Visits planets and runs away when attacked
2 - Brave Trader	Visits planets and fights back when attacked, but runs away when his attacker is out of range
3 - Warship	Seeks out and attacks his enemies, or jumps out if there aren't any.
4 - Interceptor	Seeks out his enemies, or parks in orbit around a planet if he can't find any. Buzzes incoming ships to scan them for illegal cargo. Also acts as "piracy police" by attacking any ship that fires on or attempts to board another, non-enemy ship while the interceptor is watching.
5 - Naval Fighter	Carried by another ship. Reserved for internal use - do not use!
6 - Escort	Flies with the player. Reserved for internal use - do not use!

You can set different combinations of booty to be had from ships of a certain dude class by ORing different bits into the dude's Booty field. If a dude class has a booty flag of 0x0000, then you can't get anything from the ship, and you're told that you were "repelled while attempting to board" it. The different booty flags are:

0x0001	Food
0x0002	Industrial goods
0x0004	Medical supplies
0x0008	Luxury goods
0x0010	Metal
0x0020	Equipment
0x0040	Money (depends on the ship's purchase price)

The last field tells EV what kind of text to display when you hail a ship of this dude type and send a greeting: (note that these values can be added together to have EV display multiple different types of information)

InfoTypes	What kind of info to display
1000	Good prices
2000	Disaster info
4xxx	Specific advice (the last three digits of this value are added to 7500 to get the ID of the STR# resource from which to get the quote)

One thing to remember when setting up a dude resource is that EV will choke if any of the ShipType fields are filled incorrectly (i.e. less that 128 or greater than 159) or any of the probabilities are not between 1 and 100. Therefore, if you want a dude type to point to only one ship class, you'd set all four ShipType fields the same and put 25% in each Prob field. (see the "Confed Liner Only" dude class for an example of this)

• The flët resource

A flet resource definies the paramaters for a fleet, which is a collection of ships that can be made to appear randomly throughout the galaxy. Fleets can also be made to appear by including them in dude resources.

LeadShipType	ID of the fleet's flagship's ship class
EscortType (x4)	IDs of the flagships escorts' ship classes. If you don't want to use four different escort types, you should still set the unused fields to a valid ship class ID. (you can set the min & max fields to 0 and just have the extra ships not appear)
Min (x4)	The minimum number of each type of escort to put in the fleet.
Max (x4)	The maximum number of each type of escort to put in the fleet.
Govt	ID of the fleet's government, of -1 for none.
LinkSyst	Which systems the fleet can be created in

-1	Any system
128-1127	ID of a specific system
10000-10063	Any system belonging to this specific government
15000-15063	Any system belonging to an ally of this govt
20000-20063	Any system belonging to any but this govt
25000-25063	Any system belonging to an enemy of this govt

Note: When setting up misn resources, you shouldn't make reference to a dude resource which contains pointers to fleets, as this will cause EV to behave strangely. Use only normal, single–ship dude resources in missions.

• The gövt resource

A govt resource defines the parameters for a government, which is in turn defined as "any collection of ships and planets that react collectively to the actions of the player and other ships." Governments keep track of how they feel toward you, and they can also have set enemies and allies. The govt resource's fields are:

Unused Field	A placeholder (ignored)
Flags	Sets a variety of characteristics (see below)
Ally	The ID number of the govt's ally.
Enemy	The ID number of govt's enemy.
CrimeTol	The maximum amount of evilness the player can accumulate before warships of this govt start to beat on him.
SmugPenalty	The amount of evilness a player gains for being detected smuggling illegal cargo (defined in a misn resource) past this government's ships.
DisabPenalty	The amount of evilness for disabling one of this govt's ships.
BoardPenalty	Evilness from pirating one of this govt's ships
KillPenalty	Evilness from killing this govt's ships
ShootPenalty	Evilness from shooting one of this govt's ships (currently ignored)
InitialRec	The player's initial legal record in systems controlled by this govt (0 is neutral, positive is good, negative is bad)

The different bits that can be set in a govt's Flags field are:

0x0001	Xenophobic (Warships of this govt attack everyone except their allies. Useful for making pirates and other nasty dudes.)
0x0002	Ships of this govt will attack the player in non-allied systems if he's a criminal there (useful for making one govt care only about the player's actions on its home turf, while another is nosy and enforces its own laws everywhere it goes)
0x0004	Always attacks player
0x0008	Never attacks player
0x0010	Warships of this govt will retreat when their shields drop below 25% - otherwise they fight to the death
0x0020	Ignore ships of this govt in the DoGoodSamaritan function
0x0100	'pers' ships of this govt won't use escape pod, but will act as if they did
0x0200	Warships will take bribes.
0x0400	Can't hail ships of this govt
0x0800	Ships of this govt start out disabled (derelicts)
0x1000	Warships will plunder non-mission, non-player enemies before destroying them
0x2000	Freighters will take bribes.
0x4000	Planets of this govt will take bribes
0x8000	Ships of this govt taking bribes will demand a larger percentage of your cash supply, and their planets will always take bribes (useful for pirates)

Doing evil deeds to one government will improve your rating with its enemies, and vice versa. Allied governments also communicate your actions, so attacking one government will make its allies hate you too.

One important thing to note is that two governments don't both have to have the other defined as their ally to have them be friendly toward one another. As an example: John says "I like Cajun," and Cajun says "John? Who's that?" Cajun and John are automatically allies, because Cajun's love for John is implied in John's statement of undying devotion to Cajun. :) This way, governments can have multiple allies. Enemies work in a similar fashion, except that any government *not* allied with a xenophobic government is automatically considered its enemy.

• The jünk resource

Junk resources store info on specialized commodities that can be bought and sold at only one location (each). The fields are:

SoldAt	ID number of the stellar object where the commodity is sold
BoughtAt	ID number of the stellar object where the commodity is purchased (or -1 for none)
BasePrice	The average price of the commodity (works much like the base prices for "regular" commodities)
Flags	Misc control bits (all but one unused)
	0x0001 Tribbles flag - When in your cargo bay, the commodity multiplies like tribbles.

• The mïsn resource

Missions are the crown jewel of the EV datafile, as well as the largest and most complex resources in the game. Each misn resource corresponds to a single mission that the player can undertake, with the name of the mission (which the player sees in the mission list) being the name of the associated misn resource. The first six fields in a misn resource help EV determine where and when the mission is available:

AvailStel	Which stellar objects (i.e. planets) the mission is available at
	-1 Any inhabited stellar
	128-1627 ID number of a specific stellar
	5000-5999 Stellar in a system adjacent to specific system
	9999-10127 Specific govt's stellar
	15000-15127 Specific govt's ally's stellar
	20000-20127 Stellar of anybody *but* this specific govt
	25000-25127 Specific govt's enemy's stellar
AvailBitSet	Which one of the mission flag bits, (see below) if any, must be set for this mission to become available
	-1 ignored
	0-255 this mission bit must be set
	1000-1255 this mission bit must be clear (functions similarly to AvailBitClear, described below)
AvailLoc	Where on a planet this mission is available
	0 From the mission computer
	1 In the bar
	2 Both
AvailRecord	What your legal record in this system must be for this mission to become available
	0 ignored
	positive value record must be at least this high
	negative value record must be at least this low
	-32000 when the player has dominated the stellar in question
	-32001 when the player has dominated at least one stellar
AvailRating	What your combat rating must be for this mission to be available
	-1 ignored
	0+ rating must be at least this high
AvailRandom	A randomization factor, to ensure that some missions aren't available all the time. Mission randomizing values are recalculated each time you warp into a system.
	100 always available
	1-99 available this % of the time

A quick word on mission bits: EV stores 256 flags that can be set by your missions when they fail or succeed. The mission bits can then be checked to see whether a mission is allowed to be available. There are three uses for this:

1. Ensuring that a mission is only available once.
2. Creating branching plotlines, in which the types of missions offered depend on your past successes and failuires.
3. Creating a number of missions that are available at the same time, but which are mutually exclusive. (e.g. you wouldn't want the player to go on more that one of a set of three missions) These missions could be made unavailable if bit 37 was set, for example, and then would set bit 37 on completion.

The next two fields in the misn resource define where the player needs to go to complete the mission:

TravelStel	Which stellar object the player must go to during the mission	
	-1	No specific stellar destination
	-2	A random inhabited stellar
	-3	A random uninhabited planet
	128-1627	ID number of a specific stellar
	9999-10127	Random stellar of a specific govt
	15000-15127	Random stellar of a specific govt's ally
	20000-20127	Random stellar of anybody *but* this specific govt
	25000-25127	Random stellar of specific govt's enemy

ReturnStel	Where the player must return to in order to complete the mission and receive payment	
	-1	No specific stellar destination
	-2	A random inhabited stellar
	-3	A random uninhabited stellar
	-4	The initial stellar, where the mission was accepted
	128-1627	ID number of a specific stellar
	10000-10127	Random stellar of a specific govt
	15000-15127	Random stellar of a specific govt's ally
	20000-20127	Random stellar of anybody *but* this specific govt
	25000-25127	Random stellar of specific govt's enemy

The next five fields tell EV about any special cargo associated with a mission:

CargoType	What type of cargo must be carried	
	-1	No special cargo for this mission
	0-63	Specific cargo type
	1000	Random cargo of types 0-5 (the standard types)

CargoQty	What amount of cargo must be carried	
	-1	Ignored (no cargo)

	0 and up	This many tons of cargo
	-2 and below	abs(CargoQty) tons, ± 50%

PickupMode	Where the cargo is to be picked up	
	-1	Ignored
	0	Pick up at mission start
	1	Pick up at TravelStel
	2	Pick up when boarding special ship

DropOffMode — Where the cargo is to be dropped off [**Note:** don't set your cargo to be picked up and dropped off at the same place, as it may cause EV to behave strangely]

	-1	Ignored
	0	Drop off at TravelStel
	1	Drop off at mission end (ReturnStel)

ScanGovt — Which government considers your cargo illegal

	-1	Ignored
	128-255	ID number of a government that considers this cargo illegal. If you're scanned by a ship of this government, or any government that's not its enemy (important!) you'll get that government's SmugglePenalty added to your record.

FailIfScanned — Sets whether the mission fails if you're detected carrying the cargo

	0	Mission doesn't fail if scanned
	Nonzero	Mission fails if you're scanned

The next field tells EV what to give you if you're successful in your mission:

PayVal — What you get if you're successful and you return to ReturnStel

	0	No pay, just the satisfaction of a job well done
	1 and up	This number of credits
	-10128 to -10255	Clean legal record with the govt with this ID
	-20128 to -20255	Give the player an item with this ID
	-30128 to -30255	Give the player an item with this ID at the *start* of the mission

The next six fields contain information on the special ships associated with this mission, if any:

ShipCount — The number of special ships for this mission

	-1	Ignored (no special ships)
	0-31	This number of special ships

ShipSyst — Which system the special ships will appear in

	-1	The initial system where the mission is begun
	-2	Any random system

	-3	TravelStel's system
	-4	ReturnStel's system
	-5	System adjacent to initial system
	-6	Whatever system the player is in (i.e. follow him around)
	128-1127	ID number of a specific system
	9999-10127	Specific govt's system
	15000-15127	Specific govt's ally's system
	20000-20127	System of any govt but this specific one
	25000-25127	Specific govt's enemy's system

ShipDude   What dude resource to use to determine the special ship's types and characteristics

	-1	Ignored (no special ships)
	128-255	ID number of a specific dude class

ShipGoal   The mission goal associated with the special ships

	-1	Ignored (no specific goal for the special ships)
	0	Destroy all the ships
	1	Disable but don't destroy them
	2	Board them
	3	Escort them (keep them from getting killed)
	4	Observe them (you just have to be in the same system with them)
	5	Rescue them (they start out disabled, and you must board them)
	6	Chase them off (either kill them or scare them into jumping out of the system)

ShipBehav   Defines any special actions you want the ships to take

	-1	Ignored (they use their standard AI routines)
	0	Special ships will always attack the player
	1	Special ships will protect the player
	9	Special ships will hyper in all together after a short delay
	10	Special ships will hyper in and attack the player
	11	Special ships will hyper in and protect the player

ShipNameID   Tells EV how to name the special ships

	-1	Ignored (special ships have normal names)
	128 and up	Pick a name from this STR# resource

The next three fields determine what will happen when you successfully complete the mission:

CompBitSet   Which mission bit will be set on completion

	-1	Ignored
	0-255	Set this mission bit
	1000-1255	Clear this mission bit

CompGovt	Which government to use in determining how your record changes on completing this mission
	-1 Ignored (no reward other than pay)
	0-127 Increase record with this govt

| CompReward | How much to increase your record with CompGovt |
| | (any value)   Increase record by this much |

(note: if you have a CompGovt and reward defined and you fail the mission, that govt will take it personally and decrease your record by 1/2 the amount specified in CompReward. This is useful for making missions whose success is considered vital by a certain party.)

The next field determines what happens when you fail the mission:

FailBitSet	Which mission bit to set upon failure
	-1 Ignored
	0-255 Set this mission bit on failure
	1000-1255 Clear this mission bit on failure

The next seven fields tell EV which desc resources to display at various times during the mission: (see below for more info on mission descriptions)

BriefText	The desc to show in the dialog that comes up when you accept a mission. (formats are the same for all seven fields)
	-1 No special mission briefing
	128 and up ID number of the desc resource to use (ID numbers of 5000 and up are usually the safest)

QuickBrief   The desc to show when the user hits the "Mission Briefing" (I) key.

LoadCargText   The desc to show when special mission cargo is loaded from a planet

DumpCargoText   The desc to show when special mission cargo is offloaded (not jettisoned into space as the name would suggest!)

CompText   The desc to show when you go to ReturnStel and the mission has been successful.

FailText   The desc to show when you go to ReturnStel and the mission has been a failure

The next field tells EV how long you have to complete the mission:

TimeLimit	Like it says
	-1 Ignored (no time limit)
	1 and up This number of days

The next field tells EV whether or not the mission can be aborted.

CanAbort	0	Mission can't aborted, you must go to ReturnStel in order for it to "go away" and become inactive
	Nonzero	Mission can be aborted at any time, by simply selecting another mission. Also, the mission "goes away" and becomes inactive at the moment it fails (e.g. you're scanned when you aren't supposed to be, etc.).

The next field is unused, and the next field after that is another piece of info on how and when EV should offer the mission. (sorry, I had to add this field to the template after many of the missions were already done)

AvailBitClr	-1	Ignored
	0-255	Mission is available only if this mission bit is not set.

The next few fields tell EV about any auxiliary ships you want to be placed in the universe for this mission. Auxiliary ships cannot be given specific instructions, and no goals can be set for them; they simply are "normal" ships that are placed into the universe for the purpose of adding atmosphere to a mission.

AuxShipCount	How many aux ships, if any, to activate for this mission:	
	-1	No aux ships
	1-31	Place this many aux ships in the universe
AuxShipDude	ID number of the specific dude resource to use to set up the aux ships	
AuxShipSyst	What systems to place the aux ships in:	
	-1	Any system the player is in
	-2	TravelStel's system
	-3	ReturnStel's system
	128-1127	ID number of a specific system
	5000-5999	In this system, or any systems adjacent to it
	9999-10127	Any system belonging to this govt
	15000-15127	Any system belonging to this govt or its allies
	20000-20127	Any system not belonging to this govt
	25000-25127	Any system belonging to enemies of this govt

Then there are some more fields we had to append to the end because we didn't think of them until later:

CompBitSet2	Another completion bit field, which performs identically to the other one.

Flags		Some misc. flag bits
	0x0001	Marks the mission as an auto-aborting mission, which will automatically abort itself after it is accepted. (sometimes useful to create special ships) Any mission bits pointed to by the mission's CompBitSet fields will be automatically set when the mission aborts
	0x0002	Don't show the red destination arrows on the map
	0x0004	Can't refuse the mission
	0x0010	Infinite auxShips
	0x0020	Remove pre-paid outfit item on mission failure or abort
	0x0040	Apply -5x CompReward reversal on abort
	0x0080	Global penalty when jettisoning mission cargo in space (currently ignored)
	0x0100	Show green arrow on map in initial briefing
	0x1000	Critical mission (will be offered before all others in the bar)

Whenever EV displays a desc resource related to a mission, such as the initial mission description (desc ID 4000-4255) or one of the special mission briefings (e.g. CompText and QuickBrief) it performs one other special operation on the text. It searches through the text and replaces a few special "wildcard" symbols with pertinent mission information. This is extremely useful in setting up mission briefings that include random information that wouldn't be known when the description is written. These special symbols and their expansions are:

<DSY>	The name of the destination system
<DST>	The name of the destination stellar
<RSY>	The name of the return system
<RST>	The name of the return stellar
<CT>	The name of the type of cargo to be carried
<CQ>	The quantity of cargo to be carried
<DL>	The date of the mission deadline, if any
<PN>	The player's name
<PSN>	The player's ship's name
<OSN>	The offering ship name (only works when offering a mission from a ship)
<SN>	Special ship name (Note: EV will screw up if you use this in the initial mission description, as it doesn't pick the special ship names until you actually accept the mission.)

• The nëbu resource

Nëbu resources contain info on the nebulae (or other space phenomena) which are displayed in the background of the star map. These images don't actually have any effect on events in the game, they're just there to look pretty. You can, however, combine nëbu background images with custom asteroid or interference data in the sÿst resources for cool localized effects. The PICT resources associated with the four available nëbu resources are numbered 9500-9511, and the effects of the nëbu resource's fields are as follows:

XPos   The image's position on the star map. These coordinates are
YPos   expressed in the scale of the "normal" map zoom level (not zoomed in or out) and are relative to the upper-left corner of the image.

XSize   The image's size on the star map. These values are expressed in
YSize   the scale of the "normal" map zoom level (not zoomed in or out) and tell EV how big to make the image when the map is at normal zoom.

• The öops resource

Oops resources contain info on planetary disasters. Actually, the term "disasters" is a misnomer, as these occurances simply affect the price of a single commodity at a planet or station, for good or bad. EV uses the name of the resource in the commodity exchange dialog box to indicate that a disaster is currently going on at a planet. The fields of an oops resource are:

Stellar	The stellar object this disaster is linked to	
	128-1628	ID of a stellar object
	-1	Any planet or station (use sparingly)
	-2	Nothing (used for mission-related news)
Commodity	Which commodity to affect the price of (0 = food, 1 = industrial, etc.)	
PriceDelta	How much to raise or lower the price. (negative numbers lower it)	
Duration	How many days the disaster lasts.	
Freq	Percent chance per day that the disaster will occur.	
MissionBit	Tells EV to always make this disaster active when a certain mission bit is set. (see mïsn for more info)	
	-1	Ignored
	0-191	Activate the disaster when this mission bit is set

• The oütf resource

Outf resources store information on the items that you can buy when you choose "Outfit Ship" at a planet or station. The first field tells EV whether or not the item's availability is linked to the completion of a mission:

MissionBit   Tells EV whether to offer this item only if a certain mission bit is set. (see mïsn for more info)

-1	Ignored
0-1255	Offer the item only if this mission bit is set

The next field contains info on the item's mass:

Mass   The mass in tons of the item (0 = no appreciable mass)

The next field tells EV where the item is available:

TechLevel   What the technology level of the item is. This item will be available at all spaceports with a tech level of this value or higher. (The exception to this rule invloves the SpecialTech fields of the spöb resource; see the section on spöb resources for more information.)

The next two fields tell EV what kind of modification this item performs:

If ModType is:	Then it's:	And ModVal refers to:
1	a weapon	The ID number of the associated wëap resource
2	more cargo space	The number of tons of cargo space to add
3	ammunition	The ID number of the associated wëap resource
4	more shield capacity	The number of shield points to add
5	faster shield recharge	How much to speed up (-30 = one more point per second)
6	armor	The number of armor points to add
7	acceleration booster	Amount of accel to add (see shïp for more info)
8	speed increase	Amount of speed to add (see shïp for more info)
9	turn rate increase	Amount of turn increase (1 = 30°/sec)
10	ECM (missile jammer)	ignored
11	escape pod	ignored
12	fuel capacity increase	Amount of extra fuel (100 = 1 jump)
13	density scanner	ignored
14	IFF (colorized radar)	ignored
15	afterburner	ignored
16	map	How many jumps away from present system to explore
17	cloaking device	The hue to scale the screen to while cloaked 1 = red, 2 = green, 3 = blue, 4 = cyan, 5 = magenta, 6 = yellow
18	fuel scoop	How many frames per 1 unit of fuel generated
19	auto-refueller	ignored
20	auto-eject (req. escape pod)	ignored
21	clean legal record	ID of govt to clear legal record with, or -1 for all

22	hyperspace speed mod	Number of days to increase or decrease ship's hyperspace travel time (still can't go below 1 day/jump)
23	hyperspace dist mod	Amount to increase or decrease the no-jump zone's radius by (the standard radius is 1000)

The next two fields tell EV how many of this item you can possibly have at once:

Max How many you can have (not counting weapon limitations)

Flags Miscallenous info:
0x0001 This item is a fixed gun
0x0002 This item is a turret
0x0004 This item stays with you when you trade ships

The last field, Cost, tells EV how much to charge you for the item.

Meanwhile, on the forest moon of Endor...

• The përs resource

The pers resource defines the characteristics of an AI personality - that is, a specific person the player can encounter in the game. These AI-people have their names (which are also the names of the associated përs resource) displayed on the target-info display in place of the name of their ship class. When ships are created, there is a 5% chance that a specific AI–person will also be created. (obviously, as AI-people are killed off, they cease to appear in the game.) The first field tells EV where a certain person can be encountered:

LinkSyst   Which systems the person can be created in

-1	Any system
128-1127	ID of a specific system
9999-10127	Any system belonging to this specific government
15000-15127	Any system belonging to an ally of this govt
20000-20127	Any system belonging to any but this govt
25000-25127	Any system belonging to an enemy of this govt

The next four fields define the person's character traits:

Govt   The person's governmental affiliation

-1	Ignored (person is independent)
128-255	ID of a specific government

AI Type   The person's AI type (see the section on düde resources)

1	Wimpy trader
2	Brave trader
3	Warship
4	Interceptor

Agress   The person's agression, i.e. how close ships have to be before the person will attack them, on a scale of 1 to 3.

Coward   At what percent of total shield capacity will the person run away from a fight? e.g. a value of 25 would cause the person to retreat when his shields dropped to 25%.

The next fields tell EV more about the ship that a person uses:

ShipType   ID number of the person's ship class

WeapType (x4)   ID numbers of weapon types

-1 or 0	No weapon
128-191	Add this weapon type

WeapCount (x4) How many of each of the above weapons to add (Note: This is in addition to the standard weapons already included with the ship. Standard weapons can be "removed" by entering their ID numbers in the WeapType fields and entering the negative of their standard load for the given ship class in the WeapCount field.)

-1 or 0	None

| | 1 and up | Add this many |

AmmoLoad (x4)	The standard ammo load for weapons that need it, or ignored for those that don't	
	-1 or 0	No ammo
	1 and up	Include this many rounds of ammo

Credits	How many credits the person caried	
	0	ignored (no credits)
	1 and up	This many credits, ± 25%

ShieldMod — How much to increase/decrease the person's shield capacity, in percent. For example, a value of 130 entered here would make the person's ship have shields that are 30% stronger than if he were flying a stock ship. Similarly, a value of 70 would make his shields 30% weaker.

MissionBit	Which mission bit this person is linked to.	
	0-255	Only have this person show up if this mission bit is set
	-1	Ignored

The next fields tell EV about any special quotes or missions to link to this ship:

CommQuote — Index number of an entry in STR# resource 7100, to be displayed in the communications dialog

HailQuote — Index number of an entry in STR# resource 7101, to be displayed at the bottom of the game screen (i.e. over the radio)

LinkMission — What mission to activate when the ship is boarded or hailed

Flags	Some control bits	
	0x0001	The special ship will hold a grudge if attacked, and will subsequently attack the player wherever the twain shall meet.
	0x0002	Uses escape pod
	0x0004	HailQuote only shown when ship has a grudge against the player
	0x0008	HailQuote only shown when ship likes player
	0x0010	Only show HailQuote when ship begins to attack the player
	0x0020	Only show HailQuote when ship is disabled
	0x0040	When LinkMission is accepted with a single SpecialShip, replace it with this ship while removing this one from play (only useful for escort missions)
	0x0080	Only show quote once
	0x0100	Deactivate ship (i.e. don't make it show up again) after accepting its LinkMission

0x0200	Offer ship's LinkMission when boarding it instead of when hailing it
0x0400	Don't show quote when ship's LinkMission is not available
0x0800	Make ship leave after accepting its LinkMission
0x1000	Don't offer if player is flying a wimpy freighter (aiType 1)
0x2000	Don't offer if player is flying a beefy freighter (aiType 2)
0x4000	Don't offer if player is flying a warship (aiType 3)
0x8000	Show disaster info when hailing

• The shïp resource

Spaceships are the heart of EV, so the ship resource contains a lot of info. The name of a ship class, which is seen in the targeting display, corresponds to the name of the ship resource. The first nine fields give EV some general performance info on each ship type:

Holds — Cargo capacity, in tons. Put a negative sign in front of this value if you want to prevent the player from purchasing mass expansions. (e.g. a value of -100 would mean 100 tons of hold space but no mass expansions allowed)

Shield — Shield strength. (Player's shield is a bit stronger than this)

Accel — Acceleration magnitude. 300 is considered an average value.

Speed — Top speed. 300 is also an average value here.

Maneuver — Turn rate. $1 \approx 30°/sec.$

Fuel — Fuel capacity. 100 = 1 jump.

FreeMass — Space available to add additional items and upgrades. Note that this is in addition to the space taken up by the ship's stock weapons. (e.g. a ship with 20 tons listed in FreeMass and 10 tons of stock weapons will actually have 30 tons of expansion space, with 20 available.)

Armor — Armor strength. Armor takes damage when shields go down, and when a ship's is 2/3 gone it becomes disabled.

ShieldRe — Shield recharge speed, in number of frames per shield percentage point regenerated; bigger numbers here make for slower recharging. $30 \approx 1\%$ per second.

The next twelve fields tell EV which stock weapons to put on your ship when you first buy it:

WeapType (x4) — ID numbers of weapon types
-1 or 0 — No weapon
128-191 — Add this weapon type

WeapCount (x4) — How many of each of the above weapons to add
-1 or 0 — None
1 and up — Add this many

AmmoLoad (x4) — The standard ammo load for weapons that need it, or ignored for those that don't
-1 or 0 — No ammo
1 and up — Include this many rounds of ammo

The next two fields tell EV what this ship's maximum loadout of fixed guns and turreted weapons is. Each ship has an inherent upper limit on fixed guns and turrets, in order to keep them from becoming absurdly powerful. (e.g. a bulk freighter has lots of room to add weapons, but is limited to a single turret for defense) The fields are:

MaxGun   The ship's maximum number of fixed guns, which are flagged in the WeapFlag field of the outf resource.

MaxTur   The ship's maximum number of turrets, which are flagged in the WeapFlag field of the outf resource.

The next field tells EV where this ship is available for purchase:

TechLevel   What the technology level of the ship is. This ship will be available at all shipyards with a tech level of this value or higher. (The exception to this rule invloves the SpecialTech fields of the spöb resource; see the section on spöb resources for more information.)

The next field, Cost, tells EV how much to charge you when you buy this ship. The cost of buying a ship is always the cost of the new ship minus 25% of the original cost of your current ship and upgrades. (i.e. you always "trade up" to a new ship)

The next field stores info on how the ship explodes:

DeathDelay   The number of frames the ship "disintegrates" before finally exploding.
0-59   The ship disintegrates for this number of frames and then disappears in a single fireball.
60+   The ship disintegrates for this number of frames and then disappears in a huge explosion. The exact size of the resulting fireball is proportional to the ship's mass. (see below)

The next four fields tell EV where the ship's turrets should fire from:

TurretYDisp (x4)   Exactly where on the Y-axis of the ship the turrets should fire from. EV cycles through each of these four values in turn as the turrets are fired.
0   Fire from exact center of the ship graphic
1 and up   Fire from this many pixels forward of center
-1 and below Fire from this many pixels aft of center

The next two fields store info on the physical dimensions of the ship:

Mass   The mass of the ship, in tons. This doesn't affect acceleration or speed at all, but it does affect travel time in hyperspace and the display on the density scanner. Also, the blast radius and

impact strength when the ship explodes is proportional to its mass.

1-99	1 day per jump, small blip on density scanner
100-199	2 days per jump, large blip on density scanner
200 and up	3 days per jump, large blip on density scanner

Length   The ship's length in meters. Currently unused in any calculations, but it's kinda cool, so it's displayed in the "detailed ship info" dialog.

The next field tells EV what kind of AI the ship will have if it's not created in connection with a dude resource. The only place this field is useful is when a ship is created as an escort ship; otherwise, it's ignored:

InherentAI   What AI the ship uses when it's escorting the player.

1-3	Use this kind of AI. (see the AI descriptions above)

The next field contains the number of crew members that are on each type of ship, and the field after that tells EV whether or not the item's availability is linked to the completion of a mission:

MissionBit   Tells EV whether to offer this ship only if a certain mission bit is set. (see mïsn for more info)

-1	Ignored
0-255	Offer the ship only if this mission bit is set

The next field tells EV what government is associated with a ship type:

InherentGovt	-1	No inherent govt for this ship
	128-255	ID of a government to link to this ship

• The spöb resource

Spob resources describe stellar objects, such as planets and space stations. (spob stands for space object) Each spob resource represents one stellar object, whose name is the name as the name of the resource. The first three fields tell EV where to put the stellar and what graphics to use for it:

xPos & yPos   The stellar's X and Y positions in the system (0, 0) is centered
Type   Which graphic to use, from 0 to 63.

The next field stores the flag bits that tell EV what many of the characteristics of the stellar are. Perform an OR operation on the following flags to get the final flag value:

0x00000001	Can land/dock here
0x00000002	Has commodity exchange
0x00000004	Can outfit ship here
0x00000008	Can buy ships here
0x00000010	Stellar is a station instead of a planet
0x00000020	Stellar is uninhabited (no traffic control)
0x00000040	Has bar
0x00000000	Won't trade in food
0x10000000	Low food prices
0x20000000	Medium food prices
0x40000000	High food prices
0x00000000	Won't trade in industrial goods
0x01000000	Low industrial prices
0x02000000	Medium industrial prices
0x04000000	High industrial prices
0x00000000	Won't trade in medical supplies
0x00100000	Low medical prices
0x00200000	Medium medical prices
0x00400000	High medical prices
0x00000000	Won't trade in luxury goods
0x00010000	Low luxury prices
0x00020000	Medium luxury prices
0x00040000	High luxury prices
0x00000000	Won't trade in metal
0x00001000	Low metal prices
0x00002000	Medium metal prices
0x00004000	High metal prices
0x00000000	Won't trade in equipment
0x00000100	Low equipment prices
0x00000200	Medium equipment prices
0x00000400	High equipment prices

The next field, System contains the ID number of the star system that the stellar object is in. The four flags after that tell EV what items and ships are available for purchase at this stellar:

TechLevel   What the base tech level of the stellar is. Only items and ships
                                 with TechLevels at or below this value will be available.

SpecialTech (x3)	Holds the special tech levels of this stellar. Unlike the previous field, only items and ships with exactly this TechLevel will appear here. This is useful for making low-tech worlds that also have a few high-tech items, or for flagging an item with an absurdly high TechLevel (say 15000) and then setting one of the SpecialTech fields of a particular stellar to that same value, thus making that item appear at that stellar and nowhere else.

The next two fields contain info on the stellar's governmental affiliation:

Govt	What government controls this stellar	
	-1	ignored (stellar is independent)
	128-255	ID number of the stellar's government

MinCoolness	The point on your record in the current system that you'll be denied landing clearance on this stellar.	
	-1 and below	You can be this evil before they shun you
	0 and up	They have to like you this much before they let you land

The next pair of fields tells EV which special landscape to show and which ambient sound to play.

CustPicID	Which custom landscape (333x271 PICT resource) to show	
	-1	No custom landscape
	Anything else	ID number of PICT to load instead of the standard landscape display

CustSndID	Which ambient sound to play	
	-1	No ambient sound effect
	Anything else	ID number of snd resource to load (must be 11kHz)

The last two fields tell EV what kind of ships, if any, to create for the planet's defense fleet:

DefenseDude	Which type of dude to use for the defense fleet:	
	-1	Ignored (no defense ships)
	128-255	ID number of the dude resource to use to determine the defense ships' characteristics

DefCount	The number of ships in the defense fleet. If you set this number to be above 1000, ships will be launched from the planet or station in waves. The last number in this field is the number of ships in each wave, and the first 3-4 numbers (minus 1 from the first digit) are the total number of ships in the planet's fleet. For example, a value of 1082 would be four waves of two ships for a total of eight. A value of 2005 would create waves

of five ships each, with 100 ships total in the planet's defense fleet.

• The sÿst resource

Syst resources store information on star systems, in which all combat, trading, and spaceflight take place. Each system can be linked to up to 16 other systems, and the player can make hyperspace jumps back and forth between them. System ID #128 also has a special signifigance: that system is where the player starts out when first beginning the game and if he subsequently ejects and is rescued. This system should always be made neutral, so the player isn't pursued by big beefy ships when he's just starting out.

The first two fields in the syst resource tell EV where on the map to place it:

xPos & yPos	The system's X and Y positions on the map

The next five fields store the hyperspace links to 5 other systems (the other 11 are at the bottom of the resource)

Con1-Con5	Link to another system	
	-1	No link
	128-1127	ID of a system to link to

The next fields store the stellar navigation defaults (F1-F4) for the system. It is important to always set navigation defaults for stellar objects in your systems, because that's how EV's AI routines and status display keep track of stellar objects; if you don't set a planet as a nav default, the AIs won't "see" it, it won't show up on the radar, and you can't select it.

NavDef (x4)	Navigation defaults (F1-F4)	
	-1	No nav default for this key
	128-1627	ID number of the stellar object to set as a default

The next nine fields tell EV how many ships, and of what kind, to put in the system:

DudeTypes (x4)	Which type of dude to place (best not to set this to an out-of–range value)	
	128-255	ID number of the dude type to place
% Prob (x4)	Probability that a given ship placed is of each of the above dude types	
	1-99	This percent probability
AvgShips	The average number of AI ships in the system	
	0	No ships, empty system
	1 and up	This number of ships, ± 50%

One special note: If you want a syst resource's DudeType field to point to a fleet type (see above) instead of a dude type, set its value to the negative of the fleet's resource ID. This will force EV to include that specific fleet type in the system a definite percentage of the time, instead of haphazardly as it normally does.

The next field tells EV who controls the system:

Govt   Which government owns the system
                        -1   Ignored (system is independent)
                        128-255   ID number of the controlling govt

The next tells EV which string, if any, to display as the message buoy's message when you enter a system:

Message   Which message buoy string to display
                        -1   Ignored (no special message)
                        1 and up   Use this entry in STR# resource 1000 as the
                                    text of the message buoy

The next two fields tell EV what kinds of navigation hazards to put in the system:

Asteroids   How many asteroids to put in the system, from 0 to 10

Interference   How thick the static in the system should be. 0 is no static, 100
                        is complete sensor blackout.

The next field controls how and when to make the system visible or invisble. You can pull off some cool tricks with this field, including presenting the illusion that system has changed in some way by hiding the original system and replacing it with a copy that is identical except for the desired changes.

VisBit   Which mission bit controls the system's visibility
                        -1   Ignored (system is always visible)
                        0-255   Make the system visible only when this mission
                                    bit is set
                        1000-1255   Make the system visible only when this mission
                                    bit is cleared

The next eleven fields store the hyperspace links to the other 11 systems that we didn't decide to add until EV 1.0.2 came out

Con6-Con16   Link to another system
                        -1   No link
                        128-1127   ID of a system to link to

• The wëap resource

The weap resource, surprisingly, stores info on EV's weapons. The name of the weap resource is used as the weapon name in the weaponry section of the status display. The first two fields control the duration of different aspects of the weapon:

Reload	The number of frames it takes for one of this weapon to reload. 30 = 1 shot/sec. Smaller numbers yield faster reloads.
Count	The number of frames the weapon's shots travel for before they peter out. 30 = 1 second of life.

The next two fields, MassDmg and EnergyDmg, tell EV how much damage to do when one of this weapon's shots hits something:

If the ship's shields are down: damage = MassDmg + (EnergyDmg/4)
If the ship's shields are up: damage = (MassDmg/4) + EnergyDmg

However, the weapon will always do at least one point of damage, regardless of the calculation above.

The next two fields tell EV how the weapon should behave in flight:

Guidance	The weapon's guidance mode	
	-1	Unguided projectile
	0	Beam weapon (see below)
	1	Dumb homing weapon (affected by sensor interference)
	2	Smart homing weapon (affected by ECM and asteroids)
	3	Turreted beam
	4	Turreted, unguided projectile
	5	Freefall bomb (launched at half the ship's current velocity, "weathervanes" into the "wind."
	6	Freeflight rocket (launched straight ahead, accelerates to its maximum velocity)
	7	Front-quadrant turret, (can fire ±45° off the ship's nose) fires straight ahead if no target
	8	Rear-quadrant turret (can fire ±45° off the ship's tail)
	99	Carried ship (AmmoType is the ID of the ship class)
Speed	The weapon's speed (pixels per frame * 100)	

The next field tells EV how to handle the ammunition for this weapon, assuming it's not a fighter bay:

AmmoType	What kind of ammo the weapon uses

-1	Ignored (unlimited ammo)
0-63	Draws ammo from this type of weapon. (Usually, if your Hector Cannon was of ID 131, you'd set the AmmoType to 3 so it'd use Hector Birdseed Pellets or whatever. However, you could conceivably set it to use ammo from another weapon's supply by setting the AmmoType to something else.)
-1000 & below	Weapon uses abs(AmmoType+1000) units of fuel per shot.

The next three fields tell EV which graphic and sound to use for this weapon, and how to launch it:

Graphic	What graphic set to use for this weapon
0-63	Use this graphic set (i.e. spin resources 200–263)

Inaccuracy	The weapon's inaccuracy as it leaves the ship
0	Fires straight
1 and up	Fires with up to this amount of inaccuracy (in degrees)

Sound	Which sound to play when the weapon fires
-1	Silent but deadly
0-63	Play this sound (snd ID 200-263)

The next four fields store info on how the weapon behaves when it hits something:

Impact	The magnitude of the impact when the shot hits something
0	No impact
1 and up	This amount of impact, which is inversely proportional to the ship's mass. (Missile = 30)

ExplodType	What kind of explosion to show when the weapon hits.
-1	No explosion
0	Small, sparkly explosion
1	Bigger explosion
2	Friggin' huge explosion + little sparks

ProxRadius	The radius of the weapon's proximity fuse (useful for unguided missiles and bombs)
0	Weapon requires direct hit to do damage
1 and up	This number of pixels of proximity radius

BlastRadius	The radius of the weapon's blast effect
0	No blast effect
1 and up	This number of pixels of blast radius

The last field contains some miscallenous flag info:

Flags	0x0001	Spin the weapon's graphic continuously
	0x0002	Weapon fired by second trigger
	0x0010	Weapon's sound is looped rather than played repeatedly
	0x0020	Weapon acts as a decoy for missiles
	0x0040	Multiple weapons of this type fire simultaneously
	0x0100	Weapon's blast doesn't hurt the player

Please note that if you've set the weapon to be a beam (Guidance of 0 or 3) the following fields have different functions:

Count	The number of frames the beam stays onscreen (1 is probably best)
Speed	The length of the beam (32767 is the max)
Graphic	What color the beam is

-2	red
-3	green
-4	blue
-5	cyan
-6	magenta
-7	yellow

If the beam is 3 pixels or wider, you can also use the following:

-8	magenta & red
-9	yellow & green
-10	cyan & blue

Impact	Functions normally, with one exception: if the impact is set to a negative value, the beam acts as a tractor beam whose "pull" is proportional to the acceleration strength of your ship's engines.
ProxRadius	The beam's width, in pixels. (must be greater than zero unless you want to confuse the user)
BlastRadius	Ignored

Note: You shouldn't plan on using the last weap resource (ID 163) in your game, as it's used by EV as the easter egg weapon. (i.e. it's the weapon the player gets by hitting the special key combo on the main screen)

Another Note: If you don't create an oütf resource for each weapon type, your ship's weapon loadout will be corrupted when you land on a planet. (specifically, EV will mistakenly remove any weapons for which you didn't create oütf resources)

Appendix I - Combat Ratings

Your combat rating is based on the number of kills you have made, which is the sum of the crew complements of all the ships you have destroyed. The scale is as follows:

Kills:	Rating:
0	Harmless
1	Mostly Harmless
100	Fair
200	Average
400	Above Average
800	Competent
1,600	Noteworthy
3,200	Excellent
6,400	Dangerous
12,800	Deadly
25,600	Ultimate

Appendix II - Legal Status

Your legal status in a system is based on the crime tolerance of that system's government. (if the system is independent, it is based on the first government's [ID 128] crime tolerance) On this scale, enough "good" or "evil" points to equal the government's crime tolerance is given a value of 1:

Good Scale:	Legal Status:
0	Clean
4	Decent Individual
16	Good Egg
64	Upstanding Citizen
256	Role Model
1024	Pillar of Society
4096	Honored Leader

Evil Scale:	Legal Status:
0	Clean
1	Offender
4	Criminal
16	Felon
64	Fugitive
256	Public Enemy
1024	Prime Evil
4096	Galactic Scourge

Appendix III - Patching STR# Resources

The STR# resource format used to store many of the strings in EV may seem to be incompatible with the flexible nature of plugin files. For example, a plugin for a new ship would have to replace several of the built-in STR# resources to incorporate its new name into the game. The problem arises when you want to use two plugins that try to modify the same STR# resource.

The solution is not to changes the STR# resources at all, but to use EV's handy string patching functionality by updating only select strings in a STR# resource by providing EV with a properly-numbered 'STR ' resource. For example, to change the first cargo type from food to something else, you'd simply create a 'STR ' resource with the ID 9000 and type in the name of your new commodity. A chart of 'STR ' resource numbers is provided below:

String Type:	Replacement 'STR ' ID range:
Message buoys	1000-2500
Short outfit names	3000-3127
Lowercase outfit names	3200-3327
Lowercase plural outfit names	3400-3527
Short shipyard names	3600-3663
Long shipyard names	3700-3763
Short ship names for communications dialog	3800-3863
Government abbreviations for target display	4000-4127
Government short names for comm dialog	4200-4327
Hail quotes	5000-5511
Disaster news	6000-6127
Stellar types	7000-7063
jünk abbreviations	8000-8127
Lowercase jünk names	8200-8327
Generic cargo names	9000-9063
Lowercase cargo names	9100-9163
Cargo abbreviations for status display	9200-9263
Base prices of commodities	9300-9305
Commodity abbreviations for status display	9400-9405

(A similar function exists for updating the graphics for the shipyard and outfit menus - see the note at the top of this document for details.)

The End

Escape Velocity by…	Matt Burch
Documentation by…	Matt Burch
Crashes by…	MetroWerks Corporation Apple Computer, Inc.
Clever announcements by…	MacsBug
Good ideas by…	Sheer Accident
Special Assistant to Mr. Burch…	Er, nobody really
Stunt Coordinator…	Patrick Delahanty

Filmed in Quadra-Vision
Using Genuine Apple Lenses

Soundtrack available in the lobby

In Stereo Where Available

Void Where Prohibited