

O artigo publicado em 1972, representa um marco nos estudos de Engenharia de Software ao discutir a importância da modularização e os critérios que devem ser utilizados na decomposição de sistemas em módulos. Embora o tema da modularização já estivesse presente em discussões anteriores, a contribuição de Parnas foi pioneira por propor que a divisão de um sistema não deveria se basear apenas no fluxo de processamento, mas sim no princípio do information hiding (ocultamento de informações), que posteriormente se tornou um conceito fundamental na disciplina.

Logo no início, o autor apresenta os benefícios esperados da modularização: a redução do tempo de desenvolvimento, já que equipes diferentes podem trabalhar em módulos distintos; a flexibilidade do produto, permitindo mudanças localizadas sem necessidade de reescrever o sistema inteiro; e a maior compreensibilidade, pois módulos independentes tornam o sistema mais fácil de entender e manter. No entanto, Parnas chama atenção para um problema: embora houvesse consenso sobre as vantagens da modularização, pouco se discutia sobre os critérios adequados para dividir um sistema em módulos. Seu artigo busca justamente suprir essa lacuna.

Para ilustrar sua proposta, Parnas utiliza o exemplo de um sistema de geração de índices KWIC (Key Word in Context). Ele apresenta duas alternativas de modularização. A primeira segue uma abordagem convencional, dividindo o sistema em módulos correspondentes às etapas do processamento: entrada, deslocamento circular, ordenação alfabética, saída e controle principal. Essa estrutura é intuitiva e reflete o modelo mental da maioria dos programadores da época, que costumavam pensar em termos de fluxogramas. Já a segunda alternativa adota o critério de ocultamento de informações, no qual cada módulo é projetado para esconder determinadas decisões de projeto que podem mudar ao longo do tempo, como a forma de armazenamento de linhas, o algoritmo de ordenação ou o método de deslocamento circular.

A comparação entre as duas decomposições mostra claramente a superioridade da segunda. Na primeira, alterações em aspectos básicos, como o formato de entrada ou a estratégia de armazenamento, propagam-se por todo o sistema, exigindo modificações em diversos módulos. Na segunda, essas mudanças ficam confinadas ao módulo responsável, preservando a estabilidade dos demais. Assim, a decomposição baseada em ocultamento de informações promove maior independência entre módulos, facilita o desenvolvimento paralelo e aumenta a flexibilidade do sistema frente a mudanças.

Outro ponto importante abordado por Parnas é a distinção entre decomposição de sistemas e hierarquia de programas. Embora relacionadas, essas propriedades não são equivalentes: é possível ter um sistema bem modularizado sem, necessariamente, seguir uma estrutura hierárquica rígida. Contudo, quando ambas coexistem, os benefícios são potencializados, permitindo reutilização de módulos em diferentes contextos e simplificação do processo de manutenção.

O artigo também discute desafios práticos da aplicação desse modelo, especialmente no que diz respeito à eficiência. Uma modularização orientada ao ocultamento de informações pode resultar em maior sobrecarga de chamadas de funções e, portanto, em perda de desempenho. Parnas reconhece essa limitação, mas propõe alternativas de implementação

que atenuem o problema, como o uso de montadores e técnicas de integração que preservam a independência conceitual dos módulos sem comprometer a performance.

A relevância da obra transcende o exemplo específico do KWIC. O princípio de ocultamento de informações, defendido por Parnas, antecipou de forma notável conceitos que se consolidariam posteriormente na programação orientada a objetos e na engenharia de software moderna. Sua visão de que módulos devem ser definidos pelo conhecimento que escondem — e não pelas funções que executam — estabeleceu as bases para o desenvolvimento de sistemas mais robustos, adaptáveis e de fácil manutenção.

Em síntese, o artigo de Parnas não apenas propõe um critério inovador para modularização, mas também introduz uma mudança de paradigma no modo de pensar a construção de software. Ao deslocar o foco do fluxo de controle para as decisões de projeto suscetíveis a mudanças, o autor oferece uma abordagem mais duradoura e eficaz para lidar com a complexidade crescente dos sistemas. Esse trabalho permanece como leitura obrigatória para estudantes e profissionais de Engenharia de Software, sendo considerado um dos pilares teóricos da disciplina.