



Kiến trúc phần mềm hiện đại

Giới thiệu về Kiến trúc phần mềm

Nội dung chính

- Định nghĩa kiến trúc phần mềm
- Các khái niệm cơ bản
- Các khía cạnh của kiến trúc
- Các thuộc tính chất lượng

Tài liệu tham khảo

- *Software Architecture in Practice*, chương 2, 4
- *Essential Software Architecture*, chương 1, 2, 3

Từ đơn giản đến phức tạp

- Ứng dụng A: giải phương trình $ax^2 + bx + c = 0$
- Ứng dụng B: giúp quản lý nhân sự của một công ty lớn

Từ đơn giản đến phức tạp

- Khi độ phức tạp của hệ thống cần xây dựng tăng lên, việc thiết kế không còn đơn thuần là vấn đề giải thuật và cấu trúc dữ liệu
- Thiết kế cấu trúc tổng thể của hệ thống chính là vấn đề quan trọng cần giải quyết

Kiến trúc phần mềm

- Kiến trúc phần mềm của một chương trình hoặc hệ thống máy tính là cấu trúc (hoặc các cấu trúc) của hệ thống đó, bao gồm
 - Các thành phần (phần hệ)
 - Các thuộc tính có thể thấy được từ bên ngoài của các thành phần
 - Mỗi quan hệ giữa các thành phần
- Còn rất nhiều định nghĩa khác
 - <http://www.sei.cmu.edu/architecture/start/community.cfm>

Các thuộc tính thấy được

- Chính là những giả thiết mà các thành phần khác có thể đưa ra đối một thành phần, ví dụ
 - Các dịch vụ do thành phần cung cấp
 - Các đặc tính về hiệu năng
 - Xử lý lỗi
 - Việc sử dụng tài nguyên chung
 - ...

Kiến trúc và trừu tượng hóa

- Một kiến trúc là sự trừu tượng hóa của hệ thống, che dấu nhiều điểm chi tiết không cần thiết của các thành phần mà không ảnh hưởng đến cách:
 - Sử dụng
 - Được sử dụngcủa thành phần với các thành phần khác trong hệ thống

Kiến trúc phần mềm giúp

- Hiểu hệ thống
 - Nhìn thấy được các phần chính của hệ thống, các ràng buộc
- Tái sử dụng
 - Tái sử dụng các thành phần, các mẫu tổ chức
- Xây dựng
 - Như là một bản hướng dẫn để xây dựng hệ thống

Kiến trúc phần mềm giúp

- Tiến hóa
 - Tiến hóa các thành phần trong hệ thống
- Phân tích
 - Phân tích sâu hơn về các yêu cầu, ràng buộc của hệ thống
- Quản lý
 - Xem xét quá trình phát triển hệ thống

Các khái niệm cơ bản

- *Thành phần*: còn được gọi là phân hệ, đơn vị; là một phần của hệ thống, ví dụ: bộ lọc (filters), cơ sở dữ liệu
- *Kết nối*: phương tiện trung gian cho các tương tác của các thành phần, ví dụ: gọi hàm, pipe, event broadcast

Các khái niệm cơ bản

- *Kiểu kiến trúc*: định nghĩa một tập các kiến trúc, bao gồm
 - Các thành phần và kết nối
 - Các ràng buộc cho các thành phần và các kết nối, bao gồm
 - Các luật về topology
 - Các chuẩn giao diện
 - Các thuộc tính cần thiết
 - Các phân tích (cho phép suy luận ra những thuộc tính quan trọng)

Các khái niệm cơ bản

- *Kiến trúc miền chuyên biệt* (domain specific architecture)
 - Định nghĩa các thành phần và cách tương tác
 - Áp dụng cho một miền ứng dụng cụ thể nào đó
 - Xác định các chức năng của miền ứng dụng được hiện thực ở thành phần nào
- *Mẫu thiết kế*
 - Là giải pháp thiết kế có thể dùng cho các vấn đề thường xuyên xảy ra trong thiết kế phần mềm

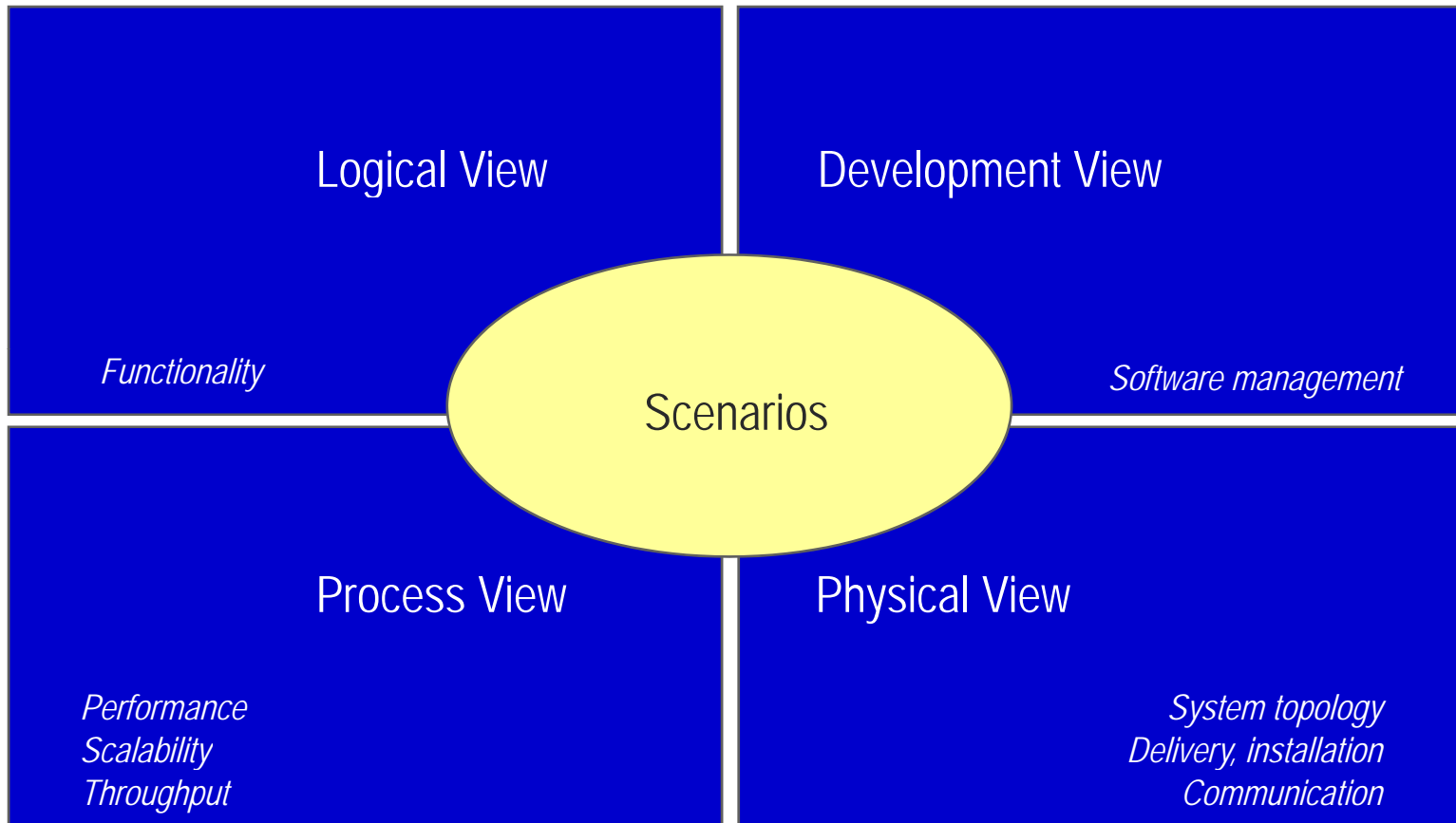
Các khái niệm cơ bản

- *Mô hình kiến trúc*
 - Là tài liệu mô tả một phần hoặc toàn bộ kiến trúc của hệ thống
- *Architecture View*
 - Là một cách quan sát kiến trúc (khía cạnh của kiến trúc)

Các khía cạnh kiến trúc

- Architectural view
- *Khía cạnh logic*: Mô tả những phần tử chính và các mối quan hệ giữa chúng
- *Khía cạnh xử lý (process view)*: Mô tả hoạt động, sự giao tiếp, đồng thời của hệ thống
- *Khía cạnh vật lý (physical view)*: Mô tả làm thế nào để các tiến trình và thành phần chính được hiện thực trên phần cứng.
- *Khía cạnh phát triển (development view)*: Mô tả tổ chức bên trong của phần mềm

Mô hình 4+1



Những người tham gia

- Quyết định về kiến trúc của một hệ thống thường chịu ảnh hưởng của nhiều người từ các khía cạnh khác nhau
- Người quản lý
 - Chi phí thấp
- Nhân viên thị trường
 - Thời gian đưa ra thị trường ngắn, chi phí thấp
- Khách hàng
 - Giao hàng đúng hạn, ít thay đổi
- Người dùng cuối (end user)
 - Hành vi, hiệu năng, bảo mật
- Nhân viên bảo trì
 - Tính có thể thay đổi

Các thuộc tính chất lượng

- Hiệu năng
 - Throughput
 - Thời gian đáp ứng (response time)
 - Deadlines
- Tính mở rộng (scalability)
- Tính có thể thay đổi (modifiability)
- Bảo mật
- Sự sẵn sàng (availability)
- Khả năng tích hợp (integration)

Các thuộc tính chất lượng

- Tính có thể di chuyển (portability)
- Tính có thể kiểm tra (testability)
- Tính có thể hỗ trợ (supportability)
- Tính khả dụng (usability)



Tóm tắt

- Định nghĩa kiến trúc phần mềm
 - Thành phần, thuộc tính nhìn từ bên ngoài, quan hệ
- Các khái niệm cơ bản
 - Kiểu kiến trúc, mẫu thiết kế,...
- Các khía cạnh của kiến trúc
 - Logic, xử lý, vật lý, phát triển
- Các thuộc tính chất lượng
 - Hiệu năng, tính mở rộng, bảo mật,...