

# VIEnCod – Uma plataforma computacional para a otimização do ciclo de desenvolvimento de sistemas automatizados de manufatura

Gilvan de Oliveira Costa (PUCPR) [gocosta@swi.com.br](mailto:gocosta@swi.com.br)  
Marco Antonio Buseti de Paula (PUCPR) [marco.buseti@pucpr.br](mailto:marco.buseti@pucpr.br)  
Eduardo Alves Portela Santos (PUCPR) [portela@ccet.pucpr.br](mailto:portela@ccet.pucpr.br)

## Resumo

*Este artigo apresenta a especificação de uma plataforma computacional denominada VIEnCod. Esta ferramenta é baseada nos conceitos de Engenharia Virtual e Manufatura Virtual, e tem por objetivo fundamental dar suporte ao desenvolvimento de sistemas automatizados de manufatura. Como resultado a ser alcançado, espera-se diminuir o ciclo de desenvolvimento de projetos de tais sistemas. O artigo descreve os módulos principais do VIEnCod bem como o detalhamento dos submódulos que os constituem. Cada um dos submódulos implementa um função específica dentro do VIEnCod, de forma que as etapas de desenvolvimento sejam adequadamente tratadas.*

*Palavras chave: Engenharia Virtual, Manufatura, Automação.*

## 1. Introdução

O crescimento tecnológico tem implicado num maior volume de problemas técnicos a serem resolvidos e na necessidade de interação entre diferentes áreas do conhecimento. Devido a diversidade de conhecimentos exigidos para a atividade de projeto de produto e sistemas industriais, raramente são realizados por um único indivíduo o projeto e desenvolvimento de um novo produto. Com frequência, faz-se necessária a criação de uma equipe de trabalho de formação diversificada para realizar esta atividade.

A multidisciplinaridade em sistemas torna o processo de projeto complexo e demanda a cooperação entre diversos engenheiros e especialistas. Cada um destes tende a observar o problema de acordo com o seu ponto de vista, surgindo então diferenças de visões e linguagens que por sua vez conduzem à incompreensão e à falta de comunicação entre os membros da equipe e, como conseqüências, tem-se o aumento do tempo de finalização do produto e sistema e a elevação de recursos utilizados (humanos, físicos e econômicos).

Para a realização do projeto de sistemas automatizados não basta empregar as atividades, meios e modelos da engenharia de automação para as partes de controle e da engenharia de produto para a parte física. Também se faz necessário integrar as diferentes áreas, principalmente na fase de concepção do sistema em que se decidem quais as tecnologias que serão empregadas. Santos (2003) apresenta o ciclo de vida de um sistema automatizado de manufatura, onde as diversas atividades são relacionadas ao tempo de execução. Esse ciclo mostra, de forma qualitativa, o esforço demandado pela empresa para a completa operação do sistema projetado. Nota-se claramente que grande esforço é realizado na busca, detecção e reparos de erros e na otimização do sistema automatizado. De uma forma ideal, grande atenção deveria ser dada às atividades de planejamento e projeto de tais sistemas, pois são estas fases, se bem sistematizadas e executadas, que determinarão o volume de trabalho nas fases subseqüentes.

Reforçando o aspecto apontado por Santos (2003), Moraes e Castrucci (2001) afirmam que historicamente até 90% do esforço total de programação são dedicados à solução de

problemas posteriores à instalação, à correção de falhas ou erros de funcionamento em sistemas automatizados. Dessa forma, é extremamente importante inserir novos métodos, técnicas e ferramentas nas metodologias de projeto existentes, visando diminuir a perda de tempo e recursos ocasionados por reparos e otimizações.

Nesse contexto, agilidade e flexibilidade são conceitos reconhecidos com os principais atributos para um sistema de manufatura satisfazer as necessidades de um mercado global competitivo. Por sua vez, este fato resulta na necessidade de produzir produtos com alta qualidade e baixo custo, com reduzido tempo de desenvolvimento e respeitando demandas para diferenciação de produtos através de customizações. Isso significa que o sistema de manufatura deve responder rapidamente a mudanças de produção tanto no que se refere a volume como a variedade, de forma a efetiva, confiável e a um baixo custo.

A Manufatura Virtual e a Engenharia Virtual são abordagens identificadas como tecnologias emergentes para a manufatura ágil e flexível. Entende-se por Manufatura Virtual a execução de atividades através da simulação do sistema real, que pode ou não existir. Esta abordagem utiliza todas as informações do processo, dados do controle e gerenciamento e dados específicos do produto. É possível que partes reais da planta coexistam com partes virtuais. A partir desta noção pode-se inferir que a integração de modelos de simulação com os dados do controle e do gerenciamento do próprio sistema real são aspectos essenciais na abordagem denominada Manufatura Virtual (Moore et al., 2003).

Assim, o presente trabalho propõe que o projeto do sistema automatizado de manufatura seja realizado num ambiente computacional específico, denominado VIEnCod. Com o uso da ferramenta proposta, vislumbra-se aqui a diminuição do ciclo de desenvolvimento do projeto, uma vez que a ferramenta computacional de suporte auxiliará o projetista nas fases de identificação, modelagem, simulação, testes, implementação e otimização do sistema de manufatura automatizado.

## **2. Ferramentas computacionais existentes**

A viabilidade do uso de metodologias de projeto para sistemas automatizados passa forçosamente pela disponibilidade de ferramentas computacionais que a implementem. Algumas destas estão hoje disponíveis porém, em geral, são desenvolvidas no meio acadêmico, e não têm as boas características de um produto, no que diz respeito a suas interfaces e capacidade de lidar com problemas de porte. O desenvolvimento de ferramentas computacionais comerciais é de fundamental importância para a disseminação e aplicação de metodologias formais de projeto de sistemas automatizados de manufatura.

Dentre as ferramentas existentes para o apoio ao projeto de sistemas automatizados pode-se citar o TCT (Wonham, 1999), DESCO (Fabian & Hellgren, 2000), UKDES (Chandra et al., 2002), GRAIL (Raymond & Wood, 1996) e VER (Balemi et al., 1993). A maioria destas implementa o modelo proposto por Ramadge & Wonham (1989), como o TCT, DESCO, UKDES e o VER. O GRAIL, originalmente concebido como uma ferramenta de manipulação de autômatos, tem agora algoritmos desenvolvidos pelo grupo de Automação da UFSC (Cury, 2001) que tratam todo o processo de síntese de controladores baseado na abordagem de Ramadge & Wonham (1989). A grande vantagem desta ferramenta reside no fato desta possuir o código fonte aberto, permitindo uma maior liberdade ao desenvolvimento de módulos de programas de acordo com uma realidade específica. Também permite ao projetista mudança e inclusões na ferramenta de uma forma mais rápida e sistemática.

Outro ponto fundamental a ser considerado nas ferramentas citadas é a possibilidade de traduzir uma linguagem formal (por exemplo, autômatos) numa linguagem normalizada de programação de Controladores Lógico Programáveis (CLPs). Nesse aspecto pode-se citar o

Supremica (Akeson, 2002), onde a partir de modelos em autômatos, é possível gerar um código de acordo com a IEC 61131-3. Em relação a uma ferramenta que integra as etapas de desenvolvimento de um sistema automatizado, implementando o conceito de Engenharia Virtual, cita-se o VIR-ENG, resultado do projeto de pesquisa ESPRIT/IST (Moore et al., 2003).

### **3. VIEnCod**

O presente artigo apresenta a especificação de uma plataforma computacional, o VIEnCod, para o suporte ao projeto de sistemas automatizados de manufatura. O objetivo fundamental desta ferramenta é dar suporte ao desenvolvimento de tais sistemas, englobando as etapas ou atividades de identificação, modelagem, síntese, geração do código, simulação, implementação, testes, otimização e seleção de tecnologias avançadas de manufatura (AMTs). A partir de uma estratégia de manufatura e de uma especificação de projeto, deseja-se obter uma planta automatizada, englobando todo o sistema de controle sintetizado, distribuído de acordo com uma topologia selecionada de hardware, implementado e otimizado, bem como com todas as AMTs selecionadas de acordo com critérios pré-estabelecidos.

O presente trabalho foca na parte do VIEnCod que tem por objetivo implementar as fases de identificação, modelagem, síntese e implementação do sistema de controle. Nessa parte da ferramenta, o projetista deverá analisar a planta de manufatura, identificar através de modelos formais a representação comportamental dos subsistemas e equipamentos que a compõe e também representar as especificações que se deseja impor a planta global. Em seguida, o VIEnCod deve retornar controladores em linguagem própria para implementação (IEC 61131-3), especificar uma estrutura distribuída do programa em relação a uma topologia pré-definida de hardware e ser capaz de simular esta estrutura. É possível ainda simular partes do sistema de controle interagindo com partes reais do sistema físico.

Para alcançar este objetivo o VIEnCod é concebido em módulos, tendo cada um destes uma função específica. Além disso, o GRAIL é integrado ao VIEnCod como ferramenta central para a modelagem e a síntese dos controladores. A opção por esta ferramenta se justifica principalmente pela abertura do seu código, onde pode-se inserir novas funções e modificar outras de acordo com o caso tratado e necessidades específicas. O formalismo matemático adotado no projeto VIEnCod é descrito na próxima seção.

#### **3.1 Formalismo matemático utilizado no VIEnCod**

Desde que uma grande parte da atividade de projeto de sistemas de manufatura automatizados é consumida na elaboração do sistema de controle, muitos trabalhos têm sido desenvolvidos no sentido de formalizar a obtenção de tal sistema. Existe uma intensa atividade de pesquisa voltada à busca de modelos matemáticos adequados a representação de sistemas de manufatura, sem que se tenha conseguido encontrar um modelo que seja matematicamente tão conciso e computacionalmente tão adequado como o são as equações diferenciais para os sistemas dinâmicos de variáveis contínuas (por exemplo, a representação de variáveis como pressão, temperatura, velocidade, no decorrer do tempo).

Diversos modelos que tratam de sistemas de manufatura podem ser citados, sendo que tais modelos refletem os diferentes tipos bem como diferentes objetivos na análise dos sistemas em estudo. Pode-se citar os principais modelos utilizados: Redes de Petri, Cadeias de Markov, Teoria das Filas, Processos Semi-Markovianos Generalizados e Simulação, Álgebra de Processos, Álgebra Max-Plus, Lógica Temporal, Teoria de Linguagens e Autômatos. Dentre os modelos citados anteriormente, pode-se destacar o modelo de Ramadge & Wonham (1989) baseado na teoria de Linguagens e Autômatos. Diferentemente dos outros modelos que

ênfatisam a análise de sistemas de manufatura, o modelo citado é dotado de procedimento de síntese de controladores, o que é uma grande vantagem em relação as outras abordagens citadas.

No entanto, quando um grande número de tarefas deve ser executado pelo sistema de controle, a abordagem de Ramadge & Wonham (1989) pode ter um desempenho computacional bastante desfavorável, uma vez que considera a obtenção de um único controlador que observa e atua sobre toda a planta. Uma forma de diminuir a complexidade computacional do problema é dividir a tarefa de controle em várias sub-tarefas e explorar a modularidade natural da planta. Dessa forma, Queiroz & Cury (2002) propõem extender o modelo de Ramadge & Wonham (1989), propondo a denominada abordagem modular local. Esta abordagem propõe uma arquitetura de controle distribuída em que cada módulo de controle atua somente sobre os subsistemas envolvidos. Por este motivo, o VIEnCod tem como formalismo matemático para o tratamento de sistemas de manufatura o modelo proposto por Ramadge & Wonham (1989) e a extensão proposta por Queiroz e Cury (2002).

### 3.2 Módulos do VIEnCod

O VIEnCod é concebido em módulos que executam tarefas específicas do ciclo de desenvolvimento do sistema automatizado. A figura 1 ilustra o VIEnCod em UML (Rumbaugh et al., 1999), onde pode-se verificar que a citada ferramenta computacional possui três módulos principais, quais sejam: o Ambiente de Desenvolvimento de Projetos (PDE), o Ambiente de Engenharia Virtual e o Ambiente de Implementação. É no primeiro módulo (PDE) que existe a interface entre o usuário e o GRAIL, sendo o ambiente em que o usuário realiza a entrada de informações. Sob o ponto de vista do software, o VIEnCod, a partir dos dados selecionados ou especificados pelo usuário, acessa o GRAIL para este realizar as operações (composição, verificação, síntese) necessárias. Ainda, o VIEnCod tem acesso a informações específicas do ambiente de manufatura de forma a suprir continuamente o seu banco de dados. Cada um dos módulos do VIEnCod é constituído de diversos submódulos que implementam as atividades relacionadas a identificação, modelagem, síntese, simulação, implementação e testes. O detalhamento dos módulos é apresentado nas seções seguintes.

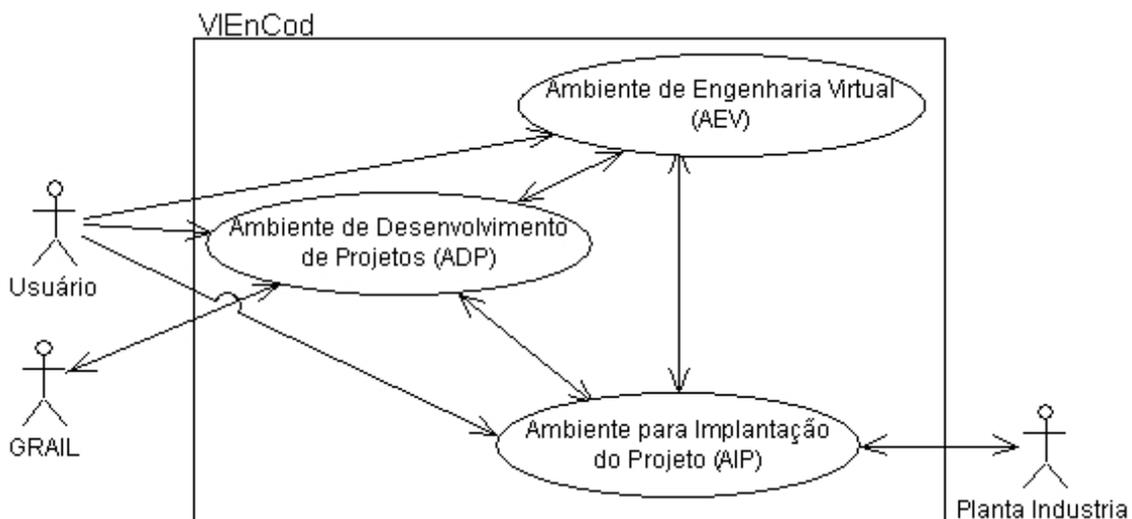


Figura 1 – Representação dos módulos principais do VIEnCod.

#### 3.2.1 Ambiente de Desenvolvimento de Projetos (ADP)

O ADP consiste basicamente de um local onde são inseridos os modelos formais de representação do sistema de manufatura, modelos estes baseados na teoria de linguagens

formais e autômatos. De acordo com a abordagem utilizada, discutida na seção 3.1, a aplicação desta pressupõe a construção de modelos adequados que representam a planta (formada geralmente por diversos subsistemas) e as especificações que se deseja impor a ela. Após a construção destes modelos, pode-se sintetizar controladores de forma automática que garantam o correto funcionamento definido pelo conjunto de especificações. As ferramentas computacionais discutidas na seção 2 na sua maioria tratam especificamente do processo de síntese, que engloba a modelagem da planta e especificações em autômatos. Entretanto, são plataformas dedicadas a estas operações, não fornecendo ao projetista o suporte necessário para lidar com a complexidade inerente a esta etapa do ciclo de desenvolvimento. Nesse sentido, o módulo ADP possui sete submódulos que dão o suporte adequado ao usuário, de acordo com a descrição a seguir.

*I - Modelagem do Sistema (MoSys):* Este submódulo dá suporte à modelagem da planta e das especificações através de autômatos. Esta modelagem pode referir-se a diversos níveis de abstração do sistema, cabendo ao usuário organizar e especificar cada nível da maneira mais conveniente para o projeto em questão.

*II – Biblioteca de Modelos em Autômatos de Subsistemas e Especificações (BMAS):* Consiste de uma biblioteca contendo diversos modelos de subsistemas e especificações comumente encontradas em sistemas de manufatura. Este submódulo torna a modelagem mais rápida e sistemática, pois proporciona modelos já utilizados e testados e fornece ao usuário informações sobre uma nova modelagem baseada numa já existente. Este submódulo consiste de um banco de dados que continuamente recebe novas informações do usuário a partir de projetos já realizados, ampliando assim o número de modelos.

*III – Leitura e Interpretação das Tabelas do GRAIL (RInT):* Este submódulo lê e interpreta o formato das informações recebidas (MoSys → GRAIL) e enviadas (GRAIL → GeSCon ou GRAIL → ViAut) pelo GRAIL. A ferramenta GRAIL manipula os modelos num formato específico (através de tabelas), sendo necessário para os outros submódulos uma tradução das tabelas num formato adequado.

*IV – Visualização dos Autômatos (ViAut):* Ambiente para visualização dos autômatos, tanto no que se refere aos modelos dos subsistemas e das especificações como ao resultado do processo de síntese (controladores modulares locais obtidos).

*V – Geração da Estrutura de Controle (GeSCon):* Este submódulo cria estruturas de controle adequadas para aplicações específicas. A função básica deste submódulo é criar opções de distribuição do programa de acordo com topologias de hardware, a partir de duas situações distintas. Na primeira situação, tem-se previamente os dispositivos (CLPs ou PCs) onde serão implementados os controladores locais obtidos. Nesse caso, o GeSCon deverá propor uma distribuição mais adequada dos controladores de acordo com a situação física encontrada na planta em questão. Na segunda situação, inexistem dispositivos onde serão implementados os controladores. O GeSCon fornecerá então opções de configuração de distribuição, de acordo com alguns critérios pré estabelecidos.

*VI – Geração do Código Fonte (GeCode):* Tendo como base as tabelas já interpretadas do GRAIL, o GeCode traduz os controladores locais existentes numa linguagem específica de programação (segundo a IEC 61131-3). Considera ainda a estrutura de controle proposta em GeSCon. O GeCode poderá ainda traduzir os controladores obtidos em linguagens para plataformas de baixo custo, por exemplo, quando usa Micro Controladores como dispositivo de implementação.

*VII – Seleção das Tecnologias Avançadas de Manufatura (Sel-AMT):* Ambiente para seleção de AMTs a serem implementadas no decorrer do desenvolvimento de um dado projeto.

### **3.2.2 Ambiente de Engenharia Virtual (AEV)**

O AEV consiste em um módulo de validação do projeto através de um ambiente virtual. Tal módulo é capaz de reproduzir ações e representá-las em um ambiente virtual através de simulações dos controladores gerados anteriormente. Para alcançar esta funcionalidade, o AEV é formado por quatro submódulos descritos a seguir.

*I – Criação da Interface Gráfica (CGraphI):* Aqui se dá o desenvolvimento de uma interface gráfica para o usuário. Tem-se a opção de simular os controladores obtidos com a distribuição selecionada, subconjuntos de controladores, camadas específicas da estrutura de controle, dentre outros. De maneira geral, é um ambiente que dá flexibilidade ao usuário criar interfaces de acordo com necessidades específicas.

*II – Biblioteca de Interfaces e Equipamentos (BIE):* Este submódulo consiste de uma bibliotecas de interfaces gráficas de componentes padrões de uma planta de manufatura e também partes de interfaces gráficas de outros projetos que possam ser reutilizadas.

*III – Configuração das Tecnologias Avançadas de Manufatura (ConAMT):* Configuração de todos os equipamentos e soluções existentes em um determinado projeto.

*IV – Execução do Projeto num Ambiente Virtual (EDVE):* O EDVE consiste no principal submódulo do AEV. É aqui que se dá a execução do projeto em um ambiente virtual, através da simulação do ambiente físico, de forma a validar as funções de controle e controladores projetados, a lógica dos programas gerados bem como os supervisores locais obtidos.

### **3.3 Ambiente para Implementação do Projeto (AIP)**

O módulo AIP dá suporte as últimas atividades do desenvolvimento do sistema automatizado de manufatura. Estas atividades são relacionadas a implementação, validação e testes do sistema de controle obtido a partir dos módulos descritos anteriormente. Dessa forma, é através desse módulo que existe uma primeira interação do ambiente virtual com a planta física. O usuário, no ambiente AIP, pode interagir com a planta através da simulação integrada com subsistemas reais, testar a distribuição proposta pelo módulo GeSCon e realizar modificações em tempo real no sistema de controle obtido. Para alcançar estes objetivos, o módulo EID é constituído de três submódulos descritos a seguir.

*I – Transmissão do Código Fonte (Tcode):* Este submódulo tem por objetivo transferir o programa gerado pelos módulos anteriores para a memória de um determinado hardware (CLP, PC, Micro Controlador).

*II – Hardware in the Loop (HIL):* Rotina de execução em tempo real de um ou mais módulos dos controladores gerados pelo sistema VIEnCod. Através desse módulo, será possível a implantação gradual do sistema de controle obtido até então no sistema físico global, possibilitando dessa forma uma melhor validação dos controladores projetados bem como uma melhor sistematização da detecção e da correção de erros.

*III – Dispositivo de Suporte para a Comunicação em Redes (DcomBus):* Interface para conexão com uma rede de comunicação específica. Consiste em um submódulo separado do VIEnCod que tem por objetivo fundamental capturar os dados via TCP/IP e convertê-los para um determinado barramento industrial. O objetivo deste submódulo é possibilitar a comunicação entre os dispositivos da planta e o VIEnCod.

### **3.4 Fluxo de informação da ferramenta VIEnCod**

Conforme verificado na figura 2, numa representação em UML, o VIEnCod trata do desenvolvimento de soluções de automatização para manufatura desde o nível da



globalizado. Deseja-se com a utilização desta ferramenta um maior tempo de serviço do sistema automatizado, reduzir o tempo de desenvolvimento de tais sistemas e tornar o tempo de entrada de um novo produto mais curto. Ao mesmo tempo, deseja-se reduzir custos em comissionamento da planta, treinamento de operadores, documentação do sistema e otimizar planos de manutenção.

## Referências

- AKESSON, K. (2002) - Methods and Tools in Supervisory Control Theory. Thesis For The Degree Of Doctor Of Philosophy, Chalmers University of Technology, Sweden.
- BALEMI, S., HOFFMANN, G. J. and GYUGYI, P. et al (1993). Supervisory control of a rapid thermal multiprocessor. *IEEE Transactions on Automatic Control*, 38(7), 1040-1059.
- CHANDRA, V., ORUGANTI, B. and KUMAR, R. (2002). UKDES: A graphical software tool for the design, analysis & control of discrete event systems. [clue.eng.iastate.edu/~rkumar/PUBS/ukdes.ps](http://clue.eng.iastate.edu/~rkumar/PUBS/ukdes.ps).17:34.
- CURY, J. E. R. (2001). Teoria de Controle Supervisório de Sistemas a Eventos Discretos, V Simpósio Brasileiro de Automação Inteligente, Gramado, RS.
- FABIAN, M. & A. HELLGREN (2000). *Desco – a Tool for Education and Control of Discrete Event Systems*. Kluwer Academic Publishers.
- IEC 61131-3 International Electrotechnical Commission (1993). Programmable Controllers Part 3: Programming Languages.
- MOORE, P.R; PU J.; NG, H.C. et al. (2003) – Virtual engineering: an integrated approach to agile manufacturing machinery design and control. *Mechatronics*, 13, p. 1105–1121.
- MORAES, C. C. & CASTRUCCI, P. L. (2001). *Engenharia de Automação Industrial*, Livros Técnicos e Científicos (LTC).
- QUEIROZ, M. H. & CURY, J. E. R. (2002). Controle Supervisório Modular de Sistemas de Manufatura. *Revista controle & automaçã*, SBA, 13, pp.115-125.
- RAMADGE, P.J. & W. M. WONHAM, W. M. (1989). The control of discrete event systems, Proc. of IEEE, Special Issue on Discret Event Dynamic Systems, **77**, pp. 81-98.
- RAYMOND, D. & WOOD, D. (1996). Grail: Engineering automata in C++: Version 2.5, <http://www.csd.uwo.ca/research/grail/> verificada em 31 de janeiro de 2003.
- RUMBAUGH, J.C.; BOOCH, G. and JACOBSON, I. (1999). *The UML Reference Guide*. Addison Wesley Longman.
- SANTOS, E. A. P. (2003). Contribuições ao projeto conceitual de sistemas de manipulação e montagem automatizados. Tese de Doutorado (Engenharia Elétrica), Universidade Federal de Santa Catarina.
- WONHAM, W. M. (1999). Notes on controls of discrete event systems. Course notes for ECE 1636F/1637S. Dept. Of Electrical and Computer Engineering, University of Toronto, Canadá.