Ship CAD systems – past, present and possible future

Nick Danese

1. Introduction

Since the dawn of time, visualization has been used by humans for communication. Visualization techniques have been extended over time to represent complex conceptual information in a generally understandable fashion. Drawings can communicate many types of information: shapes, quantities, sizes, procedures, methods, logical data, planning, implementation, desire, intention, etc. For ships, only drawings of the finished product remain from the early days of ship design and building. But this is not to say that plans, however rudimentary, were not actually drawn.

In ship design, records of ship drawings appear in the late Middle Ages. They seem to have established themselves as a standard document a little later and have since remained the one universal way to communicate all types of information. Then, at some point in the 1960s, computer technology gained recognition. It started replacing the tools used to draw and calculate, but not the drawings themselves.

Abstract

The evolution of computer technology employed in ship design and building is traced from pre-graphic to current database-centric data management systems. The four most employed ship modelling techniques are presented, and their contribution to design, production, and PLM (product life-cycle management) phases described. The contributions to the ship model by each method are reviewed, and a comparison made with technologies, methods and processes in other industries. Possible evolutions of current approaches are discussed.

Die Entwicklung der Informationstechnologie im Schiffsentwurf und Schiffbau wird von nicht-grafischen zu heutigen datenbank-zentrierten Systemen nachvollzogen. Die vier derzeit wichtigsten Ansätze zur Schiffsmodellierung werden vorgestellt und ihre Beiträge im Entwurf, Produktion und PLM (Produkt Lebenszyklus-Management) beschrieben. Ihre jeweiligen Beiträge zum Schiffsmodel werden diskutiert und ein Vergleich gezogen zu Techniken, Methoden und Prozessen in anderen Industrien. Mögliche zukünftige Weiterentwicklungen der heutigen Ansätze werden diskutiert.

Verkürzter Nachdruck eines Artikels der COMPIT 2010 (www.compit.info)

2. A brief history of the use of computers in ship design and ship building

This paper is not intended to exhaustively document the technological development over the past decades in the fields of computer aided ship design and ship building, but rather to highlight is providing enough background to put the conclusions into context.

Although television had been around since the mid-1920s, early computers were in general exclusively dedicated to computation, kept in specific computer centres, and offered no easily available graphical interface or display until the late 1970s. Some may recall the blind punch cards on the DEC-10 terminals, the 1978 IBM advertising »we take the computer out of the back room«, and the 1980 arrival of the awe inspiring Tektronics green on black, very high resolution display.

Petroglyph of canoe with crew and paddles, ca. 8000 B.C.

Source: US Forest Service, Tongass National Forest, Alaska

The lack of a graphical user interface (GUI), video graphics or computer generated pictures probably dictated the birth of CAD as a logical modelling tool. Software development for CAD took place concurrently in several countries around the world, and some of those programs continue to be developed and remain commercially available today.

As a measure of the speed of development in the field of computers, by 1983 touch screens were used to create and edit distributed systems models, and in some countries desktop computers had become almost commonplace. This is the time when a first fork in the road appeared: one way led to the much cheaper desktop computers with their small screens and fast growing peripheral accessory list, the other led to the far more powerful mainframe (and subsequent still quite large mini computer) and its significantly fewer peripherals. Concurrently, easy-to-use modem-accessed databases, early (internet) search engines, spreadsheets, and actual drawing programs became widely available. Crash-proof laptops with solid state memory, integrated modem and network capability were only a few months behind. Shoebox size ink-jet colour printers appeared in 1987.

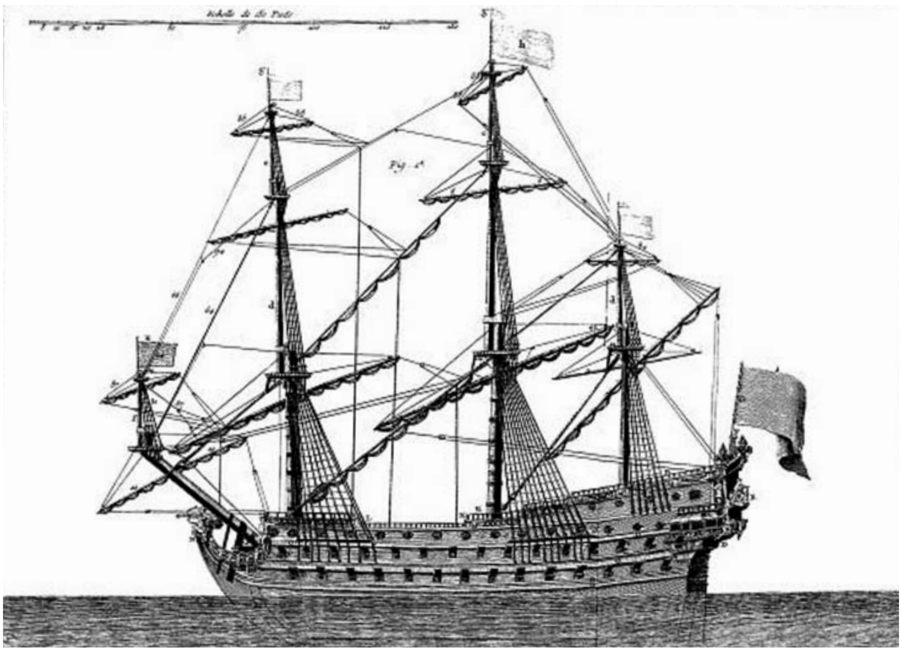


Planche de Marine, one of several illustrations from the Encyclopédie de Diderot et D'Alembert, 1765, http://diderot.alembert.free.fr/

Tektronics green-on-green 4000 dpi monitor

Source: University of Waterloo

The HP110 second generation portable computer set includes ink-jet printer and portable disc drives for a total weight of under 4 kg and several hours battery autonomy

By the late 1980s, the choice of how to use the now commonplace graphical environment created a second fork in the road: one way led to CAD-based modelling systems, the other to those using graphics as the display of logical modelling systems.

Another significant factor driving the development of certain programs was the poor quality and limited size of printed output. While computer generated drawings could be of lesser graphical quality than handmade ones without significantly impacting the industry, cutting machines required far better. Hand tracing and cutting, as well as hand-drawn sheets for optical cutting machines were replaced by transferring the cutting instructions directly from the computer to the machine, hence the birth of NC machines. The drawing was eliminated from the process, probably the only such instance.

3. The four most used methods to create ship data models

Today, four methods seem to be the most used in creating ship data models: plain drafting, logical modelling, CAD-supported logical modelling, and logic-supported graphical modelling. Some continue to exist since the early days; others have been born of advances in technology, graphical display technology in particular.

Due to the widespread and heterogeneous use of the terms, before discussing the methods in themselves, it is appropriate to define what the terms »data model«, »CAD«, and »logical« stand for in this paper:

- Data model: a collection of data and information defining, describing or otherwise referring to one object, or collection of objects, such as a ship.
- CAD: software, methods and technology in general providing functions and tools used in the creation or generation of graphically representable, original data model compo nents. CAD modelling is carried out using an interactive CAD environment, such as drawing programs, graphical libraries, etc.
- Logical: definitions or entities related to object, generally non-graphical, such as attributes, properties, relations, associations, etc. Logical modelling is carried out using macros, scripts and other generally non-graphical methods.

3.1. Plain drafting

Plain drafting is just that. This method consists of drawings, essentially in pure 2D. It was the only method existing before the inception of computers, then completely manual. Today it is generally implemented via drawing programs. While data and information are rather limited in scope, drawings are in general very portable and can be shared among different software programs.

3.2. Logical modelling

Logical modelling involves describing data and information (e.g. objects, their properties and attributes), using text organized in scripts, macros, tables, syntactically coded files, etc. This was the only computer based method before video displays became common. As an example, consider the definition of a line by a name and its end point in absolute coordinates:

Line »name of line« x_1,y_1,z_1 x_2,y_2,z_2

For a line called »edge of beam« and going from 0,0,0 to 1,1,1, this would read:

Line »edge of beam« 0,0,0 1,1,1

It can easily be seen how a variant of the command would allow the bounding of one object by another object, etc. Logical modelling requires an intimate knowledge of the overall model and sharing of a lot of information with others working on the same model to avoid duplications, misrepresentations and other errors. Logical models are virtually impossible to share among different programs, even if they are otherwise identical in scope.

3.3. CAD supported logical modelling

Here, the modelling takes place in a CAD environment, for example drawing, solid modelling, etc. The software program sources data and information directly from the CAD interface and / or model itself, the best programs work interactively and transparently, and integrate the harvested information with other non-graphical information and data.

Software implementing this method may use a database for storage of data and information, or use the CAD files themselves for storage. While a database is less portable, it is far more robust and reliable. Moreover, a database can be easily interfaced with other data management systems, while CAD files hardly lend themselves to such practices.

3.4. Logic supported graphical modelling

This method involves logical programming, providing a rather comprehensive display of the defined entities. Modern logic supported graphical modelling sometimes allows the direct use of graphical entities, once they have been defined logically. This way, the software helps in the writing of macros and scripts.

4. Contribution of the methods to design, production and PLM

All methods contribute to all phases of a ship's life, but the use of computers does not per se translate into better products or processes.

4.1. Contribution to design

In this phase, an important contribution is made by the immediate availability of data and information. In this respect, plain drafting and CAD-based modellers surpass other, more logical oriented methods. In addition to the availability of drawings by definition, drafting and CAD-based methods also generally support immediate conversion of drawings, data and information to neutral formats, for broad and easy sharing.

While more flexible in case of changes in advanced design phases, logical modelling oriented methods often require a significant amount of setting up and definition work before their models can be exploited and drawings, data and information can be produced. Drawings are also often produced in very rigid formats, which severely restricts further processing and exploitation.

Some CAD-based systems build a relational / associative model via the CAD interface itself, possibly providing the best of all worlds during this phase. Many advantages offered by certain methods are similar to those pertaining to the production phase, and will be discussed there.

4.2 Contribution to production

The production phase, often simplistically identified with detailed engineering, culminates with the modelling of each component of the ship that

- will be manufactured from stock (e.g. plate and pipe parts),
- requires a construction drawing for its fabrication (e.g. a railing or a ladder), or
- is to be sourced as a finished product (e.g. equipment and machinery).

The methods' features which most contribute to this phase include:

- propagation of repetitive and relational information, thereby logically linking similar and associated objects,
- ease of sharing data and information of all types with other programs,
- ease of integration of data and information coming from external, heterogeneous sources,
- ease of model creation and drawing production,
- ease of data and information management,
- ease of cooperative work,
- ease of data and information extraction and reporting.

Plain drafting will contribute the least to this phase, as it will require the largest amount of manual, error-prone and repetitive work, while providing neither data nor information management facilities, nor allowing any derivation of data and information whatsoever.

Interfacing or better still integrating CAD systems with other relevant data and information management systems (ERP, project management, etc.) is a major, so far under-exploited advantage offered by some systems. An open architecture, API endowed database is fundamental in this respect.

Another important requirement in this phase is the ability to track the ship's construction, integrate production feed back, data and information into the model's environment, and analyse it for a dynamic critical path review. This is doable with an open, extensible or interfaceable database.

4.3. Contribution to the PLM phase

This is the phase in the ship's life where most systems do not contribute as much as one would expect or hope. This is due to various factors, from the inability to provide the model, related data and information in commonplace or even open file formats, to the modelling software not being portable or requiring significant training to be used.

Even more than the production phase, PLM will benefit from the integral ship model, combining CAD, logical and non-CAD information via an open architecture, commercial database, which allows integration with other data and information management systems.

5. File formats, data structure, databases

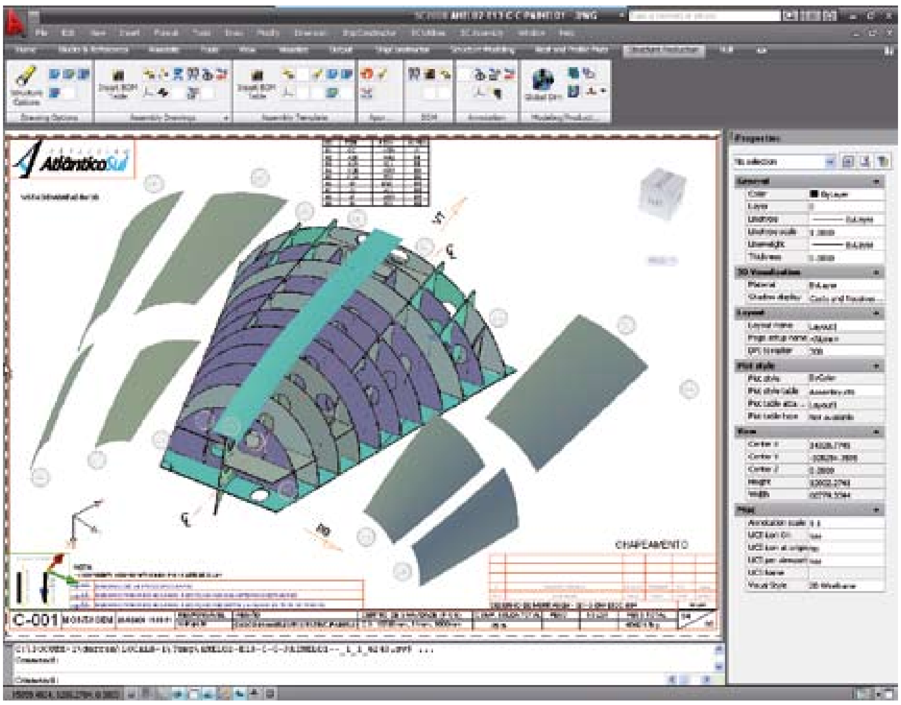
Due to historical reasons, some software development decisions remain heavily influenced by past necessities. The risk, or choice, is to remain isolated in a data communication and management sense.

File formats, arbitrary object definitions and hence the inability to share models among ship modelling programs are a quasi-insurmountable obstacle in the exploitation of the ship model by the end user, the ship owner or manager.

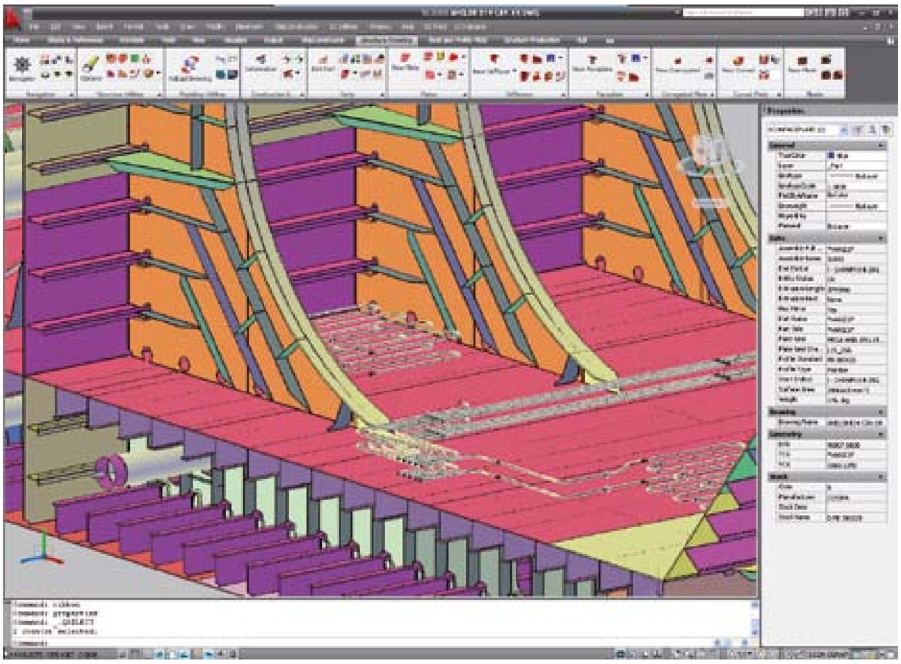
The pure drafting method does not even produce a model to speak of. Its final product (drawings) simply provides reference documentation, with the additional caveat that more often than not drawing sets are not kept up to date during ship construction and later ship modifications.

For ship modelling systems, object definition and file format are historically intimately related. In many cases the early development environment has remained fundamentally unchanged. Save for one or two more recently born software programs, ship modelling systems were developed at a time when CAD and video were either simply not commonly available or felt to be inadequate for the task, hence the perhaps forced choice of logical modelling already discussed. Moreover, until very recently, ship modelling applications were intended to serve the requirements of production only, with limited or no attention to design and PLM.

Great importance was placed on modelling and NC production of the steel structure, and lesser efforts dedicated to software for the modelling and production of distributed systems, which were instead handled



Bildlegende



Bildlegende

by software developed for the process and energy industries. Such industrial programs were often used in the ship industry, but not integrated with the structure modelling system. Distributed system modelling has also suffered from the poor cousin's syndrome; they seem to have unavoidably reserved the back seat when allocating resources or scheduling.

The lack of »standards« and the rather different cultural and technological environments in which software was developed led to programs performing the same tasks in a variety of incompatible ways. As the goal of ship modelling was reached with completion of the ship, model sharing or re-use was probably never even considered.

Over the years, modelling programs grew more capable and more powerful. The concept of pre-outfitting was born out of the need to lower costs and increase capacity through shorter building time. Greater sophistication often meant reduced flexibility in system adaptation, forcing the bridging of the gap between ship structural modelling and non-ship distributed systems modelling applications. This was achieved mostly by creating dedicated data exchange formats serving only two programs.

Many programs were sold commercially, and development continued along existing lines, increasing software isolation and losing any chance there may have been of catering to some sort of object definition standards.

Software using a database to store geometry, topology and other data, as opposed to files, could provide the environment in which to potentially create a model complete in all its components, or at least complete to the extents allowed by the available modelling capabilities. The importance and ensuing benefits of sharing models was eventually recognised, and attempts were made to create modern neutral file standards, such as STEP, an exercise which – after several years of work – has not yet produced an industry-wide practical outcome. Today, model sharing is limited to some programs exporting and / or importing solid or similarly fashioned geometry, using CAD standards such as ACIS or Parasolids. However, no data or information is shared in this way. At least one program goes a few steps farther, by acquiring native geometry and, if exposed, corresponding logical information. The intention to create the all-encompassing model was the birth of the much abused »product model« punch line, and a true result has yet to materialize.

6. The model

The »model«, or to evoke the far more fashionable definition »product model«, has been referred to a lot in this paper, but different readers will have attributed quite different meanings to this label. One general definition could be »a CAD model which includes all the parts making up the ship«. However, »CAD« and »all« remain very generic ways of defining anything, let alone something as complex as a ship. A recent, even richer sounding, evolution of this definition is »building information model«.

6.1. CAD data

»CAD« could be anything from a simplified sketch to a fully detailed, realistic, 3D model and »all« generally refers to those portions of the ship which the software being considered can model, in whichever way it can. »Information« is also a purpose-suited word, often used to convey the impression that there is more than one will really find, particularly in that the generally limited data set available tends to pertain solely to a sub-set of the »CAD« portion of the model. So, in order to avoid the debatable accuracy of many claims made in the commercial world, the less-specific but potentially more encompassing definition »ship model« is used here.

Moreover, commercial promotion of software products and ground-breaking research tends to highlight the contents per se of the model and largely omit any consideration of the availability, usability and specific usefulness of such contents aside from the generation of only a part of the finished ship.

6.2. Non-CAD data

Stopping short of proposing to integrate the model with all the even remotely related available data in existence during the design, building and PLM phases, there still is a remarkable amount of non-CAD information which could and should be combined with the CAD model, within the model itself or to produce a non-CAD, integrated data set.

ERP is the most evoked such data set, but a lot more is already available and commonly in use, albeit separately from the ship model. Examples include production feedback, production analysis, critical path forecasting and analysis, cost estimation and analysis, contingency planning, etc. A lot of this data is directly related to the design, production and PLM phases, but it is not directly attributable to the contents of the CAD model. On the other hand, non-geometrical relations can easily be established and exploited between CAD and non-CAD objects, data and information.

6.3. The ship model

Combining the ship's CAD model with related non-CAD data would yield the »ship model«. In turn, the ship model could be the live, integral part of a wider model. Imagine for example a database of up-to-date ship loading and unloading capabilities, current machinery breakdowns, etc., being used to schedule port activities.

So, let us now use the term »ship model« to represent a collection of ship-related data including CAD, CAD-related logical, and non-CAD-related data, without restrictions.

7. Relevance and role of ship model components

Throughout the discussion so far, the components of the ship model have been loosely collected into three very broad categories: CAD, logical and non-CAD. This seems to work well, and lends itself as an effective data set classing approach identifying three categories. We could then assign attributes to each category, such as »time critical«, »static«, »dependent«, »independent«, and so on.

7.1. CAD components

CAD components can be attributed all the attributes listed above, for example:

- time critical: a drawing, or a computer model used to derive a drawing, must be completed before a certain date in order to not hold up the classification process or even the building of the ship.
- static: once issued, a drawing is considered to be static; one needs not further verify its contents.
- dependent: the contents of a drawing or the objects making up a model are related and will influence each other.
- independent: certain CAD quantities are independent of other data. For example the breadth of a Suez-max ship will not change even if a greater load carrying capacity were required and catered to.

The role of CAD is to show, symbolically or realistically, what the product being represented is to look like.

7.2. Logical components

Logical components also can take on all the attributes listed above:

- time critical: the power and torque curves of an engine will influence engine room design and propeller selection. The choice of the engine is therefore a time critical component.
- static: a given object can be mounted only in one way. This orientation will not be changed by anything else in the ship model.
- dependent: a weld detail will depend on the thickness and metal grade of the components being welded. Weld detail is a dependent component.
- independent: for a given ship, the wattage of the ship lights will remain unchanged irrespective of anything else.

Mostly due to the actual implementation scheme in software, the role of logical components is mainly limited to qualify CAD objects, although their role is actually far more varied and far reaching.

7.3. Non-CAD components

Non-CAD components also can take on all the attributes listed above:

- time critical: knowing which surface treatment must be applied to an object must be known in good time for the provisioning of the appropriate supplies and for the timely availability of that object during construction. Supply of the required consumables and ability to apply them is time critical.
- static: suppose an object can be delivered by rail only. The railroad's traffic schedule, sometimes fixed months in advance, is a static component.
- dependent: the material grade of a plate will depend on the thickness it can take at the location where it will be placed, and on the load it will be carrying. For example, consider a thin, heavily loaded plate which must therefore be of high-tensile steel. The ability to actually mount the plate depends on the availability of welders qualified for this type of metal.
- independent: something as simple as production feedback often influences what goes on in the technical office and beyond, during the production phase itself. Consider the case of a fabrication gone wrong due to a design mistake, and the ensuing need to re-design it, re-source or re-build its parts, and re-fabricate it.

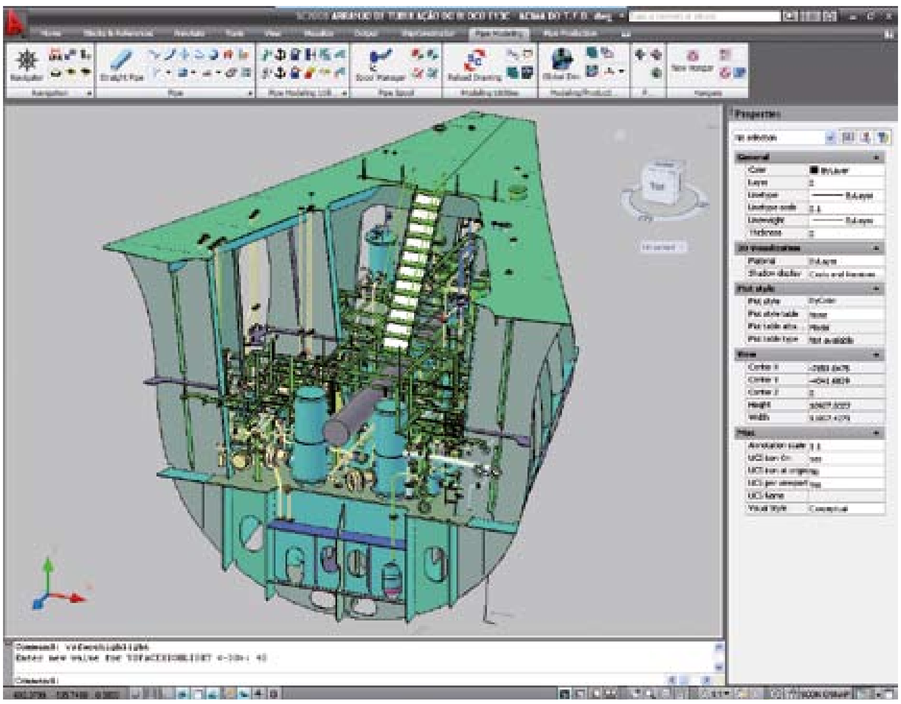
The role of non-CAD components is to add »external« data and information to the model which will influence any of the ship's phases, hence the CAD components as well. Anyway, the purpose of the exercise above is not to subdivide components into intersecting groups or classes, but rather to simply show how any of the above carefully worded definitions could be plausibly changed just slightly and that component becomes assigned to another group or changes attributes.

Possibly with very few exceptions, every component of a ship model is relevant and potentially of critical importance due to the deep interaction between all components, direct or indirect. Therefore, a ship model should include the greatest quantity of components from the categories mentioned above and they should be accessible to anyone needing that information, in as neutral a format as possible in order to avoid needing to use the native software application.

8. Other industries

Methods, processes, analyses, etc. are all different ways to describe tools, or means to reach a goal. Tool is a convenient, modern word, and will be used here as the generic definition of choice. Many examples can be brought forward to illustrate methods being used in other industries that would benefit the ship industry via the creation and management of the ship model. But in the end, it is easy to condense them all by drawing an exquisite analogy with the search engine, a tool that permeates our modern world much deeper than most realize.

The very concept of the search engine provides the link between the opening discussion on communications, the processing of information during three main stages in a process (very early planning, under-way and emergency) and the varying nature of the data itself. This ancient tool, already implemented in Babylon via clay tablets and much later with card indexes, can provide significant help far and beyond its current implementation, and this without requiring revolutionary computers or other not yet existent technology.



Bildlegende

8.1. Very early planning

During planning, use is made of uncertain or estimated data and information. Provisions are drawn and safety margins applied. The lesser the certainty about what is thought to be known, the greater the spectrum of possible outcomes is, and the greater the potential deviation from the objective goal. At this stage, the ability to identify and quantify information and its quality is crucial: estimated data can be used effectively if a measure is known of its presumed accuracy, inaccuracy, and impact. Ship weight estimation is a good example of this.

Drawings, low-detail CAD models and macroscopic data analysis are easily identified characteristics of the planning phase. In some circles, this is known as the concept or feasibility phase. This is when fundamental mistakes can be made. Let us use as an example the industrial process which cannot yield the product at the forecast cost, a surprise discovered well after production got under way.

8.2. Under-way

This is when everything is supposed to proceed smoothly, according to plan. Once a process is under way, the main activity is monitoring and checking. Once again, drawings, more detailed CAD models and data analysis at various levels of detail are used to assess progress and the quality thereof.

8.3. Emergency

Many situations qualify as emergencies, but the common denominator is generally the immediate need for all potentially relevant data and information to be available and exploitable. Much can come from the CAD model, much cannot and will not. So, once again, drawings, CAD models, non-CAD data and data analysis tools will be sought after. Qualification of the need to have data and information of all types, have access to it, and be able to exploit it effectively and in a way useful to everyone needing it will be the last factor supporting the assessment and vision to follow.

9. What can be done today

A lot of technology exists today that can be put to use in creating, maintaining and exploiting the ship model: drawing software, 3D modelling software, open-architecture databases, etc. And, not least, the search engine analogy used above is easily applied in reverse, when dealing with input to the model.

Input can take many forms, and data can be of disparate nature: drawing entities, logical attributes and properties, non-CAD information, etc. Then, if input were identified, its relations to other objects in the model could be established, regardless of their respective nature. Logical modellers and some CAD modellers work this way, but input of non-CAD or not CAD-related data is often hampered, if not impossible. So, input facilities are an integral part of the data management process. Graphical input techniques are probably by far the easiest and most intuitive channel, allowing direct, transparent input identification and management.

Let us consider the search engine again. The »finds« returned include both exact and approximate matches, as the search engine will purposely use inaccuracies to cater for what could be loosely defined input search criteria. (The engine is programmed to operate in a stringent fashion). Nonetheless, it is easy to gather substantial amounts of relevant information from a vast, unformatted amount of source data. In the case of a ship model, the environment is far more regimented than the contents of the internet. This favours cross-matching and correct connections between objects, or groups of objects. In modelling work, stringent rules must be followed when defining objects, and even more so when defining dependencies and relations. In a given planning model, representative data will use fixed formats: dates, locations, etc. The list of examples of what is achieved already thanks to formal formatting is long: bar codes, report table formats, time sheets, ISO procedures, etc. These being known quantities, the following would constitute a constructive step in assembling data from different sources and exploiting it in many ways:

- having the data
- the data being available and accessible
- the data being readable and interpretable
- having information about data interconnection (this data pertains to that object but only on the day of the week defined in table so and so …).

All the above exists today, but rarely as a useable, complete set. There is no reason today to justify the persistence of such a restriction. Of course, as seen above, another factor is that in many cases similar (or even identical) objects will be defined in very different ways, thereby making the connection virtually impossible. This places an objective constraint on data sharing and exploitation, but one that can be circumvented and that need not persist either.

Search engines have been used every day since the existence of computers, for example to find a file containing a certain word. Today, there exist sophisticated query tools, file searchers, automated data assembly tools, etc. that can be employed, if the source data is available and can be searched thanks to documented formatting. So, a more effective exploitation of existing data and information, of varying nature, purpose and scope, is technically possible today, and would be more commonplace if source data were richer, more easily available and easier to access.

10. Conclusion

If the present paper were to be condensed in a few statements, these could be:

- while not a means for systematic or automated data and information management, drawings remain a fundamental corner stone in making information available to the real-world, wider ship audience, at all times, information without which not much would happen.
- non-CAD data is a dangerously overlooked, vital component of the ship model that can and should be used throughout the various phases of a ship's life. Combining and connecting this data with the CAD model provides a quasi-universal means of making it known, understood and exploitable.

Achieving the common goal of these two statements, e.g. making relevant information available in a way that is meaningful to the recipient, at all times starting from the earliest stage of concept design and until the latest stage of dismantling, requires software which easily allows:

- input of appropriate and relevant data and information, be it graphical, logical or non-CAD, via standard, non-proprietary interfaces or data tools.
- exploitation and generation of model-associated drawings, which must remain available for further formatting using standard, non-proprietary CAD tools.
- extraction of CAD related logical data for analysis, reporting, etc., also combined with non-CAD data, via standard, non-proprietary data management tools.

The software and hardware needed to accomplish the above exist today, commercially off the shelf. Some ship CAD systems already provide the facilities, if not the tools, identified in this paper as one avenue towards industry-wide exploitation of the existing, rich, ship data sets.

Abbreviations:

API Application Programming Interface
CAD computer aided design
ERP Enterprise Resource Planning
ISO International Standard Organisation
NC numerically controlled
PLM Product Life-Cycle Management

Author:
Nick Danese, Nick Danese Applied Research, Antibes/France
ndar@ndar.com