The Navigation Economic Technologies Program

October 20, 2005

NETS
navigation · economics · technologies

HSAM

AN INTERACTIVE, IMMERSIVE ANIMATION OF DEEP-DRAFT MARITIME TRAFFIC SIMULATIONS

US Army Corps of Engineers®

IWR Report 05-NETS-P-06

Navigation Economic Technologies

The purpose of the Navigation Economic Technologies (NETS) research program is to develop a standardized and defensible suite of economic tools for navigation improvement evaluation. NETS addresses specific navigation economic evaluation and modeling issues that have been raised inside and outside the Corps and is responsive to our commitment to develop and use peer-reviewed tools, techniques and procedures as expressed in the Civil Works strategic plan. The new tools and techniques developed by the NETS research program are to be based on 1) reviews of economic theory, 2) current practices across the Corps (and elsewhere), 3) data needs and availability, and 4) peer recommendations.

The NETS research program has two focus points: expansion of the body of knowledge about the economics underlying uses of the waterways; and creation of a toolbox of practical planning models, methods and techniques that can be applied to a variety of situations.

Expanding the Body of Knowledge

NETS will strive to expand the available body of knowledge about core concepts underlying navigation economic models through the development of scientific papers and reports. For example, NETS will explore how the economic benefits of building new navigation projects are affected by market conditions and/or changes in shipper behaviors, particularly decisions to switch to non-water modes of transportation. The results of such studies will help Corps planners determine whether their economic models are based on realistic premises.

Creating a Planning Toolbox

The NETS research program will develop a series of practical tools and techniques that can be used by Corps navigation planners. The centerpiece of these efforts will be a suite of simulation models. The suite will include models for forecasting international and domestic traffic flows and how they may change with project improvements. It will also include a regional traffic routing model that identifies the annual quantities from each origin and the routes used to satisfy the forecasted demand at each destination. Finally, the suite will include a microscopic event model that generates and routes individual shipments through a system from commodity origin to destination to evaluate non-structural and reliability based measures.

This suite of economic models will enable Corps planners across the country to develop consistent, accurate, useful and comparable analyses regarding the likely impact of changes to navigation infrastructure or systems.

NETS research has been accomplished by a team of academicians, contractors and Corps employees in consultation with other Federal agencies, including the US DOT and USDA; and the Corps Planning Centers of Expertise for Inland and Deep Draft Navigation.

For further information on the NETS research program, please contact:

Mr. Keith Hofseth
NETS Technical Director
703-428-6468

Dr. John Singley
NETS Program Manager
703-428-6219

U.S. Department of the Army
Corps of Engineers
Institute for Water Resources
Casey Building, 7701 Telegraph Road
Alexandria, VA 22315-3868

The NETS program was overseen by Mr. Robert Pietrowsky, Director of the Institute for Water Resources.

October 20, 2005

navigation · economics · technologies

HSAM

AN INTERACTIVE, IMMERSIVE ANIMATION OF DEEP-DRAFT MARITIME TRAFFIC SIMULATIONS

Prepared by:

Cory Rogers
CDM

William K. Woelbeling
CDM

Richard Males
RMM Technical Services, Inc.

Keith Hofseth
USACE

Shana Heisey
USACE

For the:

Institute for Water Resources
U.S. Army Corps of Engineers
Alexandria, Virginia

IWR Report 05-NETS-P-06

www.corpsnets.us

HSAM: AN INTERACTIVE, IMMERSIVE ANIMATION OF DEEP-DRAFT MARITIME TRAFFIC SIMULATIONS

Cory Rogers and William K. Woelbeling
CDM
PO Box 1316, Carbondale, Illinois 62903
Email: rogerscm@cdm.com

Richard Males
RMM Technical Services, Inc.
3319 Eastside Avenue, Cincinnati, Ohio 45208
Email: males@iac.net

Keith Hofseth and Shana Heisey
US Army Corps of Engineers Institute for Water Resources
7701 Telegraph Road, Alexandria, Virginia 22315
Email: Shana.a.Heisey@usace.army.mil

KEYWORDS: Animation.

ABSTRACT

HSAM, the HarborSym Animation Module, is an interactive, immersive visualization of deep draft maritime traffic simulations developed by the U.S. Army Corps of Engineers. HSAM is designed as an extension to the rendering framework known as the Object-oriented Graphical Rendering Engine (OGRE). It is completely data driven, which allows the operator to configure the animation to visually represent any harbor configuration and dynamically update the animation to reflect simulation modifications without programmatic alterations. Users are able to select a graphical representation of the harbor as the surface on which HSAM animates vessel movements. Visually distinctive 3D models, or avatars, are selected to represent each vessel class, as are the textures applied to the avatars. Once the representations are selected, HSAM is directed to make vessel movements and environmental alterations though a time sequenced queue of event commands. The program is provided access to the internal databases of the simulation model, HarborSym, allowing advanced querying of individual vessel details, calls, and cargo transactions. The interactive nature of HSAM provides a rich environment for simulation analysis while its data driven structure provides a flexible and cost effective solution for planning level analyses.

INTRODUCTION

The cost-benefit analysis of policy and infrastructure modifications in deep-draft harbors is complex not only in terms of the analysis itself but also in the determination of validity, feature isolation, and presentation of results to a non-technical, authoritative audience. A tool for the analysis and presentation of deep-draft traffic simulation data must encompass several basic features and tenets. The solution must assist the analyst in determining the validity of the simulation data. It must be cost effective relative to experimental and repetitive changes to the simulation data. The solution must be capable of enhancing the perception of specific facets of the data set. It must enhance the cognitive ability of the analyst to perceive complex interactions in the simulation data, and finally, the solution should not be tightly coupled to a specific simulation package so as to avert premature functional obsolescence. E.R. Tufte, in his book *The Visual Display of Quantitative Information*, posits that two characteristics of excellence in statistical graphics are that the graphics should *reveal* data and that the graphics should "induce the viewer to think about the substance rather than about methodology, graphic design, the technology of graphic production, or something else."[1] The HarborSym Animation Module (HSAM), developed by the U.S. Army Corps of Engineers, Institute for Water Resources, provides a solution that meets these requirements.

The interactive visual depiction of complex multi vessel movement streams over linear time in HSAM allows the analyst to quickly and intuitively identify anomalies in simulation data. HSAM is completely data-driven, provides a zero cost asset for alterations to the simulation

[1] E.R. Tufte, "The Visual Display of Quantitative Information", Graphics Press, Chesire, CT, 1983.

data streams and facilitates experimentation in the simulation itself. HSAM facilitates visual feature extraction during animation through avatar and texture selection during configuration. Careful selection of texture schemes is extremely effective in focused analysis and presentation. The foundational principle of HSAM is to enhance the cognitive abilities of the analyst through the engagement of human visual perception and pattern recognition at the subconscious level though the depiction of large datasets in the form of familiar and significant avatars animated over linear time in an environment appropriate to the data being presented. This principle is advanced by Card, Mackinlay and Shneiderman where they state that information visualization amplifies human cognition through increased resources, reduced search, enhanced recognition of patterns, perceptual inference, perceptual monitoring, and manipulable medium.[2] Finally, HSAM implements a well documented and open interface providing a loosely coupled solution that is agile with respect to simulation technology. This open interface allows HSAM to be used in conjunction with any deep draft vessel movement simulation that is capable of producing output meeting the interface specifications. HSAM provides an analysis and presentation environment well suited for the cost-benefit analysis of policy and infrastructure modifications in deep draft harbors through the condensation and transformation of large datasets into a rich animated visual medium.

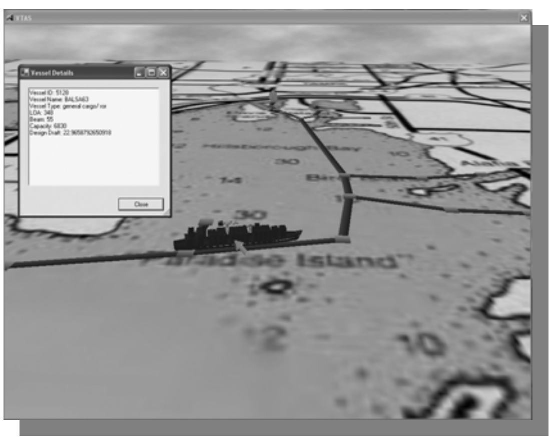

Figure 1. HarborSym Animation Module (HSAM)

ANIMATION FRAMEWORK

HSAM is based on an object-oriented, event driven animation framework. The choice of this paradigm was based on the need for a clean, open interface, extensibility, and strong specification of behaviors and characteristics. The HSAM framework is tightly linked to the Object-oriented Graphics Rendering Engine (OGRE) to provide the middle level visual rendering objects and Microsoft's DirectX™ or OpenGL for the low level rendering. OGRE is an object-oriented, open-source 3-dimensional rendering engine library written in C++ and is available without cost under the GNU limited public license. The initial prototype of HSAM was linked to Axiom, a Managed DirectX™ derivative of OGRE but proved to be less than satisfactory for reasons of stability and speed. OGRE was ultimately selected as the preferred implementation framework for its rendering speed, stability of code base, extensibility, cross-platform compatibility, and open-source nature.

The HSAM animation framework consists of a series of related objects working in concert to respond to a time ordered series of events to provide an animated depiction of the simulation data streams. The foundational objects in the HSAM framework are vessels, navigation nodes, and navigation reaches. These objects represent physical locations, the routes between those locations, and the vessels that move along those routes.

VESSEL OBJECTS

The vessel object, which represents the distinct vessels that will appear in the animation, has characteristic properties including a unique identifier, name of vessel, flag of registry, draft, beam, physical dimensions, cargo manifest, position, orientation, status, visibility, an avatar which is the physical representation of the object in the simulated environment, the vessel's class and sub-class identifiers, and a texture or simple color to be applied to the vessel avatar. A sampling of vessel avatars developed for HSAM are shown in Figure 2.

[2] S.K. Card, J.D. Mackinlay, B. Shneiderman, "Readings in Information Visualization: Using Vision to Think," Morgan Kaufmann Publishers, San Francisco, CA 1999.

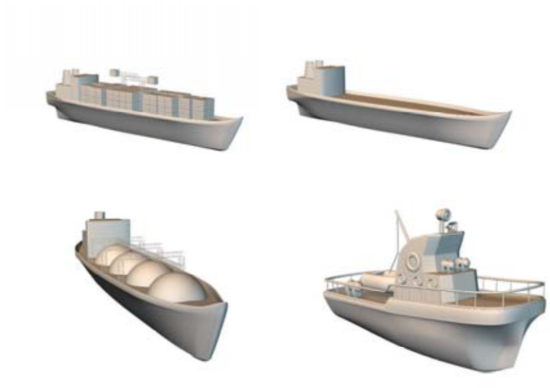

Figure 2. Sample HSAM Vessel Avatars

The vessel object retains a time sequence ordered queue of commands that pertain to itself and its cargo manifest. Vessel commands are directives to alter one or more of the vessel's properties either at a specific point in time or, in the case of a movement command, over a span of time. Properties that can be altered through a vessel command event are position, status, visibility, orientation (independent of default movement orientation), and avatar texture. Vessels can also be issued cargo manifest alteration commands. These commands take the form of a commodity code, a unit of measure, and a transaction quantity. At the indicated trigger time, such as after exchanging commodities at a dock, the vessel's cargo manifest is adjusted in accordance with the parameters of the command. Finally, as an optimization point, there is a "destruct" command that is issued to an instance of a vessel when it is no longer significant in the simulation, i.e., the calling vessel leaves the harbor. The destruction of the vessel immediately frees system resources and is used to improve average frame rate capabilities of HSAM.

NAVIGATION NODE AND NAVIGATION REACH OBJECTS

The HSAM animation framework maintains a hash table collection of those objects forming the navigation network – navigation nodes and navigation reaches. The navigation node object encapsulates a location in the simulated world that vessels may depart from, arrive at, or pass through. The navigation node object has properties encompassing a unique identifier, a name, status, a cargo throughput list, position, orientation, an avatar, a texture, and a category or type indicator designating the node object as dock, way point, entry point, or a turning basin. The category types are user defined during animation configuration as are the default avatars for each defined category as well as specific node avatar and texture overrides. The navigation node object retains a collection of commodity transaction volumes capable of tracking summary utilization in terms of cargo throughput. In addition to executing commands to allow the texture of the node, the status, and the visibility of the node to be modified, the node object processes commands to maintain the summary cargo lists. These commands allow resetting or clearing of the list and pass-through transactions indicating the cargo commodity type, the unit of measure and the quantity passing through the node. The pass-through transactions also indicate if the cargo was loaded or unloaded and the lists of loaded and unloaded cargo are maintained separately so as to allow the analyst to visualize trends or even anomalies in commodity flow.

The Navigation Reach object is a far simpler construct. A navigation reach is defined as a directed line connecting exactly two navigation nodes. The properties for reaches include a reach class identifier, a specific reach instance avatar, a texture or simple color for that avatar, and a status. The commands that can be issued to a reach object allow the texture of the reach, the status, and the visibility of the entire reach to be modified. Much like the vessel objects, both the node and reach objects maintain a time sequenced queue of commands pertaining to their specific instances. The nature of the commands available to the objects in the navigation network is more restrictive than those for the vessels as the arrangement of the network must be fixed prior to the appearance of the first vessel in the animation. Once time has begun to pass, the basic appearance of the network is no longer malleable. The command set and properties of the navigation node and navigation reach objects form the navigation network over which vessels transit in the animation and is well suited to the visual depiction of the changing status, conditions, and cargo movement patterns in deep-draft harbors.

MOVEMENT COMMANDS

In the table below, a tabular representation of a portion of a sample command stream, we see a series of commands that move two vessels and highlight a rule implementation. In the command stream, the first three fields (start time, stop time, command) are a common field structure with the arguments field being dependent upon the nature of the command issued. For example, in the first record, we have a vessel movement command (VMOVE) that will begin moving the vessel whose ID is 21 along the reach identified by 6 in the forward direction (IN) at time index 1000. The vessel will arrive at the terminal node of reach 6 at time index 1200. In this example, a vessel is moved along a reach, the vessel's status (VSTAT) is changed as well as the color of that vessel (VCOLOR) to indicate a "PRIORITY" rule activation. This rule reflects the real world operating practices applied to passage of cruise ships or gaseous tankers in which they are given unrestricted movement through the system. In the simulation, the reach is also visually triggered to show the rule activation (RSTAT) during the passage. Similar behavior occurs within the animation for all transit restrictions analyzed in HarborSym.

Table 1. Movement Commands

START TIME	**STOP TIME**	**COMMAND**	**ARGUMENTS**
1000	1200	VMOVE	21 6 IN
1005	1115	VMOVE	87 12 OUT
1005		VSTAT	87 “PRIORITY”
1005		VCOLOR	87 1.0 0.0 0.0 1.0
1005		RSTAT	12 “PRIORITY RULES”
1005		RCOLOR	12 1.0 1.0 1.0 1.0
1115		VSTAT	87 “UNDERWAY”
1115		VCOLOR	87 0.5 0.5 0.5 1.0
1200	1250	VMOVE	21 7 IN
1300		DESTROY	21

THE SYNTHETIC WORLD

The base or world object in the HSAM animation framework is a bubble of simulated time holding all of the component objects and forming a complete self-contained simulated world. The elements of the simulated world are:

- The world time clock
- The OGRE root scene node
- Collections and catalogs:
    - Two hashed collections of vessel objects (Fleet and Armada)
    - The collection of navigation node and reach objects
    - Catalogs of vessel categories, navigation node types, and navigation reach types
- Environmental commands
    - A time sequenced queue of commands
    - A set of pointers to configuration files

WORLD TIME CLOCK

The world time clock in the HSAM architecture is based on a user definable unit of time. The choice to unbind the unit of time was made to support simulations of vastly differing temporal scope and granularity. In the HarborSym implementation of the HSAM animation framework, the unit of time is dictated by the simulation kernel and is specified to be hours. The world time clock has a time scale property that controls the relationship of passage of time between the simulated world and that of the real world. The time scale factor is a decimal number restricted to non-negative values. The consequence of this range restriction is that the operator can vary the passage of time in the simulated world between stoppage and a rate that has months of simulated time passing in a single real-world second. This enables the analyst/operator to quickly pass through uninteresting portions of the simulation and focus valuable time and attention on animated segments at a rate that gives a maximum visual experience.

ROOT SCENE

The OGRE root scene node is an element of the world object and serves as the linkage point between the HSAM animation framework and the OGRE rendering engine.

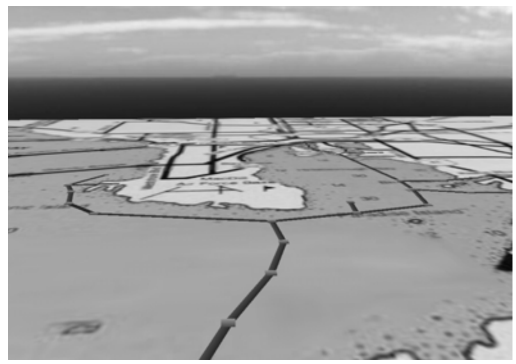

Figure 3. Fully Rendered Scene

OGRE maintains a scene graph comprised of meshes, textures, cameras, and other supporting elements commonly found in a rendering engine. The root scene node is the access point to the scene graph. The HSAM animation framework modifies and expands the OGRE scene graph by adding a plane mesh with an associated, user defined texture or image forming the surface over which the navigation network is laid and upon which the animation will ultimately play out. While both the HSAM animation framework and OGRE support height field defined 3-Dimensional terrain implementations, the cost, in terms of performance degradation, was excessive when compared to the additional visual impact in the simulation of vessel calls in a deep-draft harbor. Once the animation surface has been created in the OGRE scene graph, the textured mesh objects specified in the world objects collection of navigation nodes are added. The reach meshes are then added with the mesh being oriented and scaled to span between the nodes indicated in the world collection of navigation reaches. A fully rendered scene, showing the navigation network of nodes and reaches overlaid upon the planar mesh is shown in Figure 3.

Offset and orientation of the mesh with respect to the actual position of the navigation nodes may be specified to allow the effective use of more detailed meshes. An example of this would be a detailed dock mesh being offset from the actual node position so that vessels arriving at the dock would appear to "dock" instead of being collocated and intersecting, which can be distracting to the operator and less than satisfactory in a formal presentation. Another aspect of the simulated world that is set directly though the OGRE root scene node is the horizon texture. The HSAM animation framework implements an infinite skybox technique to provide this texture. In this texturing technique, six image files are loaded in the form of cube faces where the assembled cube is rendered such that the cube walls are always beyond any other object in the camera's viewport. The user is essentially "inside the cube", with the ability to maneuver and position the camera at will, always stopping at the cube faces or limiting walls of the simulated world. This technique enables the illusion that the user is inside a 3-dimensional world, with a navigation network below and a sky (complete with stars) above. Other interactions with the root scene node, including vessel instantiation and textural changes occur as the result of command execution once the animation is initiated and world time begins to flow.

COLLECTIONS AND CATALOGUES

Two slightly different collections of vessels are maintained by the world object in the HSAM framework – Fleet and Armada. The Fleet collection contains the specifications for every vessel that will appear during the life of the animation. It is used as the catalog of templates from which vessels will be instantiated once the animation begins. The vessel specifications in the Fleet collection do not have corresponding avatars or textures under the OGRE root scene node. Instead, they have specifications for the creation of those mesh and texture objects. This is done as a point of optimization for the animation and the OGRE engine in specific. Conversely, Armada is the collection of those vessels that are currently active in the animation at any given point in world time. The separation of the two collections is also an optimization point in that vessels that are not active in the simulation cannot be in motion and do not have visually modifying commands applied to them. Those vessels in Armada are fully instantiated and have avatar meshes and associated texture objects implanted in the OGRE scene mesh. It should be noted that due to the desire to optimize the processing of commands, vessels may be present in the Armada collection prior to actual visibility in the simulated environment and likewise may be hidden rather than removed once they leave the harbor but are destroyed at the first frame event after the trigger time code of a destruction command.

The HSAM animation framework maintains type descriptive catalogs for vessel classes, navigation reach types, and navigation node types. While each vessel, reach, and navigation node are allowed to specify unique textures and avatar meshes, those that choose not to specify fall back to the default mesh and texture information for the class or type found in the class/type catalogs. Within the HSAM framework, individual instances do not specify meshes or textures but rely fully on the class and type default information. These catalogs are established during the configuration phase of the animation and are not changeable once world time has begun to advance in order to protect visual consistency throughout the life of the animation.

ENVIRONMENTAL COMMANDS

The environmental command set that HSAM implements compliments common rule and policy trigger conditions with distinctive visual cues designed to work with the analyst on the borders of the subconscious. HSAM maintains a time sequenced queue of commands that affect the appearance of the environment as a whole. Predominantly these commands are used to indicate various environmental conditions that affect the activation or enforcement of alternate harbor policies or vessel movement restrictions and rules. The environmental commands supported by the HSAM animation framework are:

- Sun Up
- Sun Down
- Fog Conditions
- Clear Conditions
- Tide High
- Tide Low

Each of these commands affects the entire visual world being animated. The commands Sun Up and Sun Down cause the horizon texture set applied to the skybox to be replaced with a texture set specified during configuration of the animation. HSAM does not allow a horizon that is not either Sun Up or Sun Down and the default is Sun Down as the world clock runs in time units since midnight of the first day of the simulation. Visibility commands, Fog Conditions and Clear Conditions, apply a logarithmic density fog effect to the camera causing distant objects to be obscured by grey haze. Tidal commands are implemented by altering the texture on the image used as the planar movement surface as opposed to the horizon oriented textures modified by Sun Up/Down commands. The implementation of this has been to use modified versions of the same surface image with the water color made darker in the high tide texture. This provides a much stronger visual cue than a physical change in apparent water height, which would be difficult to discern given the scale of the simulated environment and the planar nature of the animation surface. HSAM also provides a default tide-neutral condition in the absence of any tide related commands.

BREATHING LIFE INTO THE SYNTHETIC WORLD

The HSAM environment, once the animation has begun, is affected by two data streams. The first stream that interacts with the running animation is the command stream generated by the loosely coupled simulator. This time sequenced stream of commands control or alter the content and appearance of the simulated world and includes the environmental commands, reach commands, node commands, and vessel commands.

- Vessel commands
  - Appear at designated node
  - Move along designated reach
  - Change status
  - Change orientation
  - Visibility
  - Alter texture or color
  - Cargo manifest transaction
  - Destruct
- Reach commands
  - Change status
  - Visibility
  - Alter texture or color
- Node commands
  - Change status
  - Visibility
  - Alter texture or color
  - Cargo throughput log addition
- Environmental commands
  - Sun up / down
  - Fog on / off
  - Tide low / high

The second control stream is the interactive input from the analyst through the keyboard and mouse. The HSAM viewport is a virtual camera in the simulated environment under the direct control of the operator. The keyboard is used to control pitch, yaw, and acceleration or zoom. The mouse can also be used to control pitch and yaw. Roll has been excluded from the control set as testing has shown that the majority of operators could not maintain coherent "flight" when this degree of freedom was allowed. Those operators that could maintain control of the camera appeared to be focused on the flight controls to the detriment of any analysis task at hand. The operator also has keyboard control of the vessel avatar scale and the time scale factors. Altering these parameters dynamically allows the operator to control the fluid pace of time in the simulated world and to alter the significant appearance of the vessels in motion so that information is not lost at vastly different viewing vantages. Finally, the user has control over the quality of the display, which can be adjusted to provide the best quality of animation on a wide range of display devices.

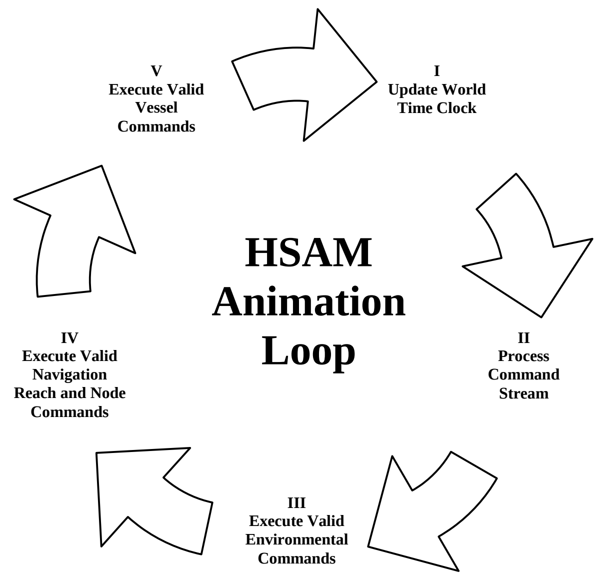

Figure 4. HSAM Animation Loop

The heartbeat of the HSAM synthetic world is the animation loop. HSAM overrides the OGRE event handler that is called each time a frame is rendered. Within this event handler HSAM performs all of the interactions between the command stream, the user input stream, and the synthetic world. The first thing done is the update of the world clock based on the amount of actual time passed since the last frame and the time scale factor. After the clock has been updated, the command stream is examined to determine if the current command should be processed. Commands are only processed if the current system time is equal to or beyond the trigger time for the command at the top of the stream. If the command is determined to be active, it is dispatched to the appropriate command queue. The next phase of the rendering loop passes control to the environmental command queue. The queue will modify the environment based on the command(s) that are valid at the current time. Control is then passed to the navigation network for the processing of command queues for navigation nodes and reaches. The final step in the animation loop is the execution of vessel commands. Each vessel in the active Armada collection is polled to execute the commands in their queues that have reached trigger points. It should be noted that the vessels reject certain invalid commands that would result in an alteration to the command stream, such as overlapping movement commands. Once HSAM has finished the animation loop, control is passed back to the base OGRE frame rendering event handler in order to allow the rendering engine to maintain internal operations.

A DATA DRIVEN WORLD

The HSAM animation framework is loosely coupled with the simulator that drives it. This was specified in the design to maximize the utility of HSAM without the need for code-level customization to the framework when changes are made to the simulation kernel. The linkage between HarborSym (the simulator) and HSAM is a well defined interface specification. That interface consists of two ASCII text files – the configuration stream and the command stream. Both files consist of command level records, one per line, in a fixed field width format. The command stream is time sequenced post-processed information produced by the HarborSym discrete-event simulator. The command stream must be sequential in non-decreasing fashion based on either start of time span or trigger time. Failure to maintain the proper order can cause the animation to stall or commands to trigger outside of indicated time parameters. As discussed earlier, the command stream contains events for controlling vessel movement/appearance, navigation node and navigation reach appearance/status and environmental controls including daylight, fog and tide. The configuration stream is not time sequenced but does have structure to the order in which commands can be processed. The configuration stream layout is as follows:

1. Vessel type catalog entries
2. Navigation node type catalog entries
3. Navigation reach type catalog entries
4. Animation surface size declaration
5. Animation surface texture specification (optional)
6. Skybox texture specification (day and night) (optional)
7. Tide high/low texture specification (optional)
8. Navigation node instantiations
9. Navigation reach instantiations
10. Vessel (fleet) declarations

The information creating both data streams (configuration and command) is directly exported by the HarborSym discrete-event simulator. The configuration stream contains information which is user-defined and may vary project to project or even purpose to purpose within the HarborSym model. The command stream is created by HarborSym within-simulation as an export of select discrete events as they occur within the simulation. With respect to both data streams, HSAM will produce animated visualizations of processed HarborSym simulations with no alterations to the HSAM code base or its data sources. It is completely data driven, requiring no user intervention of any kind to produce interactive, immersive visualizations of deep-draft maritime traffic simulations.

CUSTOMIZATION AND DISTRIBUTION

The cost of deployment for HSAM after initial research and development have been completed is zero. The HSAM animation framework employs the OGRE rendering engine in its design for two very important reasons: accessibility and agility. Design specifications for HSAM dictated that the product had to be fully accessible to the HarborSym user base in a freely distributable manner, requiring no third-party software components and being free of the need to customize the software simply because the simulation package had been applied to a new study area. Further, the nature of the desired visual experience called for a powerful, full-featured rendering solution. This left the development team with the decision to either build a full-featured frame rendering engine from scratch or query the open-source community for applicable products. After quickly contemplating the astronomical costs of developing a powerful frame rendering engine from the ground up, the development team decided to make extensive use of libraries, components and file formats all without distribution costs or applicable license fees. Axiom, an open-source rendering engine based on .NET framework and implementing both managed DirectX™ 9.0 and OpenGL rendering solutions served as the foundation for the initial proof-of-concept phase of the HSAM project. The Axiom implementation was acceptable with marginal performance. Three major problems caused Axiom to be abandoned as the rendering engine in the HSAM animation framework: marginal performance which severely limited the complexity of the simulations that the framework was capable of animating, a highly irregular and unreliable core development team which brought into question the stability and maintainability of the product, and finally a rather unnerving problem with compatibility issues with the managed DirectX™ 9.0 dynamic link libraries causing overly complex installation procedures. OGRE, the Object oriented Graphical Rendering Engine, is the parent project that spawned Axiom. OGRE, due to its larger and more established development team, presented a very stable and reliable product. Benchmark testing showed OGRE to surpass Axiom in all tested areas. Finally, OGRE did not suffer from the managed DirectX™ 9.0 DLL versioning problems that Axiom did. While the loss of the functionality inherent in the .NET framework was a drawback, OGRE and C++ have replaced Axiom and C# in the HSAM animation framework primarily for performance and stability benefits.

The HSAM framework implements open and well defined interface and data standards allowing the owner/operator to couple HSAM to a variety of simulators or data sources external to the HarborSym suite of tools. Only a basic understanding of the necessary configuration and command stream file formats would be required. Customization of the visual components such as vessel and navigation node avatars along with associated textures can be accomplished with a very small investment in tools. The avatars used in the HSAM animation framework are implemented using an open format that is not linked to any specific toolsets – the OGRE .MESH format. While the .MESH format specification is extremely robust, the avatars provided with the HSAM product are simple triangulated meshes. Tools that can be used to generate these kinds of avatars include Hash Animation Master™, Milkshape 3D, Lightwave™, Discreet 3D Studio™, and Maya™. Animation Master™ was used to generate all of the vessel and navigation node avatars for HSAM due to the powerful spline based modeling paradigm and intuitive work flow as well as the realistic cost. One of the interesting and powerful features of the OGRE .MESH file format is the accommodation of multiple levels of detail (LOD) for a single mesh object that are swapped into the rendering pipeline based on the distance of the camera from the mesh in question. This allows the creation of highly detailed meshes that will not tax the rendering engine until the camera is at an appropriate distance to allow the viewer to observe the presented detail while a perfectly acceptable mesh of far lower detail is presented when at a greater distance. Each of the node and vessel avatars include five distinct LOD providing a smooth detail gradient based on camera distance. The avatar file format is powerful, non-proprietary, and is supported by a large pool of creation and maintenance tools.

CONCLUSION

Display of simulated or historical vessel movements in a deep-draft harbor using an interactive 3-Dimensional environment as implemented in the HSAM animation framework has multiple benefits to the analyst and production staff. The most straight forward use of the HSAM product is in the presentation of findings to non-technical audiences. The audience can be "shown" the results of the simulation with the presenter-operator narrating the animation and varying the pace of the animation while gauging the response of the audience. Additional benefits lie in the ability of human vision to quickly identify aberrations in motion patterns. The interactive animation of simulation results can be used to spot errors in the simulator or other data sources. Some of the problems that can be quickly identified visually are abandoned vessels and vessels "teleporting" in or out of the environment at spots other than designated entry/exit locations. Abandoned vessels are those that enter the simulation, make a few movements, and then stay at the same navigation node for an extended period of time. Vessels that "teleport" or appear at an internal node on the navigation network as well as the abandoned vessel are quickly spotted and can lead the analyst to a fruitful and well focused investigation of the data source or simulator. Relatively unique to the easily customized and data driven animation frameworks as implemented in HSAM is the ability to highlight and investigate certain motion trends. A simple example of this would be to highlight the movement patterns of a certain class of vessels. The operator would specify a bright color for the class of vessels that has piqued his or her curiosity while leaving all of the other textures set to a neutral grey. While all of the other vessel classes are still visible and the vessel interactions are still evident, the movement of the selected class of vessels is clearly enhanced and subtle patterns can more easily emerge.

Proposed changes to harbor configurations and policy are often analyzed and validated through simulations. The complex results of such simulations can be made more accessible to both analysts and non-technical audiences by transforming those results into a familiar visual depiction that varies with time – animation. Visualizations of this type increase data analysis capabilities while enhancing the capacity to demonstrate the results of harbor structure, condition, and/or policy changes to non-technical audiences in a cost effective manner. HSAM is designed to be an interactive, immersive visualization of deep-draft maritime traffic simulations produced by the HarborSym Monte Carlo simulation model. HSAM is completely data driven which allows the operator to configure the animation to visually represent any harbor configuration being simulated as well as dynamically update the animation based on modifications to the simulation without the need for programmatic alterations. This represents a substantially cost effective presentation solution. HSAM has been shown to be an efficient tool for simulation data analysis as well as an effective presentation tool to audiences with a wide-range of technical backgrounds. A key design feature of the HSAM is the ability to control the appearance and content of the animation from HarborSym without the need to recompile any portion of the system or the need for a multi-media production engineer. HSAM is configured through loosely coupled, well defined data streams. The HSAM animation framework is object-oriented in design with objects bearing a strong similarity in both structure and behavior to their real-world analogs. Interaction with HSAM is easy, powerful, and visually engaging. The interactive nature of the HSAM animation provides a rich environment for simulation analysis and rich dynamic presentation.

REFERENCES

Card, S.K; J.D. Mackinlay, B. Shneiderman. 1999. "Readings in Information Visualization: Using Vision to Think". Morgan Kaufmann Publishers, San Francisco, CA.

Moser, David; K. Hofseth; S. Heisey; R. Males; C. Rogers. 2004. "HarborSym: A Data Driven Monte Carlo Simulation Model of Vessel Movement in Harbors" In Proceedings of the 2004 International Workshop on Harbor Maritime and Multimodal Logistics Modelling & Simulation (Rio de Janeiro, Brazil, Sep. 16-18). Universidade Federal do Rio de Janeiro, 3-15.

OGRE 3D: Open Source Graphics Engine. 2005. The OGRE Team. http://www.ogre3d.org/

Tufte, E.R.. 1983. "The Visual Display of Quantitative Information". Graphics Press, Cheshire, CT.

navigation · economics · technologies

The NETS research program is developing a series of practical tools and techniques that can be used by Corps navigation planners across the country to develop consistent, accurate, useful and comparable information regarding the likely impact of proposed changes to navigation infrastructure or systems.

The centerpiece of these efforts will be a suite of simulation models. This suite will include:

- A model for forecasting international and domestic traffic flows and how they may be affected by project improvements.
- A regional traffic routing model that will identify the annual quantities of commodities coming from various origin points and the routes used to satisfy forecasted demand at each destination.
- A microscopic event model that will generate routes for individual shipments from commodity origin to destination in order to evaluate non-structural and reliability measures.

As these models and other tools are finalized they will be available on the NETS web site:

http://www.corpsnets.us/toolbox.cfm

The NETS bookshelf contains the NETS body of knowledge in the form of final reports, models, and policy guidance. Documents are posted as they become available and can be accessed here:

http://www.corpsnets.us/bookshelf.cfm

navigation · economics · technologies