

## Uma perspectiva sistêmica sobre a Engenharia de Software

Gernot Richter\*

PUC/RJ e GMD/Alemanha

Bruno Maffeo

Pontifícia Universidade Católica – PUC/RJ

### Sumário

Criar, alterar e eliminar sistemas com alto nível de concorrência entre suas atividades, quando esses sistemas fazem parte de sistemas mais abrangentes com as mesmas características, são operações que constituem enorme desafio para uma *Engenharia de Sistemas* lato sensu. Desta, cada vez mais, exige-se uma visão integral abrangendo o sistema hospedeiro, que tanto pode ser um sistema puramente técnico quanto uma organização entendida como um sistema sócio-técnico. Neste trabalho, são abordados, de modo bastante sucinto, tópicos de pesquisa visando o atendimento dessa exigência e alguns de seus resultados. Além dos meios (formais) de expressão necessários para representar sistemas — atividades de processamento e estruturas de informação —, são também consideradas questões referentes à análise, construção, implantação e manutenção de sistemas, sob a ótica da aplicação de técnicas de modelagem.

### Abstract

To create, to change or to eliminate systems which form part of comprising systems is one of the most challenging tasks of *Systems Engineering* in the broadest sense of the word. This is even more true if the systems distinguish themselves by a high degree of concurrency between their activities. To cope with the challenge requires an integrated view which includes the comprising host system, whether it be a purely technical system or an organization perceived as a socio-technical system. This article deals quite concisely with some research topics and results which aim at meeting such requirements. Apart from (formal) means of expression needed for the representation of systems, i.e. of information processing activities and information structures, also questions referring to analysis, construction, installation and maintenance of systems are considered from a systems modelling point of view.

---

\*Trabalho parcialmente financiado pelo CNPq e IB/GMD (Projeto FORTIS)

**Palavras-chave:** sistema, engenharia de software, engenharia de sistemas, sistema sócio-técnico, informática, sistema computacional, modelagem de sistemas, Redes de Petri, sistema informático

## 1. Orientação a Sistemas

O objeto da Informática é “sistemas”.

Apesar de nenhum profissional da área negar essa assertiva, sua abrangência é, frequentemente, limitada pelo entendimento restrito do termo “sistema”. Com efeito, o foco desse entendimento convergiu historicamente para os processos de automação implementados através da tecnologia de computadores digitais programáveis. Como esse entendimento decorre de uma escolha, em larga medida, arbitrária, é possível questioná-lo e, eventualmente, substituí-lo por outro.

Que razões podem ser invocadas para esse questionamento?

Por um lado, há uma questão de coerência interna que pode ser percebida utilizando uma definição abstrata do termo “sistema”: *conjunto, identificável e coerente, de elementos que interagem coesivamente, onde cada elemento pode ser um sistema*. Essa definição, genérica o suficiente para aplicar-se a todas as áreas em que o termo é empregado — Ciências Naturais e Sociais, Engenharias etc. —, permite identificar alguns aspectos básicos do conceito. Primeiro, a “interatividade” dos elementos elimina a possibilidade de sistemas estritamente isolados — o Universo seria o único sistema isolado conceitualmente admissível. Obviamente, certas interações podem ser comparativamente menos intensas do que outras, o que viabiliza o emprego de aproximações do tipo “sistema isolado” para o estudo simplificado de sistemas complexos. Segundo, a “recursividade” do conceito implica que um elemento — diferente do Universo — entendido como sistema seja necessariamente parte de um sistema mais abrangente.

Nestas condições, verifica-se que o entendimento restrito do termo “sistema”, como o tradicionalmente adotado na área de Informática, pressupõe que:

- os sistemas da Informática podem, como aproximação simplificadora adequada, ser considerados “quase”-isolados, isto é, apenas suas interfaces mais imediatas com elementos do ambiente externo devem ser objeto de estudo;
- o sistema mais abrangente, que demanda os sistemas da Informática, deve ser analisado a partir da perspectiva do computador digital programável.

Por outro lado, há uma questão de coerência externa, relacionada com os objetivos da Informática. As necessidades de tratamento sistemático — estruturação e processamento — de informação decorrem de problemas relacionados, em última análise, com sobrevivência e/ou crescimento de sistemas sociais humanos. Tais problemas, em geral, antecedem o advento da tecnologia dos computadores digitais programáveis. Em consequência, deve ser admitido que

o objetivo da Informática é encontrar solução para esses problemas; os quais nem sempre são satisfatoriamente resolvidos através do emprego de alternativas automatizadas baseadas nessa tecnologia.

O conflito entre essas duas visões — uma, decorrente do entendimento restrito do termo “sistema”, e outra, relacionada com os objetivos “naturais” da Informática — explica-se por razões históricas. Após o advento da tecnologia dos computadores digitais programáveis, seu aprimoramento contínuo, em resposta a uma demanda crescente por informatização decorrente da percepção do poder e da abrangência de sua utilização, vem implicando enorme mobilização de esforços para superar os desafios técnico-científicos inerentes à nova tecnologia. Essa mobilização redundou em progressos tecnológicos extremamente rápidos e expressivos, porém à custa de alguma perda de perspectiva em relação aos objetivos últimos a serem atingidos. Além disso, a complexidade da tecnologia tem dificultado, durante seu processo de consolidação técnico-científica, uma interação mais qualitativa entre o “sistema Informática” e o ambiente externo — cliente e usuário da tecnologia. A debilidade dessa interação tem aprofundado o fosso entre as duas visões e levado a área de Informática a um isolamento conceitualmente inconsistente. Predomina, nessa área, a questão relativa ao “melhor uso da máquina” sobre a questão relativa à “melhor solução para o problema”; apesar do inter-relacionamento óbvio das duas questões, a segunda deve, sem dúvida, subordinar a primeira.

## 2. Sistemas com Componentes Computacionais: Engenharia de Software

O objeto da Informática é “sistemas” — o que significa isso?

Quando um sistema é implantado em determinado ambiente, o “sistema hospedeiro” é alterado, criando-se um novo sistema que possui o sistema implantado como componente. Ao eliminar-se um sistema implantado em outro sistema mais abrangente, este último é alterado, criando-se um novo sistema — agora, sem o sistema eliminado. Percebe-se, então, que a criação e a eliminação de (sub-)sistemas implantados — isto é, em operação — sempre acarretam alteração do “sistema hospedeiro”. Essa alteração é obtida pela eliminação do antigo sistema abrangente imediatamente sucedida pela criação de um novo sistema abrangente.

Essa visão “sistêmica” dos processos de criação, alteração e eliminação de sistemas, consistente com a conceituação genérica apresentada na seção precedente para o termo “sistema”, sugere uma possível definição para o tipo específico de sistema que é o objeto da Informática. Tratar-se-ia de *sistema sócio-técnico*, entendido como um *sistema aberto (total ou parcialmente) automatizado, concebido para atender a necessidades do ambiente em que está inserido, comportando-se como um mecanismo de estímulo-resposta* (ver Figura 1).

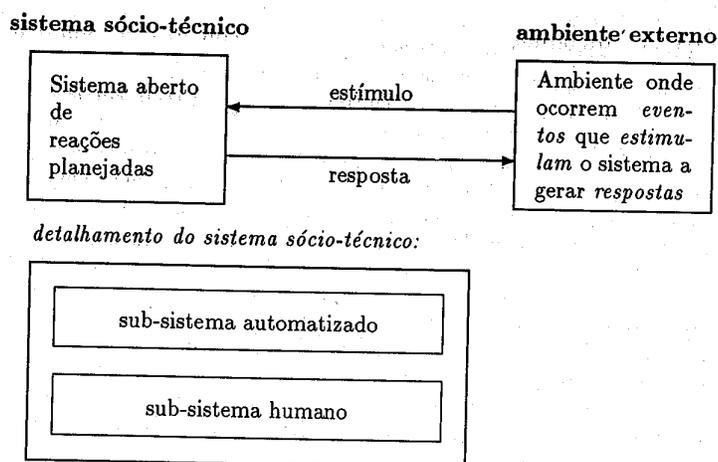


Figura 1

Essa definição requer alguns comentários adicionais.

A existência de um componente humano implica a recusa de perceber o sistema objeto da Informática como algo que possui um comportamento totalmente mecanicista, desprovido de qualquer forma de indeterminismo. Na realidade, o aspecto "planejado" do sistema requer que o planejamento das reações a um dado estímulo incorpore o fato de que nem sempre é possível ou desejável identificar a priori todos os motivos que influenciam o processo decisório que determina a(s) resposta(s) gerada(s) por um sistema sócio-técnico.

Além disso, convém ressaltar que o sub-sistema automatizado não fica necessariamente restrito à tecnologia dos computadores digitais programáveis. Computadores analógicos, componentes (em geral, sensores e acionadores de sistemas de controle e/ou monitoração de processos) eletro-eletrônicos, eletro-mecânicos, elétricos, eletrônicos ou mecânicos são legítimos candidatos a elementos de um sub-sistema automatizado.

Mencionar explicitamente a tecnologia de automação — a rigor, computadores digitais programáveis — obedece ao imperativo de pragmatismo e não a algum imperativo lógico: mesmo que indivíduos da área de Informática dediquem-se exclusivamente à modelagem de sistemas independentemente de qualquer forma de automação, o produto final de um esforço de "informatização" incluirá sempre, em grau variável, algum componente automatizado em termos dessa tecnologia. Entretanto, a definição subordina o esforço a interesses exter-

nos a esse componente e enfatiza o caráter não-isolado — aberto — do sistema gerado.

Empregando recursivamente o enfoque sistêmico e restringindo a tecnologia de automação a computadores digitais programáveis, os elementos automatizados podem ser vistos como pertencentes a dois (sub-)sistemas: hardware e software. O sistema sócio-técnico abrange, ainda, elementos pertencentes a uma terceira categoria principal de componentes, referida como o (sub-)sistema humano. Assim, os elementos que definem a “composição imediata” de um sistema sócio-técnico ficam agrupados em, no máximo, três sub-sistemas: humano, hardware e software (ver Figura 2).

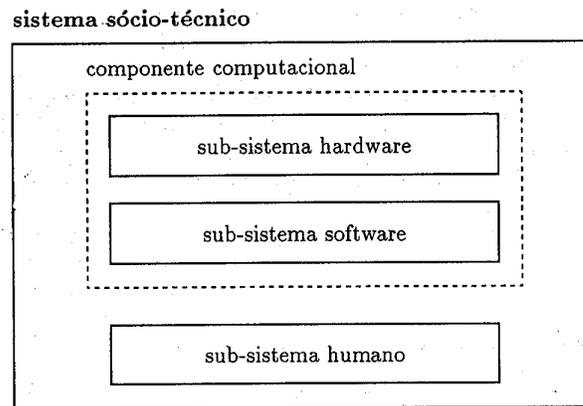


Figura 2

O componente humano constitui, sem dúvida, o (sub-)sistema mais complexo. Representado, em geral, por organização específica (empresa, universidade, autarquia, departamento, setor, ...), distingue-se dos demais por uma propriedade única: a capacidade de “decisão autônoma” associada a qualquer um de seus elementos, na presente análise, atômicos — as pessoas. Essa distinção, tendo em vista que esse (sub-)sistema está na origem da demanda por recursos da Informática, torna extremamente delicada a tarefa de construir um sistema abrangente harmonioso. É nesse (sub-)sistema que devem ser decididas autonomamente questões relevantes referentes a interesses a satisfazer, objetivos a atingir, metas a cumprir, intenções a respeitar etc. e cada decisão adotada (re-)define, em geral, a posição de equilíbrio do sistema abrangente bem como as interações envolvendo os elementos que o compõem.

O hardware — entendido como (sub-)sistema, possui componentes espe-

## Artigo Técnico

cíficos (por exemplo, CPU, memória principal, memória secundária, monitor de vídeo, teclado, ...) que podem também, cada um, ser considerados como (sub-sub-)sistemas — é objeto de uma engenharia específica, fundamentada nas Ciências Naturais — predominantemente, Física — e constitui, quase sempre, sub-sistema de propósito geral. Tendo em vista suas interações, comparativamente pouco intensas, com o ambiente externo ao sistema abrangente composto pelos três sub-sistemas, pode ser adequadamente aproximado por um modelo de sistema isolado.

O software constitui o sub-sistema que adapta as características genéricas do hardware à solução de problemas específicos, estabelecendo uma capa abstrata envolvendo o hardware. Frequentemente, essa capa deve ser analisada sistemicamente em, pelo menos, duas camadas (sub-sub-sistemas) de níveis distintos de abstração relativamente aos recursos de máquina: software básico e software aplicativo. É esse envoltório abstrato do hardware, proporcionado pelo software, que justifica o emprego de um modelo de sistema isolado como boa aproximação para abordar as características do sub-sistema hardware. Jámais essa aproximação seria adequada para o sub-sistema software — sobretudo para o sub-sub-sistema aplicativo.

Essa inevitável, geralmente muito intensa, interação do sub-sistema software com o ambiente externo e com o sub-sistema humano, quando este existe, impõe à disciplina que trata dos temas relativos ao software — a Engenharia de Software — características necessariamente abrangentes do ponto de vista sistêmico. Essa engenharia tem, frequentemente, que estabelecer uma ponte abstrata entre elementos funcionais distintos, associados ao sistema abrangente composto pelo sistema sócio-técnico e o ambiente externo:

- de um lado, peças de hardware, sub-sistema “físico” de um sistema sócio-técnico que esteja sendo construído,
- de outro lado, peças de hardware e software, pessoas, idéias etc. pertencentes seja a outros sub-sistemas situados no ambiente externo, seja ao sub-sistema humano do próprio sistema sócio-técnico considerado.

E essa ponte tem que cumprir o papel de (parte da) solução de um problema, proporcionando interface para respostas planejadas a estímulos específicos vindos do ambiente externo (ver Figura 3).

Nesse papel de ponte, a Engenharia de Software não pode restringir-se às ferramentas e técnicas adequadas ao sub-sistema computacional, isto é, não pode fundamentar-se apenas na Ciência da Computação. Ela deverá investir na criação e aperfeiçoamento de ferramentas e técnicas específicas para estabelecer a representação, e sua validação, para:

- a definição e a solução abstratas — relativamente ao emprego da tecnologia de automação — de um problema a ser resolvido (questões referentes ao ambiente externo ao sistema sócio-técnico);

sistema sócio-técnico

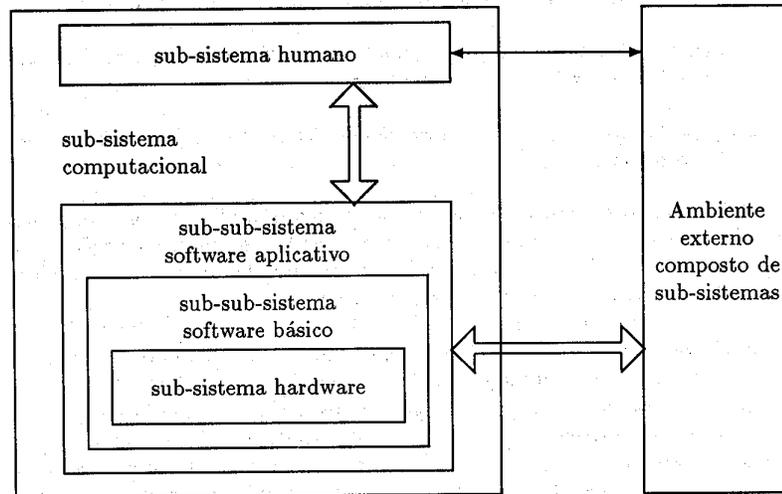


Figura 3

- a melhor alternativa de implementação para uma solução (parcial ou totalmente) automatizada (questão referente tanto ao sistema sócio-técnico quanto ao ambiente externo).

Esse esforço transcende os aspectos estritamente relacionados com o sub-sistema software e aponta inequivocamente para uma Engenharia de Sistemas lato sensu, fundamentada numa, a ser consolidada, Ciência de Sistemas.

### 3. Modelos Informais e Formais de Sistemas

Tendo em vista a discussão preliminar das seções precedentes, como pode ser definida a finalidade dos trabalhos a serem desenvolvidos pela área de Informática? A que objetivos devem orientar-se as contribuições da Informática para o aperfeiçoamento da sociedade humana? Quais são as tarefas ao perseguir esses objetivos?

Respostas a tais perguntas relacionam-se com atividades que visem:

- Investigar sistemas, em operação ou projetados, com o objetivo de avaliar e/ou prever seu comportamento;

### Artigo Técnico

- Melhorar sistemas existentes, tendo em vista critérios estabelecidos a partir de resultados de investigação, possivelmente re-organizando-os com base em princípios da Ciência da Computação;
- Analisar, projetar, implementar, implantar e manter, isto é, corrigir, adaptar ou aperfeiçoar, sistemas sócio-técnicos;
- Controlar e/ou monitorar a operação de sistemas, ou seja, dominar sistemas complexos.

Considerando o conjunto de atividades acima especificado como o escopo de uma Engenharia de Sistemas mais ampla, verifica-se uma associação desta engenharia com a aplicação rotineira de um conjunto de ferramentas e técnicas

- conceituais (para pensar, raciocinar, analisar)
- expressivas (para representar e comunicar)
- metodológicas (para proceder, abordar, corrigir, verificar, validar)
- computacionais (para automatizar).

Decorre dessa visão que o *paradigma básico* dessa engenharia constituiria um processo de *modelagem de sistemas*. Entendendo modelo como uma *representação abstrata que permite descrever e/ou prever comportamentos específicos de um sistema, através do estudo de um número reduzido de características relevantes do sistema*, a perspectiva de uma Engenharia de Sistemas lato sensu destaca a disciplina que visa obter percepções abstratas de uma realidade existente ou imaginada (por exemplo, planejada), permitindo raciocinar sobre aspectos dessa realidade e deduzir conclusões, através de processos imateriais (analíticos) e/ou de modelos físicos (concretos, materiais) para investigar sistemas utilizando métodos da Informática experimental (simulação, observação etc.).

O uso de modelos (matemáticos) para descrever e/ou prever comportamentos do mundo "real" faz parte do paradigma básico das Ciências Naturais. Esse uso permite a investigação de sistemas naturais complexos, auxiliando os processos de análise da realidade — segmentação e abstração aplicadas ao sistema em estudo — e de validação experimental de teorias — comparação entre resultados numéricos obtidos a partir do modelo com resultados numéricos obtidos a partir de medições de propriedades da realidade. Esses processos são fundamentados em manipulações formais, qualitativas e quantitativas, do modelo. Estender esse paradigma à Ciência e Engenharia de Sistemas é uma decorrência lógica da ampliação de escopo que a definição genérica do termo "sistema" exige.

*Programas de computadores digitais* são exemplos de modelos de sistemas — a maioria deles modelando apenas processos sequenciais — que, no estado de execução em determinado hardware, compõem com este um sistema computacional real.

*Diagramas de Fluxos de Dados* constituem outros exemplos de modelos de sistemas. São modelos ainda informais — não especificam de forma rigorosa nem refletem todas as características relevantes dos sistemas de interesse —

que servem para a modelagem exploratória e aproximada do comportamento de sistemas.

A *modelagem aproximada* através de diagramas apresenta, em geral, ambiguidades, e não incorpora todos os aspectos necessários a uma simulação do comportamento do sistema. Suas legendas, expressas informalmente, servem para explicar o significado dos elementos ativos e passivos do modelo em termos da realidade modelada.

A *modelagem exata* através de diagramas exige inscrições formalmente estruturadas, adicionadas à linguagem gráfica, e regras rígidas de manipulação (simulação, execução, ...) do modelo que permitam revelar a dinâmica do sistema modelado.

A passagem de uma modelagem aproximada para uma modelagem exata requer:

- aumento do grau de estrutura das legendas de modo a produzir um conjunto completo e coerente de inscrições que identifiquem os estados possíveis do sistema e a forma de efetuar transições entre os diferentes estados;
- estabelecimento de regras rígidas para reconhecer-se as ocorrências de transições entre os estados do sistema em termos dos processos que este executa.

Há muitos outros exemplos, ou seja, muitas outras ferramentas e abordagens para o processo de modelagem de sistemas. Geralmente, carecem de base conceitual matematicamente definida e/ou não têm aplicabilidade suficientemente geral para tratar a variedade e complexidade dos sistemas em que se baseia a sociedade humana. Esses sistemas são percebidos em diferentes áreas — culturais, científicas e tecnológicas — e podem englobar disciplinas tão distintas quanto Artes, Psicologia, Física, Química, Matemática, Biologia, Economia e Administração.

Convém assinalar que o conceito de sistema, ao ressaltar a existência de interações entre elementos de um conjunto ou o funcionamento do conjunto como uma “estrutura organizada”, não estabelece ipso facto a hipótese de estados globais e de eventos que determinem a evolução temporal do sistema através de “ocupações” sucessivas de estados globais. Pode-se visualizar essa evolução, tal como é feito na Teoria de Redes de Petri, como decorrente de alterações independentes de estados locais sem que seja necessário referir-se a algum feito global de um acontecimento. Em particular, essa teoria permite o tratamento formal de:

- eventos que ocorrem um após o outro (sequência de eventos);
- eventos que ocorrem concorrentemente (paralelismo de eventos);
- eventos que não podem ocorrer juntos, ou seja, cujas ocorrências sejam mutuamente exclusivas (conflito entre eventos).

## Artigo Técnico

A Teoria de Redes de Petri, cuja origem é a tese de doutorado do pesquisador alemão Carl Adam Petri, tem como objetivo criar, a partir de conceitos primitivos fundamentais e matematicamente definidos, um sistema de conceitos derivados e de meios de expressão que permitam modelar sistemas sem o uso de artifícios, isto é, sem apelar para regras e restrições alheias à natureza dos sistemas. Isso inclui, também, a adoção de resultados obtidos em outras abordagens, sejam elas formais ou informais. Essa adoção implica um mapeamento para os conceitos da Teoria de Redes, visando construir uma base conceitual comum para integrar o conhecimento sobre a natureza dos sistemas, tendo em vista que esse conhecimento está, atualmente, distribuído em vários lugares das ciências sem uma forma, geralmente aceita, de integração.

## 4. Construção de Modelos e seu Uso

A partir da definição proposta, na seção 2, para sistema sócio-técnico e tendo em vista a discussão, na seção precedente, que orienta o paradigma da Informática para um processo de modelagem de sistemas, é possível detalhar-se mais o enfoque adotado.

Um sistema informático — sistema sócio-técnico no qual predominam aspectos relacionados ao tratamento de informação com estrutura complexa e/ou aspectos relacionados à monitoração / controle de dinâmica complexa associada ao ambiente externo ou ao próprio sistema sócio-técnico — pode ser analisado em termos de ações a serem realizadas e estruturas de informação a serem processadas, submetidas a condições / restrições / a serem respeitadas.

A modelagem dessa categoria de sistemas, de modo geral, visa:

- descrever uma percepção de aspectos de uma dada realidade;
- descobrir e explorar propriedades qualitativas e quantitativas associadas à porção de realidade percebida;
- prescrever aspectos essenciais de uma realidade fabricada;
- prescrever alternativas de implementação de uma realidade fabricada.

Algumas características da modelagem de sistemas informáticos são consideradas a seguir.

### 4.1 Construção de Modelos

Os ramos de pesquisa que procuram modelar os elementos de um sistema informático — ações, estruturas de informação e condições / restrições — baseiam-se na Matemática e na Lógica, usando-as intensiva e extensivamente quando trata-se de empregar linguagens de representação formais e limitadamente no caso de abordagens aproximadas e informais. Nesse contexto, a modelagem

aproximada e informal de sistemas é considerada etapa importante do processo global pois constitui o momento em que devem ser exercitadas predominantemente a intuição e a criatividade do modelador, ainda sem a disciplina imposta por métodos rigorosos.

A qualidade das representações dos modelos é essencial. Por isso, dois requisitos são exigidos da linguagem de representação que apoia a construção de um modelo formal:

- o significado dos símbolos deve ser rigidamente definido;
- seu uso deve ajudar a intuição para facilitar a compreensão do modelo representado.

O segundo requisito é melhor atendido por linguagem gráfica, onde a possibilidade de estruturação das expressões construídas, o cuidado com as abreviaturas usadas, a forma e a variedade dos símbolos, bem como o esmero na escolha do “layout” dos diagramas, conjugam-se para formar representações com grande poder expressivo e inteligibilidade.

A disponibilidade de ferramentas para a modelagem aproximada e informal justifica-se por vários motivos:

- proporciona o meio mais adequado para a abordagem exploratória do problema a ser resolvido e para o esboço de sua solução; nessa fase inicial, exceto para problemas triviais, não seria justificável o emprego de linguagens formais, tendo em vista o “overhead” imposto pela obediência a regras, sintáticas e semânticas, rígidas.
- torna possível, a um custo tolerável, a comunicação com pessoas não treinadas em formalismos e terminologias da Informática;
- facilita a modificação de projetos em sua fase inicial;
- facilita o desenvolvimento de visões generalizantes, a partir de especificações detalhadas.

Entretanto, deve ser ressaltado que o processo deve culminar com um modelo exato — não necessariamente detalhado — e formal, sem o que o valor dos modelos aproximados e informais fica bastante questionável. Por conseguinte, devem ser estabelecidos caminhos sistemáticos para a passagem de modelos aproximados e informais a modelos exatos e formais.

## 4.2 Uso Prescritivo e Descritivo de Modelos

Há outro caminho sistemático a ser investigado no processo de modelagem de sistemas. É aquele que permita passar, de um modelo exato e formal que represente a especificação abstrata, a *essência*, de um sistema, para outro modelo exato e formal que represente a especificação de uma alternativa de implementação em

## Artigo Técnico

termos de uma dada tecnologia de automação. Em outras palavras, deseja-se sistematizar o processo de incorporação de elementos concretos de uma tecnologia específica de modo a ser possível a “encarnação” (materialização, realização) dos aspectos essenciais em um “sistema real”; essa “encarnação” constituirá a realidade capaz de exibir o comportamento desejado para o sistema.

Esse processo compreende tanto uma atividade de alto nível de abstração, que consiste na alocação de elementos da essência a elementos de nível mais alto — processadores — da tecnologia empregada, quanto uma atividade de refinamento dessa alocação onde procura-se uma organização que otimize o desempenho do “sistema real” a ser construído. Um aspecto importante dentro desse caminho, inequivocamente associado aos chamados “sistemas de informação” ou “sistemas de bancos de dados”, refere-se ao processo de estruturação de objetos de informação complexos e das operações, geralmente complexas, que atuam sobre esses objetos. Tanto os objetos quanto as operações devem ser gerados num processo envolvendo composições / agregações e decomposições / refinamentos sucessivos, através de transformações atuando sobre modelos formais; não devem ser considerados existentes a priori.

A necessidade de estabelecer-se esse tipo de caminho relaciona-se com duas formas de uso de modelos:

- *uso prescritivo*, que pretende definir uma especificação abstrata para um “sistema real” a ser construído, onde características essenciais de uma realidade existente ou imaginada são representadas independentemente de outras formas alternativas de “encarnação”, as quais dariam origem a “sistemas reais” distintos (isto é, a outras “encarnações” da essência desses sistemas), todos obedecendo às prescrições contidas no modelo da essência.

Ao usar-se um modelo de maneira prescritiva, a intenção é exigir que um sistema a ser construído conforme-se à descrição contida no modelo. Através desse uso, consubstancia-se o conceito de “requisito”, aplicável a um sistema a ser construído. Nesse caso, é indispensável que o modelo distinga claramente os componentes “a serem construídos” — sobre os quais o construtor tem grande liberdade de ação — dos componentes “dados a priori” — que constituem restrições à liberdade de ação do construtor. Na maior parte das vezes, tais restrições decorrerão de condições ambientais, externas ao sistema a ser construído, a serem respeitadas por qualquer alternativa de implementação;

- *uso descritivo*, onde pretende-se representar as características essenciais indispensáveis para compreender todos os aspectos estáticos e dinâmicos de uma realidade, independentemente da intenção de construir-se um sistema baseado nessa representação.

Quando o uso é descritivo, a intenção é representar factualmente o comportamento de um sistema existente ou imaginado. Através desse uso do

modelo, investiga-se a estrutura causal dos eventos que a dinâmica da realidade modelada encerra além, é claro, dos elementos ativos e passivos que compõem o modelo.

## 5. Perspectivas

Estabelecer uma visão sistêmica da Engenharia de Software requer um duplo esforço de abstração. Por um lado, essa área do conhecimento deve ser vista como um componente indispensável da área de Informática. Por outro lado, a Informática, mais abrangente do que uma engenharia fundamentada na Ciência da Computação, seria possivelmente melhor entendida como a engenharia fundamentada numa Ciência de Sistemas.

Assim sendo, exprimir perspectivas para a área de Engenharia de Software requer a visualização de perspectivas de uma Engenharia de Sistemas *latu sensu*.

### O quê temos?

Atualmente, dispomos de inúmeras ferramentas para modelagem de sistemas, nem sempre com a maturidade desejável mas, frequentemente, adequadas para percorrer alguns trechos dos caminhos acima mencionados. Sem ser exaustivos e selecionados pela preferência dos autores, são exemplos:

#### a) ferramentas conceituais

Matemática e Lógica de predicados;

Teoria de Redes para sistemas e processos (estados locais, transições locais, sequencialidade e concorrência, indeterminismo);

abordagens para um tratamento formal do relacionamento entre modelos de um sistema (equivalência comportamental, especificação / implementação, ...);

sistema de conceitos formais para a modelagem composicional de informação (IMC).

#### b) ferramentas expressivas

notação da Matemática e da Lógica;

linguagem gráfica da Teoria de Redes com semântica formal;

linguagens lineares para representar redes;

linguagens de anotação formal de redes;

linguagens gráficas de IMC e uma linguagem linear de primeira ordem para IMC (IMCL);

módulos pré-fabricados para atividades de modelagem.

## Artigo Técnico

### c) ferramentas metodológicas

idéias acerca de como passar de um modelo informal / aproximado para um modelo formal / exatô;

abordagens para a integração de Redes de Petri com DER (Diagramas de Entidades-Relacionamentos), DFD (Diagramas de Fluxos de Dados / Controle) e DET (Diagramas de Estados-Transições);

procedimentos para a análise qualitativa e quantitativa de modelos de sistemas.

idéias acerca de métodos para o desenvolvimento de software a partir de modelos de sistemas em redes;

### d) ferramentas computacionais

editores para diagramas de redes;

ambientes de simulação de modelos em redes;

ambientes de desenvolvimento de software;

software para análise de modelos.

## O quê falta?

Alguns aspectos relevantes relacionam-se com:

- metodologia completa e coerente para trabalhar com modelos de sistemas (criação de modelos, relacionamento entre modelos, interfaces etc);
- relacionamento entre os modelos de sistemas e a realidade percebida como sistema, ou seja, uma metodologia de percepção e realização de sistemas;
- integração de resultados provenientes de distintas abordagens para a modelagem de sistemas;
- ferramentas computacionais abrangentes (ambientes de projeto de sistemas: modelagem informal, modelagem formal, análise de modelos);
- conhecimentos mais profundos sobre a natureza de sistemas e uma terminologia adequada.

Acreditamos que a utilização consistente de uma perspectiva sistêmica sobre a Engenharia de Software deva produzir impactos sobre essa disciplina, a nível de pesquisa e, sobretudo, no ensino.

Por um lado, continuar fundamentando essa engenharia na Ciência da Computação representará uma limitação de escopo arbitrária e nefasta. Cada vez mais, o profissional dessa área deverá munir-se de ferramentas e técnicas, mais gerais do que as fornecidas pela Ciência da Computação, visando abordar, competente e seletivamente, a complexidade do sistema abrangente no qual cada

sistema sócio-técnico a ser construído estará inserido, além, é claro, da complexidade do próprio sistema sócio-técnico a ser construído. Através dessa abordagem, será possível definir o conjunto completo de requisitos para o sistema sócio-técnico e estabelecer a melhor interface para a fronteira entre:

- o sub-sistema computacional e o sistema abrangente;
- o sub-sistema computacional e o sub-sistema humano do sistema sócio-técnico.

Por outro lado, não se trata de substituir uma fundamentação científica por outra já existente. Trata-se de ampliar a base científica da Engenharia de Software, agregando à Ciência da Computação uma Ciência de Sistemas, para cujo desenvolvimento e consolidação é desejável a participação de pesquisadores da área de Engenharia de Software.

No campo do ensino, apresenta-se a necessidade de expor o estudante, em fase bem inicial de sua formação profissional, a conceitos, ferramentas e técnicas orientadas para um "pensamento sistêmico", visando dotá-lo de perspectivas mais abrangentes sobre as realidades com as quais será defrontado. Em particular, é importante, nessa fase inicial, impedir que sua mente seja dominada pelo paradigma da "programação sequencial", expondo-o sistematicamente a situações reais onde predominam aspectos de paralelismo e concorrência.

### Bibliografia

As idéias apresentadas neste artigo não são necessariamente originais. Elas podem ser encontradas, de modo diluído, em diversas publicações técnicas recentes. Neste trabalho visou-se organizá-las de modo conciso e, sem a pretensão de tratar exaustivamente o assunto, relacionar conceitos básicos associados à visão sistêmica apresentada. A bibliografia a seguir constitui nosso reconhecimento a diversos autores que vêm abordando esse tema ao longo dos últimos anos.

- P. Checkland (1981): *Systems Thinking, Systems Practice*. John Wiley
- U. Briefs, C. Ciborra, L. Schneider (1983) (eds.): *Systems Design for, with and by the Users*. North-Holland
- B. Wilson (1984): *Systems: Concepts, Methodologies and Applications*. John Wiley
- R. S. Pressman (1987): *Software Engineering: A Practitioner's Approach*. Second Edition. McGraw-Hill
- G. Schäfer with R. Hirschheim et al. (1988) (eds.): *Functional Analysis of Office Requirements: A Multiperspective Approach*. — (John Wiley information systems series) John Wiley
- Proceedings of IFIP WG 8.i Working Conferences. North-Holland

### Artigo Técnico

K. Nygaard (1986): *Program development as a social activity*. In H.-J. Kugler (ed.), *Information Processing 86 (IFIP)*. North-Holland, pp. 189-198.

R. Hirschheim (1986): *Understanding the Office: A Social Analytical Perspective*. *ACM Transactions on Office Information Systems* 4, no. 4

A bibliografia específica para redes de Petri e IMC é:

W. Reisig (1986): *Petri Nets: an Introduction*. Springer Verlag

C. A. Heuser (1989): *Modelagem Conceitual de Sistemas*. (edição preliminar para a IV EBAl) Editorial Kapelusz

R. Durchholz and G. Richter (1989): *Information Modelling by Composition*. The IMC/IMCL Reference Manual. Notas de aula, PUC-RJ/Departamento de Informática. 91 páginas

### Curriculum vitae (Gernot Richter)

Engenheiro Diplomado em Física Técnica (1963) e Doutor em Ciências Técnicas (1966) pela Universidade Técnica de Viena, Áustria. Pesquisador sênior da GMD (Centro Nacional de Pesquisa em Ciência da Computação da R. F. Alemanha), atualmente professor visitante do Departamento de Informática da PUC/Rio. Autor de várias publicações sobre modelagem formal de sistemas com Redes de Petri e de estruturas de informação no contexto de bancos de dados. Coordenador de dois projetos de cooperação entre a Alemanha e o Brasil. Participou em várias comissões técnicas do DIN e da ISO para normas técnicas (sistemas operacionais, bancos de dados, arquitetura de documentos eletrônicos).

### Curriculum vitae (Bruno Maffeo)

Bacharel em Física (PUC/Rio, 1965), Mestre em Física (PUC/Rio, 1967) e Doutor em Ciências (Universidade de Grenoble, França, 1973). Como professor-pesquisador associado do Departamento de Física da PUC/Rio (1973-1982), foi autor de várias publicações envolvendo modelagem matemática e tratamento computacional da estrutura eletrônica de matéria condensada, tendo orientado várias teses de mestrado e doutorado nessa área. Como pesquisador do LNCC (1983-1984), dedicou-se a estudos sobre modelagem matemática e tratamento computacional de ecossistemas lacustres. Como analista de sistemas do SERPRO (1984-1988), dedicou-se a investigações referentes a Metodologias de Desenvolvimento de Software. Atualmente, como professor-pesquisador do Departamento de Informática da PUC/Rio, dedica-se a ensino e pesquisa nas áreas de Engenharia de Software e Modelagem de Sistemas Sócio-Técnicos.