

BioWare Aurora Engine

Faction System

1. Introduction

A **Faction** is a control system for determining how game objects interact with each other in terms of friendly, neutral, and hostile reactions.

Faction information is stored in the **repute.fac** file in a module or savegame. This file uses BioWare's Generic File Format (GFF), and it is assumed that the reader of this document is familiar with GFF. The GFF FileType string in the header of repute.fac is "FAC".

2. Faction System Structs

The tables in this section describe the GFF Structs contained within repute.fac.

2.1. Top Level Struct

Table 2.1: Faction Top Level Struct

Label	Type	Description
FactionList	List	List of Faction Structs (StructID = list index). Defines what Factions exist in the module.
RepList	List	List of Reputation Structs (StructID = list index). Defines how each Faction stands with every other Faction.

2.1.2. Faction Struct

The Table below lists the Fields that are present in a Faction Struct found in the *FactionList*.

Table 2.1.2.1: Fields in Faction Struct (StructID = list index)

Label	Type	Description
FactionGlobal	WORD	Global Effect flag. 1 if all members of this faction immediately change their standings with respect to another faction if just one member of this faction changes it standings. For example, if killing one Guard Faction member causes all Gaurd Faction members to hate you, then the Guard Faction is Global. 0 if other members of a faction do not change their standings in response to a change in a single member. For example, killing a deer will not cause all deer to hate you.
FactionName	CExoString	Name of the Faction.
FactionParentID	DWORD	Index into the Top Level Struct's <i>FactionList</i> specifying the Faction from which this Faction was derived. The first four standard factions (PC, Hostile, Commoner, and Merchant) have no parents, and use 0xFFFFFFFF as their <i>FactionParentID</i> . No other

		Factions can use this value.
--	--	------------------------------

2.1.3. Reputation Struct

The Table below lists the Fields that are present in a Reputation Struct found in the *RepList*. Each Reputation Struct describes how one faction feels about another faction. Feelings need not be mutual.

For example, Exterminators might be hostile to Rats, but Rats may be neutral to Exterminators, so that a Rat would only attack a Hunter or run away from a Hunter if a Hunter attacked the Rat first.

Table 2.1.3.1: Fields in Reputation Struct (StructID = list index)

Label	Type	Description
FactionID1	DWORD	Index into the Top-Level Struct's <i>FactionList</i> . "Faction1"
FactionID2	DWORD	Index into the Top-Level Struct's <i>FactionList</i> . "Faction2"
FactionRep	DWORD	How Faction2 perceives Faction1. 0-10 = Faction 2 is hostile to Faction1 11-89 = Faction 2 is neutral to Faction1 90-100 = Faction 2 is friendly to Faction1

For the *RepList* to be exhaustively complete, it requires $N*N$ elements, where N = the number of elements in the *FactionList*.

However, the way that the PC Faction feels about any other faction is actually meaningless, because PCs are player-controlled and not subject to faction-based AI reactions. Therefore, any Reputation Struct where FactionID2 == 0 (ie, PC) is not strictly necessary, and can therefore be omitted from the *RepList*.

Thus, we revise our original statement and say that for the *RepList* to be sufficiently complete, it requires $N*N - N$ elements, where N = the number of elements in the *FactionList*, assuming that one of those Factions is the PC Faction.

In practice, however, the *RepList* may contain anywhere from $(N*N - N)$ to $(N*N - 1)$ elements, due to a small idiosyncrasy in how the toolset generates and saves the list. When a new faction is created, up to two new entries appear for the PC Faction.

If a Faction Struct does not yet exist for the feelings of the PC Faction toward the new faction's parent, then that Struct is created:

FactionID1	<Parent ID>
FactionID2	0
FactionRep	100

Regardless of whether the above was created or already existed, a new Faction Struct is created for how the PC Faction feels about the new faction.

FactionID1	<New ID>
FactionID2	0
FactionRep	100

The reputations are set to 100 in both cases, but remember that the actual reputation value does not matter if FactionID2 is 0.

From all the above, it follows that a module that contains no user-defined factions will have exactly $N*N - N$ Faction Structs, where $N = 5$. Modules containing user-defined factions will have more. The

maximum number of Faction Structs in the *RepList* is $N*N - 1$, because the Player Faction itself can never be a parent faction.

3. Faction-related 2DA Files

3.1. Default Faction Standings

Table 3.1: repuete.2da

Column	Type	Description
LABEL	String	programmer label; name of faction being considered by the faction named in each of the other columns. Row number is the faction ID. The rows are: Player, Hostile, Commoner, Merchant, Defender. Do not add new rows. They will be ignored.
HOSTILE	Integer	How the Hostile faction feels about the other factions
COMMONER	Integer	How the Commoner faction feels about the other factions
MERCHANT	Integer	How the Merchant faction feels about the other factions
DEFENDER	Integer	How the Defender faction feels about the other factions

3.2. Reputation Adjustment

The file **repadjust.2da** describes how faction reputation standings change in response to different faction-affecting actions, how the presence of witnesses affects the changes, and by how much the changes occur.

Note that certain things affect whether a witness does in fact serve as a witness for its own faction, including whether the witness is dominated, charmed, is a henchman, or is some other associate, as well as what faction the master belongs to. These considerations are not part of the Faction file format, however, and are not discussed further here.

Table 3.2: repadjust.2da

Column	Type	Description
LABEL	String	programmer label; name of an action. The rows are: Attack, Theft, Kill, Help. These action types are hardcoded game constants. Do not change the order of rows in this 2da. Adding new rows will have no effect.
PERSONALREP	Integer	Personal reputation adjustment of how the target feels about the perpetrator of the action named in the <i>LABEL</i> .
FACTIONREP	Integer	Base faction reputation adjustment in how the target's Faction feels about the perpetrator. This reputation adjustment is modified further by the effect of witnesses , as controlled by the columns described below. Note that a witnesses only affects faction standing if the witness belongs to a Global faction.
WITFRIA	Integer	Friendly witness target faction reputation adjustment. If there is a witness from a <i>global</i> faction that is friendly to the target of the action, then adjust the target's faction adjustment by this amount.
WITFRIB	Integer	Friendly witness personal reputation adjustment.

		If there is a witness from a faction that is friendly to the target of the action, then adjust the witness's personal reputation standing toward perpetrator by this amount.
WITFRIC	Integer	Friendly witness faction reputation adjustment. If there is a witness from a <i>global</i> faction that is friendly to the target of the action, then adjust the witness's faction standing toward the perpetrator by this amount.
WITNEUA	Integer	Neutral witness target faction reputation adjustment. If there is a witness from a <i>global</i> faction that is neutral to the target of the action, then adjust the target's faction adjustment by this amount. Do not use this if a friendly global witness was found.
WITNEUB	Integer	Neutral witness personal reputation adjustment. If there is a witness from a faction that is neutral to the target of the action, then adjust the witness's personal reputation standing toward perpetrator by this amount.
WITNEUC	Integer	Neutral witness faction reputation adjustment. If there is a witness from a <i>global</i> faction that is neutral to the target of the action, then adjust the witness's faction standing toward the perpetrator by this amount.
WITNEEA	Integer	Enemy witness target faction reputation adjustment. If there is a witness from a <i>global</i> faction that is an enemy of the target of the action, then adjust the target's faction adjustment by this amount. Do not do this if there is already a friendly or neutral global witness.
WITNEEB	Integer	Enemy witness personal reputation adjustment. If there is a witness from a faction that is hostile to the target of the action, then adjust the witness's personal reputation standing toward perpetrator by this amount.
WITNEEC	Integer	Enemy witness faction reputation adjustment. If there is a witness from a <i>global</i> faction that is hostile to the target of the action, then adjust the witness's faction standing toward the perpetrator by this amount.