Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 1997 Proceedings | Americas Conference on Information Systems (AMCIS)

8-15-1997

Computer Automated Reconciliation of Activity Models for Group Supported Organizational Modeling

Mark O. Pendergast
Washington State University, mop@tricity.wsu.edu

Computer Automated Reconciliation of Activity Models for Group Supported Organizational Modeling

Mark O. Pendergast
Washington State University
mop@tricity.wsu.edu

Abstract

Research at the University of Arizona and elsewhere have suggested that Electronic Meeting Systems augmented to support construction of Activity models can effectively support groups during development of semantic models. Group-enabled tools developed at the University of Arizona allow large groups of individuals from various functional units of an organization to actively participate in model development and to build models two to three times faster than traditional modeling approaches. Rapid model development shortens project times and makes it more likely that key personnel can participate during model development and analysis. To date, the tool set has been successfully used a number of groups for the collaborative development and analysis of business models using the IDEF0 Activity modeling methodology. This paper describes two software methods used to insure model correctness and shorten the model reconciliation process.

Introduction

Awareness of the need for business analysis has grown faster than the evolution of tools to support the collaborative development of business models. A number of authors have extolled the need for careful business process analysis and for alignment of process activities with current business objectives [2,7]. Appropriate models may support conceptualization, communication, analysis, and design of improved business processes and information systems. In addition, involvement of key personnel during model development and analysis is important for model accuracy as well as for buy-in [2]. Traditionally, however, models have been developed by individuals and small groups because of the complexity and difficulties involved when larger groups participate in the modeling process. As a result, models are often developed slowly and with limited, restrictive forms of participation.

Recent studies at the University of Arizona and elsewhere have suggested that an Electronic Meeting System (EMS)[8] designed to support construction of IDEF0 models can effectively support groups during rapid model development [3, 5]. Dean, Orwig, and Vogel [4] reported that the IDEF0 EMS tool described in this paper combined with appropriate facilitation methods allowed models to be developed two to three times faster than non-EMS supported groups and yet had comparable or superior quality. Moreover, significantly larger numbers of subject matter experts were able to actively and productively participate in model development and analysis than non-EMS supported groups.

This paper details ongoing research at the University of Arizona and Washington State University regarding a computer-supported collaborative modeling environment. Specifically, this paper describes two software methods used to insure model correctness and shorten the model reconciliation process. The next section provides some background on Electronic Meeting Systems and IDEF0 modeling. Following that is a brief review of the software prototypes and their potential.

Background

Electronic Meeting Systems (EMS) have shown promise in helping large groups contribute directly and productively during meetings [8]. In addition to verbal communication participants using EMS contribute and review information through networked computers. The primary benefits of EMS include: parallel communications, anonymity to support frank discussion of issues, and electronic memory that records meeting results. Traditional EMS tools including electronic brainstorming, idea categorization, and group vote have proven extremely useful in supporting business analysis teams. These tools, however, were not designed to support semantic model development. An objective of EA Project researchers at the University of Arizona over the last six years have been to develop EMS software tools and facilitation techniques that will allow a larger number of individuals to contribute directly and productively during semantic model development. During this time several collaborative tools using the IDEF0 modeling methodology were developed toward this end. The development of these tools was influenced by their use in controlled laboratory situations, by field testing them with groups, and by previous research on collaborative drawing tools [1,9].

The IDEF0 definition method is one of a number of methods commonly used to develop business process models. The term "IDEF" comes from ICAM (Integrated Computer-Aided Manufacturing) DEFinition language[6, 7]. IDEF0 is based largely on a well-established graphical language known as SADT (Structured Analysis and Design Technique). IDEF0 embodies a hierarchical structure so that content may be modeled at various and increasing levels of detail which helps manage model complexity. Business processes (activities) are decomposed into their component parts (children). A group of activities that share the same parent are referred to as a "sibling set". Inputs, controls, outputs and mechanisms (ICOMs) are specified for each activity. These ICOMs represent the relationships of information and resources between activities and from the external environment into the model.

Development of an IDEF0 model is an iterative task of defining activities, defining ICOMs, and reconciling errors and redundancies[6]. All activities are not identified at first. Rather, new activities may be identified at any time as a result of defining other activities or while identifying or defining ICOMs. A modeler may find that the same or similar objects (ICOMs) may be named differently in different parts of the model, and have somewhat different definitions. ICOMs used at one level of the hierarchy may have been omitted from or used in a different context at other levels. The IDEF0 methodology places limits on the number of activities for each level of decomposition(6) and the number of ICOMs any one activity can have (6 of each). Such problems can be especially challenging when parts of the model are developed in parallel by different individuals or subgroups. There are other syntactic rules governing the create of ICOM bundles (collapsing of detail), the use of "tunnels" for hiding detail, and the propagation of ICOMs from one level of the hierarchy to another. The task of establishing semantic and syntactic correctness is what we call "reconciliation."

Automated Methods for Model Reconciliation

The final result of a modeling session should be a fully compliant IDEF model. However, during intermediate development stages it usually necessary to deviate from the strict rules in order to allow participants to enter data freely. Allowing a certain amount of freedom also allows different facilitation styles to be used. Our goal is to minimize the distraction of restrictive enforcement of all IDEF0 rules and at the same time provide for enough guidance that problems can be resolved early enough. To that end we have built into our system two mechanisms for finding and correcting modeling errors. The first performs algorithmic tests on the model for violations of 28 different rules; the second performs a bottom up search for ICOM connections which were not propagated up to the parent decomposition.

The number of modeling rules that are enforced in a restrictive manner by our software prototypes is relatively small. For example, activity names must be unique. The vast majority of the rule tests were shifted into a "guided mode." That is, semantic and syntactic tests that are invoked by the user. Typically, a group or subgroup of users would work on one section of the model until it represented the activity in question. Then the analysis routine would be run to locate and correct modeling errors on that portion of the model. Each rule of the 28 rules can be individually enabled or disabled and the analysis can be performed on the entire model or any portion of it. This allows users to temporarily ignore certain types of errors during different stages in the modeling process. Rule violations are displayed in a dialog list from which the user may access help on each error, or select the "fix" button which will bring up the appropriate dialog for correcting the modeling error. Errors are also identified on graphical views of the model (decomposition diagrams, context diagrams, etc.) with visual markers.

The other mechanism our prototype supports is the propagation of ICOM connections up levels of abstraction. Since a parent IDEF0 diagram is a more general representation of its child decomposition, it is often helpful to see ICOM attachments at the child diagram level before creating the more generalized parent representation[6]. This is probably due to the fact that it is difficult for users with detailed knowledge of sub processes to think of aggregates of connections at the higher (more general level) of abstraction. In order to facilitate the definition of ICOM connections the prototype program allows ICOM connections to be created at any level of the decomposition tree and has an automated mechanism for propagating connections up the tree. The ICOM propagation function determines the missing ICOM connections for the entire model or any section of it. A list of connections is presented to the user who has the capability of paring down the list before committing them to the model database. The algorithm works by ordering existing ICOM connections according to sibling set and decomposition level. It then progress up the levels of decomposition looking for connections between sibling sets that are not reflected as single or bundled connections at the next higher level. As the algorithm progresses a list of missing connections is built. A decomposition tree containing 100 activities which are 50% populated with ICOMs would have approximately 600 different connections. If it takes a human 30 seconds to verify the "correctness" of a single connection, then five hours would be required to check the entire model! This would not include the time needed to create the missing connections, and recheck them.

Conclusions

A comparison of model development meetings supported by the EMS approach with meetings supported by the traditional, single-user tools has shown that the EMS approach provides considerable benefits. First, the EMS approach appears to break a significant productivity bottleneck associated with the traditional approach. The EMS supported approach allows a model to be built two to three times faster than the traditional approach. Moreover, the environment allows a larger group to be more productive without a drop off in individual productivity. Because larger groups may participate, the need for lengthy model review cycles is eliminated. Also, rapid model development makes it more likely that key personnel will participate. A drawback to supporting larger groups is the potential for creating large, detailed models riddled with syntactic errors and inconsistencies. Models which have such errors are less useful to the overall re-engineering process and make the models themselves difficult to export between software environments and impossible to store in standardized repositories. Manually checking and correcting even a moderate size model is a tedious task that takes longer than most groups have tolerance for. The use of algorithmic error checking has already proven to be an invaluable aid in model development. More experience is needed before the usefulness of the automatic connection propagation feature can be ascertained. The next step in this line of research is to incorporate these algorithms and others like them into software wizards which can offer additional guidance to users during the model development process.

References

1. Aytes, K. J. (1995). "Comparing Collaborative Drawing Tools and Whiteboard: An Analysis of the Group Process". Journal of Computer Supported Cooperative Work, Vol 4, No. 1.

2. Davenport, T.H. , etal, "Reengineering: business change of mythical proportions?" MISQ, June, 1994.

3. Dean, D.L.; etal, Technological support for group process modeling. JMIS, Vol. 11, No. 3, Winter, 1994-95.

4. Dean, D.L.; Orwig, R.E. and Vogel, D.R. "Facilitation methods for use with EMS tools to enable rapid development of high quality business process models". Proceedings of the 29th Annual HICSS, 1996.

5. Dennis, A.R.; etal. "Re-engineering business process modeling". Proceedings of the 27th Annual HICSS, 1994.

6. IDEF Users Group Standards and Specification Committee. IDEF0 FIPS Comment Document. (5 March, 1993).

7. Mayer, R.J.; Painter, M.K.; deWitte, P.S. IDEF Family of Methods for Concurrent Engineering and Business Re-engineering applications. (1994). College Station, TX: Knowledge Based Systems, Inc.

8. Nunamaker, J.F.; etal. "Electronic meeting systems to support group work". CACM, Vol 34, No 7.

9. Pendergast, M.O., "GroupGraphics: Prototype to Product," Groupware Real-Time Drawing, A Designers Guide, McGraw-Hill, edited by Roy Rada, Saul Greenberg, and Stephen Hayne, 1995.