システムの分析や設計を、いかに正確かつ迅速に、また分かりやすく行うかは、ソフトウェアエンジニアリングの主要なテーマの一つです。

Làm thế nào để phân tích và thiết kế một hệ thống một cách chính xác, nhanh chóng và dễ hiểu là một trong những chủ đề chính của kỹ thuật phần mềm.

それまでにも様々なアプローチが提案されましたが、中でも「構造化分析と構造化設計」は、広く認知され定義している代表的な手法と言えます。

Nhiều cách tiếp cận khác nhau đã được đề xuất cho đến nay, nhưng trong số đó, “phân tích cấu trúc và thiết kế cấu trúc” có thể nói là một phương pháp đại diện được công nhận và định nghĩa rộng rãi.

構造化分析と構造化設計は、せべてシステムは、それより小さな要素の組み合わせで構成され、その要素も、またさらに小さい要素の組み合わせで構成されると考えます。

Phân tích hóa cấu tạo và thiết kế hóa cấu tạo là toàn bộ hệ thống được cấu tạo bởi sự kết hợp những yếu tố nhỏ và những yếu tố này được cấu tạo từ những sự kết hợp yếu tố nhỏ hơn nữa.

例えば、車を設計する際、必要なネジの数やサイズを決めようとしても、それは極 めて難しいことです。しかし、車の構成要素をボディ、シャシー、エンジン、内装に分 け、次にエンジンの構成要素を本体、給排気、ピストンとクランク、に分け、さらに それら個別の設計を進めれば、最後には、おのずと必要なネジの数やサイズが明らかに。

Ví dụ, khi thiết kế một chiếc xe hơi, việc cố gắng xác định số lượng và kích thước ốc vít cần thiết có thể cực kỳ khó khăn. Tuy nhiên, nếu bạn chia các bộ phận của ô tô thành thân, khung, động cơ và nội thất, sau đó chia các bộ phận của động cơ thành thân chính, cấp / xả khí, piston và tay quay, và van, sau đó tiến hành thiết kế riêng của chúng, cuối cùng bạn có thể hoàn thành. Đương nhiên, số lượng và kích thước của vít cần phải rõ ràng.

このように、複雑なものを設計する際には、いきなり最小単位の構成要素の組み合 わせを考えるのではなく、まとまった単位で少しずつ細かく分割していきます。それに より、自然に最小単位の構成要素にたどり着くことができます。あるいは逆に、単純な ものを組み合わせて少しずつ複雑な構成要素を作り、その構成要素を組み合わせていく ことで、最終的に極めて複雑な設計ができる場合もあります。これらがまさに、構造化

Bằng cách này, khi thiết kế một sản phẩm phức tạp, thay vì đột nhiên nghĩ đến sự kết hợp của các thành phần của đơn vị nhỏ nhất, chúng ta chia nó thành các đơn vị nhỏ hơn từng chút một. Điều này cho phép chúng tôi đạt đến đơn vị nhỏ nhất của các thành phần. Hoặc ngược lại, bạn có thể kết thúc với một thiết kế cực kỳ phức tạp bằng cách kết hợp những thứ đơn giản để tạo ra các thành phần phức tạp từng chút một và sau đó kết hợp các thành phần. Chúng được cấu trúc chính xác

構造化分析と構造化設計を用いたシステム開発では、システムの機能をサブシステムやモジュールに分割し、サブシステム間やモジュール間のデータの流れを定義します。その際、データの流れを直感的に把握しやすい「DFD（Data Flow Diagram）」という図よく用いられます。

Phát triển hệ thống sử dụng phân tích hóa cấu tạo và thiết kế hóa cấu tạo là chia chức năng của hệ thống thành các hệ thống con và các module, mô tả luồng dữ liệu giữa các hệ thống con và các module. Trong trường hợp đó, người ta sử dụng sơ đồ để dễ nắm bắt trực quan luồng dữ liệu (DFD)。

視点を変えることにより全体の構造を見渡したり、特定の部分を詳細に分析したり できる構造化分析と構造化設計は、我々人間にとって、自然で理解しやすい考え方と言 えるでしょう。