INTERFACE CONTROL DOCUMENT for the COMMON IMAGE GENERATOR INTERFACE (CIGI)

Version 2.0: 8 March, 2002

Report Number: TST02I015

W. B. Phelps

PO Box 516, St. Louis, MO 63166 Phone (314) 232-8888 Fax (314) 232 2187 e-mail willard.b.phelps@boeing.com

LIST OF PAGES

USE AND DISCLOSURE OF DATA

This document is intended to be a public domain document. As such it does not contain any proprietary notices. The Boeing Company may make improvements and changes to the contents of this document at any time without notice. The Boeing Company assumes no responsibility for the use of the information in this document. This document may contain technical inaccuracies or typographical errors. Periodic changes are made to the information contained herein: these changes will be incorporated in new editions of the document. To maintain configuration control of the interface it is requested that this document not be reproduced in part for the purpose of modification to the interface. Please forward all modification suggestions to the maintainer shown at the bottom of the cover page for discussion and possible incorporation into the interface.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INDEX OF CHANGE PAGES						
Revision Letter	Pages Affected			Remarks	Revised By	Approval
	Revised	Added	Removed			
New					W. B. Phelps	
1.0	all			Serious updates	W. B. Phelps	
2.0	Cover pages			Added note for copying and redistribution authority	W. B. Phelps	
	1, 2			Re-write of introduction material	W. B. Phelps	
	3, 4			Re-write of interface theory material	W. B. Phelps	
	5, 98, 99			Added User definable data packets	W. B. Phelps	
	5,6			Added new packet titles to data packet identification table	W. B. Phelps	
	8, 10			Added Basic Definitions, Principles and Nomenclature section	W. B. Phelps	
	15			Updated data packet description section	W. B. Phelps	
	16, 82			Changed version number to 2	W. B. Phelps	
	16, 17			Added timing value to IG Control data packet	W. B. Phelps	
	18, 19			Re-wrote Entity Control data packet narrative	W. B. Phelps	
	21, 25, 72, 73, 74, 75, 77, 78, 85, 86, 87, 88			Changed all altitudes to double float values and adjusted packet padding accordingly	W. B. Phelps	
	21, 24			Added Percent Transparency to Entity Control data packet	W. B. Phelps	
	21, 22, 24			Redefined Entity state and effects state switches in the Entity Control data packet	W. B. Phelps	
	27, 28, 29			Added examples of various uses of Component controls	W. B. Phelps	
	30, 31			Added Sensor ID to the Component Control data packet.	W. B. Phelps	
	33, 34, 35, 36			Changed Articulated Parts offset and rotational values from fixed point to floating point values	W. B. Phelps	
	41, 42, 43			Introduce MODTRAN into the Environmental Control data packet	W. B. Phelps	
	44			Added diagram to Weather Control data packet narrative	W. B. Phelps	
	55, 56			Removed Offsets in Trajectory definition data packet already accommodated in the Entity Control data packet.	W. B. Phelps	
	65, 66, 67			Renamed Collision Detection Definition to Collision Detection Segment Definition to distinguish it from Swept Volume technique and clarified narrative	W. B. Phelps	
	65, 67			Redefined Material Mask to Collision Mask	W. B. Phelps	
	69, 70, 71			Added Swept Volume Collision Detection Definition data packet	W. B. Phelps	
	74, 75, 76			Added clarification for the LOS occult request	W. B. Phelps	
	80, 81			Added HOT request data packet.	W. B. Phelps	
	82, 84			Added timing value to Start of Frame data packet	W. B. Phelps	
	89, 90			Renamed Collision Detection Response to Collision Detection Segment Response to distinguish it from Swept Volume technique	W. B. Phelps	
	93, 94			Added HOT response data packet	W. B. Phelps	
	95, 96			Added Swept Volume Collision Detection Response data packet	W. B. Phelps	
	97			Added Image Generator Response Massage data packet	W. B. Phelps	
	Many			Corrected grammatical errors	W. B. Phelps	

1. Scope

This document describes the open-source Common Image Generator Interface. This document does not contain any proprietary notices as this interface is intended for unrestricted public use.

1.1 Purpose

This Interface Control Document (ICD) is to be used by software engineers to aid in the integration of an image generator (IG) with a Host Simulator using the Common Image Generator Interface (CIGI). This document contains a description of all data parameters, event sequences, and Input / Output (I/O) protocols necessary to accomplish this task.

This interface is meant to be generic in nature and provide a Host simulator the capability to communicate with an image generator equipped with the CIGI. As such, typical IG control functions are provided, but unique control functions are absent. A generic control (i.e. Component Control) is provided for unique functions such that given a mapping of the specific controls or functions to the generic control, the integration engineer has all the information necessary to program these functions. This document should be accompanied by a control function definition document for these unique controls. That document should contain the function identification and parameter value assignments pertinent to a given control function. A majority of these functions may have default values programmed on the IG that are sufficient as defined and may never require alteration by the host.

The CIGI provides controls to manage entities by type designation. In order to complete the integration of the host with the IG, the type assignment needs to be defined. In addition, specific terrain features and terrain sets may have controls and IDs that must be known. These identifications should be captured in a database definition document. That document should contain parameter value assignments and peculiarities pertinent to a given database and the entities, moving or stationary, used with it.

1.2 Instructions for revising this document

For version correlation purposes this interface has the CIGI version number in the IG Control and Start of Frame data packets. This means that any time a change is made to this document that affects data formatting the designated version number in these blocks must be incremented.

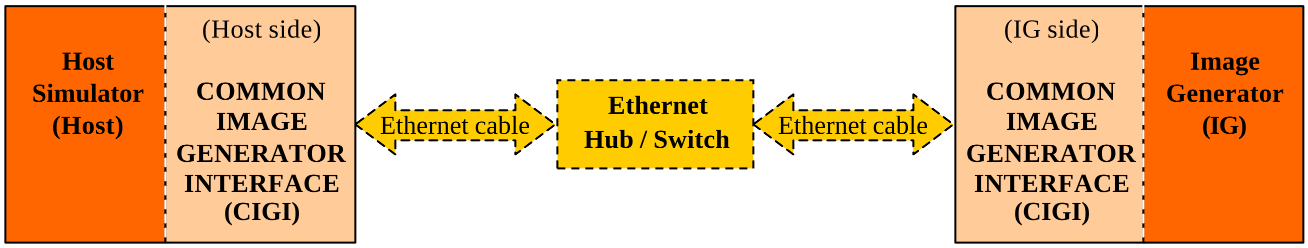
1.3 Introduction

The Host simulator discussed herein, also referred to as the "Host," communicates with the IG via a bi-directional Ethernet connection. The data contained in these communications consists of information to perform data synchronization and mission scenarios. The data are formatted per the CIGI protocol. Each of these data and their associated formats are contained within the CIGI will be explained and discussed in this document.

The CIGI is a data packaging protocol. To date it has been implemented using an Ethernet medium. That is not to say that the CIGI is limited to Ethernet applications. The communication medium can be whatever is appropriate for the situation; an optical interface, mirrored memory, shared memory, etc. The following description assumes the use of Ethernet for ease of discussion. It should not be difficult for implementers to imagine how the CIGI would be implemented using other communication medium. As with any formally configured application, the version of the communications medium must be established among the users. For the Ethernet implementation of the CIGI, the standard is Internet Protocol Specification RFC 791 and UDP RFC 768.

The Ethernet connection between the Host and the IG should be a dedicated Ethernet connection, as illustrated in Figure 1. The connection may be made using a single crossover Ethernet cable (i.e. were the transmit and receive signals are swapped so that an Ethernet hub or switch is not required) or via an Ethernet hub or switch. The IP address and send and receive ports for each device are configurable.

Connection using Ethernet Hub or Switch



Connection using Cross-over Ethernet Cable

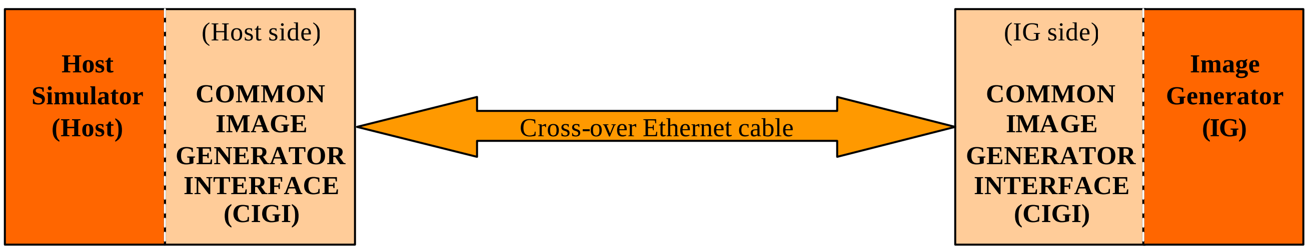


Figure 1 – Data interface connections

2. Interface theory

2.1 Message protocol

2.1.1 Ethernet message synchronization

CIGI supports both synchronous and asynchronous operation. Synchronous operation is the preferred mode because it provides an interface with the least amount of latency or message delay and avoids cumbersome interpolation or extrapolation computations. Asynchronous operation is accommodated by placing timing values in the mandatory Start of Frame and IG Control data packets to allow for definition of the data timing methods.

Asynchronous operation demands that the positional and attitude data provided from a Host be corrected to compensate for disparities of when the data was derived in the Host to when the data is to be used in the IG. In synchronous operation this is not necessary as the IG video frame and data creation in the Host is "locked" via a sync operation. In an asynchronous operation the compensation for the disparity between when data is derived and the time it is used is typically done using an interpolation or extrapolation process. Because there are many interpolation and extrapolation mechanisms available the particular implementation will not be discussed here. There are several pieces of information that are needed for the interpolation or extrapolation mechanisms. First there is a time factor. This time factor is provided for in the CIGI by the timing values in the mandatory Start of Frame and IG Control data packets. It is intended that the IG will compute any necessary frame-to-frame time deltas using this information. Depending on the fidelity required for the simulation either velocity (rate) or acceleration information may be used to derive position and attitude information within the IG. Velocity (rate) information can be provided from the Host simulation via the Rate Control data packet described in section 5.7. As of this writing the CIGI does not implement a data packet containing acceleration information, however a User defined data packet described in section 5.29 may be used to implement such a packet. The remainder of this section discusses the CIGI's synchronization mechanism.

During synchronous operation, the IG via the CIGI begins each frame by transmitting a start of frame signal to the Host. This signal is in the form of a Start of Frame data packet and should be the first data packet in the Ethernet message. See section 5.21 for details on the use of the Start of Frame data packet. Any other data packets destine for the Host may follow the Start of Frame data packet in the Ethernet message. This start of frame produces a "heartbeat" which dictates the timing of data transfers between the IG and the Host. The Host should immediately respond to the start of frame with an Ethernet message containing as its first data packet an IG Control data packet. See section 5.3 for details on the use of the IG Control data packet. Other data packets destine for the image generator may follow the IG Control data packet in the Ethernet message. Because of the timing of the interface, these data will represent the state of the simulation during the previous frame. When the IG receives this response, it can then begin performing its computational and rendering cycles. Meanwhile, the Host updates its simulation parameters such as entity states, atmospherics, mission function requests, etc., then waits for the next start of frame signal to transmit these new values to the image generator. Figure 2 illustrates this sequence:

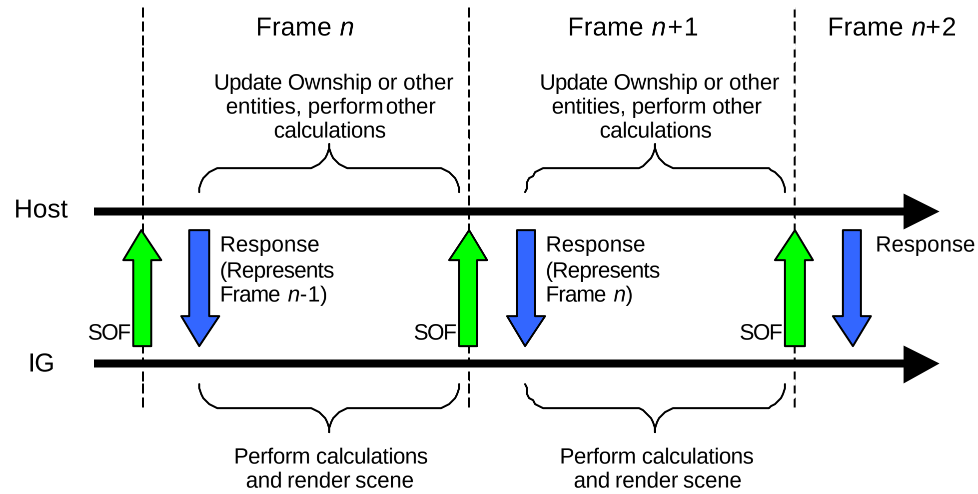


Figure 2 - CIGI start of frame / response cycle

Depending on bandwidth limitations and the amount of Ethernet traffic on the system, the IG may not receive a response early enough to complete its computational cycle before the start of the next frame. To alleviate this situation, a time offset can be introduced to offset the Host and IG frame periods. Because the IG controls the start of the frame and the frame rate, the idea is to send a start of frame signal before the actual beginning of each IG frame. This mechanism will allow Host to IG data to arrive at such a time as to allow the IG its entire frame time for computations and rendering. Because the Ethernet bandwidth may vary from frame to frame, this offset can be adjusted to allow for worst case network loads so that no late arrivals occur. Figure 3 illustrates the start of frame offset technique:

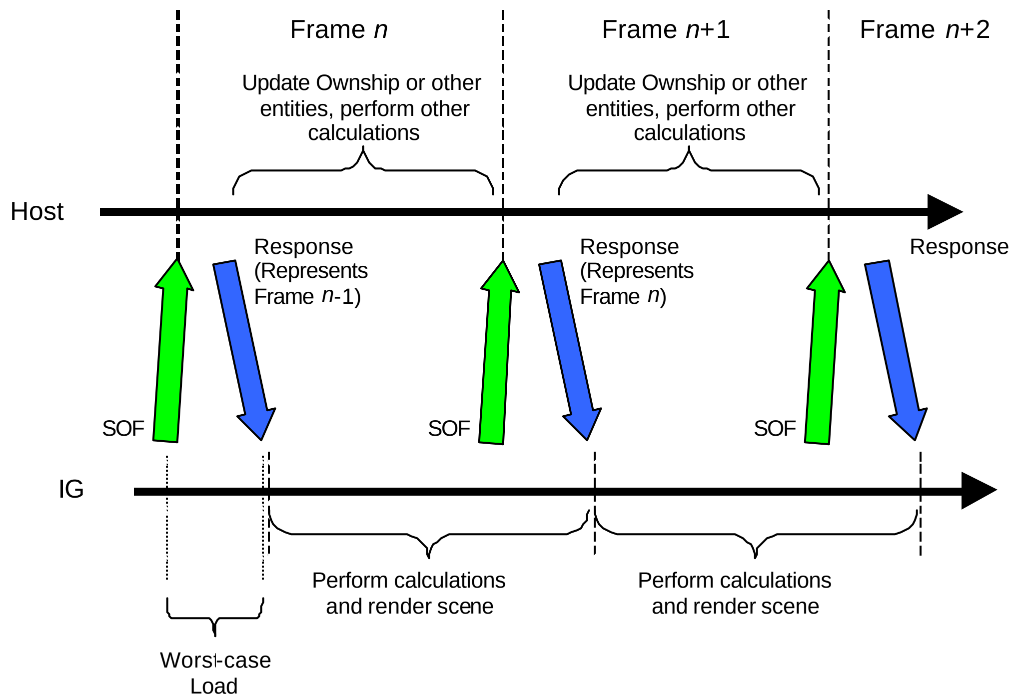


Figure 3 - IG to Host frame offset

As a capability to track Ethernet messages, the data packets contained in the IG-to-Host and Host-to-IG Ethernet transfers are tagged with sequence numbers. The sequence number originates within the IG and is passed to the Host in the IG to Host Frame Counter parameter of the Start of Frame data packet. The Host should extract this number from the Start of Frame data packet and place it into the Host to IG Frame Counter parameter of the IG Control data packet that is returned from the Host. In this way, Ethernet communications can be checked for one-to-one correspondence.

2.1.2 Ethernet message frequency

The IG software can be configured to run at any reasonable frequency. Typically, the IG data update rate should match the display refresh rate to alleviate stepping and other undesirable effects caused by asynchronous data updates. This means that for synchronous operation, the Host must also be bound to a multiple of the display update rate or must extrapolate its data each frame to meet the specified IG update rate. Common update rates are 30 and 60 Hz.

2.2 Start up sequence

The Host should only communicate with the IG in response to a Start of Frame message. Upon initial power-up the IG may wait a predetermined amount of time before communicating with the Host to allow the display system and other components to initialize. It is recommended that this time be configurable within the IG. The IG should be considered mission-ready when it sends its first start of frame signal. Normal communications should proceed from this point. If the host attempts to manipulate mission data before this time, the IG will not process the data and the information will be lost.

The IG will be set to the standby (reset) mode of operation after completion of the startup cycle. The Host is sent the first start of frame signal at this juncture. The IG will remain in this mode until it encounters a change in the IG Mode parameter of the IG Control data packet contained within a Host response message. The Host must change the IG operational mode to *operate* and wait for the mode change to be acknowledged in the IG Mode parameter of the Start of Frame data packet before attempting to send initialization or mission data to the IG.

Upon power-up, the IG may pre-load a default database or test pattern as specified in the IG configuration. If another database is desired, the Host can request that database once the IG is mission-ready. Because the IG must be reinitialized during a database load, data relating to previous operations will be lost and should be re-instantiated by the Host, as necessary. See the Database Number parameters of the IG Control and Start of Frame data packets in sections 5.3 and 5.21 for further details.

After completion of operational training sessions, the Host should command the CIGI back to the standby (reset) mode. This is done so that all entities that were instantiated during the previous training session are removed from the display before a new training session begins.

To use the IG maintenance mode, the Host must first command the IG to the *standby* via the *IG Mode* parameter of the IG Control data packet described in section 5.3.

2.3 User Definable Data Packets

Although the CIGI is a robust interface, there may be times when a developer wishes to define a unique data packet format for a specific purpose. For this reason, CIGI has been designed to be extensible. Data packet identifiers 236 through 255 have been reserved for user defined data packets. See section 5.29 of this document for further details on user definable data packets.

2.4 Data packaging

The data required to operate the IG are logically organized into data packets. The data packet identification number contained in the first byte of each data packet uniquely identifies that data packet's type. Refer to Table 1 for a list of packet types. Please note that some data packets are mandatory for each frame. Communication messages may be comprised of a number of data packets within a particular frame.

Because the IG Control data packet may contain information that will determine how other data in the message will be used, this shall be the first data packet in the Host to IG message. If not, an error will be returned to the Host and no further action will be taken by the IG in that frame.

An entity must exist before it can have parameters or attributes applied to it. For example, if a Component Control is to be applied to an entity in the same message in which the entity is first specified, the associated Component Control data packet must follow the corresponding Entity Control data packet in the message. Other than these requirements, no restrictions are place on packet ordering and all other data packets can move in relative position in the data buffer from frame to frame.

To reduce the risk of overloading the IG computational frame, an attempt should be made to minimize the amount of data contained in each message supplied to the IG. To accomplish this goal, the CIGI interface is capable of varying in size from frame to frame during real-time operation. If a data packet is not mandatory, it should be contained in the message only if it describes data changes to the IG; reference Table 1.

Table 1 – Data packet identification

Data Packet Identification number	Data Packet Name	Mandatory Every Frame	Communication Direction
Host to IG			
1	IG Control	Yes	Host to IG
2	Entity Control	No	Host to IG
3	Component Control	No	Host to IG
4	Articulated Part Control	No	Host to IG
5	Rate Control	No	Host to IG
6	Environment Control	No	Host to IG
7	Weather Control	No	Host to IG
8	View Control	No	Host to IG
9	Sensor Control	No	Host to IG
21	Trajectory Definition	No	Host to IG
22	Special Effect Definition	No	Host to IG
23	View Definition	No	Host to IG
24	Collision Detection Segment Definition	No	Host to IG
25	Swept Volume Collision Detection Definition	No	Host to IG
41	Height Above Terrain Request	No	Host to IG
42	Line of Sight Occult Request	No	Host to IG
43	Line of Sight Range Request	No	Host to IG
44	Height of Terrain Request	No	Host to IG
IG to Host			
101	Start of Frame	Yes	IG to Host
102	Height Above Terrain Response	No	IG to Host
103	Line of Sight Response	No	IG to Host
104	Collision Detection Segment Response	No	IG to Host
105	Sensor Response	See section 5.25	IG to Host
106	Height of Terrain Response	No	IG to Host
107	Swept Volume Collision Detection Response	No	IG to Host
108	Image Generator Response Message	No	IG to Host
User Definable Data Packets			
236 - 255	User Definable Data Packets	N/A	N/A

During real-time operation only a subset of these data packets are required in any given message to describe data changes to the IG. As an example, Table 2 shows the data packet IDs of a hypothetical Host to IG sequence of frames. Note that this example contains at least one Entity Control data packet each frame indicating dynamic movement of one entity, possibly the Ownship.

Table 2 - Example of Entity IDs in an Ethernet message

Ethernet Message frame n	Ethernet Message frame n+1	Ethernet Message frame n+2	Ethernet Message frame n+3	Ethernet Message frame n+4	Ethernet Message frame n+5	Ethernet Message frame n+6	Ethernet Message frame n+7
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
7	2	2	2	2	3	2	
8	3	9	4	9	2	2	
41	2	9	7	9	9	5	
9	4	42	9		42		
9	2	42	9		43		
	4						

2.4.1 Floating-point format

Data represented as a floating-point number is formatted in IEEE format.

2.4.2 Fix-point format

Data represented as a fixed-point number is formatted as follows:

This notation is used to express the range and resolution of a fixed-point format. It consists of an uppercase B followed by two numbers in parentheses, such as B(n,m). The first number, n, defines the power of two that the most significant bit represents. The second number, m, defines the power of two that the least-significant bit represents. Formats used in this document are:

Scaled distance format (16 bit scaled at B6)

	2^8	2^0	2^{-1}	2^{-6}
15	14	6	5	0

S Most significant byte	Least significant byte

Format: 16-bit, two's complement, fixed-point B(9, -6)
Conversion:
Resolution: 2^{-6}
Range: -2^9 through $2^9 - 2^{-6}$

Angle format (16 bit)

180^o	90^o	45^o	22.5^o	2^{-16} X 360^o
15	14	13	12	0

S Most significant byte	Least significant byte

Format: 16-bit, unsigned, fixed-point B(-1, -16)
Conversion:
Resolution: 2^{-16} X 360^o
Range: 0^0 through 360^o – (2^{-16} X $360)^o$

3. Basic Definitions, Principles and Nomenclature

The following section will describe some commonly used definitions, nomenclature and basic principles that are used when discussing the CIGI.

3.1 Definition Data Packets

Definition data packets are used to define or alter the characteristics of a CIGI feature. There are definition packets for defining trajectories, special effects, views, collision detection segments and collision detection volumes. For instance the Trajectory Definition data packet can be used in conjunction with the Entity Control data packet, the Special Effects Definition data packet, and the Rate data packet to define aspects of an object's trajectory, such as a bullet's flight. The Special Effects Definition data packet can be used in conjunction with the Entity Control data packet to override the default-modeled parameters within an effect. View Definition data packets can be used to define the characteristics of a view and/or override the IG default configuration of a view. Collision Detection Segment Definition and Swept Volume Collision Detection Definition data packets can be used to define collision structures that can be tested for intersections with other objects or surfaces within the simulation environment. See their respective sections for detailed discussion on these definition data packets.

3.2 Control Data Packets

Control data packets are used to control features within the IG. There are data packets for controlling the image generator, entities, components, articulated parts, entity rates, environmental parameters, weather parameters, views, and sensors. For example, the IG Control data packet contains information that controls, among other things, the IG states, database selection, frame count, and data timing values. Entity Control data packets control the physical parameters of an object in the simulation scenario such as its appearance, temperature, transparency characteristics, and its position and attitude. The Component Control data packet is provided as a generic control mechanism to manipulate components contained within the synthetic environment or an entity. It is used to control the aspect of components such as light intensities, symbology placement, polygon and texture component states, etc. The Articulated Part Control data packet contains parameters to control parts that can articulate such as flaps, slats, speed brakes, etc. The Articulated Part Control data packet can be used to move such parts in up to six degrees of freedom. The Rate Control data packet can be used to specify the rate parameters of an entity or supplement other packets' information, such as the Trajectory Definition data packet. Atmospheric phenomena are controlled using the Environmental and Weather Control data packets. With these packets, the user can control global environment parameters and override default local or layered weather phenomena. View Control data packets are used to control the position and orientation of a view that has either been previously defined in the IG or has been defined by a View Definition data packet. For detailed discussions on these data packets see their respective sections.

3.3 Request Data Packets

Request data packets are used to make requests of the image generator. There are data packets for requesting the height above terrain, height of terrain, line of sight occulting and line of sight ranging information. The Height Above Terrain data packet will invoke a response from the image generator that contains information about the height above the terrain for a specific point specified in the request. The Height of Terrain data packet will invoke a response from the image generator that contains information about the height of the terrain at a specified location. The Line of Sight Occult data packet is used to determine intervisibility or occulting between a source and destination point. The Line of Sight Range data packet is used to determine the range from a source point to an object within the environment. For detailed discussions on these request data packets see their respective sections.

3.4 Response Data Packets

Response data packets are sent from the image generator to the Host in response to a request data packet. There are response data packets that correspond to each type of request data packet discussed above, plus a sensor response data packet. The details of the use of these data packets can be seen in their respective sections of this document.

3.5 Entities

Within the CIGI, an entity is defined as an object that has a separate and distinct instance within the synthetic environment. Entity types can include stationary and repositionable objects. Some examples are vehicles such as aircraft, ships and ground vehicles; special effects such as explosions, missile and vapor trails, and smoke, etc.; ground features such as buildings, towers, and bridges; or attributes such as lights and steerable lobes.

A unique instance of an Entity is identified by its Entity ID. An Entity ID is merely a way to identify a single dynamic coordinate system. Any object known to the CIGI can be assigned to an Entity ID by its Entity Type. In this way, the user can instantiate up to several instances of the same object by assigning the same Entity Type to several unique Entity IDs. This principle can be seen in the missiles and missile trails in Figure 4. The number assignments in this example are hypothetical.

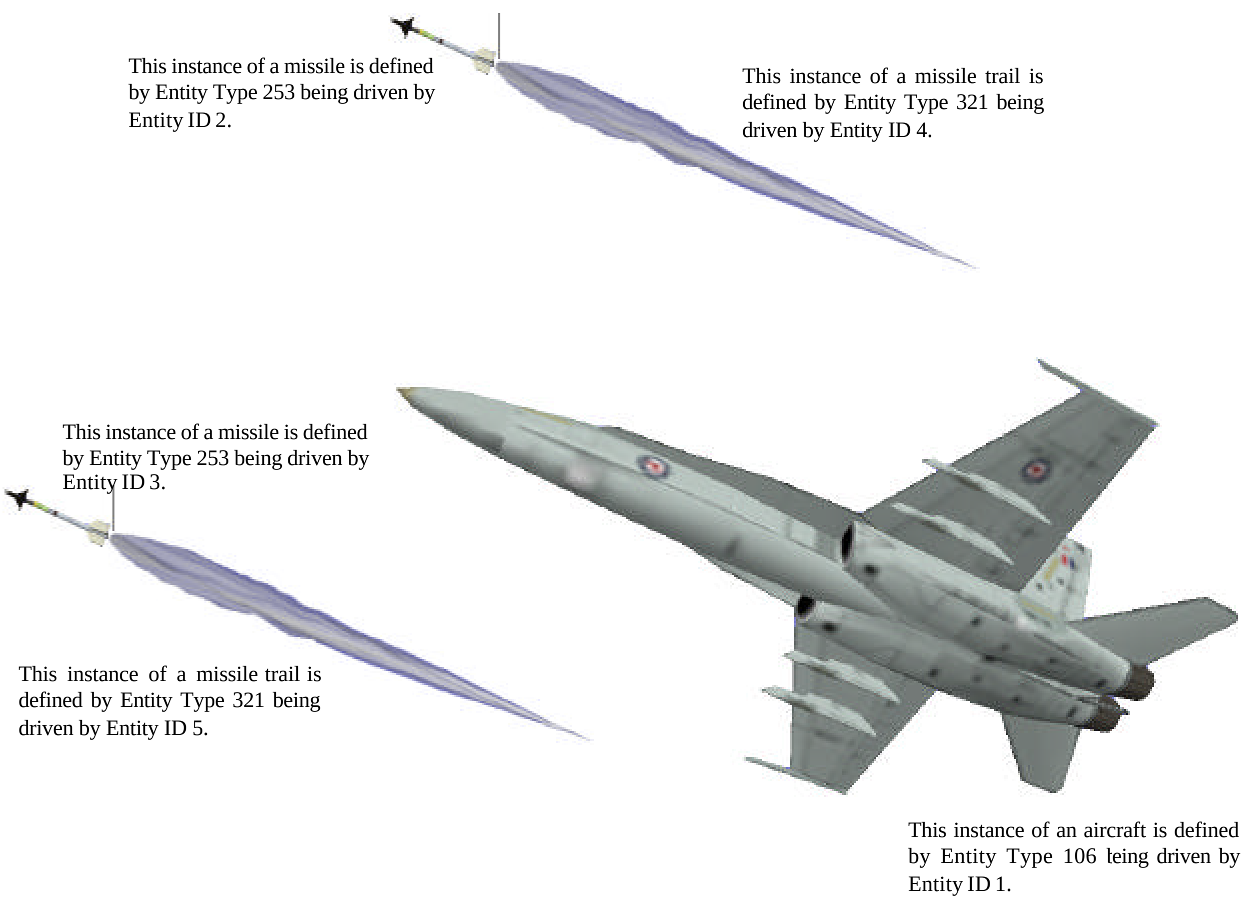


Figure 4 - Entity Definitions

Entities can also be attached to one another. Attached entities are said to have a parent/child relationship. The depth of the parent / child chain is only limited by the computational capabilities of the Host and IG platforms. In Figure 4, the missiles and missile trails could be maneuvered in one of two ways. First, each missile and missile trail could be driven uniquely, requiring the host to provide position and attitude data for each of the four entities. The simpler and typically preferred way would be to establish a parent/child relationship between each missile and missile trail pair. This would be done by establishing Entity ID 4 (missile trail) as a child of the Entity ID 2 (missile), and likewise for Entity IDs 5 and 3. When entities are parented, the host is only obligated to control the top-level parent entity via the CIGI. All children will be "chained" to the parent within the image generator, and no unique manipulation of a child entity is necessary unless its position and attitude change with respect to its parent. For more details on parenting entities, see section 5.4.

3.6 Views

Views are used to create the viewing portals for a display system. The display system can be of any variety, such as an Out-the-Window visual system, a sensor channel display, a Night Vision Goggle scope, etc. Views can be defined in size and depth and can be positioned and rotated. Views can also be grouped together to create contiguous panoramic views. Views can be attached to entities such as the Ownship, or attached to a weapon's seeker to provide a sensor view. Figure 5 gives an idea of how views can be manipulated. See the View Control and View Definition data packet descriptions in sections 5.10 and 5.14 for a detailed discussion on view manipulation.

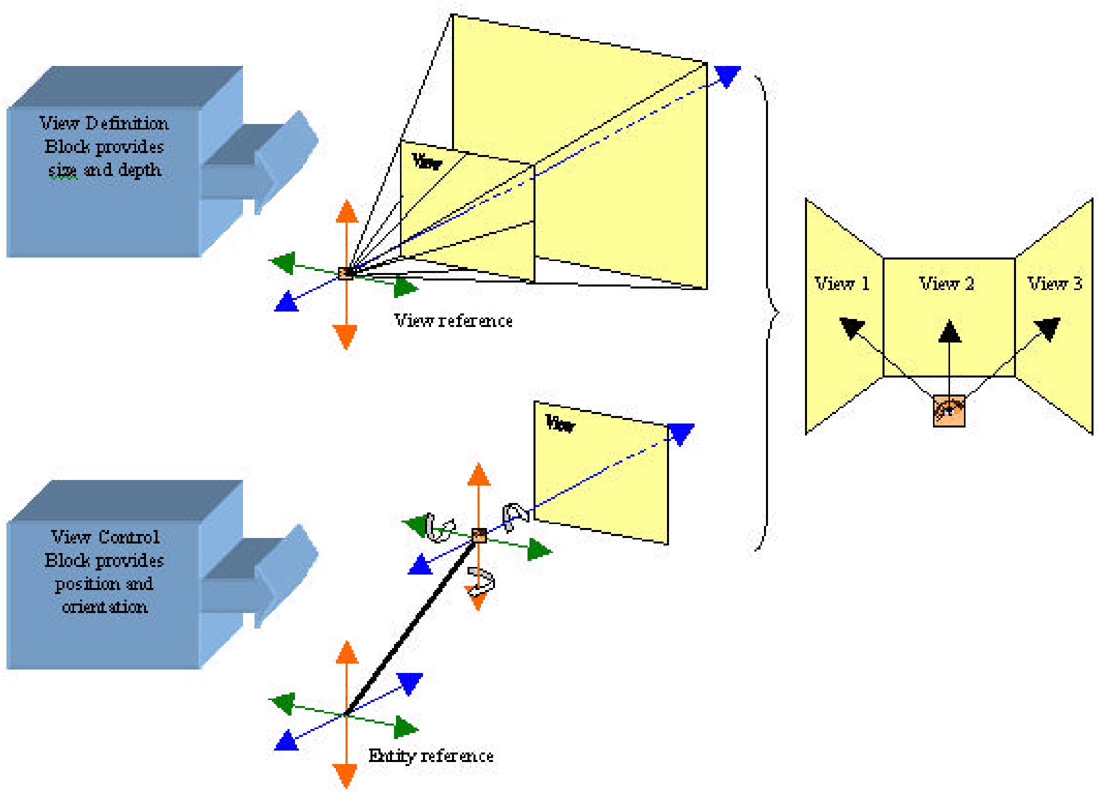


Figure 5 - View Manipulation

4. Coordinate systems

4.1 Entity positioning

CIGI specifies an entity's position in Geodetic Coordinates. The Geodetic Coordinate system specifies a location as a latitude, longitude, and altitude. The altitude is the distance from a point in space to the closest point on the Earth's ellipsoidal surface. This altitude line will be perpendicular to the flat plane that is tangent to the earth at this point. Altitude is measured positive above the surface of the reference ellipsoid, and negative below it. As this line is extended toward the polar axis (Z-axis) it intersects the equatorial plane, giving the latitude angle, *lat*, as shown in Figure 6, measured positive north of the equator and negative south, limited to ±90°. The altitude line then intersects the Z-axis to give the longitude angle, *lon*, measured positive east of the prime meridian and negative west, limited to ±180°.

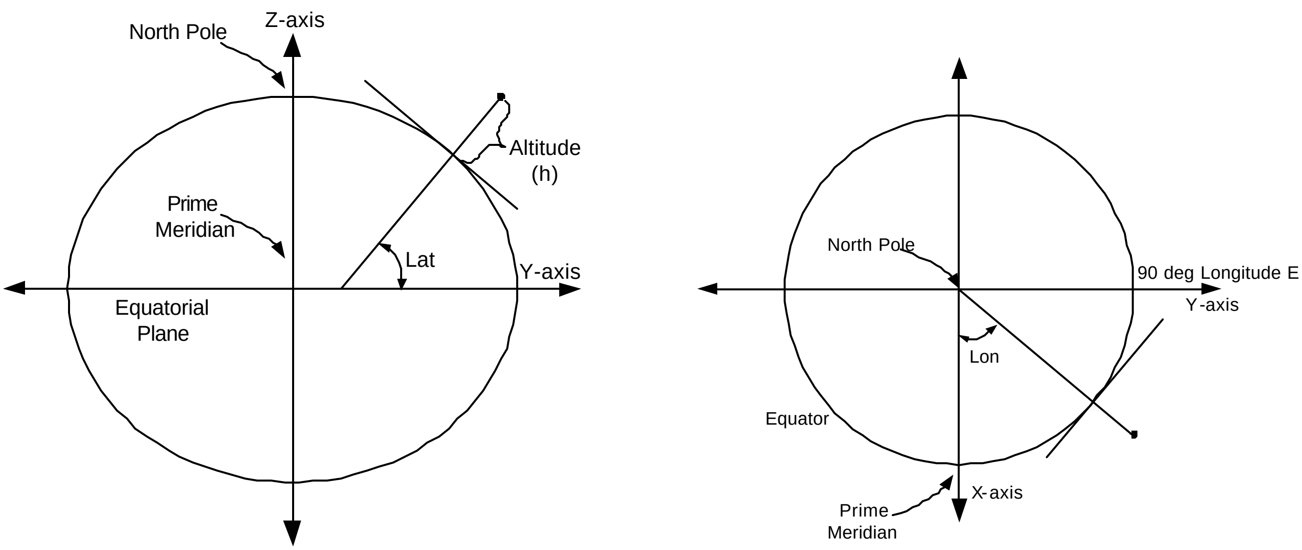


Figure 6 - Entity Position in Geodetic Latitude and Longitude

4.2 Entity orientation

The orientation of an entity is with respect to a plane tangent to the ground at a point directly beneath it. The entity coordinate system is parallel with the Local North, East, Down coordinate system when the entity's Heading, Pitch, and Roll are all zero. Figure 7 illustrates the entity coordinate system. The order of rotation is defined in CIGI to be about the Z, Y and then X axes (i.e. heading, pitch, and then roll), as seen in Figure 8.

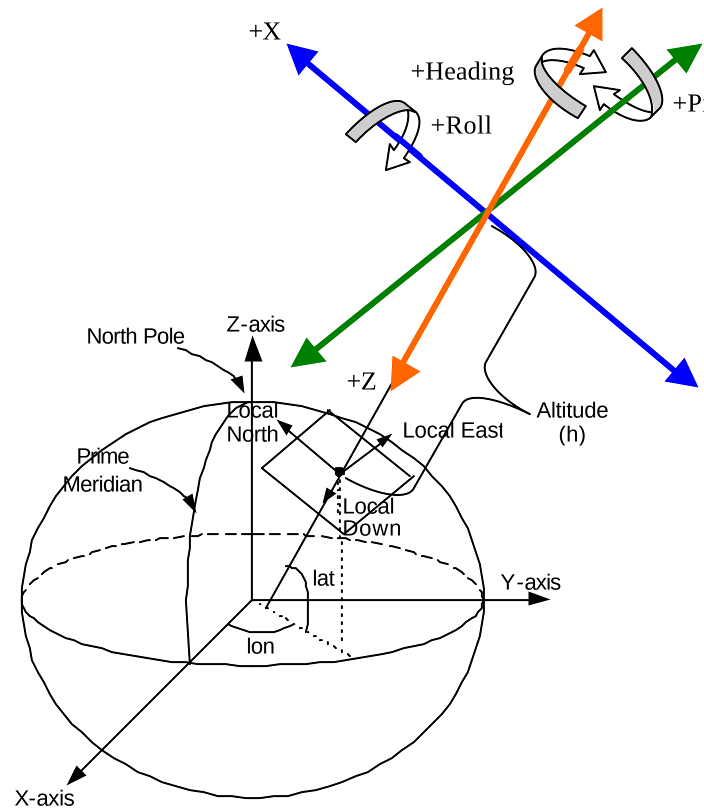


Figure 7 - Entity Rotation

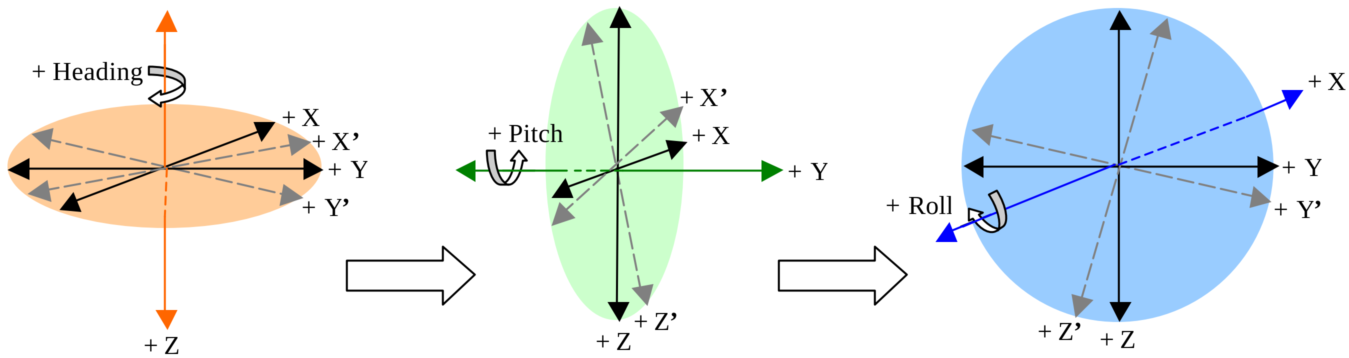


Figure 8 - Order of Rotation

4.3 Entity Coordinate System

For convenience, a typical aircraft reference system is used to describe the coordinate axis used for an entity, as shown in Figure 9, with +X out through the nose, +Y out the starboard side, and +Z through the ventral side.

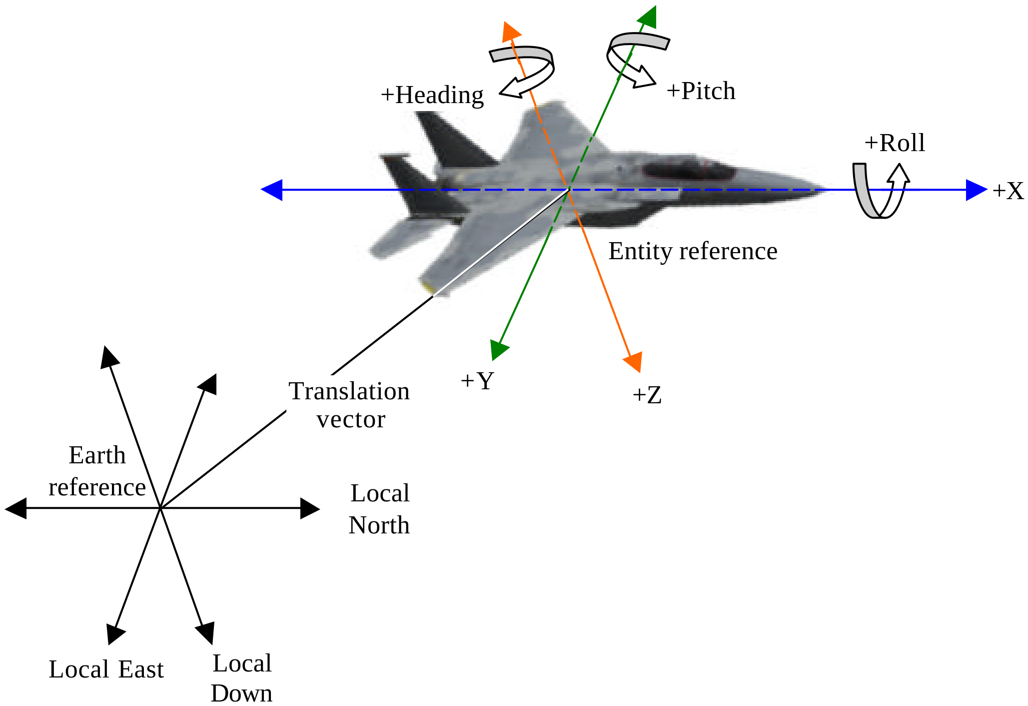


Figure 9 - Entity coordinate system

5. Data packet nomenclature

5.1 Data packet relationships

The CIGI uses a principle of base objects and redefinition theory. That is to say objects, including entities, special effects and views, are defined based on a unique identification for each instance. An entity, including special effects, is created via the Entity Control data packet. A View is created via the View Control data packet. The creation of an entity or view establishes its base definition. After these entities or views are established they can be modified via other data packets. A diagram showing some of the possible relationships between data packets accompanies the description of each data packet described as necessary. The nomenclature used in these diagrams is explained in Figure 10.

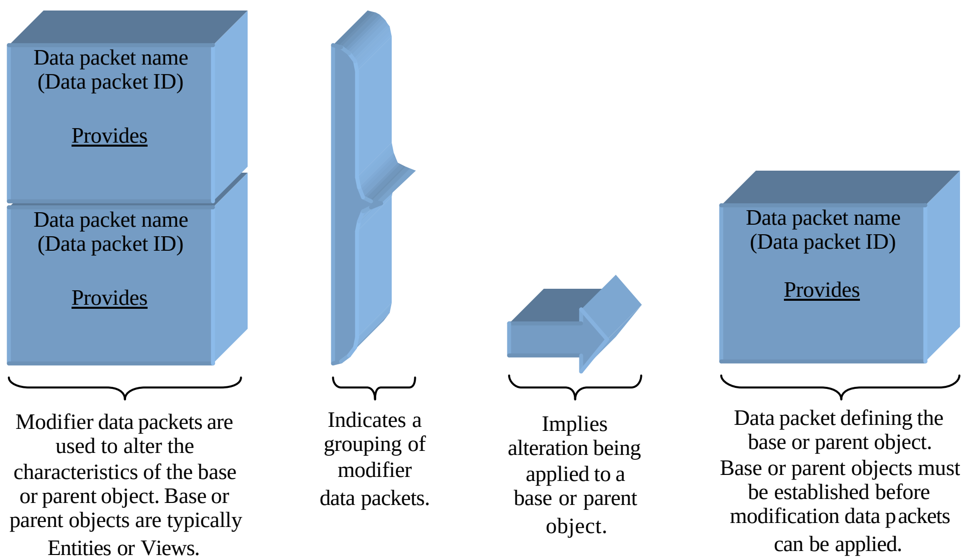


Figure 10 - Data packet relationship nomenclature

5.2 Data packet description

Each data packet format is discussed in the following sub-paragraphs. The parameter assignments for each data packet are shown in a diagram similar to the one in Figure 11. All CIGI packets use big-endian byte ordering.

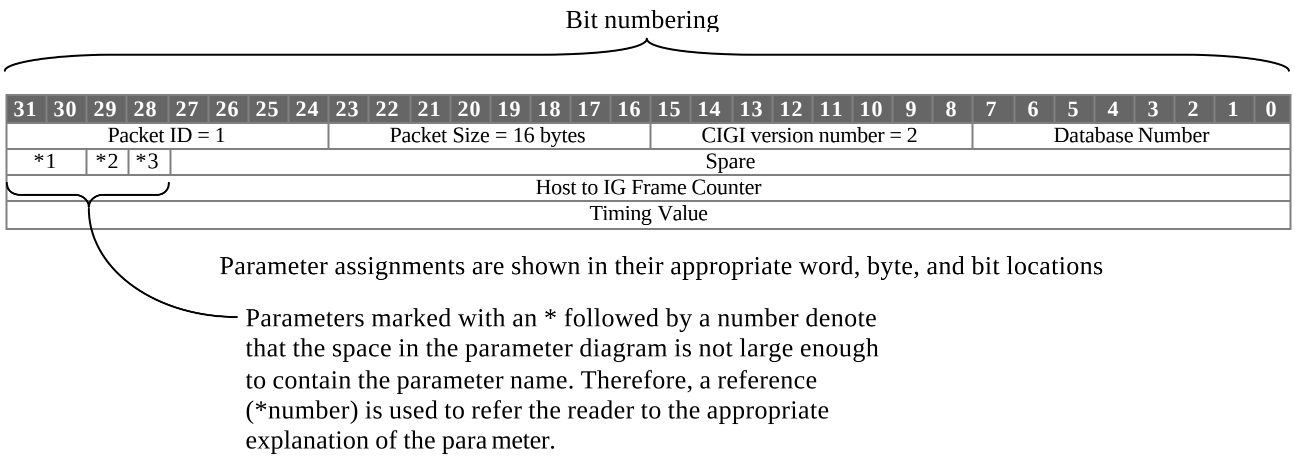


Figure 11 - Example of Data Packet Parameter Diagram

A narrative description of each datum is presented in a section below the data packet diagram as seen in the example in Figure 12.

Formats and Ranges	Description
Packet ID = 1	This area identifies the data packet. This area also identifies any restrictions on the usage of the data packet.
*1 Name : Type : Units valid values: Default: N/A Datum:	This area identifies the data parameter's name, type and any applicable units. It also identifies any restrictions on the values of a data parameter and the default value, if any, that the IG will assign the parameter until a data packet containing that parameter is sent to the IG. The datum for a parameter, such as Mean Sea Level for altitude will also be provided, if appropriate. This area will also provide a narrative of the intended use for the data parameter and how it may interact with other parameters in the Ethernet message.

Figure 12 - Example of Data Packet Narrative

5.3 IG Control

The IG Control data packet is contained in the Ethernet message sent from the Host to the IG. This data packet is mandatory in each Ethernet message and is used to control various operations of the IG. Because the IG Control data packet may contain information that will determine how other data in the Ethernet message will be used, it shall be the first data packet in the Host to IG Ethernet data buffer. If this rule is not followed, an error will be returned to the Host and no further action will be taken for that frame.

The contents of the IG Control data packet are shown below.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Packet ID = 1								Packet Size = 16 bytes								CIGI version number = 2								Database Number							
*1		*2	*3	Spare																											
Host to IG Frame Counter																															
Timing Value																															

IG Control parameter definitions:

Formats and Ranges	Description
Packet ID = 1 : unsigned char : N/A	This parameter identifies this data packet as the IG Control data packet. There can be only one instance of this data packet per frame. If more than one data packet is received the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
CIGI version number : unsigned char : N/A valid values: 0 – 255 Default: N/A	This parameter indicates the version of the CIGI interface that is currently running on the host. The image generator can use this number to determine concurrency.
Database Number : signed char : N/A valid values: -128 to -1 Not used 0 No load requested +1 to +99 Request load of this database See the Database Number table in the applicable Database and Entity Attribute Definition Document(s). Default = N/A	This parameter indicates the number associated with the database requiring loading. Placing a valid database number in this field will cause the IG to commence loading of the requested database. The IG will respond with the negated value of the database number that is requested indicating that the database load is under way. This indication is provided in the Database Number data field of the Start of Frame data packet, section 5.21. When the Host receives this indication, it should return this parameter to zero. This must be done to prevent the IG from loading the database again upon completion of the previously requested load. Also, during the time that the IG is returning the negated value, mission data will be ignored. Therefore the Host should not send any data packets to the IG other than the IG Control data packet during a database load.

*1 IG Mode Change Request : 2 bit field : N/A valid values: 0 = standby (reset) 1 = operate 2 = debug 3 = Not used Default: 0	This parameter is used by the Host to command the IG to enter its various modes. When the IG receives a request for a mode change via this parameter it will return the corresponding mode in the Current IG Mode parameter of the Start of Frame data packet once the new mode has been accomplished. **Standby (reset):** See the discussion on the standby (reset) mode in the Current IG Mode parameter description in the Start of Frame data packet in section 5.21. **Operate:** See the discussion on the operate mode in the Current IG Mode parameter description in the Start of Frame data packet in section 5.21. **Debug:** See the discussion on the debug mode in the Current IG Mode parameter description in the Start of Frame data packet in section 5.21.
*2 Tracking Device Enable : Boolean : N/A valid values: 0 = disable tracking inputs 1 = enable tracking inputs Default: 0	This parameter is used by the Host to enable or disable an external tracking device connected to the image generator. An example would be a head tracker used to drive head position. Currently only one tracking device is supported by this interface
*3 Tracking Device Boresight : Boolean : N/A valid values: 0 = No action 1 = Boresight Default: 0	This parameter is used by the Host to enable the boresight mode (zeroing out of view offset positions and angles) for an external tracking device connected to the image generator. Boresight mode will remain active until this bit is cleared.
Host to IG Frame Counter : unsigned integer : N/A valid values: 0 to 4,294,967,295 Default: N/A	This parameter contains a number representing a particular frame. The Host should copy the corresponding value from the IG to Host Frame Counter parameter in the Start of Frame data packet, section 5.21 and place it in this parameter to show that this Host Ethernet message is in response to a particular IG Ethernet message.
Timing Value : Float IEEE: usec valid values: 0 to 86,399,999,999.99 Default: 0	This parameter is optional for synchronous operation, but required for asynchronous operation. It contains a timing value that is used to time-tag the Ethernet message when asynchronous operation is instituted. When asynchronous operation is used, the synchronous timing scheme described in section 2.1.1 is superceded. In order to preserve floating-point accuracy, this timing value is limited to a 24-hour simulation period. At the end of 24 hours, the counter will reset to zero.

5.4 Entity Control

The Entity Control data packet is contained in the Ethernet message sent from the Host to the IG. An entity is defined as an object that has a separate and distinct instance within the synthetic environment. Entity types can include moving or repositionable objects such as aircraft, ships, ground vehicles, special effects, ground models, lights, steerable lobes, etc. An entity can also be created that represents nothing more than a “camera” for the purpose of controlling a view.

The Entity Control data packet is used to instantiate an entity in one of two ways: 1) as a unique entity where this data packet is used to manipulate its attitude and geodetic position, and 2) as a child of a parent entity where this data packet is used to manipulate the child’s attitude and positional offset relative to its parent’s reference point. All positional data represent the position of the entity’s reference point, which corresponds to the model origin. This is typically, but not necessarily, the entity’s center of gravity.

This data packet applies to all entities that are required for the simulation, including the Ownship.

In order to reduce the load on Ethernet messages and the IG computational frame, only Entity Control data packets that contain data changes should be included in the Ethernet message. Once an Entity Control packet describing an entity is sent to the IG, the state of that entity will not change until another Entity Control packet containing that entity ID is received. For example, packets describing the Ownship and a wingman may be sent every frame to indicate continuous positional changes, while a packet describing an inactive SAM site may be sent once during mission initialization.

The Entity State field is used to control when an entity is visible and when its geometry is loaded and unloaded. When an entity is created, the Entity State field can be set to *Load/Show* to specify that the entity should be added to the scene as soon as the model geometry is loaded. The entity can then be temporarily removed from the scene graph, or made invisible, by setting Entity State to *Load/Hide*. When the entity is no longer needed, Entity State can be set to *Unload* to direct the IG to unload the geometry and free any memory allocated for the entity.

Models can also be preloaded to increase the speed at which they can be initially displayed. For example, when the Ownship fires a missile, a new entity would need to be created for that missile. Unless the missile geometry is cached, the IG must load the model from disk. Because of its tremendous speed, the missile might fly beyond visual range before the disk I/O can be completed. By preloading the entity, the geometry can already exist in memory and be instantly loaded into the scene graph when needed. To accomplish this, an Entity Control packet with the Entity State flag set to *Load/Hide* would be sent to the IG during mission initialization or at some other point prior to firing of the missile. When the missile is needed, another Entity Control packet for that entity would be sent containing the proper positional data and with the Entity State flag set to *Load/Show*.

Child entities inherit the entity state of their parent. In other words, the Entity State parameter of a parent entity affects not only that entity, but also all its children. When a parent’s Entity State parameter is set to *Load/Hide*, all the children will be hidden. Likewise, when a parent’s Entity State is set to *Load/Show*, all the children that have not been explicitly hidden will be shown. When a parent’s Entity State parameter is set to *Unload*, the entity and all its children will be destroyed by the IG.

The Effect Animation State field is used to control the animation state of entities representing special effects. When an effect is preloaded, the Effect Animation State parameter should be set to *Stop* and the effect will remain in its initial state. To start the animation sequence at any time thereafter, the host would send an Entity Control packet with its Entity State and Effect Animation State parameters to *Load/Show* and *Play*, respectively. Setting the Effect Animation State to *Stop* simply stops the animation sequence at the current frame. Setting the parameter to *Play* in a subsequent frame will resume the animation; setting it to *Restart* will play the animation from its initial state. If an effect is modeled as momentary (i.e., having limited duration), it will stop automatically. The host may reactivate a momentary effect by setting the Effect Animation State to *Restart* without having to first set the parameter to *Stop*.

Continuing the example above, if the missile were to hit an enemy aircraft, the aircraft would likely explode. During mission initialization, to preload the effect, the host would have first sent an Entity Control packet specifying the ID of the explosion and the Entity State (*Load/Hide*), and then an Effect Control packet describing the direction, size, and other attributes of the effect. During the exercise, if the missile were to hit its target, the host would set the Entity State for the missile's Entity Control packet to *Load/Hide* or *Unload*. The host would then send another Entity Control packet for the explosion. In this packet, the Entity State field would be set to *Load/Show* and the Effect Animation State field would be set to *Play* or *Restart*. After the explosion dissipates, the host could destroy the entity, or it could set Entity State to *Load/Hide* and reuse the explosion later.

Table 3 summarizes the actions possible with the Entity State and Effect Animation State fields.

Table 3 - Entity / Effect State Action Table

Entity Type	Action	Entity State	Effect Animation State
Non-effect	Load but don't show	Load/Hide	-
Non-effect	Load and show	Load/Show	-
Non-effect	Show	Load/Show	-
Non-effect	Hide	Load/Hide	-
Non-effect	Hide and unload	Unload	-
Effect	Load but don't play (preload) or if already loaded, Stop and hide	Load/Hide	Stop
Effect	No action, effect is hidden	Load/Hide	Play
Effect	No action, effect is hidden	Load/Hide	Restart
Effect	Pause	Load/Show	Stop
Effect	Load and play from beginning , Play loaded effect (1st time), Continue playing, or resume if paused	Load/Show	Play
Effect	Load and play from beginning , Play loaded effect (1st time), Restart loaded effect from beginning	Load/Show	Restart
Effect	Stop and unload	Unload	Stop
Effect	Unload	Unload	Play
Effect	Unload	Unload	Restart

Opacity of an entity can be applied using this data packet. Opacity will modify the apparent transparency of an entity. Each individual entity, i.e. any entity not attached to another, must have it's opacity changed using the data packet controlling that entity. For attached entities, i.e. those with a parent/child relationship, the opacity is dictated by the entity's type. For attached entities, representing objects, the parent and child entities will fade according to the top-level parent's opacity. For attached entities that are special effects the top-level parent's opacity will not affect the child's opacity.

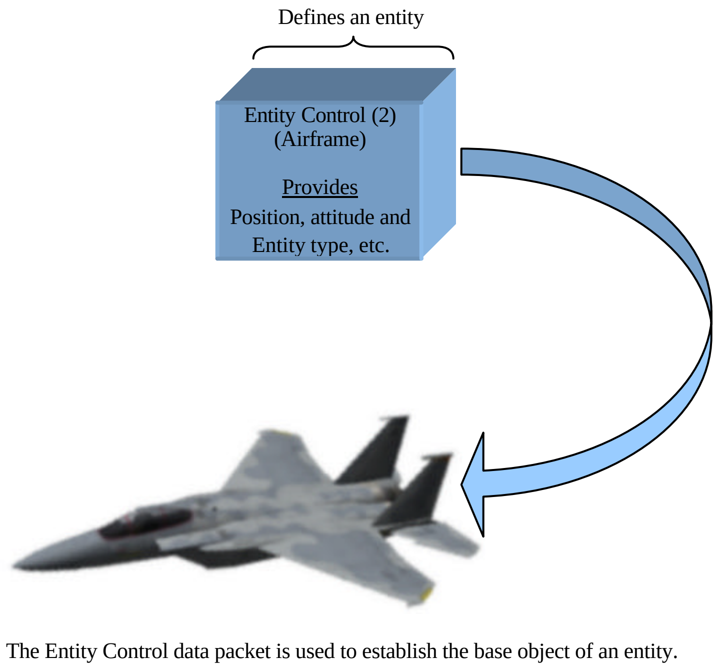


Figure 13 - Object creation using an entity

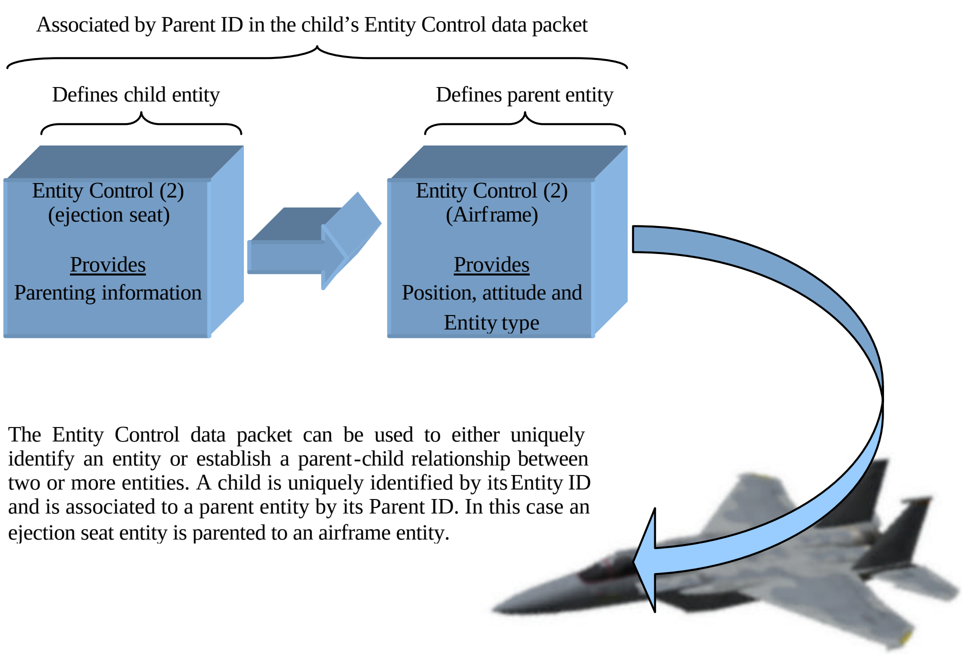


Figure 14 - Single Parent / Child Components

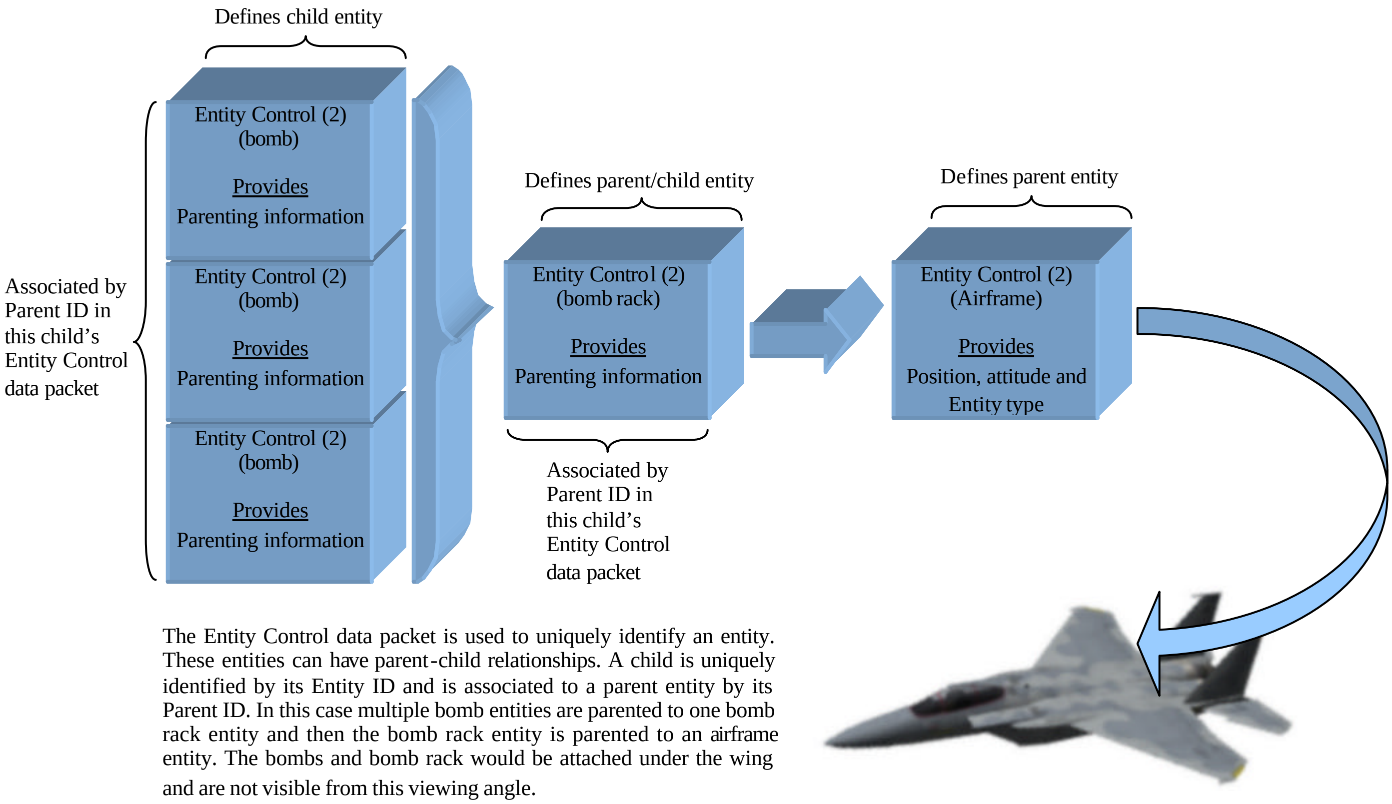


Figure 15 - Multiple Parent / Child Components

In order to reduce the load on Ethernet messages and the IG computational frame, only Entity Control data packets that contain data changes should be included in the Ethernet message.

The contents of the Entity Control data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 2</td><td colspan="8">Packet Size = 56 bytes</td><td colspan="16">Entity ID</td></tr>
<tr><td colspan="2">*1</td><td>*2</td><td>*3</td><td colspan="2">*4</td><td colspan="26">Spare</td></tr>
<tr><td colspan="16">Entity Type</td><td colspan="16">Parent Entity ID</td></tr>
<tr><td colspan="32">Percent Opacity</td></tr>
<tr><td colspan="32">Internal Temperature</td></tr>
<tr><td colspan="32">Entity Roll</td></tr>
<tr><td colspan="32">Entity Pitch</td></tr>
<tr><td colspan="32">Entity Heading</td></tr>
<tr><td colspan="32">Entity Altitude / Z Offset (MSW)</td></tr>
<tr><td colspan="32">Entity Altitude / Z Offset (LSW)</td></tr>
<tr><td colspan="32">Entity Latitude / X Offset (MSW)</td></tr>
<tr><td colspan="32">Entity Latitude / X Offset (LSW)</td></tr>
<tr><td colspan="32">Entity Longitude / Y Offset (MSW)</td></tr>
<tr><td colspan="32">Entity Longitude / Y Offset (LSW)</td></tr>
</table>

Entity Control parameters parameter definitions:

Formats and Ranges	Description
Packet ID = 2 : unsigned char : N/A	This parameter identifies this data packet as the Entity Control data packet. There can be multiple instances of this data packet per frame, but each unique entity should only be specified once per frame. If more than one data packet with the same Entity ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
Entity ID : unsigned short : N/A valid values: 0 = Ownship entity 1 to 65535 Default: N/A	This parameter indicates the entity motion system this data packet represents. If the specified entity id contains zero the parameters in this data packet will be applied to the Ownship. If the specified entity id contains a number greater than zero the parameters in this data packet will be applied to the specified entity.
*1 Entity State : 2 bit field : N/A Valid Values: 0 = Load/Hide 1 = Load/Show 2 = Unload Default: N/A	This parameter specifies whether an entity's geometry should be visible/invisible, or whether the entity should be destroyed. When an entity is required, this parameter should be set to *Load/Hide* or *Load/Show*. This will cause the IG to create a new instance of the type specified in the Entity Type parameter, and this new instance will initially be invisible or visible, respectively. Once the new entity's geometry has been loaded, changing this parameter sets the visibility of the geometry accordingly. When the entity is no longer required, this parameter should be set to *Unload* to remove the entity's hierarchy from the scene graph. Note that destroying an entity causes all child entities (attachments and special effects) to be destroyed, as well.

*2 Attach/Detach Switch : Boolean : N/A valid values: 0 = Detach 1 = Attach Default: 0	This parameter specifies whether the entity it represents should be attached as a child to a parent. **To specify a unique (parent) entity, or to accomplish a detachment:** ?? This parameter must be set to *Detach*. ?? The Entity State parameter must be set to *Load/Show* or *Load/Hide*. ?? The Entity ID parameter must be valid. ?? The positional information in this data packet should specify the entity's Latitude, Longitude, and Altitude. The Parent Entity ID parameter is ignored. **To accomplish an attachment:** ?? This parameter must be set to *Attach*. ?? The Entity State parameter must be set to *Load/Show* or *Load/Hide*. ?? The Entity ID parameter must be valid. ?? The Parent Entity ID parameter must be valid, i.e., the parent entity must be previously defined. ?? The positional information in this data packet should specify the X Offset, Y Offset, and Z Offset from the parent. Once the assignment is made, the IG will retain the parent-child relationship until the parent entity is destroyed or the child is detached from the parent.
*3 Collision Detection Request : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter enables/disables collision detection for this entity. See the Collision Detection Segment and Swept Volume Collision Response data packet descriptions, sections 5.24 and 5.27 for details of the return data.

*4 Effect Animation State : 2 bit field : N/A Valid Values: 0 = Stop 1 = Play 2 = Restart Default: 0	This parameter specifies the animation state of a special effect. This parameter applies only when the value of the Entity Type parameter corresponds to an effect. *Stop* – Stops the animation sequence. Has no effect if the animation is currently stopped. *Play* – Begins playback from the current animation frame. If an animation was previously stopped mid-sequence, playback continues from that point. If an animation has not yet been played, or if it has been played through its entirety and stopped, playback begins at the start of the sequence. *Restart* – Starts playback from the beginning of the animation sequence. Note that this value is a momentary state; leaving this parameter set to *Restart* in subsequent frames will cause playback to restart with each CIGI frame, making the effect appear to "hang" at its initial animation state.
Entity Type : unsigned short : N/A valid values: See the entity identification table in the applicable Database and Entity Attribute Definition Document(s). 0 = Not visible Default: N/A	This parameter indicates the type for the entity being represented by this data packet. If the integration engineer wishes to attach a view to a position without a model present, a 0 can be used in this field to signify that no type be used. This will effectively cause the entity to not be shown in the visual scene. If the specified Entity Type is invalid, an error will be generated and the data packet will be disregarded.
Parent Entity ID : unsigned short : N/A valid values: 0 = Ownship entity 1 to 65535 Default: N/A	This parameter indicates the parent to which this entity should be attached. This field is only valid when the Attach/Detach Switch is set to *Attach*. If the specified Parent Entity ID is not active, an error will be generated and the data packet will be disregarded.
Percent Opacity : Float IEEE : N/A valid values: 0.0 to 1.0 where: 0.0 Entity is fully invisible, or 100 % faded. 1.0 Entity if fully visible, or 0 % faded. Default: 0.0	This parameter specifies the degree of opacity of the Entity. A fully visible entity will have a Percent Opacity of 100.0 (no fade applied). A fully invisible entity will have a Percent Opacity of 0.0 (full fade applied).

Internal Temperature : Float IEEE : degrees Celsius valid values: Minimum to maximum allowed by the data format Default: N/A Datum: 0° C	This parameter specifies the internal temperature of the Entity. It is used to show internal contrast such as engine warming on thermal views.
Entity Roll : Float IEEE : degrees valid values: 0 to +180 right wing down 0 to –180 left wing down Default: N/A Datum: See Figure 7.	This parameter specifies the roll angle of the Entity. If the Attach/Detach Switch of this data packet is set to *Detach* the Entity Roll is relative to the coordinate system shown in Figure 7. If the Attach/Detach Switch of this data packet is set to *Attach*, the Entity Roll is relative to the parent coordinate system shown in Figure 9.
Entity Pitch : Float IEEE : degrees valid values: 0 to +90 nose up 0 to –90 nose down Default: N/A Datum: See Figure 7.	This parameter specifies the pitch of the Entity. If the Attach/Detach Switch of this data packet is set to *Detach* the Entity Pitch is relative to the coordinate system shown in Figure 7. If the Attach/Detach Switch of this data packet is set to *Attach*, the Entity Pitch is relative to the parent coordinate system shown in Figure 9.
Entity Heading : Float IEEE : degrees valid values: 0 to +360 clockwise Default: N/A Datum: See Figure 7.	This parameter specifies the heading of the Entity. If the Attach/Detach Switch of this data packet is set to *Detach* the Entity Heading is relative to the coordinate system shown in Figure 7. If the Attach/Detach Switch of this data packet is set to *Attach*, the Entity Heading is relative to the parent coordinate system shown in Figure 9.
Entity Altitude : Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A Datum: Mean Sea Level, See Figure 6. -------------- Or ---------------- Z Offset: Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A Datum: Parent Reference Point	This parameter specifies the altitude position of the reference point of the Entity. This parameter specifies the Z Offset of a child entity's reference point from its parent's reference point.

Entity Latitude : Double IEEE : degrees valid values: 0 to +90 (north positive) 0 to –90 (south negative) Default: N/A Datum: equator, See Figure 6. -------------- Or ---------------- X Offset: Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A Datum: Parent Reference Point	This parameter specifies the latitude position of the reference point of the Entity. This parameter specifies the X Offset of a child entity's reference point from its parent's reference point.
Entity Longitude : Double IEEE : degrees valid values: 0 to +180 (east positive) 0 to –180 (west negative) Default: N/A Datum: prime meridian, See Figure 6. -------------- Or ---------------- Y Offset: Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A Datum: Parent Reference Point	This parameter specifies the longitude position of the reference point of the Entity. This parameter specifies the Y Offset of a child entity's reference point from its parent's reference point.

5.5 Component Control

The Component Control data packet is contained in the Ethernet message sent from the Host to the IG. The Component Control data packet is provided as a generic control mechanism to manipulate components contained within the synthetic environment. Components are identified by first specifying the particular class they belong to, then identifying the particular instance of the class, and then identifying the particular component within the instance. Examples of these associations can be see in Table 4.

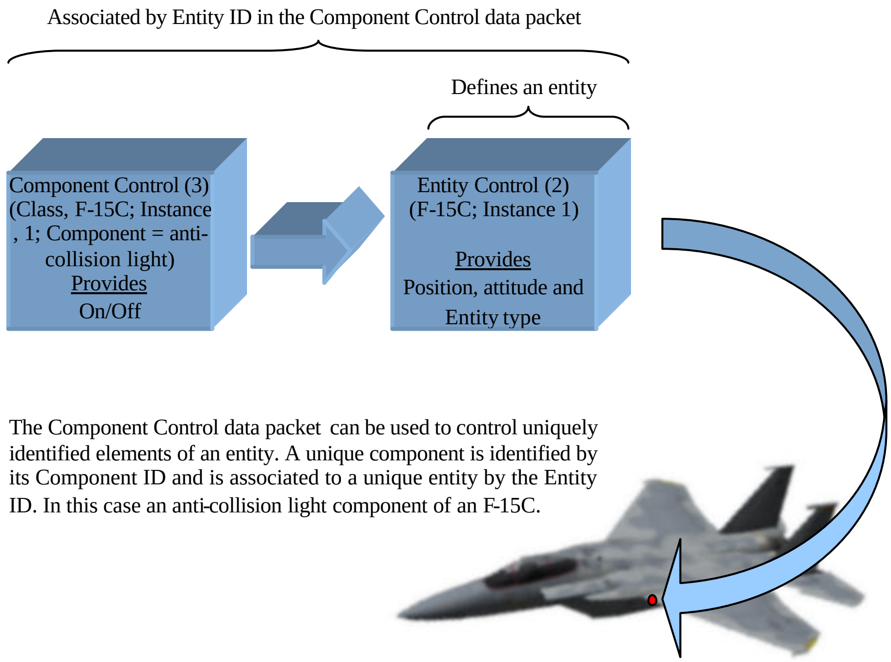
After the component has been identified values can be applied to manipulate its state or sliding scale value. An example of a state manipulation is shown in Table 4 where the entity F-16C can have air break states of 25, 50, 75 and 100 open. An example of a sliding scale value is shown in Table 4 where the entity F-15C formation light's intensity can be selected from 0 to 100 percent. Another example of a sliding scale value is shown in Table 4 where a sensor gate symbol can be placed within the sensor display by using both sliding scale values provided in the component control data packet, one for X position and the other for Y position.

This data packet contains both a discrete integer value and two float-point values. A component control may use any or all values, depending on the control. The IG and database will provide a configuration mechanism that maps components and controls to the appropriate Component Class, Instance ID and Component ID.

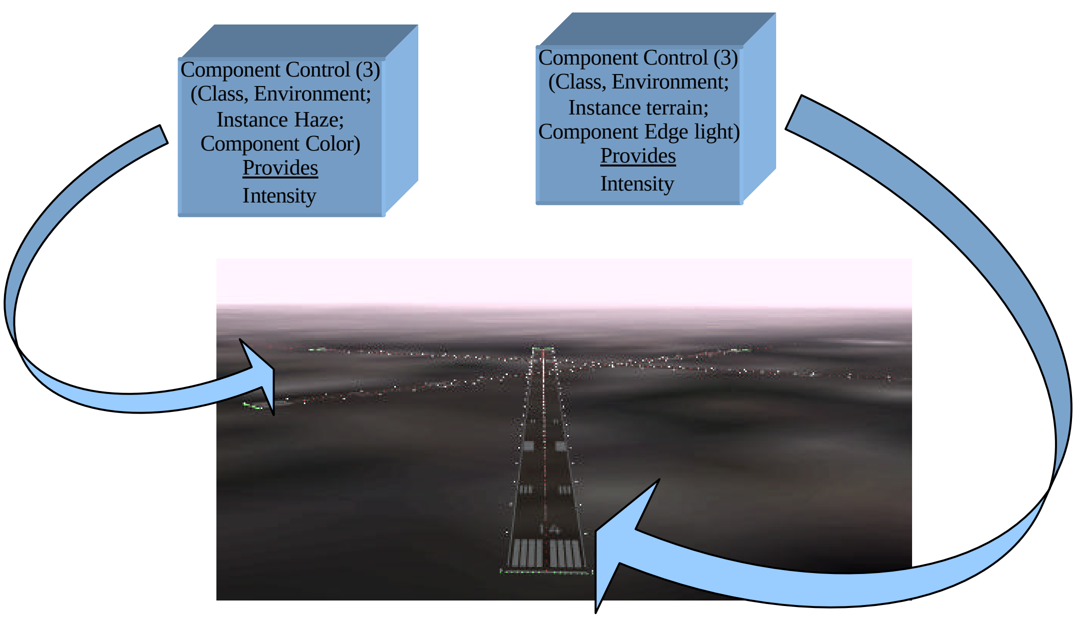
Table 4 - Component Identification Scheme

Component Class	Instance ID	Component ID
Entity	F-15C	Anti-collision light on/off Position lighting on/off Formation lighting intensity, 0 to 100% Air break position partial open Air break position full open
Entity	F-16C	Anti-collision light on/off Position lighting on/off Formation lighting intensity, 0 to 100% Air break position 25% open Air break position 50% open Air break position 75% open Air break position 100% open
Environment	Sky	Red color Blue color Green color Sun on/off Moon on/off
Environment	Haze	Red color Blue color Green color
Environment	Terrain	Runway centerline intensity, 0 to 100% Runway edge light intensity, 0 to 100% Runway threshold light intensity, 0 to 100%
View	View ID	Zoom
View Group	View Group ID	Zoom
Sensor	Gate Symbol	Screen position in X Screen position in Y
Sensor	Cursor Symbol	Screen position in X Screen position in Y
System	Image Generator	Screen blanking (Crash indicator)

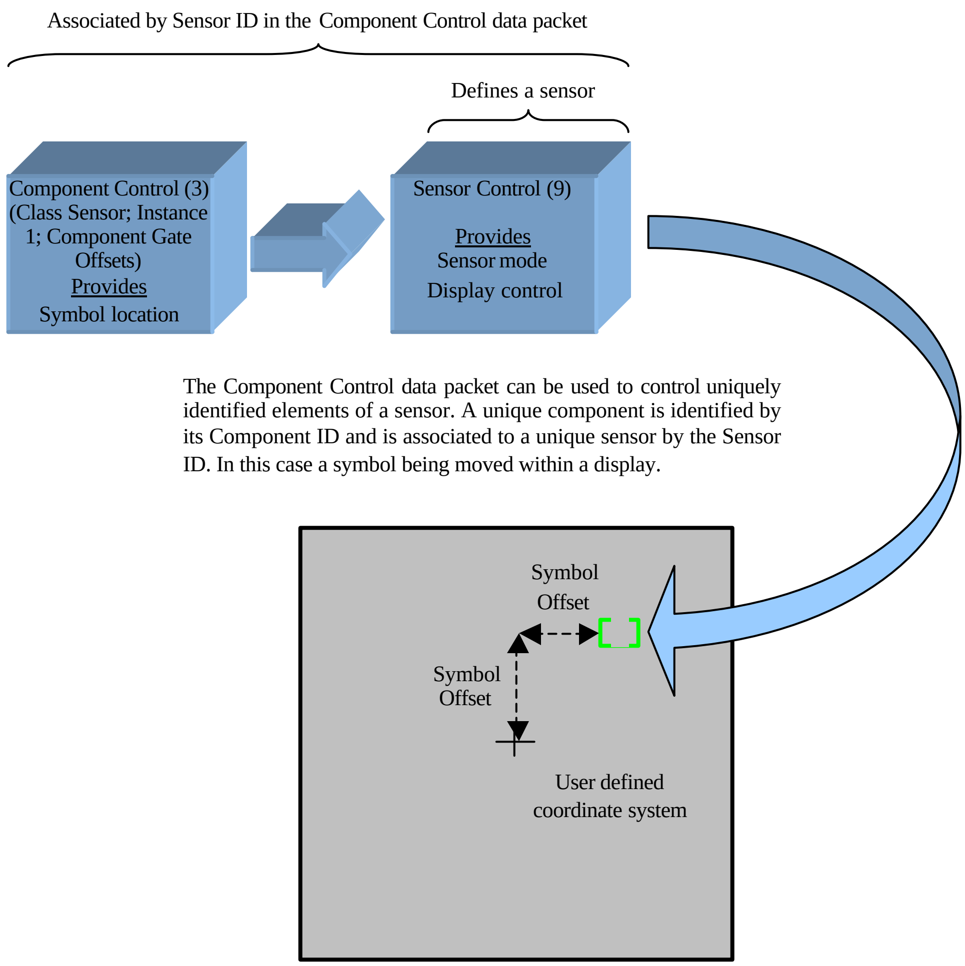
Example of a Component Control data packet controlling an entity related attribute:



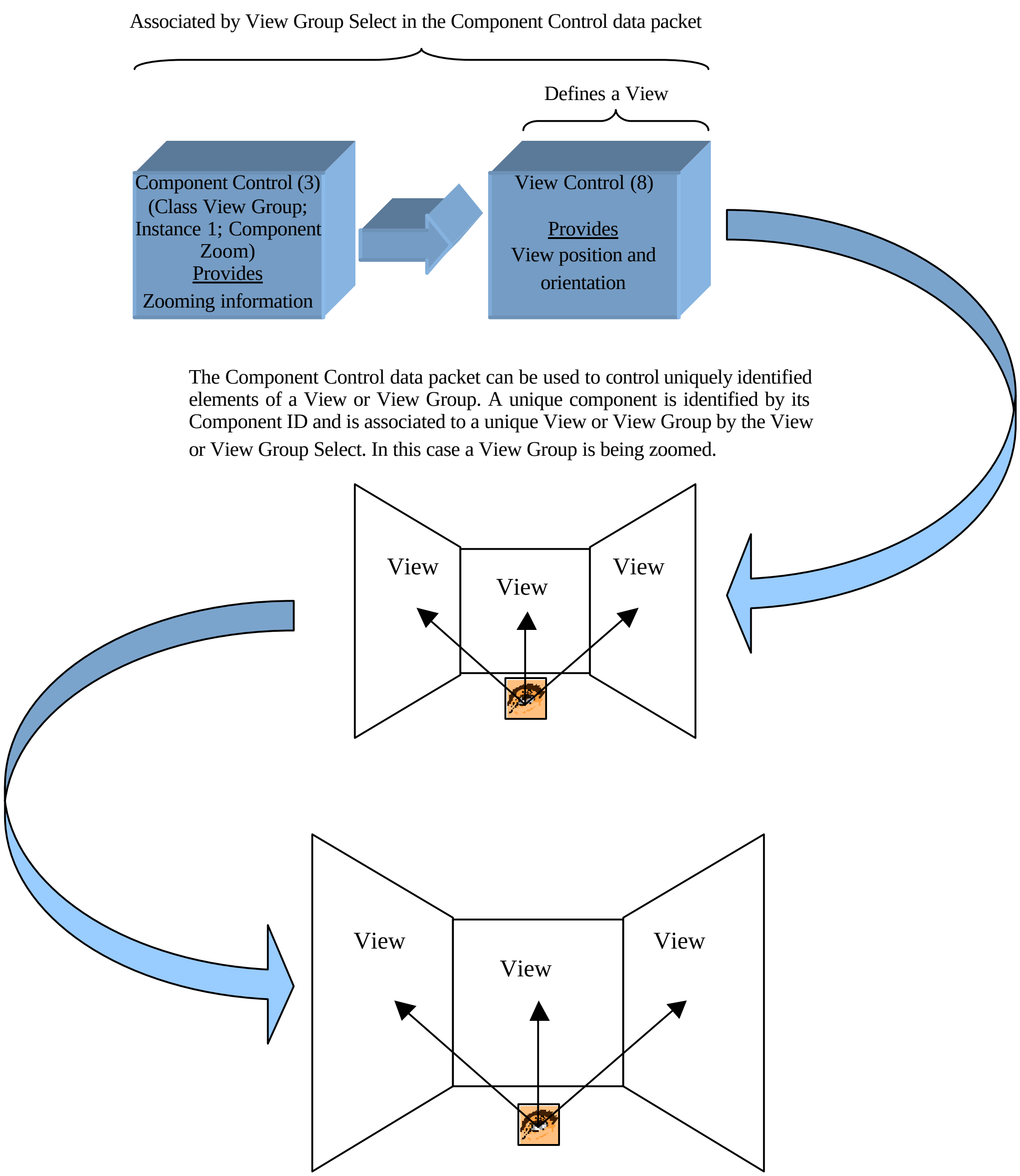
Example of a Component Control data packet controlling terrain lights and haze color attributes:



Example of a Component Control data packet controlling a sensor related attribute:



Example of a Component Control data packet controlling a View or View Group related attribute:



(It should be noted that a zooming feature may also be implemented using a View Definition data packet.)

Figure 16 – Examples of Component Control application

In order to reduce the load on Ethernet messages and the IG computational frame, only Component Control data packets that contain data changes should be included in the Ethernet message.

The contents of the Component Control data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 3</td><td colspan="8">Packet Size = 20 bytes</td><td colspan="16">Instance ID</td></tr>
<tr><td colspan="8">Component Class</td><td colspan="24">Spare</td></tr>
<tr><td colspan="16">Component ID</td><td colspan="16">Component State</td></tr>
<tr><td colspan="32">Component Value 1</td></tr>
<tr><td colspan="32">Component Value 2</td></tr>
</table>

Component Control parameters parameter definitions:

Formats and Ranges	Description
Packet ID = 3 : unsigned char : N/A	This parameter identifies this data packet as the Component Control data packet. There can be multiple instances of this data packet per frame. Component information for a unique component, can only be specified once per frame. If more than one is received per frame the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
Instance ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter indicates the instance of a class the component being controlled belongs to. It is used in conjunction with the Component Class and Component ID to uniquely identify a component. The relationship between Component Class, Instance ID, and Component ID is explained in the narrative and Table 4 of this section.
*2 Component Class : unsigned char : N/A valid values: 0 = Entity 1 = Environment 2 = View 3 = View group 4 = Sensor 5 = System Default: 0	This parameter indicates what class the component being controlled is in. It is used in conjunction with the Instance ID and Component ID to uniquely identify a component. The relationship between Component Class, Instance ID, and Component ID is explained in the narrative and Table 4 of this section.
Component ID : unsigned short : N/A valid values: 0 to maximum allowed by the data format See the Component Control assignments in the applicable Database, Entity Attribute, and IG functions Definition Document (s). Default: N/A	This parameter identifies the Component of a Component Class and Instance ID this data packet will be applied to. It is used in conjunction with the Component Class and Instance ID to uniquely identify a component. The relationship between Component Class, Instance ID, and Component ID is explained in the narrative and Table 4 of this section. If an invalid Component ID is specified, an error will be generated and the data packet will be disregarded.

Component State : unsigned short : N/A valid values: 0 to maximum allowed by the data format See the Component Control assignments in the applicable Database, Entity Attribute, and IG functions Definition Document (s). Default: N/A	This parameter specifies the commanded state of a Component. If an invalid Component State is specified, an error will be generated and the data packet will be disregarded.
Component Value 1 : Float IEEE : Component defined valid values: minimum to maximum allowed by the data format See the Component Control assignments in the applicable Database and Entity Attribute Definition Document (s). Default: N/A	This parameter specifies a sliding scale value to be applied to a Component. If an invalid Component Value is specified, an error will be generated and the data packet will be disregarded.
Component Value 2 : Float IEEE : Component defined valid values: minimum to maximum allowed by the data format See the Component Control assignments in the applicable Database and Entity Attribute Definition Document (s). Default: N/A	This parameter specifies a sliding scale value to be applied to a Component. If an invalid Component Value is specified, an error will be generated and the data packet will be disregarded.

5.6 Articulated Part Control

The Articulated Part data packet is contained in the Ethernet message sent from the Host to the IG. This data packet contains parameters to manipulate articulated parts such as flaps, slats, etc., that require articulation in six degrees of freedom.

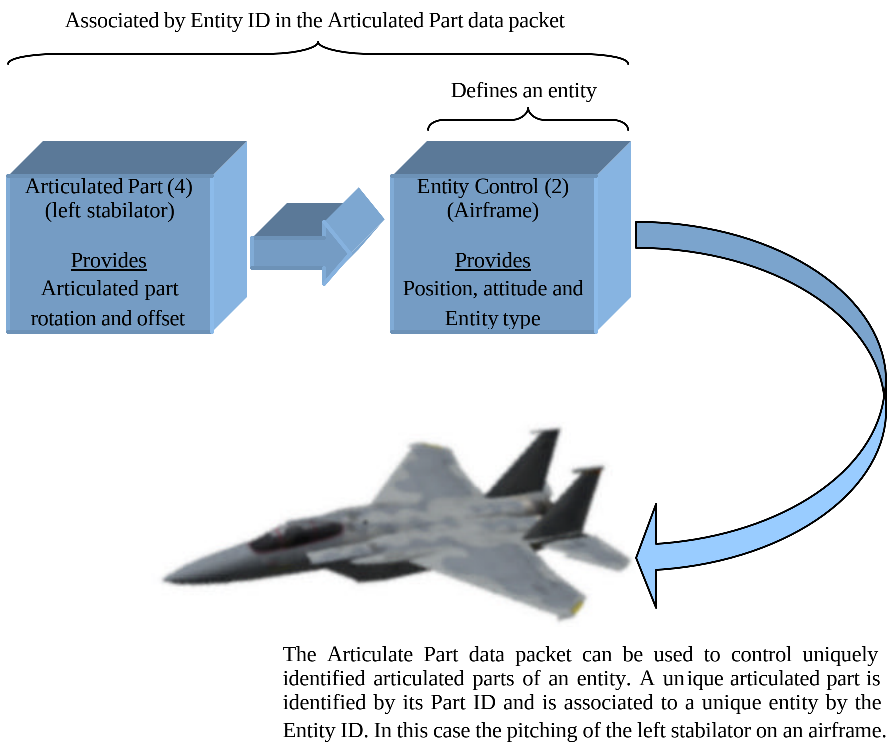


Figure 17 - Articulated Part Control Components

Based on the switch setting of the data fields designated by *2 through *7 in the diagram below, the host can change any or all degrees of freedom of the articulated part. If the host chooses not to activate a particular degree of freedom, that degree of freedom will be defaulted to the modeled default(s). If, however, the host activates a particular degree of freedom, the modeled default value will be overwritten and lost until the IG is restarted.

In order to reduce the load on Ethernet messages and the IG computational frame, only Articulated Part Control data packets that contain data changes should be included in the Ethernet message.

The contents of the Articulated Part Control data packet are shown below.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Packet ID = 4								Packet = 32 bytes								Entity ID															
Articulated Part ID								*1	*2	*3	*4	*5	*6	*7	Spare																
Articulated Part X Offset																															
Articulated Part Y Offset																															
Articulated Part Z Offset																															
Articulated Part Roll																															
Articulated Part Pitch																															
Articulated Part Heading																															

Articulated Parts parameter definitions:

Formats and Ranges	Description
Packet ID = 4 : unsigned char : N/A	This parameter identifies this data packet as an Articulated Part data packet. There can be multiple instances of this data packet per frame, but each unique articulated part should only be specified once per frame. If more than one data packet with the same Articulated Part ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
Articulated Part ID: signed char : N/A valid values: 0 – 127 identifies a unique articulated part See the articulated part identification assignments in the applicable Database and Entity Attribute Definition Document (s). Default: N/A	This parameter indicates which articulated part is controlled with this data packet. If the specified Articulated Part ID is not a valid part of the entity specified by the Entity ID, an error will be generated and the data packet will be disregarded.
Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter specifies the entity to which this data packet will be applied. If the specified entity id contains zero the parameters in this data packet will be applied to the Ownship. If the specified entity id contains a number greater than zero the parameters in this data packet will be applied to the specified entity. If the specified Entity ID is not active, an error will be generated and the data packet will be disregarded.
*1 Articulated Part State : Boolean : N/A valid values: 0 = Inactive (removes the part from the display) 1 = Active (introduces the part into the display) Default: 1	This parameter indicates whether an articulated part is to be shown in the display (active) or not shown in the display (inactive).
*2 Enable/Disable Articulated Part X Offset : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Articulated Part X Offset value contained in this data packet is manipulated from the Host (enabled). If the Host previously changed this degree of freedom and this switch is disabled the Articulated Part X Offset will remain where it was last placed.

*3 Enable/Disable Articulated Part Y Offset : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Articulated Part Y Offset value contained in this data packet is manipulated from the Host (enabled). If the Host previously changed this degree of freedom and this switch is disabled the Articulated Part Y Offset will remain where it was last placed.
*4 Enable/Disable Articulated Part Z Offset : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Articulated Part Z Offset value contained in this data packet is manipulated from the Host (enabled). If the Host previously changed this degree of freedom and this switch is disabled the Articulated Part Z Offset will remain where it was last placed.
*5 Enable/Disable Articulated Part Roll : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Articulated Part Roll value contained in this data packet is manipulated from the Host (enabled). If the Host previously changed this degree of freedom and this switch is disabled the Articulated Part Roll will remain where it was last placed.
*6 Enable/Disable Articulated Part Pitch : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Articulated Part Pitch value contained in this data packet is manipulated from the Host (enabled). If the Host previously changed this degree of freedom and this switch is disabled the Articulated Part Pitch will remain where it was last placed.
*7 Enable/Disable Articulated Part Heading : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Articulated Part Heading value contained in this data packet is manipulated from the Host (enabled). If the Host previously changed this degree of freedom and this switch is disabled the Articulated Part Heading will remain where it was last placed.
Articulated Part X Offset : Float IEEE : meters valid values: Minimum to maximum allowed by the data format Default: As set in the models default Datum: Entity coordinate system, see Figure 9.	This parameter specifies the distance along the X-axis by which the articulated part should be moved. This parameter is ignored if the Enable/Disable X Offset field is set to Disable.

Articulated Part Y Offset : Float IEEE : meters valid values: Minimum to maximum allowed by the data format Default: As set in the models default Datum: Entity coordinate system, see Figure 9.	This parameter specifies the distance along the Y-axis by which the articulated part should be moved. This parameter is ignored if the Enable/Disable Y Offset field is set to Disable.
Articulated Part Z Offset : Float IEEE : meters valid values: Minimum to maximum allowed by the data format Default: As set in the models default Datum: Entity coordinate system, see Figure 9.	This parameter specifies the distance along the Z-axis by which the articulated part should be moved. This parameter is ignored if the Enable/Disable Z Offset field is set to Disable.
Articulated Part Roll : Float IEEE : degrees valid values: 0 to +180 clockwise 0 to –180 counter clockwise Default: As set in the models default Datum: see Figure 9.	This parameter specifies the roll of this part with respect to the submodel coordinate system. This parameter is ignored if the Enable/Disable Roll field is set to Disable.
Articulated Part Pitch : Float IEEE : degrees valid values: 0 to +90 up 0 to –90 down Default: As set in the models default Datum: see Figure 9.	This parameter specifies the pitch of this part with respect to the submodel coordinate system. This parameter is ignored if the Enable/Disable Pitch field is set to Disable.
Articulated Part Heading : Float IEEE : degrees valid values: 0 to +360 clockwise Default: As set in the models default Datum: see Figure 9.	This parameter specifies the heading of this part with respect to the submodel coordinate system. This parameter is ignored if the Enable/Disable Heading field is set to Disable.

5.7 Rate Control

The Rate Control data packet is contained in the Ethernet message sent from the Host to the IG. This data packet contains rate information that supplement the Entity Control data packet or the Articulated Part data packet as needed. An entity is normally placed using the attitude and positional data received in the Entity Control data packet. If reception of an Entity Control data packet is discontinued and a Rate Control data packet was received, the rate information in the Rate Control data packet will be used to continue the entity's movement by extrapolation of the entity's position along the given velocity vector with the specified angular rates. The IG will use the placement and rate information from a prior frame as the basis for the extrapolation. In the event that a new Entity Control data packet is received at some later time, the entity position will be updated and extrapolation will continue from that location. Setting all rate components to zero will cause the entity to be stationary

This data packet may also be used to animate articulated parts modeled within an entity, or child parts attached to a parent model. Given proper angular rates, the IG will extrapolate these components to simulate such things as spinning propellers, rotating wheels, etc.

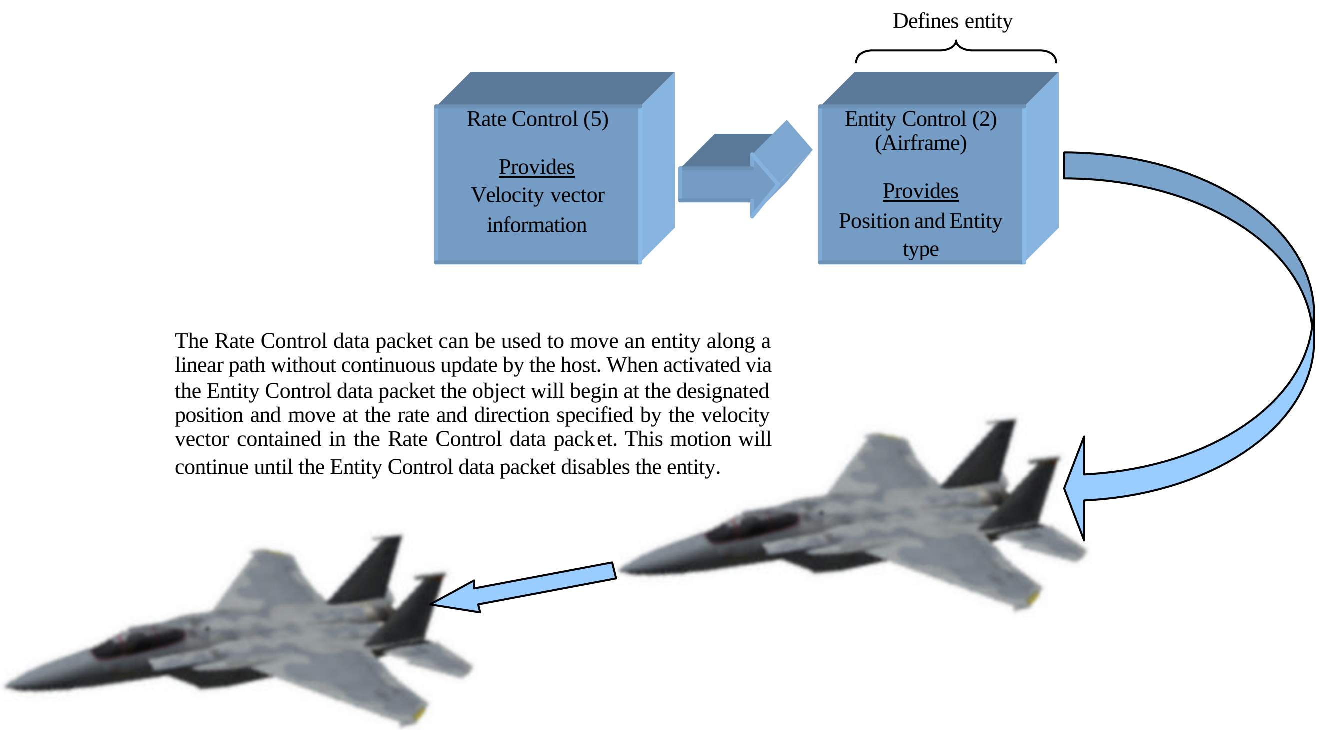


Figure 18 - Rate Control Components

In order to reduce the load on Ethernet messages and the IG computational frame, only Rate Control data packets that contain data changes should be included in the Ethernet message.

The contents of the Rate Control data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 5</td><td colspan="8">Packet Size = 32 bytes</td><td colspan="16">Entity ID</td></tr>
<tr><td colspan="8">Articulated Part ID</td><td colspan="24">Spare</td></tr>
<tr><td colspan="32">Vx Component of the Velocity Vector</td></tr>
<tr><td colspan="32">Vy Component of the Velocity Vector</td></tr>
<tr><td colspan="32">Vz Component of the Velocity Vector</td></tr>
<tr><td colspan="32">Roll Angular Rate</td></tr>
<tr><td colspan="32">Pitch Angular Rate</td></tr>
<tr><td colspan="32">Heading Angular Rate</td></tr>
</table>

Rate Control parameters parameter definitions:

Formats and Ranges	**Description**
Packet ID = 5 : unsigned char : N/A	This parameter identifies this data packet as the Rate Control data packet. There can be multiple instances of this data packet per frame, but each unique Rate Control data packet should only be specified once per frame. If more than one data packet with the same Entity ID, or the same Entity ID and associated Articulated Part ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter specifies the entity to which this data packet will be applied. If the specified Entity ID is not active an error will be generated and no further action will be taken. If the specified entity id contains zero the parameters in this data packet will be applied to the Ownship. If the specified entity id contains a number greater than zero the parameters in this data packet will be applied to the specified entity. If the specified Entity ID is not active, an error will be generated and the data packet will be disregarded.
Articulated Part ID: signed char : N/A valid values: -1 = Apply rates to entity only 0 – 127 identifies a unique articulated part See the articulated part identification assignments in the applicable Database and Entity Attribute Definition Document (s). Default: N/A	This parameter indicates which articulated part is controlled with this data packet. If the data is meant to control the entity only, a –1 should be placed in this field. If the specified Articulated Part ID is not a valid part of the entity specified by the Entity ID, an error will be generated and the data packet will be disregarded.

Vx Component of the Velocity Vector : Float IEEE : meters per second valid values: Minimum to maximum allowed by the data format (+) forward direction (-) backward direction Default: N/A Datum: see Figure 9.	This parameter specifies the X component of the velocity vector for the entity being represented. The velocity vector is specified in the entity reference system as shown in the datum.
Vy Component of the Velocity Vector : Float IEEE : meters per second valid values: Minimum to maximum allowed by the data format (+) right wing direction (-) left wing direction Default: N/A Datum: see Figure 9.	This parameter specifies the Y component of the velocity vector for the entity being represented. The velocity vector is specified in the entity reference system as shown in the datum.
Vz Component of the Velocity Vector : Float IEEE : meters per second valid values: Minimum to maximum allowed by the data format (+) downward direction (-) upward direction Default: N/A Datum: see Figure 9.	This parameter specifies the Z component of the velocity vector for the entity being represented. The velocity vector is specified in the entity reference system as shown in the datum.
Roll Angular Rate : Float IEEE : degrees per second valid values: Minimum to maximum allowed by the data format (+) right wing down (-) left wing down Default: N/A Datum: see Figure 9.	This parameter specifies the roll angular rate for the entity being represented. The angular rate is specified in the entity reference system as shown in the datum.
Pitch Angular Rate : Float IEEE : degrees per second valid values: Minimum to maximum allowed by the data format (+) nose up direction (-) nose direction Default: N/A Datum: see Figure 9.	This parameter specifies the pitch angular rate for the entity being represented. The angular rate is specified in the entity reference system as shown in the datum.

Heading Angular Rate : Float IEEE : degrees per second valid values: Minimum to maximum allowed by the data format (+) clockwise direction (-) counterclockwise direction Default: N/A Datum: see Figure 9.	This parameter specifies the heading angular rate for the entity being represented. The angular rate is specified in the entity reference system as shown in the datum.

5.8 Environment Control

The Environment Control data packet is contained in the Ethernet message sent from the Host to the IG. The Environment Control data packet allows the Host to control the global environment parameters for a given mission scenario. The image generator provides a simulation of the position of the sun and moon based on its internal ephemeris model. The time of day is continuously incremented based on this ephemeris model unless the model is turned off, in which case the time of day will remain at the exact values provided in the Hour and Minute parameters of this data packet. If the Host submits time-of-day information while the ephemeris model is enabled, the current time of day will be changed to the values supplied by the Host, and the ephemeris model will continue to update the value thereafter. Care should be taken when sending this data packet because when the IG receives it, all data parameters contained in the data packet will be updated. If the Environment values are outside the range specified, an error will be returned to the Host and the data packet will be disregarded.

The CIGI Environmental Control data packet contains parameters to support the Moderate Resolution Transmittance (MODTRAN) Code. MODTRAN is a fully validated, government standard software package for calculating important atmospheric quantities such as transmittance and radiance. MODTRAN calculates the atmospheric path transmittance and path radiance values for frequencies from 0 to 50,000 cm-1, with a spectral resolution of approximately 2 cm-1 (20 cm-1 in the UV portion of the spectrum).

Sun and moon positions, moon phase, and horizon glow are computed by the image generator and do not require Host control.

In order to reduce the load on Ethernet messages and the IG computational frame, only Environment data packets that contain data changes should be included in the Ethernet message.

The contents of the Environment data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 6</td><td colspan="8">Packet Size = 36 bytes</td><td colspan="8">Hour</td><td colspan="8">Minute</td></tr>
<tr><td>*1</td><td colspan="7">Humidity</td><td>*2</td><td colspan="23">Spare</td></tr>
<tr><td colspan="32">Date</td></tr>
<tr><td colspan="32">Air Temperature</td></tr>
<tr><td colspan="32">Global Visibility</td></tr>
<tr><td colspan="32">Wind Velocity</td></tr>
<tr><td colspan="32">Wind Direction</td></tr>
<tr><td colspan="32">Barometric Pressure</td></tr>
<tr><td colspan="32">Aerosol</td></tr>
</table>

Environment Control parameter definitions:

Formats and Ranges	Description
Packet ID = 6 : unsigned char : N/A	This parameter identifies this data packet as the Environment Control data packet. There can be only one instance of this data packet per frame. If more than one data packet is received the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
Hour : unsigned char : hours valid values: 0 – 23 Default: 0 Datum: Local time	This parameter indicates the hour of the day for the ephemeris program within the image generator.

Minute : unsigned char : minutes valid values: 0 – 59 Default: 0 Datum: Local time	This parameter indicates the minute of the hour for the ephemeris program within the image generator.
*1 Ephemeris on/off : Boolean : N/A valid values: 0 = Static Time of Day 1 = Continuous Time of Day Default: 1, Ephemeris program active	This parameter controls whether a continuous time of day or static time of day is used for a mission. If set to continuous, the image generator will update the time of day.
Humidity : unsigned 7 bit field : percent valid values: 0 to 100 101 to 127 are invalid Default: N/A	This parameter indicates the global humidity of the environment.
*2 Modtran on/off : Boolean : N/A valid values: 0 = Modtran off 1 = Modtran on Default: 0	This parameter determines whether atmospherics will be included in the calculations. An “Off” setting means that source radiance will be calculated, whereas, an “On” setting means that apparent radiance will be calculated.
Date : integer : MMDDYYYY valid values: MMDDYYYY = (month number * 1000000) + (day number * 10000) + year number Default: N/A	This parameter indicates the desired date for use by the ephemeris program within the image generator.
Air Temperature : Float IEEE: degrees Celsius valid values: Minimum to maximum allowed by the data format Default: N/A Datum: 0^{o} C	This parameter indicates the global temperature of the environment.
Global Visibility : Float IEEE: meters valid values: 0 to maximum allowed by the data format Default: 0	This parameter indicates the global visibility.

Wind Velocity : Float IEEE: meters per second valid values: 0 to maximum allowed by the data format Default: 0	This parameter indicates the global velocity of the wind.
Wind Direction : Float IEEE: degrees valid values: 0 to +360 clockwise Default: N/A Datum: True North	This parameter indicates global direction of the wind.
Barometric Pressure : Float IEEE: millibars (mb) valid values: 0.0 to maximum allowed by the data format Default: 0.0	This parameter controls the atmospheric pressure input into MODTRAN. Typically, this will default to the value used when defining the model atmosphere.
Aerosol : Float IEEE: gm/m^3 valid values: 0.0 to maximum allowed by the data format Default: 0.0	This parameter controls the liquid water content for the defined atmosphere. Typically, this will default to the value used when defining the model atmosphere.

5.9 Weather Control

The Weather Control data packet is contained in the Ethernet message sent from the Host to the IG. This data packet is used to control and/or override default local or layered weather phenomena. The parameters within this data packet allow for the descriptions of haze, ground fog, rain, cloud layers, etc. When the data packet is used to represent ground fog, the Host is responsible for maintaining the relationship between visibility and Runway Visual Range via the Runway Visibility Range parameter of this data packet.

Figure 19 - Cloud Elevation and Thickness

Particular weather phenomena may be assigned to an entity. Hence, position/orientation controls for weather are controlled via the Entity Control data packet for these types. When a Weather Definition data packet is used to control a particular instance of an entity, some parameters within this packet may not be used. For instance when a Weather Definition data packet defines the attributes of a Thunderstorm entity only the following parameters of this data packet will typically be used, all others will be ignored.

Packet ID	Severity
Packet Size	Phenomenon Type = 0
Entity ID	Air Temperature
Weather State	Opacity

In order to reduce the load on Ethernet messages and the IG computational frame, only Weather Control data packets that contain data changes should be included in the Ethernet message.

The contents of the Weather Control data packet are shown below.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Packet ID = 7								Packet Size = 44 bytes								Entity ID															
*1	*2	*3	*4			Spare										Phenomenon Type															
Air Temperature																															
Opacity/Runway Visibility Range																															
Scud																															
Coverage																															
Elevation																															
Thickness																															
Transition Band																															
Winds Aloft Velocity																															
Winds Aloft Direction																															

Weather Control parameter definitions:

Formats and Ranges	Description
Packet ID = 7 : unsigned char : N/A	This parameter identifies this data packet as the Weather Control data packet. There can be multiple instances of this data packet per frame. Each instance should uniquely identify a weather phenomenon via the Entity ID parameter in this data packet.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter indicates which Entity ID, i.e. Entity Control data packet to which this weather model is assigned. This field may be valid only for particular local weather phenomena (e.g. thunderstorms, fronts, and sandstorms). This parameter is only used when the Phenomenon Type of this data packet is zero.
*1 Weather State : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter indicates whether the phenomena specified by this data packet is visible (*Enable*) or not (*Disable*).
*2 Scud : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter indicates whether there will be scud effects applied to the phenomenon specified by this data packet. If this parameter is applied to the ground fog layer, it will cause a patchy fog effect.
*3 Random Winds Aloft: Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter indicates whether a random frequency and duration should be applied to the Winds Aloft value. This is meant to provide for gusting winds. Winds Aloft will enable phenomenon drift contrasting with the global winds parameters defined in the Environment Control data packet.
*4 Severity : 3 bit field : N/A valid values: 0 – 5 Least to most severe Default: N/A	This parameter indicates the severity of the weather phenomenon. This parameter can be used to control such things as thunderstorm severity or sea state.

Phenomenon Type : unsigned short: N/A valid values: 0 = Use Entity ID 1 = Cloud Layer 1 2 = Cloud Layer 2 3 = Ground Fog 4 = Rain 5 = Snow 6 = Sand 7 to 65535 defined by the IG Default: N/A	This parameter indicates the type of weather described by this data packet. Values zero through six are provided to establish a common numbering scheme for standard layered weather.
Air Temperature : Float IEEE: degrees Celsius valid values: Minimum to maximum allowed by the data format Default: N/A Datum: 0^o C	This parameter indicates the local temperature inside the weather phenomenon.
Opacity : Float IEEE: Percent valid values: 0 to 100% Default: 0 ------------------ Or ------------------ Runway Visibility Range : Float IEEE: meters valid values: 0 to < global visibility Default: 0	This parameter indicates the opacity of the weather phenomenon. One hundred percent opacity produces zero visibility through the phenomena. This control is meant to provide a transmissive or density effect for the weather (e.g. wispy clouds, rain severity, snow severity) This parameter indicates the distance from the eye point to a point where the scene is completely fogged. This parameter is valid only when the Phenomenon Type is set to Ground Fog (3).
Scud : Float IEEE: Percent valid values: 0 to 100% Default: N/A	This parameter indicates the frequency of the scud effect. Valid values range from 0% meaning no scud affect, to 100%, which indicates a solid effect. If this parameter is applied to the ground fog layer, it will cause a patchy fog effect.
Coverage : Float IEEE: Percent valid values: 0 to 100% Default: N/A	This parameter indicates the amount of area coverage a particular phenomenon has over the specified global visibility range given in the Environment Control data packet (e.g., 100% for a cloud layer produces a solid cloud layer).

Elevation : Float IEEE: meters valid values: 0 to maximum allowed by the data format Default: 0 Datum: MSL	This parameter indicates the base (bottom) altitude of the weather phenomenon.
Thickness : Float IEEE: meters valid values: 0 to maximum allowed by the data format Default: 0 Datum: Elevation upward as defined in this data packet.	This parameter indicates the vertical thickness of the weather phenomenon. When applied to clouds or fog, the altitude at the top of the layer is equal to the elevation plus the thickness.
Transition Band : Float IEEE: meters valid values: 0 to maximum allowed by the data format Default: N/A Datum: As defined in this data packet, Elevation downward for the bottom and Elevation plus Thickness upward for the top.	This parameter indicates a vertical transition band both above and below (if applicable) a phenomenon. Within this band, visibility transitions from the specified opacity to the global visibility value given in the Environment Control data packet.
Winds Aloft Velocity : Float IEEE: meters per second valid values: 0 to maximum allowed by the data format Default: 0	This parameter indicates the local velocity of the wind applied to the phenomenon. Setting this parameter to zero disables Winds Aloft.
Winds Aloft Direction : Float IEEE: degrees valid values: 0 to +360 clockwise Default: N/A Datum: True North	This parameter indicates local direction of the wind applied to the phenomenon.

5.10 View Control

The View Control data packet is contained in the Ethernet message sent from the Host to the IG. It is used to attach a view to an entity and to define the position and orientation of the view relative to the entity's reference point. This concept can be used to specify view offsets that may be used for such things as pilot eye, weapon/sensor viewpoints, and stealth views. It should be noted that the number of views might be limited by the IG configuration. It is also assumed that the characteristics of a view found in the View Definition data packet of this interface are defaulted within the IG or must be specified by the View Definition data packet in section 5.14.

The order of operation to move and rotate the view must be such that the view is translated with respect to the entity's body axis by first applying the X, Y, and Z offset values and then rotating the view about the Z, Y, and the X axes (i.e. applying heading, pitch and then roll). This data packet should be supplied to the IG at mission initialization and then again any time the view must be moved relative to an entity's reference point.

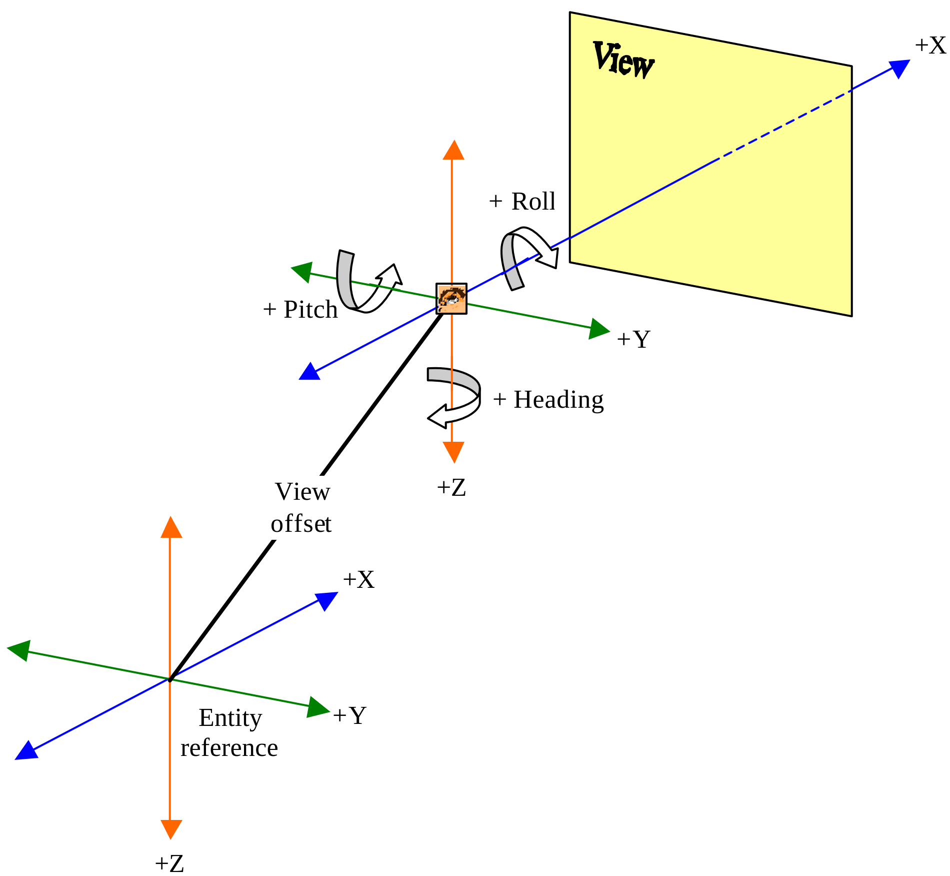


Figure 20 – View Point Offset and Orientation from Entity Reference

The contents of the View Control data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 8</td><td colspan="8">Packet Size = 32 bytes</td><td colspan="16">Entity ID</td></tr>
<tr><td colspan="5">View ID</td><td colspan="3">*1</td><td>*2</td><td>*3</td><td>*4</td><td>*5</td><td>*6</td><td>*7</td><td colspan="18">Spare</td></tr>
<tr><td colspan="32">X Offset</td></tr>
<tr><td colspan="32">Y Offset</td></tr>
<tr><td colspan="32">Z Offset</td></tr>
<tr><td colspan="32">View Roll</td></tr>
<tr><td colspan="32">View Pitch</td></tr>
<tr><td colspan="32">View Heading</td></tr>
</table>

View Control parameter definitions:

Formats and Ranges	Description
Packet ID = 8 : unsigned char : N/A	This parameter identifies this data packet as a View Control data packet. There can be multiple instances of this data packet per frame, but each unique view control should only be specified once per frame. If more than one data packet with the same Entity ID and View ID or View Group Select is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: 0	This parameter indicates the entity to which this view should be attached.
View ID: 5 bit field: N/A valid values: 0 – 31 View ID Default = 0	This parameter specifies which view position is associated with offsets and rotations specified by this data packet. If the Host requests a view that has not been configured on the IG, an error will be generated and the data packet will be disregarded
*1 View Group Select: 3 bit field: N/A valid values: 0 = invalid 1 – 7 Group select Default = 0	This parameter specifies which view group is to be controlled by the offsets specified by this data packet. View groups are defined using the View Definition data packet. When this parameter is 0, the field is disregarded, and position control is performed on an individual view specified by the View ID in this data packet. Otherwise, the transformation is applied to each view in the specified group. If the Host requests a group that has not be defined by the View Definition data packet or pre-configured on the IG, an error will be generated and the data packet will be disregarded.

*2 Enable/Disable View X Offset : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the View X Offset value contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.
*3 Enable/Disable View Y Offset : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the View Y Offset value contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.
*4 Enable/Disable View Z Offset : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the View Z Offset value contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.
*5 Enable/Disable View Roll : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the View Roll value contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.
*6 Enable/Disable View Pitch : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the View Pitch value contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.
*7 Enable/Disable View Heading : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the View Heading value contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.

X Offset: Float IEEE : meters valid values: Minimum to maximum allowed by the data format Default: 0 Datum: Entity coordinate system, see Figure 9.	This parameter is used to define the X component of the view offset vector along the entity's longitudinal axis, as shown in Figure 20.
Y Offset: Float IEEE : meters valid values: Minimum to maximum allowed by the data format Default: 0 Datum: Entity coordinate system, see Figure 9.	This parameter is used to define the Y component of the view offset vector along the entity's lateral axis, as shown in Figure 20.
Z Offset: Float IEEE : meters valid values: Minimum to maximum allowed by the data format Default: 0 Datum: Entity coordinate system, see Figure 9.	This parameter is used to define the X component of the view offset vector along the entity's vertical axis, as shown in Figure 20.
View Roll: Float IEEE: degrees valid values: 0 to +180 clockwise 0 to –180 counter clockwise Default: As set in the models default Datum: Entity coordinate system, see Figure 9.	This parameter specifies the rotation about the view's X axis, as shown in Figure 20.
View Pitch: Float IEEE: degrees valid values: 0 to +90 up 0 to –90 down Default: As set in the models default Datum: Entity coordinate system, see Figure 9.	This parameter specifies the rotation about the view's Y axis, as shown in Figure 20.
View Heading: Float IEEE: degrees valid values: 0 to +360 clockwise Default: As set in the models default Datum: Entity coordinate system, see Figure 9.	This parameter specifies the rotation about the view's Z axis, as shown in Figure 20.

5.11 Sensor Control

The Sensor Control data packet is contained in the Ethernet message sent from the Host to the IG. This data packet, in conjunction with the View Control, View Definition, Component Control, and LOS Range Request data packets, control and describe the abilities of a sensor-based weapon system.

This Sensor Control data packet will provide sensor mode-of-operation and display behavior. This data packet is associated with a particular view via the View ID parameter. This allows for positioning and orienting a sensor with respect to an entity. Field-of-view characteristics may be controlled using a View Definition data packet, which is also associated with a view using the View ID parameter of that data packet. Discrete sensor attributes are controlled using one or more Component Control data packets. These data packets are also associated with a particular view via the View ID parameter of the data packets. The Line-of-Sight Range Request data packet is used to request a Line-of-Sight Range Response data packet from the IG that contains the intersection point on the database along the Line-of-Sight vector based on the sensor position and orientation specified in the View Control data packet.

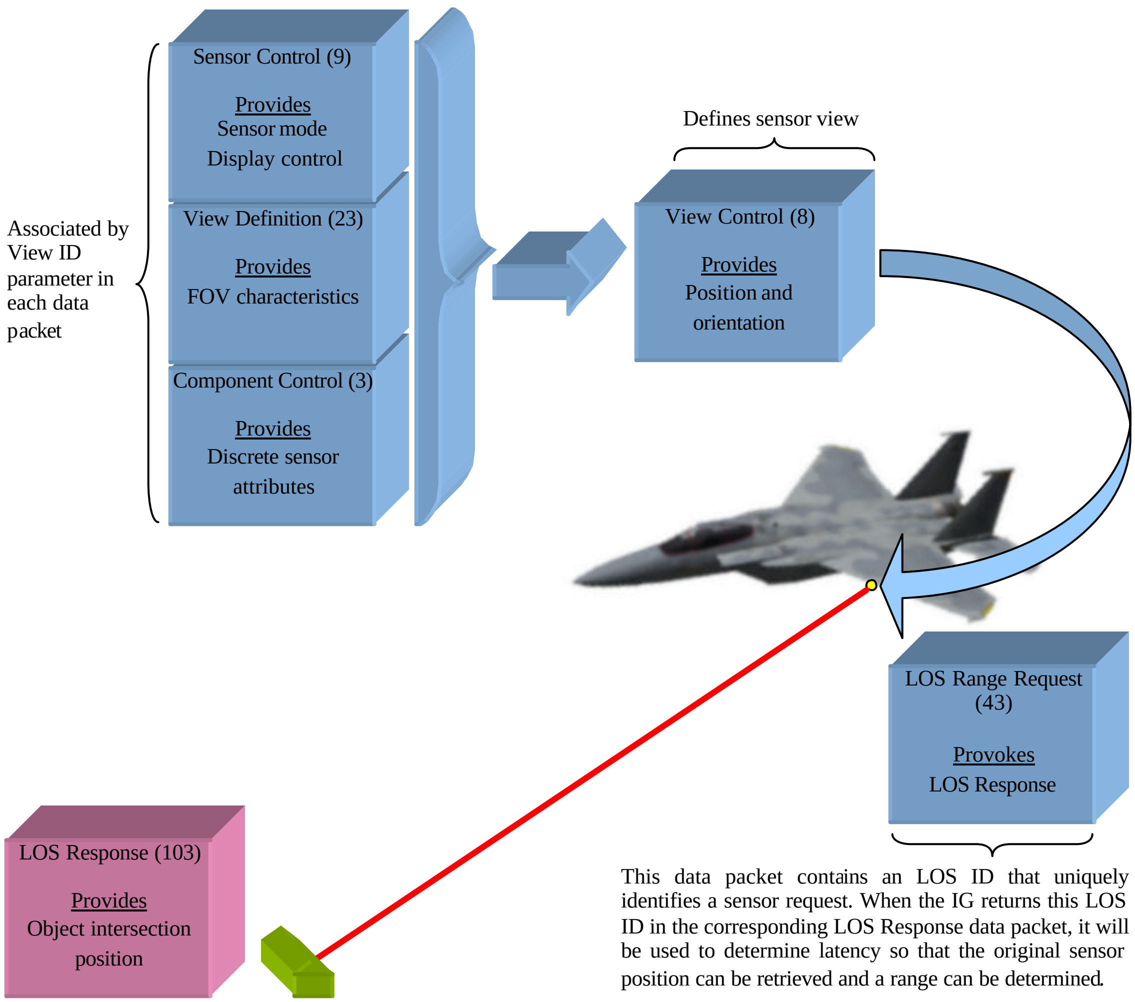


Figure 21 – Sensor Definition Components

When a Sensor Control data packet is sent from the Host, associated Component Control and LOS Range Request data packets may also be sent. It will be useful to the Host to know which series of data packets are associated with the same request. Even though the LOS Range Request has a unique LOS ID it may be useful to have the Host internally associate these packets together via the Sensor ID parameter of the Sensor Control data packet. To match up Sensor Control and other associated data packets from the host with responses from the IG, the Sensor ID parameter is used. The same number that is placed in this parameter will be returned in the Sensor ID parameter of the corresponding Sensor Response data packet as described in section 5.25. The Sensor ID value should be manipulated in such a way as to not duplicate the value in a reasonable amount of time (typically one second). This will prevent similarly identified requests from being lost by the IG.

There is no restriction on the number of Sensor Control data packets that can be sent in a single frame. The user should be aware, however, that the response time of the IG might be degraded under conditions that overload the IG Sensor computation mechanism.

The contents of the Sensor Control data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 9</td><td colspan="8">Packet Size = 24 bytes</td><td colspan="5">View ID</td><td>*1</td><td>*2</td><td>*3</td><td colspan="8">Sensor ID</td></tr>
<tr><td colspan="4">*4</td><td>*5</td><td>*6</td><td colspan="26">Spare</td></tr>
<tr><td colspan="32">Gain</td></tr>
<tr><td colspan="32">Level</td></tr>
<tr><td colspan="32">AC Coupling</td></tr>
<tr><td colspan="32">Noise</td></tr>
</table>

Sensor Control parameter definitions:

Formats and Ranges	Description
Packet ID = 9 : unsigned char : N/A	This parameter identifies this data packet as a Sensor Control data packet. There can be multiple instances of this data packet per frame, but each unique sensor control should only be specified once per frame. If more than one data packet with the same View ID and Sensor ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
View ID: 5 bit field: N/A valid values: 0 = Not Used 1 – 31 Weapon Sensor view point Default = 0	This parameter dictates to which view the corresponding sensor is assigned, regardless of the view group. The offsets associated with a view are specified via the View Control data packet.
*1 Sensor on / off : Boolean : N/A valid values: 0 = off 1 = on Default = 0	This parameter indicates whether this Sensor is turned on or off.

*2 Polarity : Boolean : N/A valid values: 0 = White Hot 1 = Black Hot Default = 0	This parameter indicates whether this Sensor is showing white hot or black hot.
*3 line-by-line drop-out : Boolean : N/A valid value: 0 = off 1 = on Default = 0	This parameter indicates whether the line-by-line drop-out feature is enabled.
Sensor ID : unsigned char : N/A valid values: 0 to 255 Default: 0	This parameter is used to identify the Sensor Control so that when a Sensor Control packet is received, it can be identified by the host. This is done via the Sensor ID parameter of the Sensor Response data packet.
*4 Track mode : 4 bit field: N/A valid values: 0 = off 1 = Force correlate 2 = Scene 3 = Target 4 = Ship 5 – 15 invalid Default = 0	This parameter indicates which track mode the sensor should use: Force Correlate – The maverick seeker image-processes a portion of the view, establishes an image pattern, and attempts to keep the seeker pointed at the center of this image pattern. Scene - The FLIR seeker image-processes a portion of the view, establishes an image pattern, and attempts to keep the seeker pointed at the center of this image pattern. Target - Contrast tracking is locked to a specific (target) area. Ship - Contrast tracking, where the tracking point is adjusted so that the weapon strikes close to the water line. If an invalid value is received, an error will be generated and not further action will be taken.
*5 Automatic Gain: Boolean: N/A valid values: 0 = off 1 = on Default = 0	This parameter, when set to “on,” causes the Weapons Sensor to automatically adjust the gain value to optimize the brightness and contrast of the sensor display.

*6 Track white or black : Boolean: N/A valid values: 0 = white 1 = black Default = 0	This parameter causes the Weapons Sensor to track either white or black.
Gain : Float IEEE : N/A Valid values: 0.0 to 100.0 0.1 Default = N/A	This parameter indicates the gain value for the weapon sensor option. Gain and Level are used together to improve the contrast of the target imagery.
Level : Float IEEE : N/A Valid values: 0.0 to 1.0 Default = N/A	This parameter indicates the level value for the weapon sensor option. Level and Gain are used together to improve the contrast of the target imagery.
AC Coupling : Float IEEE : N/A Valid values: 0.0 to 1.0 Default = N/A	This parameter indicates the AC Coupling decay rate for the weapon sensor option. This feature is only available when the IG is equipped with enhanced Weapons Sensor effects.
Noise : Float IEEE : N/A Valid values: 0.0 to 1.0 Default = 0.0 (noise off)	This parameter indicates the detector-noise gain for the weapon sensor option.

5.12 Trajectory Definition

The Trajectory Definition data packet is contained in the Ethernet message sent from the Host to the IG. This data packet is used in conjunction with the Entity Control, the Special Effects Definition, and the Rate data packets to define aspects of an object's trajectory. The Entity Control data packet will provide the initial position for the trajectory. If the entity is attached to another entity, then the positional data will be used as an offset from the parent's reference point. Other applicable Entity Control parameters may also apply. The Special Effects Definition data packet will provide effects duration, burst count, separation, color, etc.; the Rate data packet will provide the initial velocity components.

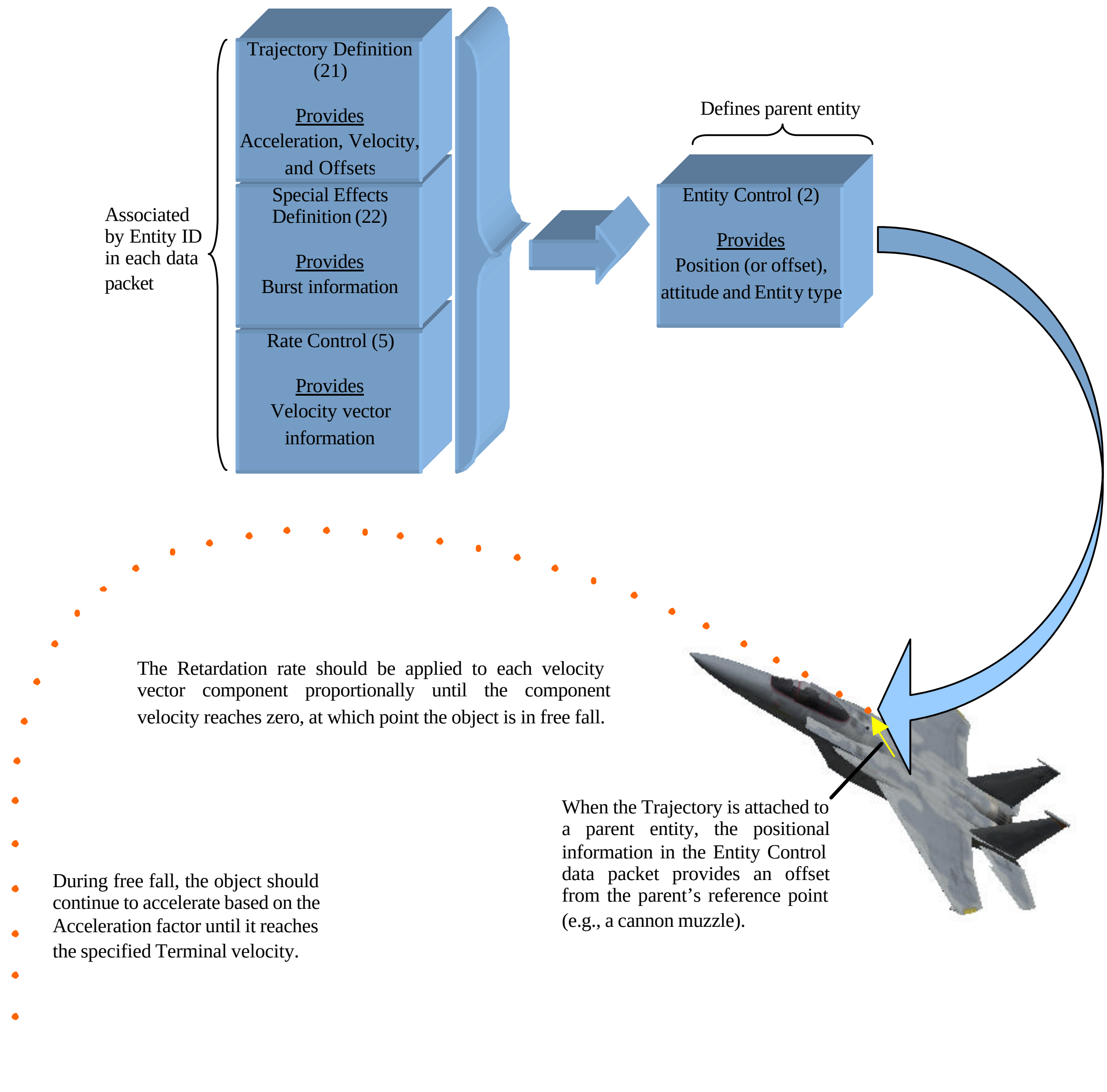


Figure 22 – Trajectory Definition Components

The contents of the Trajectory Definition data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 21</td><td colspan="8">Packet Size = 16 bytes</td><td colspan="16">Entity ID</td></tr>
<tr><td colspan="32">Acceleration factor</td></tr>
<tr><td colspan="32">Retardation rate</td></tr>
<tr><td colspan="32">Terminal velocity</td></tr>
</table>

Trajectory Definition parameter definitions:

<table>
<tr><th>Formats and Ranges</th><th>Description</th></tr>
<tr><td>Packet ID = 21 : unsigned char : N/A</td><td>This parameter identifies this data packet as the Trajectory Definition data packet.

There can be multiple instances of this data packet per frame, but each unique trajectory definition should only be specified once per frame. If more than one data packet with the same Entity ID is received in the same frame, the last one received will be used.</td></tr>
<tr><td>Packet Size : unsigned char : N/A</td><td>This parameter indicates the number of bytes in this data packet.</td></tr>
<tr><td>Entity ID : unsigned short : N/A

valid values:

0 to 65535

Default: N/A</td><td>This parameter indicates which entity is being influenced by this trajectory behavior. The specified Entity ID must have been previously defined. If this data packet is sent without a previously defined Entity ID either earlier in the same Ethernet message or in an earlier Ethernet message, an error will be generated and this data packet will be ignored.</td></tr>
<tr><td>Acceleration factor : Float IEEE: meters/second^2

valid values:

0 to maximum allowed by the data format

Default: N/A</td><td>This parameter indicates the acceleration factor that will be applied to the Vz component of the velocity vector over time to simulate the effects of gravity on the object.</td></tr>
<tr><td>Retardation rate : Float IEEE: meters/second

valid values:

0 to maximum allowed by the data format

Default: N/A</td><td>This parameter indicates what retardation factor will be applied to the object's motion. This factor will be used to proportionally reduce the Vx, Vy and Vz components of the velocity vector over time until they reach zero to simulate the effect of frictional forces acting upon the object.</td></tr>
<tr><td>Terminal Velocity : Float IEEE: meters/second

valid values:

0 to maximum allowed by the data format

Default: N/A</td><td>This parameter indicates what final velocity the object will be allowed to obtain.</td></tr>
</table>

5.13 Special Effect Definition

The Special Effect Definition data packet is contained in the Ethernet message sent from the Host to the IG. This data packet is used in conjunction with the Entity Control data packet to override the default-modeled parameters within an effect. The effect must be loaded via an Entity Control data packet either earlier in the same Ethernet message or in an earlier Ethernet message before the Special Effects Definition can be applied.

In order to reduce the load on Ethernet messages and the IG computational frame, only Special Effect Definition data packets that contain data changes should be included in the Ethernet message.

The contents of the Special Effect Definition data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 22</td><td colspan="8">Packet Size = 32 bytes</td><td colspan="16">Entity ID</td></tr>
<tr><td>*1</td><td>*2</td><td colspan="6">Spare</td><td colspan="8">Red color value</td><td colspan="8">Green color value</td><td colspan="8">Blue color value</td></tr>
<tr><td colspan="16">X Scale</td><td colspan="16">Y Scale</td></tr>
<tr><td colspan="16">Z Scale</td><td colspan="16">Time Scale</td></tr>
<tr><td colspan="16">Spare</td><td colspan="16">Burst Count</td></tr>
<tr><td colspan="32">Separation</td></tr>
<tr><td colspan="32">Burst Rate</td></tr>
<tr><td colspan="32">Duration</td></tr>
</table>

Special Effect Definition parameter definitions:

Formats and Ranges	Description
Packet ID = 22 : unsigned char : N/A	This parameter identifies this data packet as the Special Effect Definition data packet. There can be multiple instances of this data packet per frame, but each unique special effects definition should only be specified once per frame. If more than one data packet with the same Entity ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter indicates which effect is being modified. An effect must be previously assigned to an Entity ID. If this data packet is sent without a previously defined Entity ID either earlier in the same Ethernet message or in an earlier Ethernet message, an error will be generated and this data packet will be ignored.
*1 Sequence Direction : Boolean : N/A valid values: 0 = Forward 1 = Backward Default: 0	This parameter indicates whether the effect animation sequence should be sequence from beginning to end or vice versa.

*2 Color switch : Boolean : N/A valid values: 0 = Off 1 = On Default: 0	This parameter indicates whether the Red, Green, and Blue color values specified in this data packet will be applied to the special effect.
Red color value : unsigned char : N/A valid values: 0-255 Default: N/A	This parameter specifies the red component of a color bias to be applied to the effect. This bias will be multiplied with the effect's original color. If the color switch indicates off, this parameter will be ignored and the color of the effect will be modeled.
Green color value : unsigned char : N/A valid values: 0-255 Default: N/A	This parameter specifies the green component of a color bias to be applied to the effect. This bias will be multiplied with the effect's original color. If the color switch indicates off, this parameter will be ignored and the color of the effect will be modeled.
Blue color value : unsigned char : N/A valid values: 0-255 Default: N/A	This parameter specifies the blue component of a color bias to be applied to the effect. This bias will be multiplied with the effect's original color. If the color switch indicates off, this parameter will be ignored and the color of the effect will be modeled.
X Scale : scaled distance format (16 bit B6): N/A valid values: limits of scaled distance format (16 bit B6) Default: 1.0 Datum: Entity coordinate system, see Figure 9.	This parameter specifies a scale factor to apply to the effect's dimension along the longitudinal axis.
Y Scale : scaled distance format (16 bit B6): N/A valid values: limits of scaled distance format (16 bit B6) Default: 1.0 Datum: Entity coordinate system, see Figure 9.	This parameter specifies a scale factor to apply to the effect's dimension along the lateral axis.
Z Scale : scaled distance format (16 bit B6): N/A valid values: limits of scaled distance format (16 bit B6) Default: 1.0 Datum: Entity coordinate system, see Figure 9.	This parameter specifies a scale factor to apply to the effect's dimension along the height axis.

Time Scale : scaled distance format (16 bit B6): N/A valid values: limits of scaled distance format (16 bit B6) Default: 1.0	This parameter specifies a scale factor to apply to the effect's animation speed.
Burst Count : unsigned short : number of repetitions of an effect valid values: 0 = ignore 1 to Maximum allowed by data format Default: N/A	This parameter indicates how many effects are in a single burst. This allows the display of a certain number of repetitions of an effect using a single data packet. This feature may be useful for rendering gun flashes, for example; the Host could specify that 15 rounds would be fired from a particular weapon where each firing would be represented by a gun flash effect. Rather than submit 15 data packets to control those flashes, a single data packet with this field set to 15 would achieve the same result. The Host may control the frequency of the bursts by setting the Burst Rate field of this data packet to an appropriate value.
Separation : float IEEE: meters valid values: 0 = ignore >0 to Maximum allowed by data format Default: N/A	This parameter indicates the distance between effects within a burst when the Burst Count parameter of this data packet is greater than zero.
Burst Rate : float IEEE: seconds valid values: 0 = ignore >0 to Maximum allowed by data format Default: N/A	This parameter indicates the time between bursts when the Burst Count parameter of this data packet is greater than zero.
Duration : float IEEE : second valid values: -1 = always on 0 = using default >0 to Maximum allowed by data format Default: 0.	This parameter indicates how long an effect or sequence of bursts will be active. If an effect has a non-negative duration the effect will automatically be disabled after the duration elapses.

5.14 View Definition

The View Definition data packet is contained in the Ethernet message sent from the Host to the IG. It is used to define the characteristics of a view and/or override the IG default configuration.

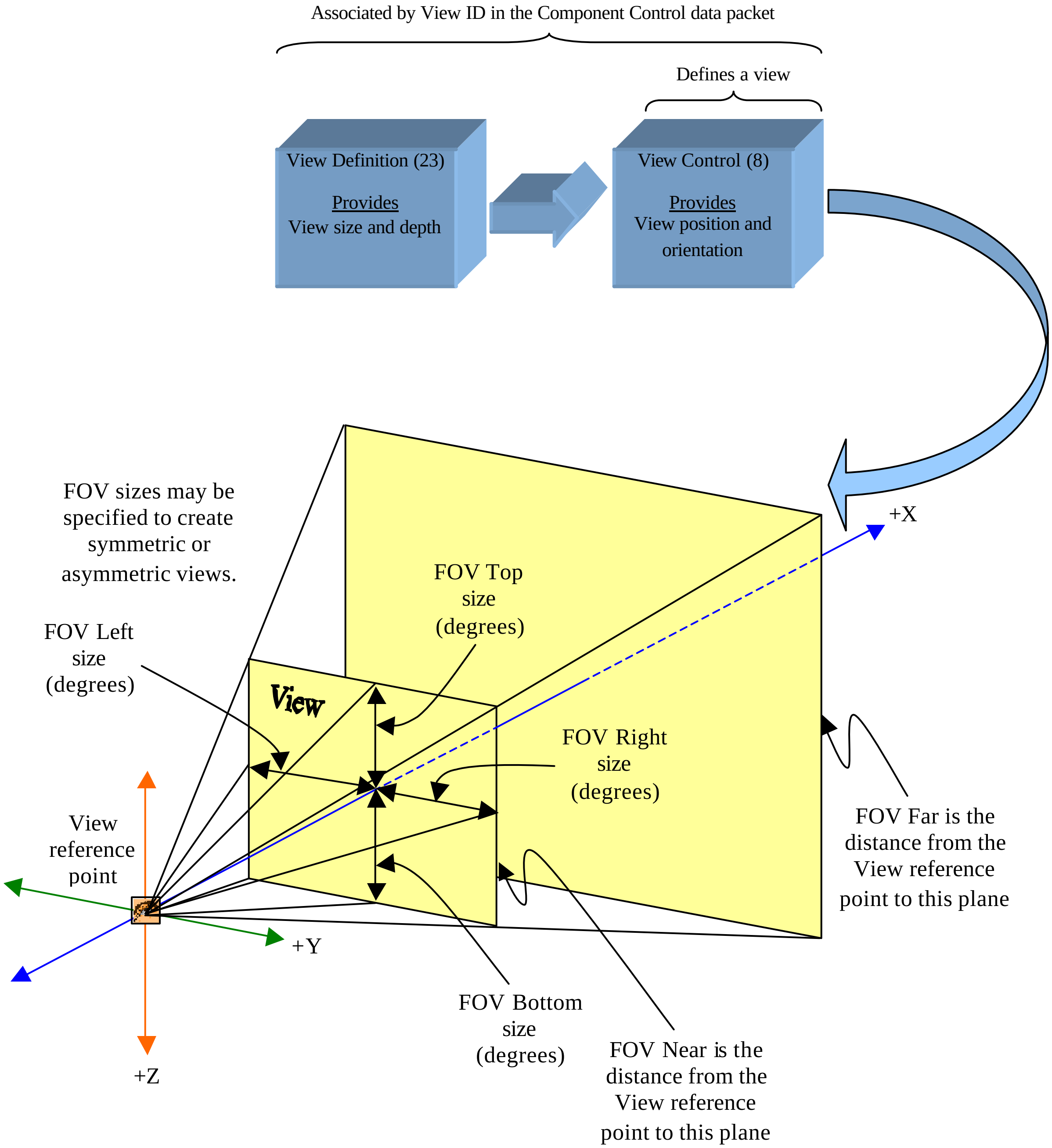


Figure 23 - View Definition Nomenclature

Indicating "None" in the View Group Assign parameter allows for the definition of an individual view. In this case, parameters within this data packet will be applied to the specified View ID. Figure 24 shows three individual views that would have been defined using three instances of the View Definition data packet. Each instance would have a unique View ID, one through three. In all of these instances, the View Group Assign would have indicated "None."

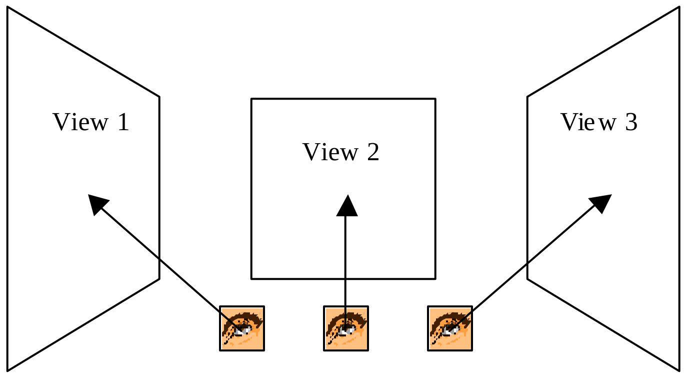


Figure 24 - Individual definitions for three views

Individual views may be grouped together by using the View Group parameter of this data packet. In this way, several views can be moved in unison through the use of a single View Control data packet. To do this, two or more View Definition data packets will be used to assign individual views to the same view group via the View Group Assign and View ID parameters of this data packet. In this case, only the View Group Assign and View ID parameters within this data packet are used. If, for example the three individual views shown in Figure 24 were group together in this fashion, the result would be as illustrated in Figure 25.

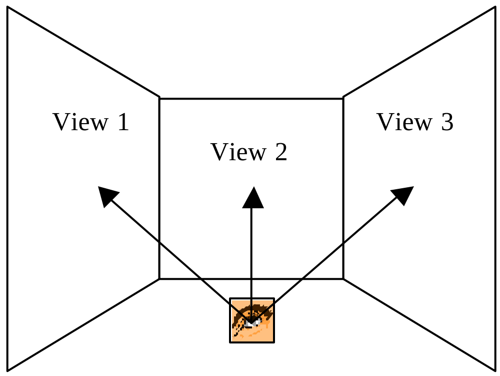


Figure 25- A grouping of three individual views

If the View Group parameter is set to "None" and the View ID parameter corresponds to a view that is a member of a view group, that view will be separated from the group as shown in Figure 26.

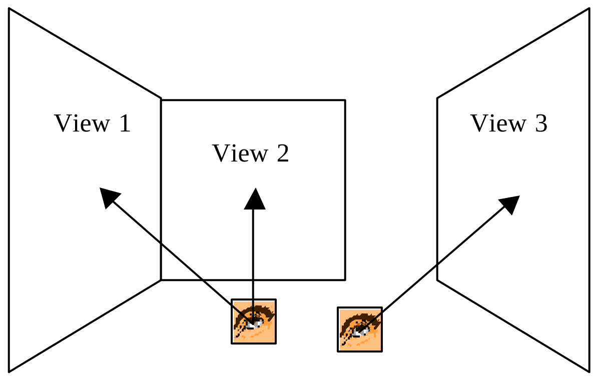


Figure 26 - View three separated from a view group

In order to reduce the load on Ethernet messages and the IG computational frame, only View Definition data packets that contain data changes should be included in the Ethernet message.

The contents of the View Definition data packet are shown below.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Packet ID = 23								Packet Size = 32 bytes								View ID					*1			*2			*3			*4	
*5	*6	*7	*8	*9	*10	*11	Spare																								
Field of View Near																															
Field of View Far																															
Field of View Left																															
Field of View Right																															
Field of View Top																															
Field of View Bottom																															

View Definition parameter definitions:

Formats and Ranges	Description
Packet ID = 23 : unsigned char : N/A	This parameter identifies this data packet as a View Definition data packet. There can be multiple instances of this data packet per frame, but each unique view definition should only be specified once per frame. If more than one data packet with the same View Group Assignment and/or a View ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
View ID: 5 bit field: N/A valid values: 0 – 31 View ID Default = 0	This parameter specifies the view to which this packet should be applied. If the Host requests a view that has not been configured on the IG, an error will be generated and the data packet will be disregarded.

*1 View Group Assign: 3 bit field: N/A valid values: 0 = none 1 – 7 Group select Default = 0	This parameter specifies the view group to which the view should be assigned. Only one view can be added to a view group per data packet. Therefore, two or more View Definition data packets will need to be used to define a group of views. When this parameter is 0, the view specified by the View ID field is separated from any groups, and the packet is applied only to that view.
*2 View Type: 3 bit field: N/A valid values: 0 – 7 Default = 0	This parameter specifies the view type. The integration engineer should consult the image generator configuration to determine what types of views are available and what their View Type assignments should be.
*3 Pixel Replication: 3 bit field: N/A valid values: 0 = No Replicate 1 = 1x2 Pixel Replicate 2 = 2x1 Pixel Replicate 3 = 2x2 Pixel Replicate 4 = TBD 5 = TBD 6 = TBD 7 = TBD Default = 0	This parameter specifies what pixel replication function should be applied to the view. This function is typically used in particular sensor applications to perform electronic zoom (pixel and line doubling function).
*4 View Mirror: 2 bit field: N/A valid values: 0 = None 1 = Horizontal 2 = Vertical 3 = Horizontal and Vertical Default = 0	This parameter specifies what mirroring functions should be applied to the view. This function is typically used to replicate the view of a mirrored surface used in display systems or rear view mirrors.
*5 Tracker Assign: 1 bit field: N/A valid values: 0 = Not Assigned 1 = Assigned Default = 0	This parameter specifies whether the view should be attached to the input controls from an external tracking device.

*6 Enable/Disable Field of View Near : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Field of View Near value contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.
*7 Enable/Disable Field of View Far : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Field of View Far value contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.
*8 Enable/Disable Field of View Left : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Field of View Left value contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.
*9 Enable/Disable Field of View Right : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Field of View Right contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.
*10 Enable/Disable Field of View Top : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Field of View Top contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.
*11 Enable/Disable Field of View Bottom : Boolean : N/A valid values: 0 = Disable 1 = Enable Default: 0	This parameter identifies whether the Field of View Bottom value contained in this data packet is manipulated from the Host, i.e. Enabled, or not manipulated by the Host, i.e. Disabled.

Field of View Near: Float IEEE : meters Valid values: >0 to maximum allowed by the data format Default: 1.0 Datum: View Reference Point, see Figure 23.	This parameter is used to define the near clipping plane for the view. Any object inside of this distance will be clipped from the view. It should be noted that 0 and negative numbers are invalid values.
Field of View Far: Float IEEE : meters valid values: >Field of View Near to maximum allowed by the data format Default: N/A Datum: View Reference Point, see Figure 23.	This parameter is used to define the far clipping plane for the view. Any object outside of this distance will be clipped from the view. This value must be a positive number greater than the Field of View Near value. It should be noted that 0 is not a valid value.
Field of View Left : Float IEEE : degrees valid values: -179.9 to 179.9 Default: N/A Datum: View Reference Point, see Figure 23.	This parameter is used to define the size of the left side of the field of view. This value should always be less than the Field of View Right. If this condition is not satisfied, an error will be generated and the data packet will be disregarded.
Field of View Right: Float IEEE : degrees valid values: -179.9 to 179.9 Default: N/A Datum: View Reference Point, see Figure 23.	This parameter is used to define the size of the right side of the field of view. This value should always be greater than the Field of View Left value. If this condition is not satisfied, an error will be generated and the data packet will be disregarded.
Field of View Top: Float IEEE : degrees valid values: -179.9 to 179.9 Default: N/A Datum: View Reference Point, see Figure 23.	This parameter is used to define the size of the top side of the field of view. This value should always be greater than the Field of View Bottom value. If this condition is not satisfied, an error will be generated and the data packet will be disregarded.
Field of View Bottom: Float IEEE : degrees valid values: -179.9 to 179.9 Default: N/A Datum: View Reference Point, see Figure 23.	This parameter is used to define the size of the bottom side of the field of view. This value should always be less than the Field of View Top value. If this condition is not satisfied, an error will be generated and the data packet will be disregarded.

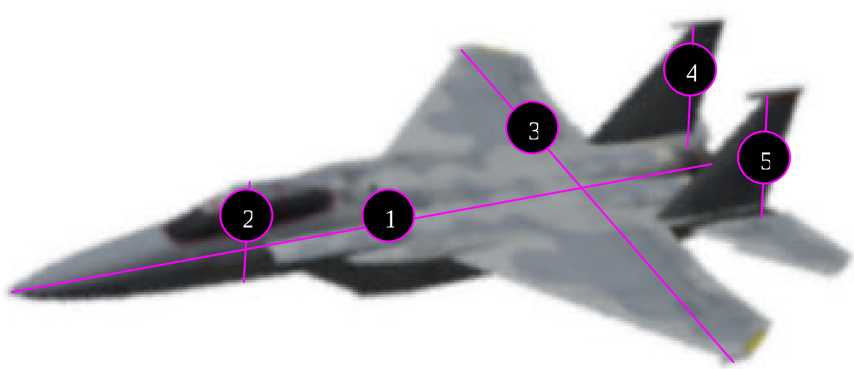
5.15 Collision Detection Segment Definition

The Collision Detection Segment Definition data packet is contained in the Ethernet message sent from the Host to the IG. Because this test does not guard against missed collisions, the Collision Detection Segment test should only be used for slow-moving objects. Since collision tests are conducted at discrete moments in time, it is possible that two entities could pass through one another between successive tests. For instance, the two collision tests might be performed just prior to and just following the moment were the two entities passed through one another, thus causing a missed indication. In cases where less precise contact location information is required the Swept Volume Collision Detection test, described in section 5.16, can be employed. This test does guard against missed collisions of fast moving objects due to the circumstances cited above, but does not return specific contact location and material type information.

The Collision Detection Segment Definition data packet is used to define a segment along which collision testing will be performed. This segment is referenced to the entity specified by the Entity ID. All enabled segments will be tested for intersection against surfaces not associated with the given entity.

The collision mask field is used to include or exclude particular features in collision detection. This provides the ability to include certain features such as terrain, buildings, trees, etc. for consideration for collision detection while excluding other features such as clouds, smoke, dust, etc. When the collision mask allows a certain feature to be considered for collision detection testing, and a collision is detected, a collision response will return a subtype for that feature. For example, if terrain is to be included in collision testing, a collision detection response might indicate that the intersection was with concrete, grass, rock, asphalt, etc.

To match collision segment definitions with responses from the IG, the segment ID parameter is used. The same value will be returned in the segment ID parameter of the corresponding Collision Detection Segment Response data packet as described in section 5.24 to uniquely identify the response. Each segment associated with a given entity must be assigned a unique ID. Thus, a segment ID of zero, one, two, etc., can be assigned to every entity if desired.



The Collision Detection Segment Definition data packet is used to establish segments that will be tested for contact with polygonal surfaces. The segments are described with start and end points within the entity's reference system. In this example, there are five test segments as follows: Segment 1 will test the length of the fuselage, segment 2 will test the vertical height of the fuselage at the cockpit area, segment 3 will test the wingspan, segments 4 and 5 will test the right and left vertical stabilizers, respectively.

Figure 27 - Collision Detection Segment Definition

The contents of the Collision Detection Segment Definition data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 24</td><td colspan="8">Packet Size = 24 bytes</td><td colspan="16">Entity ID</td></tr>
<tr><td>*1</td><td colspan="7">Segment ID</td><td colspan="24">Spare</td></tr>
<tr><td colspan="32">Collision Mask</td></tr>
<tr><td colspan="16">Segment X Start</td><td colspan="16">Segment Y Start</td></tr>
<tr><td colspan="16">Segment Z Start</td><td colspan="16">Segment X End</td></tr>
<tr><td colspan="16">Segment Y End</td><td colspan="16">Segment Z End</td></tr>
</table>

Collision Detection Segment Definition parameter definitions:

Formats and Ranges	Description
Packet ID = 24 : unsigned char : N/A	This parameter identifies this data packet as the Collision Detection Segment Definition data packet. There can be multiple instances of this data packet per frame, but each unique collision detection segment definition should only be specified once per frame. If more than one data packet with the same Entity ID and Segment ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
*1 Enable/Disable : Boolean : N/A valid values: 0 = Disabled 1 = Enabled Default: 0	This parameter indicates whether the defined segment is enabled for collision testing.
Segment ID : 7 bit field : N/A valid values: 0 to 127 Default: N/A	This parameter indicates which segment is being uniquely defined for a given entity. The range of Segment ID 0 through 127 can be reused per entity.
Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter indicates the entity to which this collision detection definition is assigned. An entity must have been previously created with the Entity ID. If this segment is applied to a non-existent entity, an error will be generated and this data packet will be ignored. The entity may be defined earlier in the same Ethernet message or in an earlier Ethernet message.

Collision Mask : 32 Booleans : N/A valid values: See the Collision Mask assignments in the applicable Database, Entity Attribute and IG Functions Definition Document (s). Default: 0x00000000	This parameter indicates which environment features will be included in or excluded from consideration for collision detection testing.
Segment X Start : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the starting point of the collision segment in the X-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.
Segment Y Start : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the starting point of the collision segment in the Y-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.
Segment Z Start : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the starting point of the collision segment in the Z-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.
Segment X End : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the ending point of the collision segment in the X-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.
Segment Y End : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the ending point of the collision segment in the Y-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.

Segment Z End : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the ending point of the collision segment in the Z-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.

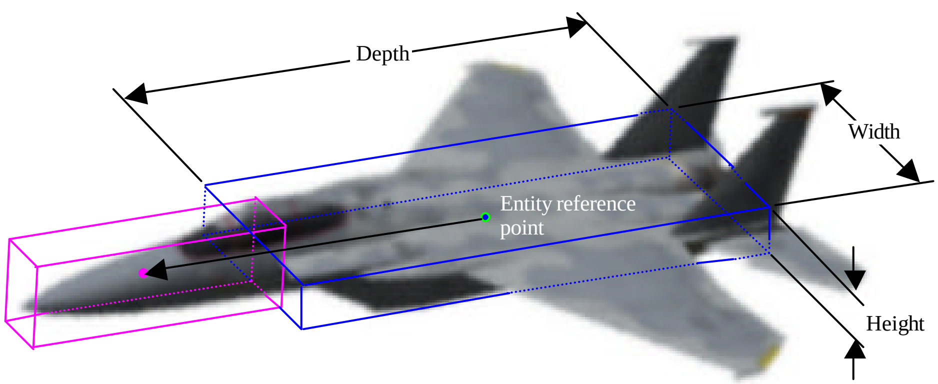
5.16 Swept Volume Collision Detection Definition

The Swept Volume Collision Detection Definition data packet is contained in the Ethernet message sent from the Host to the IG. The Swept Volume Collision Detection is used to guard against missed collisions. Because collision tests are conducted at discrete moments in time, it is possible that two entities could pass through one another between successive tests. For instance, the two collision tests might be performed just prior to and just following the moment were the two entities passed through each other, thus causing a missed indication. In cases were more precise contact location information is required, the Collision Detection Segment test, described in section 5.15, can be employed. This test however, does not guard against missed collisions of fast moving objects due to the circumstances cited above. Swept Volume Collision Detection can be used in the cases were the Host only needs to know if two objects contacted each other. The response to this request does not return a collision location or material type. If more precise collision areas are required, the volume can be reduced in size to accommodate refined collision detections.

Swept Volume Collision Detection is used to define a volume around a centroid within which collision testing will be performed. This volume is referenced to a particular entity specified by the Entity ID. All enabled volumes will be tested for intersection against surfaces not associated with the given entity.

To match collision volume definitions with responses from the IG, the volume ID parameter is used. The same value will be returned in the volume ID parameter of the corresponding Swept Volume Collision Detection Response data packet as described in section 5.27 to uniquely identify the response. Each volume must be assigned a unique ID per entity. Thus, a Volume ID of zero, one, two, etc., can be assigned to every entity if desired.

To avoid missing valid collisions due to frame-to-frame placement of a discrete volume, the volume(s) are swept rearwards as a function of the entity's velocity during collision detection processing by the IG.



The Swept Volume Collision Detection Definition data packet is used to establish the volume to be checked for collisions. The volumes are described by Height, Width, and Depth oriented to the entity's reference system and centered about a common centroid. The centroid of the volume is referenced to the entity's reference point. In this example, the aft volume has a centroid that is coincident with the entity's reference point. The forward volume is positioned using a vector referenced from the entity's centroid. The Volume Centroid X, Y, and Z Offsets define this vector.

Figure 28 - Collision volume definition

The contents of the Swept Volume Collision Detection Definition data packet are shown below.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Packet ID = 25								Packet Size = 20 bytes								Entity ID															
*1	Volume ID							Spare																							
Volume Centroid X Offset																Volume Centroid Y Offset															
Volume Centroid Z Offset																Volume Height															
Volume Width																Volume Depth															

Swept Volume Collision Detection Definition parameter definitions:

Formats and Ranges	Description
Packet ID = 25 : unsigned char : N/A	This parameter identifies this data packet as the Swept Volume Collision Detection Definition data packet. There can be multiple instances of this data packet per frame, but each unique swept volume collision definition should only be specified once per frame. If more than one data packet with the same Entity ID and Volume ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
*1 Enable/Disable : Boolean : N/A valid values: 0 = Disabled 1 = Enabled Default: 0	This parameter indicates whether the defined volume is enabled for collision testing.
Volume ID : 7 bit field : N/A valid values: 0 to 127 Default: N/A	This parameter indicates which volume is being uniquely defined for a given entity. The range of Volume ID 0 through 127 can be reused per entity.
Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter indicates the entity to which this collision detection definition is assigned. An entity must have been previously created with the Entity ID. If this volume is applied to a non-existent entity, an error will be generated and this data packet will be ignored. The entity may be defined earlier in the same Ethernet message or in an earlier Ethernet message.

Volume Centroid X Offset : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the offset of the volume's centroid along the X-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.
Volume Centroid Y Offset : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the offset of the volume's centroid along the Y-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.
Volume Centroid Z Offset : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the offset of the volume's centroid along the Z-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.
Volume Height : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the height of the volume. The height is along the Z-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.
Volume Width : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the width of the volume. The width is along the Y-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.
Volume Depth : scaled distance format (16 bit B6): meters valid values: limits of scaled distance format (16 bit B6) Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the depth of the volume. The depth is along the X-axis with respect to the entity's reference point. See section 2.4.2 for a description of the data format.

5.17 Height Above Terrain Request

The Height Above Terrain Request data packet is contained in the Ethernet message sent from the Host to the IG. It is used to request the height above terrain (HAT) at a specified location. If HAT is required for an entity, the host should insert the positional information for that entity in this data packet. To match up requests from the host with responses from the IG, the HAT ID parameter is used. The same value will be returned in the HAT ID parameter of the corresponding Height Above Terrain Response data packet described in section 5.22 to uniquely identify the response. The HAT ID value should be manipulated in such a way as to not duplicate the value in a reasonable amount of time (e.g., one second). This will prevent similarly identified requests from being lost by the IG.

There is no restriction on the number of HAT requests that can be made in a single frame. The user should be aware however that the response time of the IG might be degraded under conditions that overload the IG HAT computation mechanism.

The IG will only return valid HAT data for points within the bounds of the database being displayed. If a point beyond the database bounds is requested, an invalid answer will be returned in the Height Above Terrain Response data packet. Refer to the applicable Database and Entity Attribute Definition Document(s) for details on the database bounds.

The contents of the Height Above Terrain Request data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 41</td><td colspan="8">Packet Size = 32 bytes</td><td colspan="16">HAT ID</td></tr>
<tr><td colspan="32">Spare</td></tr>
<tr><td colspan="32">HAT Request Altitude (MSW)</td></tr>
<tr><td colspan="32">HAT Request Altitude (LSW)</td></tr>
<tr><td colspan="32">HAT Request Latitude (MSW)</td></tr>
<tr><td colspan="32">HAT Request Latitude (LSW)</td></tr>
<tr><td colspan="32">HAT Request Longitude (MSW)</td></tr>
<tr><td colspan="32">HAT Request Longitude (LSW)</td></tr>
</table>

Height Above Terrain Point Request parameter definitions:

Formats and Ranges	Description
Packet ID = 41 : unsigned char : N/A	This parameter identifies this data packet as a Height Above Terrain Request data packet. There can be multiple instances of this data packet per frame, but each unique HAT request should only be specified once per frame. If more than one data packet with the same HAT ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
HAT ID : unsigned short : N/A valid values: 0 to 65535 Default: 0	This parameter is used to identify the HAT request so that when the answer to the request is returned, it can be identified by the host. This is done via the HAT ID parameter of the Height Above Terrain Response data packet.

HAT Request Altitude : Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A Datum: Mean Sea Level	This parameter specifies the Altitude from which the HAT request was made.
HAT Request Latitude : Double IEEE : degrees valid values: 0 to +90 (north positive) 0 to –90 (south negative) Default: N/A Datum: equator	This parameter specifies the latitudinal position from which the HAT request was made.
HAT Request Longitude : Double IEEE : degrees valid values: 0 to +180 (east positive) 0 to –180 (west negative) Default: N/A Datum: prime meridian	This parameter specifies the longitudinal position from which the HAT request was made.

5.18 Line of Sight Occult Request

The Line of Sight Occult Request data packet is contained in the Ethernet message sent from the Host to the IG. It is used to determine intervisibility or occulting between a source and destination point. The result of the LOS Occult test is contained in the Line of Sight Response data packet, described in section 5.23. To match up requests from the host with responses from the IG, the LOS ID parameter is used. The same value will be returned in the LOS ID parameter of the corresponding Line of Sight data packet to uniquely identify the response. The LOS ID value should be manipulated in such a way as to not duplicate the value in a reasonable amount of time (e.g., one second). This will prevent similarly identified requests from being lost by the IG.

The user should be aware that if the source or destination points of the line of sight occult request emanate from or terminate within an entity with geometry, a valid occult response would be invoked.

There is no restriction on the number of LOS requests that can be sent in a single frame. The user should be aware, however, that the response time of the IG might be degraded under conditions that overload the IG LOS computation mechanism.

Valid LOS responses will only be returned for locations on the current database. Refer to the applicable Database and Entity Attribute Definition Document (s) for details on the extent of the Database.

The contents of the Line of Sight Occult Request data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 42</td><td colspan="8">Packet Size = 56 bytes</td><td colspan="16">LOS ID</td></tr>
<tr><td colspan="32">Spare</td></tr>
<tr><td colspan="32">LOS Request Altitude Source (MSW)</td></tr>
<tr><td colspan="32">LOS Request Altitude Source (LSW)</td></tr>
<tr><td colspan="32">LOS Request Latitude Source (MSW)</td></tr>
<tr><td colspan="32">LOS Request Latitude Source (LSW)</td></tr>
<tr><td colspan="32">LOS Request Longitude Source (MSW)</td></tr>
<tr><td colspan="32">LOS Request Longitude Source (LSW)</td></tr>
<tr><td colspan="32">LOS Request Altitude Destination (MSW)</td></tr>
<tr><td colspan="32">LOS Request Altitude Destination (LSW)</td></tr>
<tr><td colspan="32">LOS Request Latitude Destination (MSW)</td></tr>
<tr><td colspan="32">LOS Request Latitude Destination (LSW)</td></tr>
<tr><td colspan="32">LOS Request Longitude Destination (MSW)</td></tr>
<tr><td colspan="32">LOS Request Longitude Destination (LSW)</td></tr>
</table>

Line of Sight Occult Request parameter definitions:

Formats and Ranges	Description
Packet ID = 42 : unsigned char : N/A	This parameter identifies this data packet as a Line of Sight Occult Request data packet. There can be multiple instances of this data packet per frame, but each unique LOS request should only be specified once per frame. If more than one data packet with the same LOS ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.

LOS ID : unsigned short : N/A valid values: 0 to 65535 Default: 0	This parameter is used to identify the LOS request so that when the response to the request is returned it can be identified by the host. This is done via the LOS ID parameter of the Line of Sight Response data packet. Because the Line-of-Sight Response data packet is used for responding to both the LOS Occult and LOS Range requests, the LOS ID parameters assigned for these queries should be unique between the two request types.
LOS Request Altitude Source : Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A Datum: Mean Sea Level	This parameter specifies the altitude of the source point for the LOS request segment.
LOS Request Latitude Source: Double IEEE : degrees valid values: 0 to +90 (north positive) 0 to –90 (south negative) Default: N/A Datum: prime meridian	This parameter specifies the latitudinal position of the source point for the LOS request segment.
LOS Request Longitude Source: Double IEEE : degrees valid values: 0 to +180 (east positive) 0 to –180 (west negative) Default: N/A Datum: prime meridian	This parameter specifies the longitudinal position of the source point for the LOS request segment.
LOS Request Altitude Destination : Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A Datum: Mean Sea Level	This parameter specifies the altitude of the destination point for the LOS request segment.
LOS Request Latitude Destination: Double IEEE : degrees valid values: 0 to +90 (north positive) 0 to –90 (south negative) Default: N/A Datum: prime meridian	This parameter specifies the latitudinal position of the destination point for the LOS request segment.

LOS Request Longitude Destination: Double IEEE : degrees valid values: 0 to +180 (east positive) 0 to –180 (west negative) Default: N/A Datum: prime meridian	This parameter specifies the longitudinal position of the destination point for the LOS request segment.

5.19 Line of Sight Range Request

The Line of Sight Range Request data packet is contained in the Ethernet message sent from the Host to the IG. It is used to determine the range from a source point to an object within the environment. The Line of Sight test vector emanates from the source position specified in this data packet. A minimum and a maximum range are specified in order to constrain the search, if desired. The result to the LOS Range test is contained in the Line of Sight Response data packet, described in section 5.23. To match up requests from the host with responses from the IG, the LOS ID parameter is used. The same value will be returned in the LOS ID parameter of the corresponding Line of Sight Response data packet to uniquely identify the response. The LOS ID value should be manipulated in such a way as to not duplicate the value in a reasonable amount of time (e.g., one second). This will prevent similarly identified requests from being lost by the IG.

There is no restriction on the number of LOS requests that can be sent in a single frame. The user should be aware, however, that the response time of the IG might be degraded under conditions that overload the IG LOS computation mechanism.

The IG will only return valid LOS data if an intersection is detected within the active LOS segment, that is, between the minimum and maximum distances as specified in this data packet, and the request is located within the bounds of the current database. Refer to the applicable Database and Entity Attribute Definition Document(s) for details on the extent of the database.

The contents of the Line of Sight Range Request data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 43</td><td colspan="8">Packet Size = 48 bytes</td><td colspan="16">LOS ID</td></tr>
<tr><td colspan="32">LOS Request Heading</td></tr>
<tr><td colspan="32">LOS Request Pitch</td></tr>
<tr><td colspan="32">Spare</td></tr>
<tr><td colspan="32">LOS Request Minimum Range</td></tr>
<tr><td colspan="32">LOS Request Maximum Range</td></tr>
<tr><td colspan="32">LOS Request Altitude Source (MSW)</td></tr>
<tr><td colspan="32">LOS Request Altitude Source (LSW)</td></tr>
<tr><td colspan="32">LOS Request Latitude Source (MSW)</td></tr>
<tr><td colspan="32">LOS Request Latitude Source (LSW)</td></tr>
<tr><td colspan="32">LOS Request Longitude Source (MSW)</td></tr>
<tr><td colspan="32">LOS Request Longitude Source (LSW)</td></tr>
</table>

Line of Sight Range Request parameter definitions:

Formats and Ranges	Description
Packet ID = 43 : unsigned char : N/A	This parameter identifies this data packet as a Line of Sight Range Request data packet. There can be multiple instances of this data packet per frame, but each unique LOS request should only be specified once per frame. If more than one data packet with the same LOS ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.

LOS ID : unsigned short : N/A valid values: 0 to 65535 Default: 0	This parameter is used to identify the LOS request so that when the answer to the request is returned it can be identified by the host. This is done via the LOS ID parameter of the Line of Sight Response data packet. Because the Line-of-Sight Response data packet is used for responding to both the LOS Occult and LOS Range requests, the LOS ID parameters assigned for these queries should be unique between the two request types.
LOS Request Heading : Float IEEE : degrees valid values: 0 to +360 clockwise Default: N/A Datum: see Figure 7.	This parameter specifies the heading of the requested LOS.
LOS Request Pitch : Float IEEE : degrees valid values: 0 to +90 up 0 to –90 down Default: N/A Datum: see Figure 7.	This parameter specifies the pitch of the requested LOS.
LOS Request Minimum Distance : Float IEEE : meters valid values: 0.0 to < LOS Request Maximum Distance Default: N/A	This parameter specifies the distance from the source position specified in this data packet to a point along the LOS vector where intersection testing will begin.
LOS Request Maximum Distance : Float IEEE : meters valid values: > LOS Request Minimum Distance to maximum allowed by the data format Default: N/A	This parameter specifies the maximum extent from the source position specified in this data packet to a point along the LOS vector where intersection testing will end.
LOS Request Altitude Source: Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A Datum: Mean Sea Level, see Figure 6.	This parameter specifies the altitude of the source point of the LOS request vector.

LOS Request Latitude Source: Double IEEE : degrees valid values: 0 to +90 (north positive) 0 to –90 (south negative) Default: N/A Datum: equator, see Figure 6.	This parameter specifies the latitudinal position of the source point of the LOS request vector.
LOS Request Longitude Source: Double IEEE : degrees valid values: 0 to +180 (east positive) 0 to –180 (west negative) Default: N/A Datum: prime meridian, see Figure 6.	This parameter specifies the longitudinal position of the source point of the LOS request vector.

5.20 Height of Terrain Request

The Height of Terrain Request data packet is contained in the Ethernet message sent from the Host to the IG. It is used to request the height of terrain (HOT) at a specified location. If HOT is required for an entity, the host should insert the positional information for that entity in this data packet. To match up requests from the host with responses from the IG, the HOT ID parameter is used. The same value will be returned in the HOT ID parameter of the corresponding Height of Terrain Response data packet, described in section 5.26, to uniquely identify the response. The HOT ID value should be manipulated in such a way as to not duplicate the value in a reasonable amount of time (e.g., one second). This will prevent similarly identified requests from being lost by the IG.

There is no restriction on the number of HOT requests that can be sent in a single frame. The user should be aware, however, that the response time of the IG might be degraded under conditions that overload the IG HOT computation mechanism.

The IG will only return valid HOT data for points within the bounds of the database being displayed. If a point beyond the database bounds is requested, an invalid response will be returned in the Height of Terrain Response data packet. Refer to the applicable Database and Entity Attribute Definition Document(s) for details on the extent of the database.

The contents of the Height of Terrain Request data packet are shown below.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Packet ID = 44								Packet Size = 24 bytes								HOT ID															
Spare																															
HOT Request Latitude (MSW)																															
HOT Request Latitude (LSW)																															
HOT Request Longitude (MSW)																															
HOT Request Longitude (LSW)																															

Height of Terrain Point Request parameter definitions:

Formats and Ranges	Description
Packet ID = 44 : unsigned char : N/A	This parameter identifies this data packet as a Height of Terrain Request data packet. There can be multiple instances of this data packet per frame, but each unique HOT request should only be specified once per frame. If more than one data packet with the same HOT ID is received in the same frame, the last one received will be used.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
HOT ID : unsigned short : N/A valid values: 0 to 65535 Default: 0	This parameter is used to identify the HOT request so that when the response to the request is returned it can be identified by the host. This is done via the HOT ID parameter of the Height of Terrain Response data packet.

HOT Request Latitude : Double IEEE : degrees valid values: 0 to +90 (north positive) 0 to –90 (south negative) Default: N/A Datum: equator	This parameter specifies the latitudinal position from which the HOT request is made.
HOT Request Longitude : Double IEEE : degrees valid values: 0 to +180 (east positive) 0 to –180 (west negative) Default: N/A Datum: prime meridian	This parameter specifies the longitudinal position from which the HOT request is made.

5.21 Start of Frame

The Start of Frame data packet is the first packet contained in the Ethernet message sent from the IG to the Host. When the Host receives the Start of Frame data packet, it should respond immediately with the Host to IG Ethernet message containing all mandatory data packets and any other data packets necessary to describe data changes to the IG.

The contents of the Start of Frame data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 101</td><td colspan="8">Packet Size = 16 bytes</td><td colspan="8">CIGI version number = 2</td><td colspan="8">Database Number</td></tr>
<tr><td colspan="8">IG Status Code</td><td colspan="2">*1</td><td colspan="22">Spare</td></tr>
<tr><td colspan="32">IG to Host Frame Counter</td></tr>
<tr><td colspan="32">Timing Value</td></tr>
</table>

Start of Frame parameter definitions:

Formats and Ranges	Description
Packet ID = 101 : unsigned char : N/A	This parameter identifies this data packet as the Start of Frame data packet. There will be only one instance of this data packet per frame.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
CIGI version number : unsigned char : N/A valid values: 0 – 255 Default: N/A	This parameter indicates the version of the CIGI interface that is currently running on the image generator. The host can use this number to determine concurrency.
Database Number : signed char : N/A valid values: -128 The requested database is not available -99 to –1 The requested database is being loaded 0 No database is loaded or being loaded +1 to +99 The designated database is loaded See the Database Number table in the applicable Database and Entity Attribute Definition Document(s). Default = N/A	This parameter indicates load status of the requested database. The IG will return a value of –128 while the Host is requesting a database that is not available. See the Database Number parameter of the IG Control data packet, described in section 5.3 for a further discussion on database loading theory.
IG Status Code : unsigned char : N/A valid values: 0 – normal operation 1 – 255 See assignments in Table 5. Default: 0	This parameter indicates the operational status of the IG.

*1 Current IG Mode : 2 bit field : N/A valid values: 0 = standby (reset) 1 = operate 2 = debug 3 = off-line maintenance Default: 0	This parameter identifies to the Host the current operating mode of the IG. When the IG receives a request for a mode change via the IG Mode Change Request parameter of the IG Control data packet it will return the corresponding mode in this parameter once the new mode has been accomplished. **Standby (reset):** This is the mode to which the IG will be initialized during start up. In this mode, the IG will initialize the mission scenario to begin a new mission. All entities that were instantiated during a previous mission will be removed from the display. While in this mode, the IG will only send the Start of Frame data packet to the Host and will ignore Host inputs except for the IG Mode Change parameter of the IG Control data packet. Also during this mode, the IG may be put into maintenance mode via a graphical user interface provided on the IG. **Operate:** In this mode, the IG will accept all data packet types destined for the IG. The IG will also return all data packet types appropriate for real-time operation. While in this mode, the IG will report errors to the Host via the IG Status Code parameter of this data packet. Because of the real-time nature of this mode, status codes are provided only for informational purposes. If further investigation is necessary, the debug mode should be used for non-real-time operation. **Debug:** In this mode, the IG will accept all data packet types destined for the IG. The IG will also return all data packet types appropriate for real-time operation. This mode can be used as a diagnostic tool while integrating or troubleshooting the Host and IG interface. Because of error logging that takes place during this mode the IG may not always operate in a real-time fashion. While in this mode, the IG will report errors to the Host via the Image Generator Response Message data packet. Also, a log of all status messages is kept on the IG. This log can be viewed using a graphical user interface provided on the IG. Typically, status logs are regenerated each time the CIGI is started. However, log archives can be saved for later diagnosis via a graphical user interface provided on the IG. **Off-line maintenance:** The off-line maintenance mode is activated from the IG. While the IG is in off-line maintenance mode, the Host cannot change the IG mode. The IG must initiate a mode change to standby (reset) before the Host can command a mode change. While in the off-line maintenance mode the IG will only send the Start of Frame data packet to the Host and will ignore all Host commands. When the IG transitions from a mode where it has ignored Host commands, The Host must initialize the IG to the proper mission scenario start conditions.

IG to Host Frame Counter : unsigned integer : N/A valid values: 0 to 4,294,967,295 Default: N/A	This parameter contains a number representing a particular frame. This number is incremented each frame by the IG. It is intended to be used in conjunction with the Host to IG Frame Counter parameter in the IG Control data packet to assist in correlating IG and Host frames. When this parameter reaches its maximum value, it will roll back to zero.
Timing Value : Float IEEE: usec valid values: 0 to 86,399,999,999.99 Default: 0	The use of this parameter is optional for synchronous operation, but required for asynchronous operation. It contains a timing value that is used to time-tag the Ethernet message during asynchronous operation. When asynchronous operation is used, the synchronous timing scheme described in section 2.1.1 is superceded. In order to preserve floating-point accuracy, this timing value is limited to a 24-hour simulation period. At the end of 24 hours, the counter will reset to zero.

5.22 Height Above Terrain Response

The Height Above Terrain Response data packet is contained in the Ethernet message sent from the IG to the Host. It is used to respond to a Height Above Terrain Request. To match up requests from the host with responses from the IG, the HAT ID parameter is used. The value of this field is the same as that used in the HAT ID in the Height Above Terrain Request data packet used to make the original request. See the Height Above Terrain Request data packet narrative in section 5.17 for more information on this correlation scheme.

The contents of the Height Above Terrain Response data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 102</td><td colspan="8">Packet Size = 24 bytes</td><td colspan="16">HAT ID</td></tr>
<tr><td>*1</td><td colspan="31">Spare</td></tr>
<tr><td colspan="32">Spare</td></tr>
<tr><td colspan="32">Material Type</td></tr>
<tr><td colspan="32">HAT Response Altitude (MSW)</td></tr>
<tr><td colspan="32">HAT Response Altitude (LSW)</td></tr>
</table>

Height Above Terrain Response parameter definitions:

Formats and Ranges	Description
Packet ID = 102 : unsigned char : N/A	This parameter identifies this data packet as a Height Above Terrain Response data packet. There can be multiple instances of this data packet per frame. Each instance is uniquely identified by the HAT ID parameter.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
HAT ID : unsigned short : N/A valid values: 0 to 65535 Default: 0	This parameter identifies the HAT response corresponding to the associated HAT request. This parameter is provided to allow the host to match this response with the issued request.
*1 Response Validity : Boolean : N/A valid values: 0 = invalid 1 = valid Default: 0	This parameter is used to indicate whether the response is valid or invalid. A response is invalid if the test point was located beyond the bounds of the database.
Material Type : integer : N/A valid values: See the Material type assignments in the applicable Database and Entity Attribute Definition Document (s). Default: N/A	This parameter specifies the Material Type of the object intersected by the HAT test vector.

HAT Response Altitude: Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A	This parameter represents the altitude above or below the terrain for the position requested in the Height Above Terrain Request data packet.

5.23 Line of Sight Response

The Line of Sight Response data packet is contained in the Ethernet message sent from the IG to the Host. It is used to respond to a Line of Sight Request. To match up requests from the host with responses from the IG the LOS ID parameter is used. The value of this field is the same as that used in the LOS ID parameter in either the Line of Sight Occult Request or the Line of Sight Range Request data packet used to make the original request. See the Line of Sight Occult Request or the Line of Sight Range Request data packet's narrative for more information on this correlation scheme in sections 5.18 and 5.19.

The contents of the Line of Sight Response data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 103</td><td colspan="8">Packet Size = 40 bytes</td><td colspan="16">LOS ID</td></tr>
<tr><td>*1</td><td>*2</td><td colspan="30">Spare</td></tr>
<tr><td colspan="32">Material Type</td></tr>
<tr><td colspan="32">LOS Range Response</td></tr>
<tr><td colspan="32">LOS Intersection Altitude (MSW)</td></tr>
<tr><td colspan="32">LOS Intersection Altitude (LSW)</td></tr>
<tr><td colspan="32">LOS Intersection Latitude (MSW)</td></tr>
<tr><td colspan="32">LOS Intersection Latitude (LSW)</td></tr>
<tr><td colspan="32">LOS Intersection Longitude (MSW)</td></tr>
<tr><td colspan="32">LOS Intersection Longitude (LSW)</td></tr>
</table>

Line of Sight Response parameter definitions:

Formats and Ranges	Description
Packet ID = 103 : unsigned char : N/A	This parameter identifies this data packet as a Line of Sight Response data packet. There can be multiple instances of this data packet per frame. Each instance is uniquely identified by the LOS ID parameter.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
LOS ID : unsigned short : N/A valid values: 0 to 65535 Default: 0	This parameter identifies the LOS response corresponding to the associated LOS request. This parameter is provided to allow the host to match this response with the issued request.
*1 Response Validity : Boolean : N/A valid values: 0 = invalid 1 = valid Default: 0	This parameter is used to indicate whether the response is valid or invalid.

*2 LOS Occult Response : Boolean : N/A valid values: 0 = occulted 1 = visible Default: 0	This parameter is used to respond to the LOS Occult Request data packet. It indicates whether the destination point is visible from the source point. This field is not applicable to the LOS Range Request data packet.
Material Type : integer : N/A valid values: See the Material type assignments in the applicable Database and Entity Attribute Definition Document (s). Default: N/A	This parameter specifies the Material Type of the object intersected by the LOS test vector.
LOS Range Response: Float IEEE : meters valid values: -1 = beyond requested LOS Maximum Range 0 to maximum allowed by the data format Default: N/A	This parameter is used to respond to the Line of Sight Range Request data packet. If an object is not intersected within the active LOS segment, that is, between the minimum and maximum distances as specified in the Line of Sight Range Request data packet, the Response Validity will indicate a valid response and the LOS Range Response will contain a negative one (-1). Otherwise, the LOS Range Response will contain the range of the intersection from the source position specified in the Line of Sight Range Request data packet.
LOS Intersection Altitude : Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A Datum: Mean Sea Level, see Figure 6.	This parameter specifies the altitude of the point of intersection of the LOS request vector with an object. If the LOS Range Response in this data packet contains a negative one (-1) this altitude value should be ignored.
LOS Intersection Latitude : Double IEEE : degrees valid values: 0 to +90 (north positive) 0 to –90 (south negative) Default: N/A Datum: equator, see Figure 6.	This parameter specifies the latitudinal position of the intersection point of the LOS request vector with an object. If the LOS Range Response in this data packet contains a negative one (-1) this latitude value should be ignored.
LOS Intersection Longitude : Double IEEE : degrees valid values: 0 to +180 (east positive) 0 to –180 (west negative) Default: N/A Datum: prime meridian, see Figure 6.	This parameter specifies the longitudinal position of the intersection point of the LOS request vector with an object. If the LOS Range Response in this data packet contains a negative one (-1) this longitude value should be ignored.

5.24 Collision Detection Segment Response

The Collision Detection Segment Response data packet is contained in the Ethernet message sent from the IG to the Host. There can be up to 127 impact locations specified on an entity. These are uniquely identified using the segment ID parameter.

The Collision Detection Segment Response data packet will be returned if the following three conditions are met: a collision occurs, the segment enable parameter in the Collision Detection Segment Definition data packet is enabled, and the collision detection request switch in the Entity Control data packet is enabled. Explanations of these parameters can be found in their respective sections in this document.

If two defined entities contact each other, then two Collision Detection Segment Response data packets would be generated, one for each entity.

The contents of the Collision Detection Segment Response data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 104</td><td colspan="8">Packet Size = 24 bytes</td><td colspan="16">Entity ID</td></tr>
<tr><td colspan="7">Segment ID</td><td>*1</td><td colspan="8">Spare</td><td colspan="16">Contacted Entity ID</td></tr>
<tr><td colspan="32">Material Type</td></tr>
<tr><td colspan="32">Collision Point X</td></tr>
<tr><td colspan="32">Collision Point Y</td></tr>
<tr><td colspan="32">Collision Point Z</td></tr>
</table>

Collision Detection Segment Response parameter definitions:

Formats and Ranges	Description
Packet ID = 104 : unsigned char : N/A	This parameter identifies this data packet as a Collision Detection Segment Response data packet. There can be multiple instances of this data packet per frame. Each instance is uniquely identified by the Segment ID and Entity ID parameters.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter indicates which entity experienced a collision.
Segment ID : 7 bit field : N/A valid values: 0 to 127 Default: N/A	This parameter identifies the collision segment corresponding to the associated Collision Detection Segment request. This parameter is provided to allow the host to match this response with the issued request.

*1 Entity / Non-Entity Contact : Boolean : N/A valid values: 0 = contact with non entity surface 1 = contact with a defined entity Default: N/A	The parameter indicates whether another entity was contacted during this collision. If this parameter indicates contact with non-entity surface (0), then the Contacted Entity ID field is ignored. If this parameter indicates contact with a defined entity (1), then the Contacted Entity ID field shall contain the Entity ID of the entity that was contacted.
Contacted Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter indicates which entity was contacted during the collision. If the Entity / Non-Entity Contact parameter of this data packet indicates contact with non-entity surface (0), then this field is ignored. If the parameter indicates contact with a defined entity (1), then this field shall contain the Entity ID of the entity that was contacted.
Material Type: integer : N/A valid values: See the Material type assignments in the applicable Database and Entity Attribute Definition Document (s). Default: N/A	This parameter specifies the Material Type of the surface that this collision test segment contacted.
Collision Point X: Float IEEE: meters valid values: Minimum to maximum allowed by the data format limited to the extent of the segment Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the X component of a vector, which lies along the defined segment where the segment intersected a surface. When fully defined by the X, Y, and Z, collision vector components the collision point will lie upon the originally defined segment. This vector will originate from the source location of the Collision Detection Segment.
Collision Point Y: Float IEEE: meters valid values: Minimum to maximum allowed by the data format limited to the extent of the segment Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the Y component of a vector, which lies along the defined segment where the segment intersected a surface. When fully defined by the X, Y, and Z, collision vector components the collision point will lie upon the originally defined segment. This vector will originate from the source location of the Collision Detection Segment.
Collision Point Z: Float IEEE: meters valid values: Minimum to maximum allowed by the data format limited to the extent of the segment Default: N/A Datum: Entity coordinate system, see Figure 9.	This parameter specifies the Z component of a vector, which lies along the defined segment where the segment intersected a surface. When fully defined by the X, Y, and Z, collision vector components the collision point will lie upon the originally defined segment. This vector will originate from the source location of the Collision Detection Segment.

5.25 Sensor Response

The Sensor Response data packet is contained in the Ethernet message sent from the IG to the Host. It is used to report aspects of the Sensor tracker to the Host. For every frame that the sensor specified by the Sensor ID is active, this packet must be returned to the host. Otherwise information will be lost to the Host.

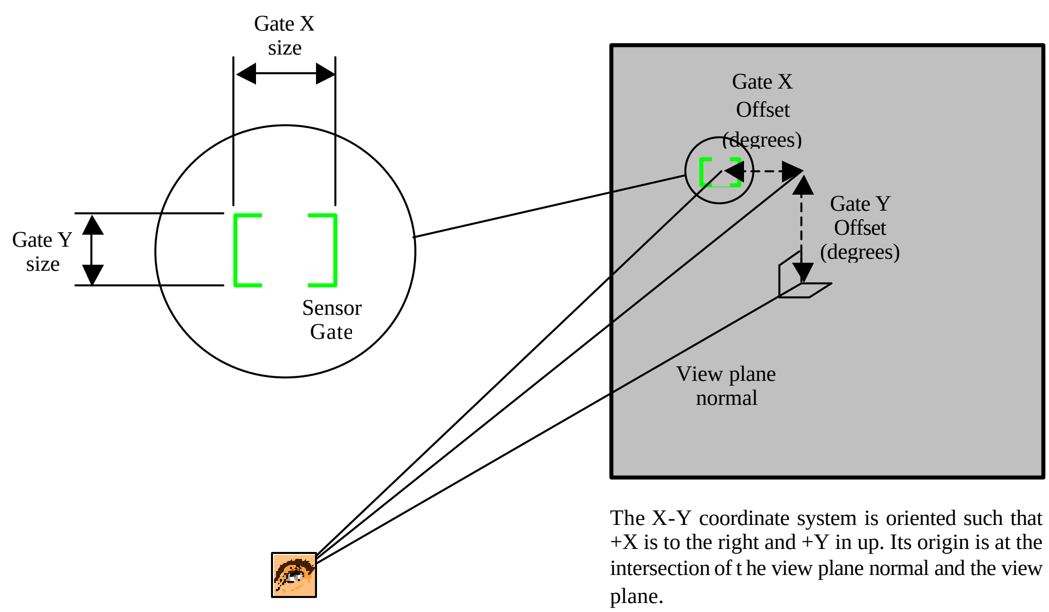


Figure 29 - Gate Offset and Size

The contents of the Sensor Response data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 105</td><td colspan="8">Packet Size = 12 bytes</td><td colspan="5">View ID</td><td colspan="2">*1</td><td colspan="1">S</td><td colspan="8">Sensor ID</td></tr>
<tr><td colspan="16">Target X Offset</td><td colspan="16">Target Y Offset</td></tr>
<tr><td colspan="16">Gate X Size</td><td colspan="16">Gate Y Size</td></tr>
</table>

Sensor Response parameter definitions:

Formats and Ranges	Description
Packet ID = 105 : unsigned char : N/A	This parameter identifies this data packet as Sensor Response data packet. There will be one of these data packets for each active Sensor view.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
View ID: 5 bit field : N/A valid value: 0 – 31 Default N/A	This parameter indicates the Sensor view to which this data packet is applicable.

*1 Sensor Status : 2 bit field : N/A valid values: 0 = Searching for target 1 = Tracking 2 = Impending Breaklock 3 = Breaklock Default = 0	This parameter indicates the current Sensor mode.
Sensor ID : unsigned char : N/A valid values: 0 to 255 Default: 0	This parameter identifies the Sensor response corresponding to the associated Sensor Control data packet. This parameter is provided to allow the host to match this response with the issued Sensor Control data packet and its associated data packets.
Gate X Offset: angle format (16 bit): degrees valid values: Positive and negative values bounded by the specified view Default = 0 Datum: see Figure 29.	This parameter specifies the target's horizontal offset from the view plane normal.
Gate Y Offset: angle format (16 bit): degrees valid values: Positive and negative values bounded by the specified view Default = 0 Datum: see Figure 29.	This parameter specifies the target's vertical offset from the view plane normal.
Gate X Size: unsigned short : See note to right valid values: 0 to maximum required Default = 0 Datum: see Figure 29.	This parameter specifies the target size in the X direction (horizontal) in pixels. Note – The units can be either pixels or lines depending on the view rotation.
Gate Y Size: unsigned short : See note to right valid values: 0 to maximum required Default = 0 Datum: see Figure 29.	This parameter specifies the target size in the Y direction (vertical) in pixels. Note – The units can be either pixels or lines depending on the view rotation.

5.26 Height of Terrain Response

The Height of Terrain Response data packet is contained in the Ethernet message sent from the IG to the Host. It is used to respond to a Height of Terrain Request. To match up requests from the host with responses from the IG the HOT ID parameter is used. The value is the same as that used in the HOT ID in the Height of Terrain Request data packet used to make the original request. See the Height of Terrain Request data packet narrative in section 5.20 for more information on this correlation scheme.

The contents of the Height of Terrain Response data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 106</td><td colspan="8">Packet Size = 24 bytes</td><td colspan="16">HOT ID</td></tr>
<tr><td>*1</td><td colspan="31">Spare</td></tr>
<tr><td colspan="32">Spare</td></tr>
<tr><td colspan="32">Material Type</td></tr>
<tr><td colspan="32">HOT Response Altitude (MSW)</td></tr>
<tr><td colspan="32">HOT Response Altitude (LSW)</td></tr>
</table>

Height of Terrain Response parameter definitions:

Formats and Ranges	Description
Packet ID = 106 : unsigned char : N/A	This parameter identifies this data packet as a Height of Terrain Response data packet. There can be multiple instances of this data packet per frame. Each instance is uniquely identified by the HOT ID parameter.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
HOT ID : unsigned short : N/A valid values: 0 to 65535 Default: 0	This parameter identifies the HOT response corresponding to the associated HOT request. This parameter is provided to allow the host to match this response with the issued request.
*1 Response Validity : Boolean : N/A valid values: 0 = invalid 1 = valid Default: 0	This parameter is used to indicate whether the response is valid or invalid. A response is invalid if the test point was located beyond the bounds of the database.
Material Type : integer : N/A valid values: See the Material type assignments in the applicable Database and Entity Attribute Definition Document (s). Default: N/A	This parameter specifies the Material Type of the object intersected by the HOT test segment.

HOT Response Altitude: Double IEEE : meters valid values: Minimum to maximum allowed by the data format Default: N/A Datum: Mean Sea Level	This parameter represents the altitude of the terrain for the position requested in the Height of Terrain Request data packet.

5.27 Swept Volume Collision Detection Response

The Swept Volume Collision Detection Response data packet is contained in the Ethernet message sent from the IG to the Host. The swept Volume Collision detection mechanism can be used when it is only important for the Host to know when two objects have made contact. This response does not contain contact location or material type information. There can be up to 127 impact volumes specified on an entity. These are uniquely identified using the volume ID parameter. This data packet will be returned if the following three conditions are met: a collision occurs, the enable parameter in the Swept Volume Collision Detection Definition data packet is enabled, and the collision detection request switch in the Entity Control data packet is enabled. Explanations of these parameters can be found in their respective sections of this document.

If two defined entities contact each other, then two Swept Volume Collision Detection Response data packets would be generated, one for each entity.

The contents of the Swept Volume Collision Detection Response data packet are shown below.

31–25	24	23–16	15–0
Packet ID = 107		Packet Size = 8 bytes	Entity ID
Volume ID	*1	Spare	Contacted Entity ID

Swept Volume Collision Detection Response parameter definitions:

Formats and Ranges	Description
Packet ID = 107 : unsigned char : N/A	This parameter identifies this data packet as a Swept Volume Collision Detection Response data packet. There can be multiple instances of this data packet per frame. The Volume ID and Entity ID parameters will uniquely identify each instance.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.
Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter indicates which entity experienced a collision.
Volume ID : 7 bit field : N/A valid values: 0 to 127 Default: N/A	This parameter identifies the collision volume corresponding to the associated Swept Volume Collision Detection Request. This parameter is provided to allow the host to match this response with the issued request.

*1 Entity / non-entity contact : Boolean : N/A valid values: 0 = contact with non entity surface 1 = contact with a defined entity Default: N/A	The parameter indicates whether another entity was contacted during this collision. If this parameter indicates contact with non-entity surface (0), then the Contacted Entity ID field is ignored. If this parameter indicates contact with a defined entity (1), then the Contacted Entity ID field shall contain the Entity ID of the entity that was contacted.
Contacted Entity ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter indicates which entity was contacted with during the collision. If the Entity / Non-Entity Contact switch of this data packet indicates contact with non-entity surface (0), then this field is ignored. If this parameter indicates contact with a defined entity (1), then this field shall contain the Entity ID of the entity that was contacted.

5.28 Image Generator Response Message

The Image Generator Response Message data packet is contained in the Ethernet message sent from the IG to the Host. It is used to notify the Host of a condition in the IG that may be of value to the user. Messages are only sent to the Host from the IG while the IG Mode Change Request parameter of the IG Control data packet is set to “Debug.” See section 5.3 for details.

This data packet is expandable depending on the number of characters contained in the message. The programmer should place the total number of bytes, including the four bytes of message header in the Packet Size field so that the Host can accept the packet properly. The character string should also be Null-terminated as shown by the last byte in the diagram below. If a terminating NULL falls in one of the first three bytes of a word the remainder of that word should be padded with NULLs. Note that if a string ends on a word boundary, an additional word is needed to contain the terminating NULL and padding.

The string should be limited to 96 characters (bytes), bringing the maximum packet length to 100 bytes.

In order to reduce the load on Ethernet messages and the IG computational frame, only Image Generator Response Message data packets that contain data changes shall be included in the Ethernet message.

The contents of the Image Generator Response Message data packet are shown below.

31–24	23–16	15–8	7–0
Packet ID = 108	Packet Size = 4 bytes + characters	Message ID	
Character 1	Character 2	Character 3	Character 4
•	•	•	•
Character n - 2	Character n - 1	Character n	Null

Image Generator Response Message parameter definitions:

Formats and Ranges	Description
Packet ID = 108 : unsigned char : N/A	This parameter identifies this data packet as a Image Generator Response Message data packet. There can be multiple instances of this data packet per frame. Its Message ID will uniquely identify each instance.
Packet Size : unsigned char : N/A valid values: Not to exceed 100 bytes	This parameter indicates the number of bytes in this data packet.
Message ID : unsigned short : N/A valid values: 0 to 65535 Default: N/A	This parameter uniquely identifies an instance of an Image Generator Response Message.

Character : char : alphanumeric character valid values: ASCII character set Default: N/A	This is the ASCII value of an alphanumeric character. The last charter in the message should be occupied by a Null to terminate the character string. Byte locations following the Null value should be padded with NULL.

5.29 User-Definable Data Packet

A User-Definable data packet can be contained in the Ethernet message sent from either the Host to the IG or the IG to the Host. The User-Definable data packet is provided as a custom packet mechanism to allow transmission of data not specifically supported by an existing CIGI packet.

When user-defined data packets are introduced into a particular CIGI application, they should adhere to the standard data packet format in order to maintain continuity across data packets. Standard data packet format includes the Packet ID in the first byte and the Packet Size in the second byte of the data packet. It is recommended that if data such as Entity ID and View ID are used in the data packet, these values be positioned and sized in similar fashion as other instances of like information within the interface. The Component Control data packet described in section 5.5 is a good example of this data formatting.

The size of each user-defined data packet depnds on the amount of data contained in the packet. The programmer should place the total number of bytes, including the two bytes of header, in the Packet Size field of this packet so that the Host can accept the packet properly.

It must be understood that when a user-defined data packet is introduced into a particular implementation of the CIGI, that implementation will no longer conform to the baseline packet definitions within CIGI and hence may not be acceptable to the general CIGI user community.

It is also important to note that an attempt has been made to contain information within a CIGI data packet that is consistent with what might be expected in a real-world object. Even though it may be convenient under some circumstances to place information into a data packet that is not pertinent or realistic to the data packet's intent, this is not recommended. The intent of the CIGI in this respect is to provide an object-oriented interface that can be used for an entire spectrum of applications. When unrealistic information or functionality is contained in the interface, it makes it more difficult to apply in an abstract sense.

To remain consistent with the CIGI, any user-defined fields should use big-endian byte ordering.

The contents of the User Definable data packet are shown below.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Packet ID = 236 to 255</td><td colspan="8">Packet Size = 4 bytes + 4*n fields</td><td colspan="16">User Definable Data</td></tr>
<tr><td colspan="32">User-Definable Data</td></tr>
<tr><td colspan="32">•
•
•</td></tr>
<tr><td colspan="32">User-Definable Data (n)</td></tr>
</table>

User Definable parameter definitions:

Formats and Ranges	Description
Packet ID = 236 to 255 : unsigned char : N/A	This parameter identifies this data packet as a User-Definable data packet. There may or may not be multiple instances of this data packet per frame depending on its intended use.
Packet Size : unsigned char : N/A	This parameter indicates the number of bytes in this data packet.

User Definable Data : formatted as needed valid values: user defined Default: N/A	The remainder of the fields in this data packet are user-defined. The data may be made up of Booleans, chars, integer shorts, integer longs, floats, and doubles as needed.

6. Status Messages

Following is a table of suggested status message definitions that may be reported from the CIGI to the Host in the IG Status parameter of the Start of Frame data packet described in section 5.21. Because this parameter can only hold one IG Status Number at a time if two or more messages are generated in the same frame an IG Status Number of 24 will be generated to indicate that multiple errors exist.

Table 5 – CIGI status messages

IG Status Number	Error Description
0	Normal Operation
1	The IG Control data packet was not the first data packet detected in the Host-to-CIGI message.
2	Invalid Entity Type in Entity Control data packet.
3	An Entity Control data packet contains an inactive Parent Entity ID.
4	A Component Control data packet contains an inactive Entity ID.
5	A Component Control data packet contains an invalid Component ID.
6	A Component Control data packet contains an invalid Component State.
7	A Component Control data packet contains an invalid Component Value.
8	An Articulated Part Control data packet contains an invalid Articulated Part ID.
9	An Articulated Part Control data packet contains an inactive Entity ID.
10	A Rate Control data packet contains an inactive Entity ID.
11	A Rate Control data packet contains an invalid Articulated Part ID.
12	An Environment Control data packet contains values are outside the specified range.
13	A View Control data packet contains an undefined view group.
14	A View Control data packet contains an undefined view.
15	A Sensor Control data packet contains an invalid value for Scene/Target track mode on/off.
16	A Trajectory Definition data packet contains an inactive Entity ID.
17	A Special Effects Definition data packet contains an inactive Entity ID.
18	A View Definition data packet contains an undefined view.
19	A View Definition data packet specified the Field of View Left value greater than the right.
20	A View Definition data packet specified the Field of View Right value less than the left.
21	A View Definition data packet specified the Field of View Top value less than the bottom.
22	A View Definition data packet specified the Field of View Bottom value greater than the top.
23	A Collision Detection Definition data packet contains an inactive Entity ID.
24	More that one status error has been generated in one frame.

7. Acronyms

AAR	Air-to-Air Refueling
CG	Center of Gravity
HAT	Height Above Terrain
HPR	Heading, Pitch, Roll
Hz	Hertz
I/O	Input / Output
ICD	Interface Control Document
ID	Identification
IFF	Identify Friend or Foe
IG	Image Generator
IP	Internet Protocol
IEEE	Institute of Electrical and Electronic Engineers
LOS	Line-of-Sight
LSW	Least Significant Word
ms	Millisecond(s)
MSL	Mean Sea Level
MSW	Most Significant Word
N/A	Not Applicable
OFS	Operational Flight Simulator
OTW	Out The Window
SOF	Start of Frame
RTM	Real Time Monitor
TBD	To Be Determined
TCP	Transfer Control Protocol
TPS	Target Projection System
UDP	User Datagram Protocol
WRT	With Respect To