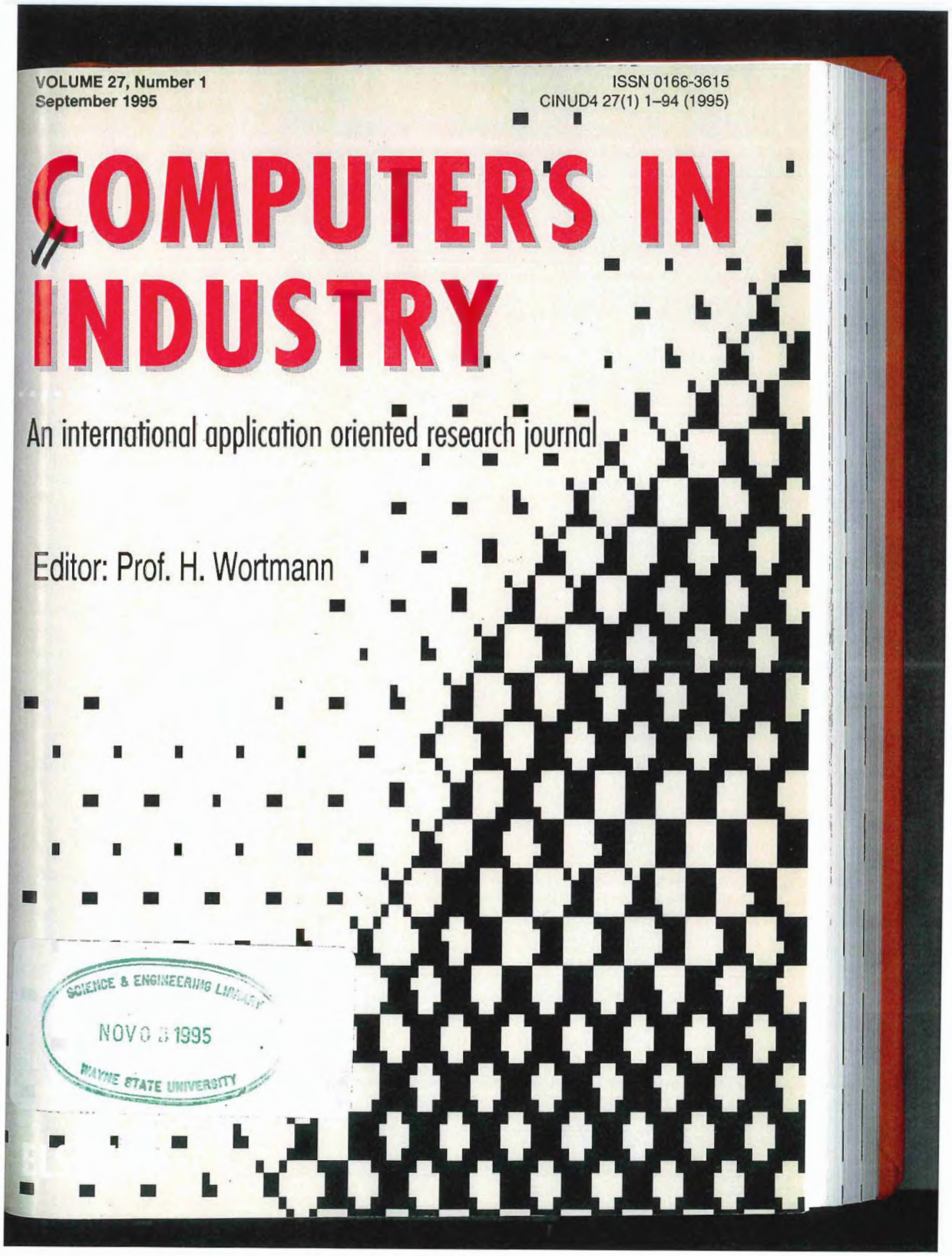

COMPUTERS IN INDUSTRY

Volume 27, 1995

ELSEVIER

Amsterdam – Lausanne – New York – Oxford – Shannon – Tokyo

ELSEVIER

Computers in Industry 27 (1995) 3–21

COMPUTERS IN INDUSTRY

Configuring computer systems through constraint-based modeling and interactive constraint satisfaction

S.M. Fohn [b], J.S. Liau [a,*], A.R. Greef [b], R.E. Young [a], P.J. O'Grady [a]

[a] *Group for Intelligent Design in Manufacturing, Department of Industrial Engineering, North Carolina State University, Raleigh, NC 27695-7906, USA*

[b] *IBM Corporation, Network Application Services Division, Thornwood, NY 10594, USA*

Received 22 November 1993; revised 22 September 1994

Abstract

A major problem in the computer industry, as with many other industries, is ensuring that the products ordered by a customer are viable products and that they can be delivered at the quoted price. The inability of companies to solve this problem results in an enormous expense for a company. In this paper we present an approach to solving this problem in the computer industry through constraint-based modeling and interactive constraint satisfaction. PC/CON, a Personal Computer Configuration System, has been built within a computer-based environment called Saturn. The development of PC/CON and related work is discussed. Saturn's architecture is presented and an example session with PC/CON is demonstrated. We show how Saturn and PC/CON have the features and capabilities necessary to support the extremely large variety of products and to support the maintenance of a configuration system for a rapidly and continuously changing product line. This approach would also appear to be applicable to other industries where there is a wide product variety.

Keywords: Computer configuration; Interactive constraint modeling; Constraint-based reasoning; Concurrent engineering

1. Introduction

A major problem in the computer industry, as with many other industries, is ensuring that the products ordered by a customer are viable products and that they can be delivered at the quoted price. The computers must be configured so that they can be readily manufactured and that they can support the desired software. The software must be configured for compatibility between hardware and other software so as to achieve good performance. Although a seemingly structured problem, the variety of options available for a computer system allows for billions of feasible product configurations making this problem difficult to solve consistently.

Trilogy [1] estimates that in the personal computer (PC) industry, between 30 and 85 percent of all orders are incorrectly configured. The result is an enormous expense for a company, especially when realizing that infeasible product configurations made at the point of sale may only be discovered much later in manufacturing, or, worse yet, after delivery to the customer. The consequences of late discovery include having to change a customer's order (thereby delaying manufacturing and increasing the lead-time),

* Corresponding author.

SSDI 0166-3615(94)00041-7

forcing a customer to return a shipment of faulty orders, and the loss of sales. Furthermore, products that are incorrectly configured are most likely incorrectly quoted. As a result, a manufacturer may be required to give away free components which were omitted from the sales quote yet are needed to supply the customer with the promised functionality. The costs involved, both monetary and the loss of customer good will, can be very high.

In order to reduce the impact from the sale of infeasible products, companies have attempted to include product configuration checks both at the point of sale and between the point of sale and start of manufacture. Configuration checking is performed either manually or automatically by computer programs. Programs like XSEL [2] and Trilogy [1] check orders at the point of sale while XCON [3] and Proof-Plan [4] check orders after a sale and before manufacture. The effectiveness of point-of-sale checking is dependent on the experience of the sales representatives and on the variety of the customers. An inexperienced sales representative dealing with a large number of different concerns from a diverse clientele may not be able to answer all questions immediately. This results in postponing sales until more information can be gathered to address the needs of a client. With post-sale checking, any detected configuration conflicts are resolved by calling customers and essentially cycling back to the point of sale before a valid work order is released to manufacturing. This cyclic approach can have a negative affect on a customer who is forced to endure long lead times, frustrating product configuration changes, and the prospect of returning products that do not meet customer expectations.

The computer product configuration problem can therefore be summarized as follows:

- 30% to 85% of all orders are incorrectly configured.
- Late discovery of configuration errors increases manufacturing lead times and produces incorrect orders.
- Configuration errors can result in a loss of confidence by customers and can result in a loss of sales.
- Incorrectly configured products can be quoted inaccurately.
- The configuration problem can be very large, taking it beyond the bounds of human ability to solve.

A system that solves the computer configuration problem must address these issues.

We present an approach that addresses the issues and solves the computer product configuration problem in the computer industry through constraint-based modeling and interactive constraint satisfaction. "Computer Product Configuration" is defined as the generation of a computer hardware and software configuration that meets the price and performance constraints imposed by a customer, and the compatibility constraints between hardware and software. This approach aims at avoiding the cycling between a manufacturer and its customers and focuses on interactively assisting order taking from a diverse clientele. It furthermore is capable of supporting a diverse number of personnel in fields such as manufacturing and marketing all of whom have an interest in computer product configuration. To accommodate this approach, we also introduce a Constraint Modeling System called Saturn, which is an interactive constraint satisfaction system that can be used to implement a computer configuration system. Saturn implements a logic-based, ATMS-supported, interactive constraint satisfaction system tightly coupled with a relational database [5]. It has a spreadsheet-like development interface that, in conjunction with a database manager, facilitates the construction of a constraint-based model representing a product line of feasible computer hardware configurations, supported software configurations, and pricing information.

In this paper, the development of a computer product configuration system is discussed, beginning with the salient features and capabilities that a computer-support environment for computer configuration should have. Next the related work on configuration systems that have addressed similar problems are presented and compared to our constraint modeling approach. This is followed by a discussion on the nature of the computer product configuration problem and a brief description of Saturn's architecture. In this context, the implementation of PC/CON, which is a personal computer configuration system built within Saturn, as a constraint-based model is shown. An example session with PC/CON is then described to show how Saturn's interactive constraint satisfac-

tion capabilities are harnessed by PC/CON to perform PC configuration. Finally, it is shown how Saturn and PC/CON have the features and capabilities necessary to support computer product configuration and to support the maintenance of a configuration system for a rapidly and continuously changing product line.

2. Embracing the task of computer product configuration

Product configuration is a critical yet a complex task in most industries. This is particularly true for the computer industry. To deliver a viable product, the computers not only need to meet the customers' functional requirements, they also must be configured so that they can be readily manufactured and they possess compatibility between hardware components and between hardware and the desired software. Although a seemingly structured problem, the variety of options available for a computer system allows for billions of feasible product configurations, making this problem difficult to consistently solve. As an illustration, we explore the computer product configuration problem in the personal computer (PC) industry. Compared to a supercomputer or a minicomputer, a PC typically has a simpler operating system, a smaller amount of memory and primary storage space, limited input and output devices, and is a single user system. However, as we will show, the product configuration problem for a PC is still a complex and challenging task.

In earlier years, manual PC product configuration was viable. This is readily seen by examining the number of feasible hardware configurations possible for the IBM model 25, with an 8086 8 MHz processor and an 8 bit PC bus architecture. This machine had only 240 feasible configurations and could run all software products. Today, consistently configuring PC orders manually is nearly impossible. The IBM model 95, with an 80486 33 MHz processor and a 32 bit MCA bus architecture has 467 billion feasible hardware configurations without considering any of the 115 cards that could plug into any of the 8 slots in its backplane. Table 1 contains the number of feasible product configurations for the PS/2 product line for 1991/1992 and for 1993. The numbers are computed from [6] and [7].

Table 1
The number of feasible configurations for the PS/2 product line

Model	Number of configurations	
	1991/1992	1993
25-086	240	240
25-286	768	768
30	21,760	11,520
35	145,317,888	238,132,224
55	835,584	2,151,552
56	235,929,600	2,615,328,768
57	115,605,504	308,625,408
70	258,048	1,347,840
80	23,181,312	131,843,712
90	5,138,022,400	2,082,103,296
95	31,085,035,520	467,362,415,520
Total	36,744,208,624	472,741,960,848

From the table, we can see that the number of configurations increased by more than 400 billion in just one year. The explosion of software products and their inherent incompatibility problems and platform-specific requirements has added to the complexity of computer product configuration. Compounding the problem is the weekly release of new products and changes to existing products. Major product introductions that require significant restructuring of the entire PC model line are now occurring in a 3 month cycle. In the PC industry it is not only essential to support the task of PC product configuration but also to support the development and maintenance of the information upon which that support is based. This calls for a configuration environment that can accommodate growing product complexity stemming from technological progress and from consumer demand.

To meet this challenge we consider computer product configuration to be a highly interactive, non-monotonic, step-wise process. A user can specify any permutation of product/price attribute values. The output is made up of consistent values for locally related attributes whose values were previously unspecified or of feedback information for resolving inconsistent partial configurations. Once all of the attributes have values and there are no inconsistencies among them, the computer product configuration is a valid working product that can be readily manufactured.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.