Proceedings of the 2008 Industrial Engineering Research Conference
J. Fowler and S. Mason, eds.

Improving the Navigation and Information Integration of Complex Process Models: A Case Study Using IDEF0

Melissa R. Hastings and Kenneth H. Funk II
School of Mechanical, Industrial, and Manufacturing Engineering
Oregon State University, Corvallis, Oregon 97331-6001, USA

Abstract

Process, or functional, modeling facilitates design and analysis of systems by providing a structure for communicating the interaction of information, resources, and activities. Automated modeling tools ease the development of complex process models and assist in navigation but there remains the problem of navigating between key areas while maintaining a detailed and contextual understanding of the relationships of interest. The web-based prototype described in this paper provides an alternate display of IDEF0 models, allowing the user to investigate and navigate low-level processes, while displaying high-level contextual information.

Keywords

Process Modeling, Functional Modeling, IDEF0, navigation of complex representations

1. Introduction

A need exists to allow users to explore detailed process models and develop meaningful conclusions without encumbering them with many diagrams from which to remember and integrate data. This results from the basic concept of process modeling, which relies on expanding processes into their sub-processes, resulting in an exponentially expanding number of processes within the model. This paper discusses a proposed solution, in the form of a prototype, which addresses this need. The prototype uses the IDEF0 modeling standard and allows users to examine the interaction of any process with low-level processes while still showing the larger context of the model. Review of process modeling literature has shown no previous efforts in alternative navigation, display, or visualization methods for complex models.

2. Process Modeling

2.1 Applications and Advantages

The literature regarding process modeling, sometimes referred to as functional or systems modeling, spans many applications and has a varying definition. In this paper, the focus is on the modeling of a system of processes that transform a given input into an output. Process modeling has many applications, from modeling of business practices [1] to simulation of health care implementation [2] and from the interdependencies of national infrastructure [3] to the interactions of web applications [4]. Process modeling is used in all of these applications for several reasons. First, it provides an unambiguous description of the system in question, helping to identify requirements and functionality. Second, the interactions between processes can be clearly defined and interdependencies investigated. Finally, it provides a basis from which to design improvements and modifications as well as build comprehensive documentation of the system.

2.2 Limitations

While process modeling has been used successfully in many situations, it has several limitations. Depending on the method used, there can be a significant time investment required for training users; both for generation and interpretation of the model [5]. The necessary time investment can be reduced somewhat with the use of automated modeling tools, allowing the model creation process to focus on information gathering as opposed to formatting. Another limitation occurs with the incorporation of more detail into the model. While this allows for a richer and more realistic model, it causes exponential growth of the interactions captured by the model. The results can quickly become unmanageable, crippling the utility of the model.

2.3 Focus for Improvement

The ability to investigate highly detailed, complex process models and focus in on key, low-level processes without losing their context is poorly supported by current representations of process models; especially those dependent on hierarchical process decomposition. This functionality would enable users to investigate low-level processes with the confidence that they are viewing the big picture, as far as it concerns the current process. Identification of this deficiency is what instigated the development of this prototype along with the concept of allowing the user to customize their view of the model and explore it in a more natural way.

3. Prototype

3.1 Modeling Methods

Development began with an investigation of how current process models and system hierarchies are displayed. It became apparent that the key was to find a method for the user to navigate within the model; to create custom and temporary system boundaries. Ideally, the user would not need to leave this alternate model view to navigate to different processes, allowing them to build a mental model of how the processes interact. Also this investigation lead to the conclusion that IDEF0 was realistic choice for the process modeling method on which to build the prototype. Of primary importance was that the IDEF0 standard produces a model whose notation was standard enough to translate easily into a normalized database [5] [6]. Additionally, the IDEF0 standard has been established in industry for well over a decade and has been successfully implemented in a wide variety of applications.

3.2 Implementation

Several software packages are available to assist in the development of IDEF0 models. Three of these were reviewed (iGrafx, System Architect, and AI0Win) to determine whether they would be useful in generating a database over which the prototype could be built. AI0Win was selected for the ease in which an IDEF0 model may be generated and for the fact it exported a human-readable database for the complete model.

Using a structured query language, two relational lists were generated for the prototype. As seen in Table 1, the first is indexed by process and contains three categories: sub-processes (if any), processes which produce the incoming information or materials, and processes which use the outgoing information or materials. The second list is of linked originating and destination processes with a sub-list of all of the information and materials that link them. There are additional look-up tables for the detailed names of all the processes, which are displayed by the prototype.

Table 1: Excerpt of primary relational list

A2:	A21, A22, A23, A24	A0I, A11, A12, A13, A14, A15, A16	A3111, A3112, A3113, A3131, A3132, A3133, A3134
A21:		A0I, A11, A12, A13, A14, A15, A16	A23
A22:		A0I, A15	A23
A23:		A0I, A12, A13, A14, A16, A21, A22	A2411, A2431
A24:	A241, A242, A243, A244, A245	A0I, A11, A23	A3111, A3112, A3113, A3131, A3132, A3133, A3134
A241:	A2411, A2412, A2413, A2414	A0I, A11, A23, A242, A244, A2451, A2452, A2453, A2454, A2455, A2456	A3111, A3112, A3113
A2411:		A0I, A11, A23, A242, A244	A2414
A2412:		A0I, A11, A23, A2451, A2452, A2453, A2454, A2455, A2456	A2414
A2413:		A0I, A11, A23, A242, A244	A2414
A2414:		A0I, A2411, A2412, A2413	A242, A244, A2451, A2452, A2453, A2454, A2455, A2456, A3111, A3112, A3113

The interface was built as a web application to maximize its usability and portability. The implementation uses JavaScript to build an HTML table in real-time for the current view requested by the user. For a copy of the current prototype code, please contact Kenneth Funk.

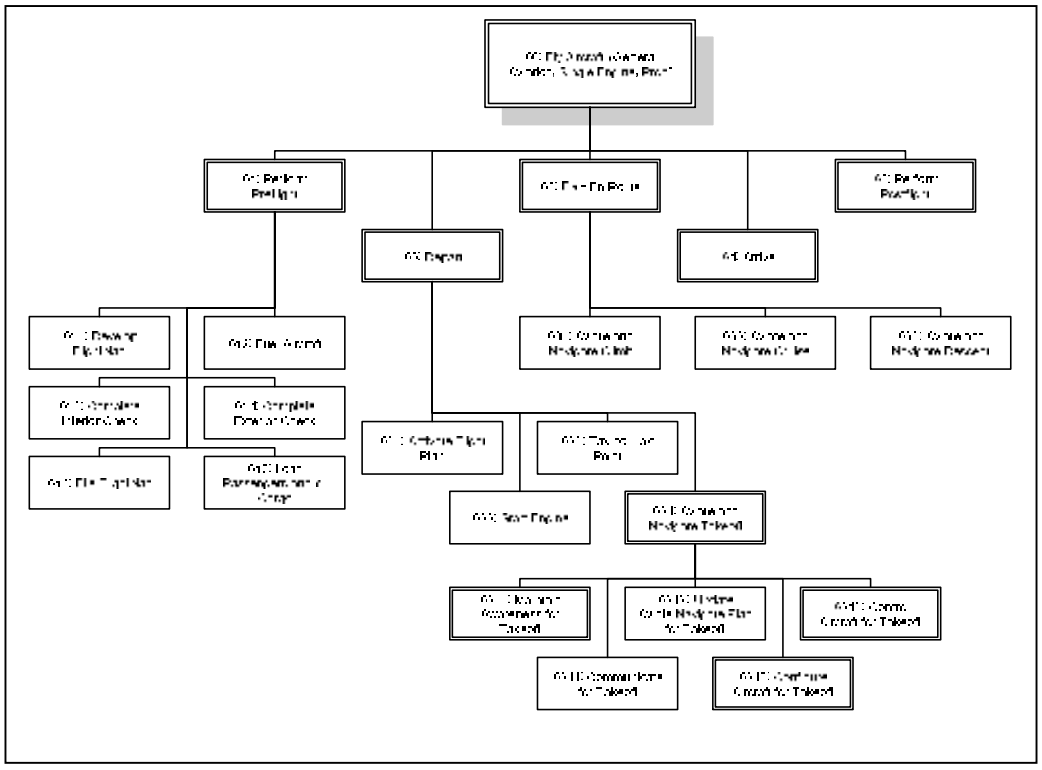

Figure 1: Excerpt of node tree diagram focusing on the *A2: Depart* process and its sub-processes.

3.3 Functionality

The prototype functionality is best explained in the context of a model. For this, a model of flying a non-commercial aircraft will be used and a portion of the node tree for this model is shown in Figure 1 giving the hierarchy of relationships of the processes. Figure 2 is the IDEF0 diagram for the *A2: Depart* process showing its sub-processes as the items within the boxes. For additional information on reading IDEF0 diagrams as well as the complete standard, please visit <*http://www.idef.com/idef0.html*>.

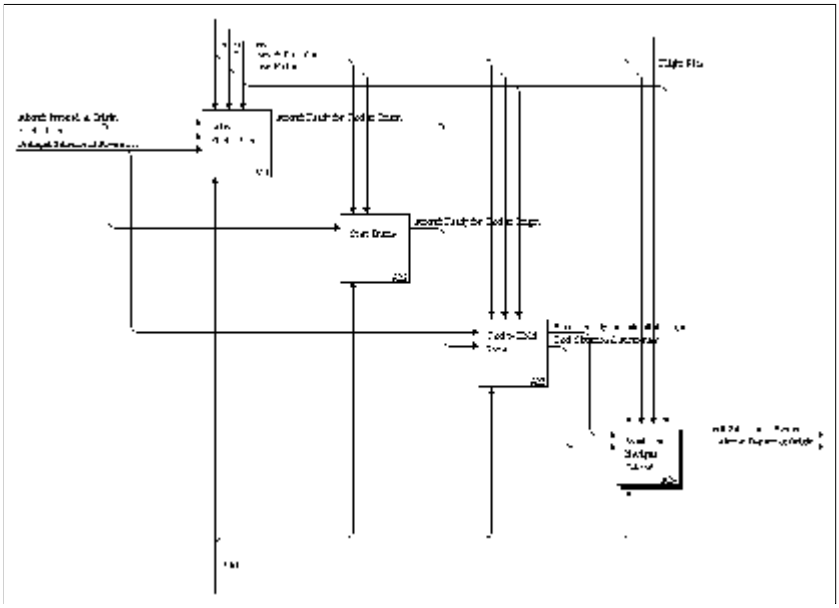

Figure 2: IDEF0 diagram for *A2: Depart* showing the *A24: Aviate and Navigate Takeoff* process.

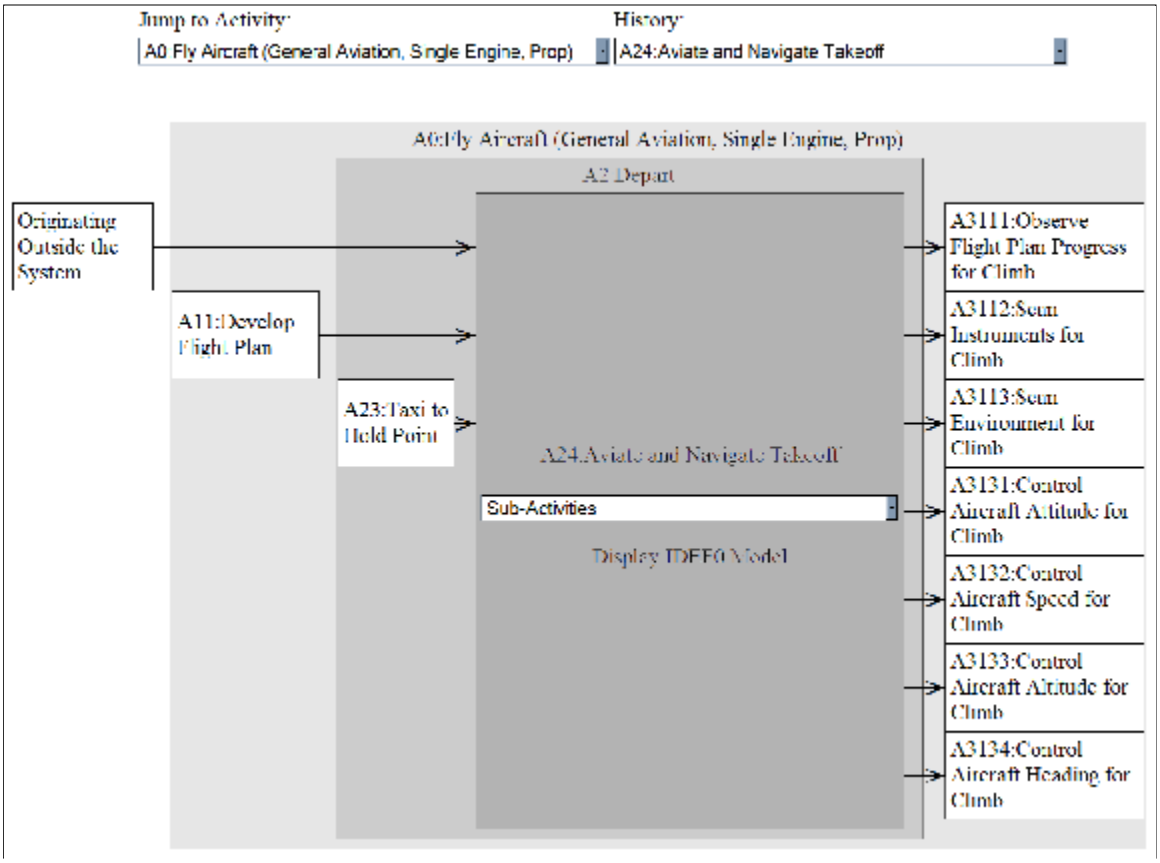

Figure 3: Screen capture of the prototype focused on the process *A24: Aviate and Navigate Takeoff*.

Inspection of the *A24: Aviate and Navigate Takeoff* process shows that there are multiple sources of information and materials for the process but it is not clear where they originate; to determine this information, a user would need to examine additional diagrams. The same is true of the destination processes for the information and material generated by the process. Contrasting this with the prototype as shown in Figure 3, a user can concurrently view the origin of all materials and information and their final destination. Additionally, the user can view the higher level processes for the current process, as shown by the nesting. This nesting also gives an idea of the distance between two processes since all processes only are contained within processes that are their parent. This can be seen with *A11: Develop Flight Plan* and *A23 Taxi to Hold Point* in Figure 3.

Several forms of navigation within the model are allowed. The first is achieved by clicking on any process title within the display, allowing the user to transition between processes that are linked by the flow of materials and information. If the user were to select the *A23: Taxi to Hold Point* process the view displayed by the prototype would shift to that shown in Figure 4. In the same manner, all of the parent processes can be directly reached from the display with a single interaction.

The second method of navigation is using the drop-down box within in the current process, which contains its sub-activities. The final two methods of navigation are the two drop-down boxes above the model display. The *Jump to Activity* drop-down contains a sorted list of all processes within the model, allowing the user to navigate directly to any process. The *History* drop-down contains all of the processes the user has navigated to since the model was loaded into the prototype.

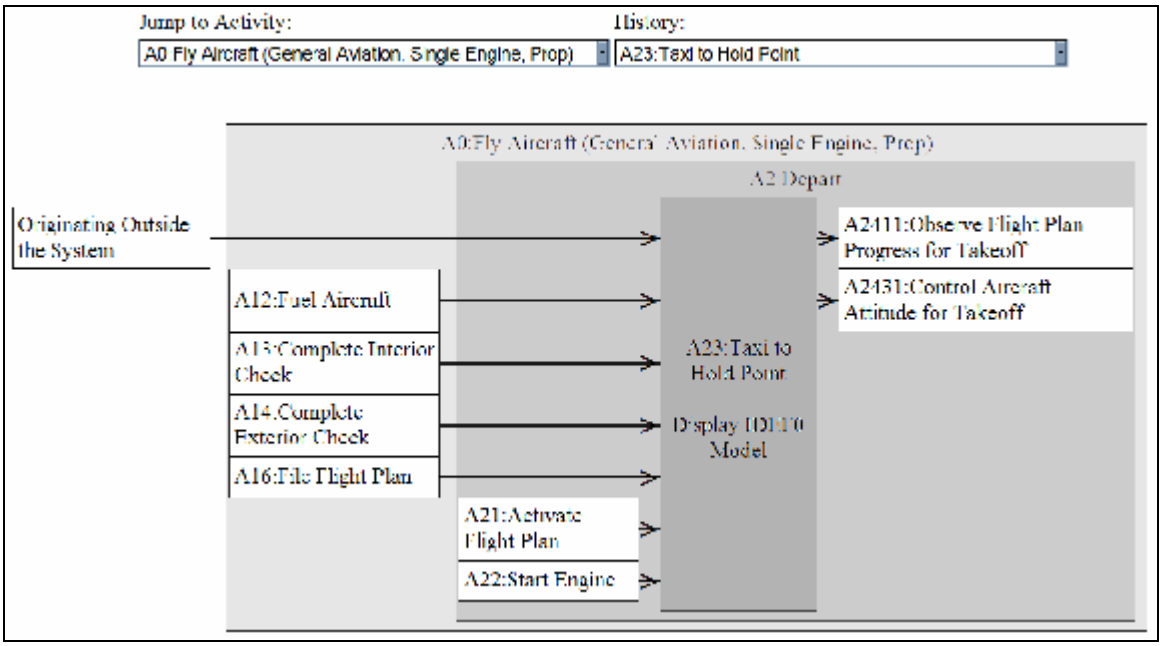

Figure 4: Screen capture of the prototype focused on the process *A23: Taxi to Hold Point*.

Another function that is not illustrated by Figure 3 occurs when the user clicks on an arrow connecting processes. This results in a report summarizing the information and materials that flow between those activities. For the case where the arrow from *A11: Develop Flight Plan* to *A24: Aviate and Navigate Takeoff* is selected, the report would reflect that the *Flight Plan* is an output of the *A11: Develop Flight Plan*, which controls, at least partially, the *A24: Aviate and Navigate Takeoff* process.

The final function available in the prototype is the ability to click the link within the current process and have the corresponding IDEF0 diagram displayed. For the display shown in Figure 3, selecting this link will redirect the user to the diagram shown in Figure 2.

4. Future Work

A validation study is currently underway with participants from a third-year Industrial Engineering class on Work Systems Engineering at Oregon State University. Their performance in navigating and interpreting an IDEF0 model will be contrasted between an electronic display of the standard model and the prototype as described above, while the underlying model for both displays will be the same. The tasks the participants will perform will vary from simple process identification to propagation of error analysis.

If the validation study is successful, the next stage will be to expand the prototype to allow for custom tagging of key processes and automatic report generation.

5. Conclusions

While the proposed prototype may not address all possible difficulties in navigation of process models, it does begin to allow the user to navigate the model in a cohesive and continuous manner. The greatest improvement is in allowing the user to investigate lower-level processes without losing the context of the model as a whole. With additional investment into development, this prototype could be expanded to allow users to identify key relationships and paths through the model resulting in an automatic report on their findings. Additionally, the prototype could be automated to identify feedback loops and other areas of interest.

Acknowledgments

Thank you to the IE 492 – Spring 2007 class for their feedback on the early version of this prototype.

References

1. Kim, C., Yim, D., and Weston, R., 2001, "An integrated use of IDEF0, IDEF3and Petri net methods in support of business process modeling," Proceedings of the Institution of Mechanical Engineers -- Part E -- Journal of Process Mechanical Engineering, 215 (4), 317-329.
2. Karakostas, B., Zorgios, Y., and Alevizos, C., 2006, "Automatic derivation of BPEL4WS from IDEF0 process models," Software & Systems Modeling, 5 (2), 208-218.
3. Min, H., Beyeler, W., Brown, T., Son, Y., Jones, A., 2007, "Toward modeling and simulation of critical national infrastructure interdependencies," IIE Transactions, 39(1), 57-7.
4. Jin, X., Kagioglou, M., and Aouad, G., 2006, "Towards a Dynamic Healthcare Process: From Requirement Capture to Simulation," Journal of Integrated Design & Process Science, 10(2), 1-19.
5. Malhotra, R. and Jayaraman, S., 1992 "An Integrated Framework for Enterprise Modeling", Journal of Manufacturing Systems, 11(6), 126–144.
6. Department of Defense, 1993, "Functional Process Simulation—A Guidebook".
7. National Institute of Standards and Technology, Computer Systems Laboratory, 1993, "Integration definition for function modeling (IDEF0): Draft Federal Information Processing Standards Publication 183: Department of Commerce".