Smart Ship System Design (S3D) Integration with the Leading Edge Architecture for Prototyping Systems (LEAPS)

Julie Chalfant[1], Blake Langland[2], Douglas Rigterink[3], Cullen Sarles[3], Peter McCauley[4], David Woodward[4], Alan Brown[5], Robert Ames[6]

[1]Massachusetts Institute of Technology, Cambridge, MA, chalfant@mit.edu
[2]University of South Carolina, Columbia, SC
[3]Naval Surface Warfare Center Carderock Division, West Bethesda, MD
[4]Naval Surface Warfare Center Philadelphia Division, Philadelphia, PA
[5]Virginia Polytechnic Institute and State University, Blacksburg, VA
[6]Office of Naval Research, Arlington, VA

***Abstract*— The Navy's early-stage design tools are integrated with the Leading Edge Architecture for Prototyping Systems (LEAPS) data repository, thus enabling a streamlined, cohesive approach to the creation, storage and access of data pertinent to a ship design. The Smart Ship System Design (S3D) design environment currently under development within the Electric Ship Research and Development Consortium (ESRDC) recently completed development of S3D as a LEAPS application and is extending its capability to include the LEAPS application framework, multi-platform compatibility, and added functionality.**

Keywords—Electric Ship Design, Software Development, Ship Systems, Data Storage Ontology

I. Introduction and Motivation

A common data repository of a hub and spoke architecture has many benefits in ship design and analysis that provide rigor and commonality among software products; among them are a common ontology, a common representation for data, a coordinated definition of the current status of data and the design data trade space, and the elimination of translation errors between one software product and another found in traditional bus architectures. The Leading Edge Architecture for Prototyping Systems (LEAPS) is the U.S. Navy's data repository for ship design data; surface-ship-related data are stored in LEAPS using the Formal Object Classification for Understanding Ships (FOCUS) product meta-model (ship ontology).

The Smart Ship System design (S3D) software brings the ability to create, visualize and analyze two-dimensional one-line diagrams of electrical, mechanical, piping and HVAC systems along with three-dimensional visualization of physical placement and interactions of equipment and structure. Integrating S3D with LEAPS and FOCUS allows ship system designs to be created, refined and analyzed in a cohesive structure such that the data developed within S3D is available to other ship design software and vice-versa.

Further, writing software using the LEAPS application framework facilitates modularization of the software thus enabling reuse of modules among various software applications and the implementation of modules in a large-scale design space exploration implemented using the Rapid Ship Design Environment (RSDE).

Figure 1 provides a screenshot of the electrical designer view in S3D. All the design views within S3D have the same look and feel. Figure 2shows a screenshot from the LEAPS editor. The integration of S3D with LEAPS uses the application framework, reusing the previously designed dockable panels available in the LEAPS editor.

This paper details the progress of the S3D and LEAPS development and integration effort, extending the work previously reported in [1] and [2]. The paper is organized as follows: Section II presents background on the Navy design tools and their software structure and data structure. Section III provides an overview of the S3D design environment. Section IV describes the anticipated work flow and S3D use cases. Section V describes the data structure for storage of systems in LEAPS. Section VI provides a summary and outline of future work.

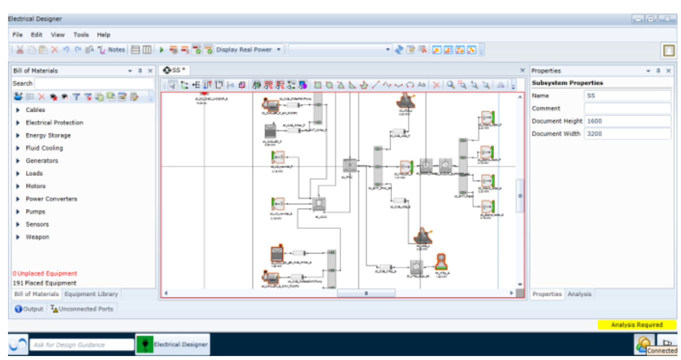

Figure 1. Screenshot of electrical designer in S3D.

This material is based upon research supported by the U. S. Office of Naval Research (ONR) under award number N00014-14-1-0166 (MIT), and N00014-14-1-0165 (USC), ESRDC – Designing and Powering the Future Fleet; N00014-16-1-2956, Electric Ship Research and Development Consortium; N00014-16-1-2945, Incorporating Distributed Systems in Early-Stage Set-Based Design of Navy Ships; and by the MIT Sea Grant College Program under NOAA Grant Number NA14OAR4170077.

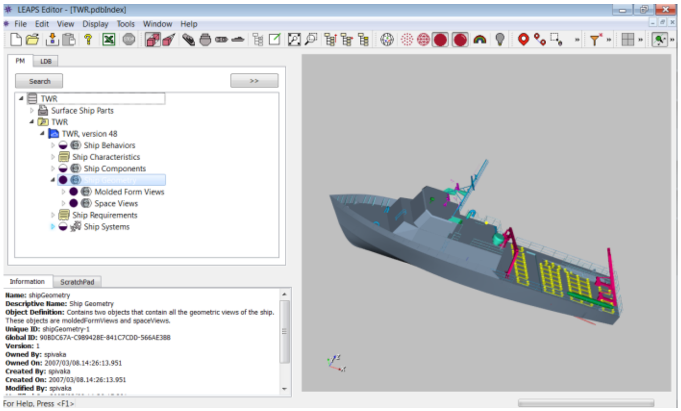
Figure 2. Screenshot of LEAPS editor

II. NAVY DESIGN TOOLS

LEAPS is a development framework that supports virtual prototyping and analysis of conceptual and preliminary ship designs through integration of many design and modeling and simulation tools. The overall goal of using LEAPS is to cut down on unnecessary data manipulation, thus increasing the speed of a design effort while minimizing the risk of introducing inconsistencies into the design data. The LEAPS Application Programming Interface (API) contains a set of generic data classes that describe physical and/or functional representations of engineered products. On top of this API, the LEAPS Application Framework has been created to streamline the process of developing LEAPS applications. Another advantage of the Framework is it allows for the easy reuse of components from one application in other applications and standardizes the user experience. Applications built using the Framework can be run on either Windows or Linux operating systems, which allows for the use of high performance computing clusters when running computationally expensive simulations.

The FOCUS product meta-model is the specific object formalization for defining a surface ship. FOCUS formalizes the physical and functional characteristics typical of a ship using the available LEAPS classes. Adherence to FOCUS is what ensures all LEAPS applications (shown in Figure 3) can read and write to a LEAPS database and use consistent values during the design and analysis process. One way to think of the relationship between LEAPS classes and FOCUS object formalization is to draw a parallel between the alphabet (classes) and English (formalization). LEAPS also uses formalized object ontologies for air vehicles (AIRSOM) and submarines (SUBSET).

LEAPS applications are typically divided into two groups: design tools and analysis tools, as shown in Figure 3. Design tools include the Advanced Ship and Submarine Evaluation Tool (ASSET) and the Rapid Ship Design Environment (RSDE), which are used for ship synthesis and design space exploration [3]. Analysis tools include the Integrated Hydrodynamics Design Environment (IHDE) which facilitates the use of high-fidelity computational fluid dynamics tools in early-stage design by non-expert users. Whereas the design tools

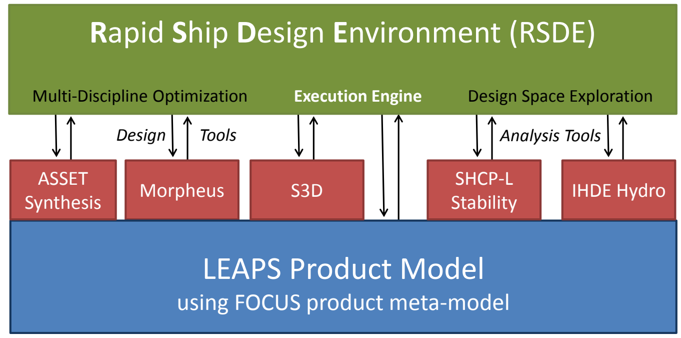

Figure 3: The LEAPS Software Environment, adapted from [5]

contain both modeling and analysis, they provide the bulk of the data used to populate a ship concept into the LEAPS database. Analysis tools are used to simulate and record the behaviors associated with those representations. The tools are represented here for convenience left to right because analysis usually depends on data created by the design tools. No directionality is implied and no tool communicates to any other tool except through LEAPS data.

Transitioning S3D to Navy use required making S3D compatible with LEAPS and built using the LEAPS application framework to allow for the analysis of the impact of system design decisions on the overall ship concept. Additionally, in [4] it was mandated that all future Navy tool development must build on the LEAPS foundation.

III. SMART SHIP SYSTEM DESIGN (S3D) SOFTWARE OVERVIEW

A. *Introduction to the Software*

S3D is a collection of tools that provide design and simulation support for various engineering disciplines such as electrical systems, mechanical systems, and air and liquid cooling systems. In addition, S3D provides a three-dimensional equipment arrangement tool so the users can ensure the physical fit of equipment and their placement while providing for a more realistic ship concept model that includes lengths of cables, pipes, ducts, etc. The primary impetus for the development of S3D was to dramatically reduce the time required to build a conceptual ship model, to allow engineers from disparate backgrounds to easily understand the impacts of the their design decisions on the ship as a whole, to reduce the risks involved with the introduction of novel technologies, and to increase the quality and efficiency of the overall ship.

The various schematic editor tools provide a view into the ship that is particular to a specific discipline. For instance, the electrical schematic editor displays only equipment that is electrical in nature, hiding equipment such as valves and gearboxes. This allows the engineers to focus on the parts of the ship that are of importance to them while ignoring superfluous information. This does not mean, however, that the interactions between systems are ignored. For instance, if the thermal engineering team decides that the cooling pumps need to increase the flow rate of water to certain devices then the electrical team will immediately be made aware of an increase in the electrical power demand from those pumps. Likewise, if the naval architect decides to relocate the transformer in 3D space farther from the main generator to which it is connected, the electrical engineer will see that the impedance of the cable connecting the two devices is increased.

In addition to the design and simulation tools, the S3D environment provides an equipment library for both notional as well as off-the-shelf products. S3D uses a multi-view modeling approach that allows distinct and separate simulation models to represent the same piece of equipment in various disciplines. For instance, a motor might have an electrical, mechanical and thermal representation. The equipment library allows all discipline-specific views to see the same device and provide equipment details from different perspectives. The properties for the various equipment types are stored in the equipment library via a set of formalized attributes types. The definition for a specific equipment type is formalized by the FOCUS ontology and can be extended as necessary. Formalization of the ontology as well as the attributes allows for other software tools to readily query the design and programmatically extract or insert data as needed, thus potentially enabling the tool to be used as part of an automated design process such as RSDE.

B. Pending Upgrades

The ESRDC continues to work on advancing the capabilities of the S3D design environment. As the research and development mature, the resulting products are integrated into the S3D environment. Currently a mission definition and analysis tool are slated to be ported to this environment. The mission definition tool provides a means for users to create specific scenarios for which a set of conceptual ship designs can be exercised against for comparison purposes. Because S3D provides a steady-state analysis, the mission definition tool allows users to create mission segments, which provide the ship operating point and the time duration. These mission segments are stitched together to create the mission definition. These missions are created in a ship agnostic manner so that they can be created once and then exercised against many different types of ships. For instance, a yearly fuel consumption mission might stipulate that the ship operate under cruise conditions for 25% of the year, operate in a patrol mode for 20% of the year, be docked with shore power for 45% of the year, and battle conditions for 10% of the year. Each ship makes a determination as to which weapons and sensors are on-line, which loads have power, which power generation modules are on-line, etc. Once this has been stipulated the ship can then be exercised against the mission profile in order to determine the actual fuel consumption for a year. This same mission is then executed against several competing designs and used for comparison purposes. Typically, a suite of mission types will be created and then exercised against a set of conceptual designs as a means to help differentiate between the designs and provide the contrasts necessary to help the stakeholders down select.

C. Future Work

The ESRDC has several active research thrusts that, as the research matures, will lead to new capabilities and enhancements to the S3D design environment.

Research into a design guidance and decision support tool has begun, with the purpose being to help users of S3D readily leverage the larger body of ship design knowledge that exists within research journals, military specification documents, and the Navy and industry's established best practices. This tool will allow users to search through the large body of documentation to answer specific design questions and to help provide assessments of existing designs. When the tool discovers exceptions in the design to recommended practices, the deficiency will be noted and brought to the attention of the design team for potential modification.

S3D currently lacks a control layer. One active area of research in the ESRDC is the exploration of the appropriate level of detail that should be captured with respect to controls during the conceptual phase of the design process. Currently the S3D users must manually configure the plant alignment for the ship in order to provide appropriate power and cooling to the various pieces of equipment for each operating point. This is a time consuming process and so the control layer will likely include some form of an optimization algorithm that is capable of assisting users in this configuration process and do so considering factors such as fuel efficiency among others.

When designing a ship, the introduction of new technologies, features, or capabilities can introduce significant risk into the design process. One way of mitigating such risk is to perform detailed time-domain analysis of the ship design. One research thrust the ESRDC is pursuing is to help enable an automated transition of a conceptual design to a more detailed time-domain model. The S3D environment will provide mechanisms to help expedite and automate the exploration of a design in greater detail and provide feedback mechanisms in order to update and improve the fidelity of the conceptual model. The ability for S3D to rapidly explore large sets of potential ship designs will help the stakeholders and end users gain confidence that the proposed design is both a feasible and good design.

S3D needs the ability to work in a set-based-design environment both in the generation of potential sets and in the evaluation of these sets. The ESRDC is exploring ways in which the S3D environment can help produce such sets from high-level user inputs such as mission loads, hull displacement, range of the ship, maximum speed, etc.

IV. Work Flow and Use Cases

A. Work Flow

This section focuses only on the workflow within the S3D tool. A discussion of how S3D can be used in conjunction with other Navy design tools in a set-based design study can be found in [3]. The basic workflow of S3D has four parts: component creation, system connection, component placement, and system analysis.

1) Component Creation

To fully model a component the user must assign a weight, an area, and a volume requirement, and model the component's electrical, mechanical, and thermal behaviors. Fuel requirements for engines must also be defined.

The user is provided with a number of simulation models for common components like electric motors or gas turbines. These computational solvers characterize the behavior of a component during simulation. Component attributes or properties are used to provide static properties like weight or simulation parameters for each component. These provided components come in two types. The first type is notional components where the user is free to modify the component values, for example creating a hypothetical gas turbine that produces 25,000kW but only weighs 10t. The second type, actual components, represent real products like a GE LM2500 which has locked output production of 24,050kW and a weight of 90t, per manufacturer's specifications.

For next generation components, like Power Electronic Building Blocks, where a simulation model is not already available in S3D, the user would need to code that component's simulation model(s), associate the model with the component, and place it in a LEAPS catalog for use by S3D.

2) System Connection

To begin building a system the components must first be connected together logically in a schematic view, similar to what has been shown in Figure 1. S3D can be used to create this logical connection in the electrical, mechanical, and thermal fluid domains. During the system connection process, the components are connected using logical connectors rather than specifying wires, cables, shafts, or pipes.

3) Component Placement

Once a system schematic has been developed, it is necessary to simulate the system to prove that the right components and connections exist and the system is feasible. Whereas physically arranging components in 3D space is not required for simulation, a simulation is incomplete without it. Creating 3D arrangements allows for the analysis of the effects of things like cable length on impedance and calculating the length of the propeller shafts. Placing the components also arranges and associates structural subdivision zones with electrical or other system zones.

The arrangement and placement of components in a ship requires the existence of a LEAPS ship concept. While any tool can populate a concept to a LEAPS database, S3D assumes that the concept was populated by the ASSET module of RSDE, and therefore includes structural subdivisions, deckhouse, propellers, and prime movers. The structural zones, as generated by RSDE, will provide arrangement boundaries for S3D. Additionally, RSDE provides S3D with a speed-power curve for determining the load on the prime movers.

4) System Analysis

Once all components are placed, the user can run an electrical, mechanical, or thermal-fluid load-flow simulation to determine if all the components are receiving their required power and cooling. Additionally, the user can run the simulation for a specific discipline at any time during the process of creating the system model.

At this time S3D only performs a steady state analysis, intended for early-stage conceptual design. This is consistent with other early-state system design practices. The possibility of adding dynamic, time-domain analysis and controls systems simulations is currently being explored.

B. Templates and Patterns

In an effort to reduce the time to develop complete system designs and to standardize at least parts of the designs the concepts of Templates and Patterns will be used by S3D. This concept was introduced in [3] and has been refined as the nature of the problem has been better understood.

Templates and Patterns exist on a continuum of design fidelity that begins with back of the envelope sketches and ends with the physical systems in a functioning naval combatant. Templates have higher levels of detail than Patterns. Patterns serve to hold general rules and connection information between component types within a design. The rules can be used to size different parts of the system as components and their properties are defined in the process of creating a template. For example a Pattern may only contain the information that a gas turbine generator is connected to a switchgear bus and that the sizing of the cable between the two follows a certain rule. During the system design process the cable is then sized appropriately to meet the generator to switchboard electrical distribution requirements.

Templates can be thought of as large, aggregate components. Each Template will have its own interface definition existing at terminals corresponding to the available disciplines within S3D. Templates can also contain relative location information of the individual components within the Template for storing information regarding zonal survivability for instance, but this is not required.

The goal for using Patterns and Templates in a Navy-led design is to have certain sets of Patterns and/or Templates verified and validated and then assigned for use in either pre-AoA studies or within the AoA themselves. It is anticipated that Templates will have sufficient detail to even be used past the AoA in any Navy-led preliminary or contract design activities. The full requirements for Patterns and Templates are still under development.

C. S3D Use Cases

A top-level overview of the anticipated use cases for S3D is provided below; these are based on the work flow described above.

1) Ship-Specific System Design Use Case

A single user works on the design of systems within a single ship design by opening or creating a FOCUS-compliant design in a LEAPS database; by adding, manipulating and/or deleting equipment; and by running simulations on the resultant designs.

The normal progression would be to use S3D after running a ship synthesis in ASSET and possibly using other programs such as SHCP or IHDE. The changes made during S3D use are stored in a FOCUS-compliant manner, so the information created is available to other LEAPS-integrated software after S3D implementation.

S3D does not create or modify ship geometry such as hullform, superstructure, decks or bulkheads. It can open and render existing ship geometry and interact with it for equipment placement.

2) Standalone System Design Use Case

In the standalone system design use case, S3D is used to create a notional system design without a ship hull, rather than loading a ship design from a LEAPS database as described in the previous use case. This use case can be used to test out new system ideas in a stand-alone atmosphere without considering a specific ship arrangement, or to model off-hull systems such as those found in a shore-based test site.

For example, a novel cooling plant might be designed that is capable of dissipating 10 MW of heat; this cooling plant could then be integrated into several designs. In this case the thermal engineer would supply all of the cooling components necessary and may simulate and test the design.

3) Pattern Creation Use Case

As described above, a Pattern is a connected topology that defines functions and interfaces and is composed of functional blocks which represent classes of equipment or functions rather than specific components.

To create a Pattern in S3D, one could follow the same general procedure as the standalone system design use case described above; however, the "components" used would be functional blocks. The additional functionality required from S3D to accomplish this is the creation of a component that represents a functional block such as an "Equipment Type Category" in S3D.

A simulation cannot be run on a Pattern; insufficient information exists in the functional blocks.

4) Instantiating a Template from a Pattern Use Case

A Pattern contains functional blocks that represent classes of equipment or functions rather than specific components. To instantiate a Template from a Pattern, the functional blocks in the Pattern are replaced with actual components in a Template. The Template's components may represent individual off-the-shelf pieces of equipment, notional components, assemblies, or they could even be "no-operation" components that do nothing other than connect. For example, there may be a "power conversion module" functional block in a Pattern that is replaced in a Template with a specific piece of equipment (e.g. a SiC rectifier rated for 30MW at 13.8 kVAC and 20kVDC with given dimensions), a notional piece of equipment (e.g. a rectifier whose size and rated voltage and power are adjustable), an assembly (e.g. several PEBB modules assembled into a converter unit) or just a connection (e.g. connecting a generator directly to a main bus when the generator produces power at bus voltage/frequency so that no conversion is necessary).

There may also be situations in which a single component replaces more than one functional block, e.g. an Integrated Power Node Center (IPNC) performs power conversion, energy storage, and fault protection functions, which could be represented by three separate functional blocks in a Pattern.

To instantiate a Template from a Pattern requires copying the Pattern into a new Concept, then replacing the functional blocks with components and defining the node data properties. Attention is required in the cases when there is not a one-to-one match between a single functional block and a single component.

The original Pattern remains unchanged during and after the Template instantiation process; thus, multiple different Templates may be instantiated from a single Pattern.

A Template contains sufficient information in the instantiated components to perform a simulation. It may be necessary to add dummy components for a full simulation; e.g. an electrical distribution template may require the addition of loads and sources for successful simulation.

5) Template Creation Use Case

A Template will usually be created as an instantiation of a Pattern, but it may be created from scratch. In this case, the Template will be created using the "Standalone System Design Use Case" described above. A Template may also be created by copying an existing Template into a new Concept and modifying the new Concept. In this case, the original Template would remain unchanged.

6) Application of a Template to a Ship Design Use Case

To apply a Template to a ship design, the user opens a ship Concept in one LEAPS database and applies a template from another LEAPS database.

Once the Template has been instantiated in the ship design, the resultant ship design and all its constituent elements (components, connections, systems, diagrams, etc.) must be editable and simulatable as a stand-alone ship design. Any changes made to the design are not propagated back to the template; the original template remains unchanged during and after the instantiation process.

In the future, it is anticipated that LEAPS database(s) will be available which contain a number of system Templates. These Templates will have fully functional, if sparse, systems containing all components and connections required for the system to function. These Templates may be fully designed systems with all constituent parts that merely require copying from one concept to another. The Templates may also be portions of a system that must be assembled into a whole within the final ship design. It is envisioned that most users will not create a new system from scratch; rather they will start with a baseline system and edit it to fit their needs.

7) Rapid Ship Design Environment (RSDE) Use Case

It is anticipated that programmatic design and simulation of a ship under the RSDE paradigm will be possible through judicious application of the Template/Pattern process. This methodology remains to be developed.

V. DATA STRUCTURE

The underlying concept for S3D data storage is that information that details a specific ship design and that is needed to run simulations is stored in the LEAPS database itself; information that is used by S3D but is unchanging from one ship design to another is an inherent part of the S3D software and not stored in LEAPS at all; information that is the result of a simulation or that can be recreated easily is persisted in an .xml file stored in the LEAPS file structure but not actually part of the LEAPS database; and information required during simulation but that can be derived from properties or operation state is not stored at all. Another way to view this structure is that information likely to be used by other software tools or to be needed in analysis of the ship designs is stored in the LEAPS database. More details about data persistence follow:

- Data pertaining to the ship components, nodes, connections, diagrams, systems, etc., is stored in a LEAPS database in accordance with the FOCUS product meta-model. This includes such information as names, properties, location, orientation, and connectivity.
- Data pertaining to the specific ship system design that is necessary for the operation of S3D but is not expected to be used by any other program is stored in the LEAPS database as tool data. The tool-specific properties are those properties that need to be persisted from one S3D use session to another, or that should be maintained with the model for future reference. For example, these would include operational settings for equipment such as switch position, valve position, or adjustable load power level that were used to run a simulation.
- The simulation engines for the individual components and the overall solvers are maintained within the S3D software itself. Each component within a ship design stored in a LEAPS database will contain, as a tool property, the ID of the simulation engine associated with that component if the component is a part of an S3D simulation. The supporting properties to run the model are all stored within LEAPS as either FOCUS-compliant data or S3D tool data.
- S3D-specific information such as schematic data that denote the location and orientation of icons on the screen for viewing a one-line diagram are stored in an .xml file within the file structure of the LEAPS database.

A. Component Specification Data

Component properties that are expected to be used by other software programs are stored as properties on the respective components, either as component-specific properties, or as shared properties that are associated with the component. Examples follow:

- Common Component Properties are pertinent to all components and include such things as location, orientation, weight, and Ship Work Breakdown Structure (SWBS) number.
- Shared Component Properties are pertinent to a wide range of components but not all components. For example, liquid-cooled components share piping properties such as fluid type, loss coefficient and rupture pressure, while air-cooled components share HVAC properties such as air flow rate.
- Component-specific Properties are pertinent to a single component type or a very small number. For example, "resistance per unit length" is applicable to electrical cables.

Component properties that are only needed by S3D are stored as tool-specific properties; these properties are defined within the PMM but are stored under the S3D tool. The tool-specific properties are those properties that need to be persisted from one S3D use session to another, or that should be maintained with the model for future reference. For example, these would include operational settings for equipment such as switch position, valve position, or adjustable load power level that were used to run a simulation.

Component properties that are results of simulations are not stored in the LEAPS database, since these results may be re-created using the operational settings stored as tool-specific properties. Simulation results may be stored as .xml data within the file structure of the LEAPS database.

B. Node Data

Node properties that are expected to be used by other software programs are stored as properties on the respective nodes; for example, voltage on an electrical node. Similar to the component properties, these data can be common properties, shared properties, or node-specific properties, although no node-specific properties have been identified to date.

Node properties that are used only by S3D are stored as tool-specific properties; these properties are still defined within the PMM but are stored under the S3D tool associated with the appropriate node. For example, switch state (open or closed) for a switchboard node would be stored as an S3D tool property.

Node properties that are simulation results are not stored in LEAPS; an example is electrical current flowing through a node during a simulation.

C. Connectivity Data

Connectivity between components is stored within LEAPS using the LEAPS connectivity convention as follows:

A component may have zero, one or more ports associated with it.

A port may have zero, one or more terminals associated with it via port-terminal linkages; each terminal may be associated with only one port using one port-terminal linkage. A port-terminal linkage has two members: a port and a terminal.

Port-terminal linkages are connected to one another via Exchange Connections. An exchange connection has two members; each member is a port-terminal linkage.

Components are connected together using Component Exchange Connections. Component exchange connections have three members: the two components and the exchange connection that links them. Components may be associated with multiple component exchange connections, for they may be parts of many different systems.

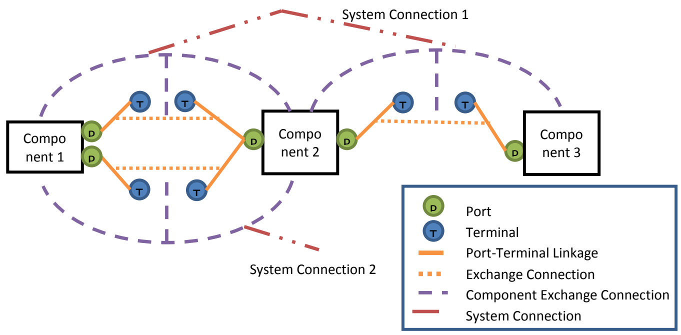

Figure 4. LEAPS Connectivity Terminology Example

Component exchange connections are linked together via System Connections. System connections may include many component exchange connections, and a single component exchange connection may be part of multiple system connections. The system connection is the root connection for a diagram.

Figure 4 shows a connectivity example with three components linked by two different system connections. System connection 1 has two component exchange connections (red dash-dot line); the one on the left connects Component 1 to Component 2 using the top component exchange connection (purple dashed line) which has as members component 1, component 2 and an exchange connection (black dashed line). That exchange connection has as members two port-terminal linkages (orange solid lines). The one on the left connects a terminal (blue circle) to a port (green circle) associated with Component 1. The one on the right connects a terminal to a port; this port has a second terminal associated with it, but that terminal is in a different system connection.

VI. Summary

This paper presents an update on the progress of the development of the Smart Ship Systems Design (S3D) development environment and of the integration of S3D with the LEAPS family of tools. Specifically, the following have been produced:

An initial desktop version of S3D that uses LEAPS as the underlying data repository was produced and distributed in 2016; this version is under update to employ the application framework and to improve FOCUS compliance.

A data dictionary that clearly delineates the required changes to the FOCUS model was produced. This vetted document provides definitions of terms and data storage requirements.

A set of use cases for employing S3D in early-stage ship design have been developed, ranging from development of ship-specific system designs, through implementation of pre-designed templates, to ship-independent system explorations.

The methodology for storage of system data proposed in [6] has been revised and implemented.

This paper addressed these advances and outlined future work required for the complete integration of S3D with LEAPS.

Acknowledgment

The authors would like to acknowledge the contributions of Larry Leibman and Rick Carlstrom for the initial strawman coding of the LEAPS Application Framework version of S3D and Norbert Doerry for his insight especially into the concepts of Templates and Patterns.

References

[1] B. Langland, R. Leonard, R. Smart, R. Dougal, "Modeling and data exchange in a concurrent and collaborative design environment for electric ships," in IEEE Electric Ship Technologies Symposium, IEEE ESTS 2015, Alexandria, VA, June 21-24, pp. 388 - 394.

[2] M. Ferrante, J. Chalfant, C. Chryssostomidis, B. Langland, R. Dougal, "Adding Simulation Capability to Early-Stage Ship Design," in IEEE Electric Ship Technologies Symposium, IEEE ESTS 2015, Alexandria, VA, June 21-24, 2015, pp. 207-212.

[3] D. Rigterink, R. Ames, A. Gray, N. Doerry, "Early-Stage Assessment of the Impacts of Next Generation Combat Power and Energy Systems on Navy Ships," ASNE Advanced Machinery Technology Symposium (AMTS) 2016, Villanova, PA, May 25-26, 2016.

[4] P. Sullivan, "Ship Design and Analysis Tool Goals", NAVSEA Memorandum Ser 05D/047, February 2008.

[5] A. Mackenna, ''Rapid design and integration (RDI),'' presented at the NDIA Conf. Physics-Based Modeling U.S. Defense, Denver, CO, USA, Nov. 6–8, 2012.

[6] R. Dellsy, M. Parker, D. Rigterink, "Multi-Scale, Interdisciplinary Systems Analysis for Naval Platforms." Naval Engineers Journal, 127(2), pp. 93-100.