

Công nghệ phần mềm

2

Các quy trình
phần mềm

Các quy trình phần mềm

1

Những đặc trưng của PM

2

Quy trình PM

3

Các mô hình phát triển PM

4

Đặc tính chất lượng PM

Các quy trình phần mềm

1

Những đặc trưng của PM

2

Quy trình PM

3

Các mô hình phát triển PM

4

Đặc tính chất lượng PM

Những đặc trưng của PM

Độ tin cậy, độ chính xác

- Độ tin cậy (Reliability) của phần mềm là gì?
 - Độ tin cậy của phần mềm là 1 thuật ngữ trừu tượng, có thể coi đó là khả năng người dùng tin tưởng được vào hệ thống:
 - Không có các lỗi nguy hiểm đến việc vận hành hệ thống mà chưa biết trước
 - Hệ thống phục hồi tốt từ lỗi
 - Thời gian ngừng (downtime) thấp
 - Độ chính xác cao
 - Tính chịu đựng cao
 - ...

Những đặc trưng của PM

Độ tin cậy, độ chính xác

- Độ chính xác (Correctness) của phần mềm là gì?
 - ☐ Độ chính xác của phần mềm là khả năng phần mềm hoạt động đúng như mong đợi
- Độ chính xác và độ tin cậy có ý nghĩa gần giống nhau, tuy nhiên, xét về mặt ngữ nghĩa:
 - 1 hệ thống đáng tin cậy chắc chắn là 1 hệ thống chính xác
 - 1 hệ thống chính xác chưa chắc là 1 hệ thống đáng tin cậy

Độ thực thi, tính vận hành

- Độ thực thi (Operability) là gì?

□ Độ thực thi là 1 giá trị định lượng phần công việc mà 1 hệ thống có thể làm được trong 1 khoảng thời gian nhất định:

- Hệ thống kiểm tra hành lí ở sân bay quét được 100 hành lí mỗi phút
- Hệ thống photocopy có thể quét, xử lí, in các bản sao với tốc độ 1 trang mỗi giây
- ...

Độ thực thi, tính vận hành

- Tính vận hành cùng nhau (Interoperability) là gì?
 - Tính vận hành cùng nhau là khả năng 1 hệ thống phần mềm có thể cùng tồn tại và tương tác với các hệ thống khác:
 - Trong 1 hệ thống nhúng, phần mềm phải có khả năng giao tiếp với các thiết bị khác sử dụng cấu trúc bus và giao thức tiêu chuẩn
 - ...

Những đặc trưng của PM

Tính dễ bảo trì

- Tính dễ bảo trì (Maintainability) của phần mềm là gì?
 - Tính dễ bảo trì của phần mềm là khả năng phần mềm có thể được bảo trì nhằm:
 - Cô lập, sửa chữa lỗi hoặc nguyên nhân gây lỗi
 - Sửa chữa, thay thế các thành phần bị hỏng hoặc hoạt động sai mà không ảnh hưởng đến các thành phần hoạt động đúng khác
 - Ngăn chặn sự cố bất ngờ
 - Tối đa hóa tuổi thọ, hiệu quả, độ tin cậy,...
 - Đáp ứng yêu cầu mới
 - Đối phó với việc thay đổi môi trường hoạt động

Các quy trình phần mềm

1

Những đặc trưng của PM

2

Quy trình PM

3

Các mô hình phát triển PM

4

Đặc tính chất lượng PM

Các quy trình phần mềm

1

Những đặc trưng của PM

2

Quy trình PM

3

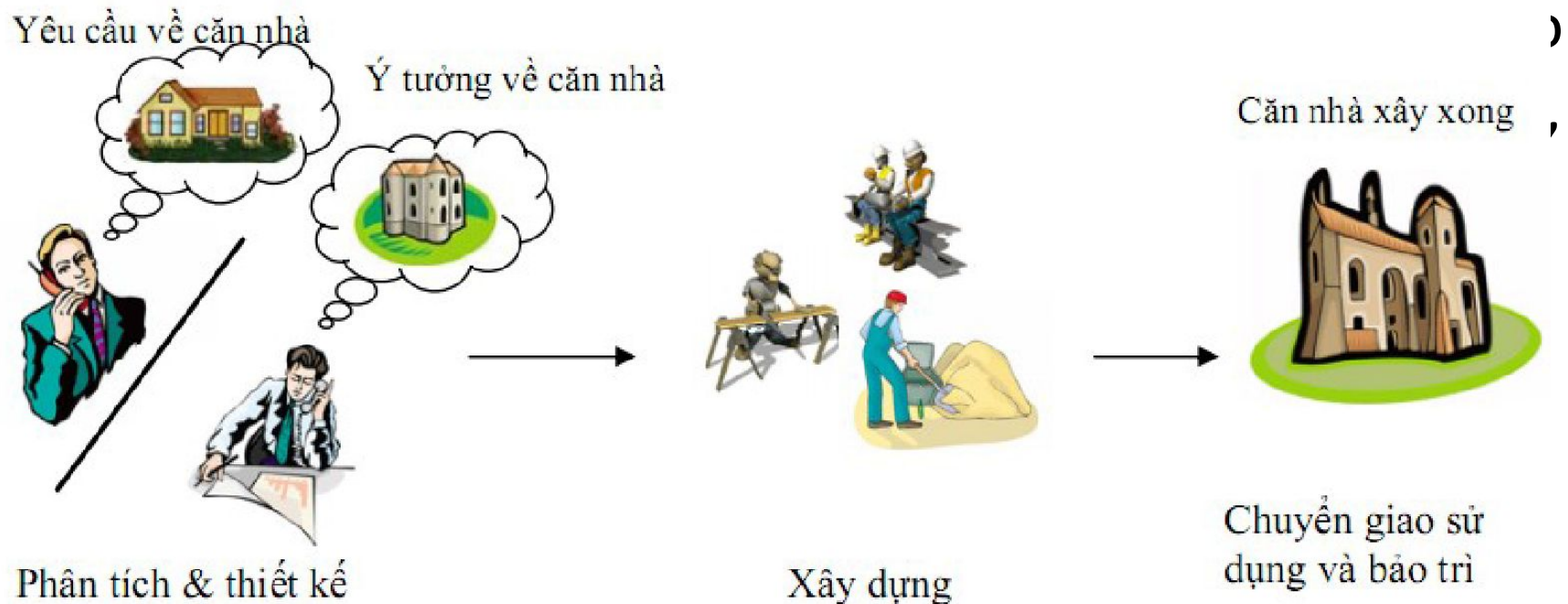
Các mô hình phát triển PM

4

Đặc tính chất lượng PM

Vòng đời của PM

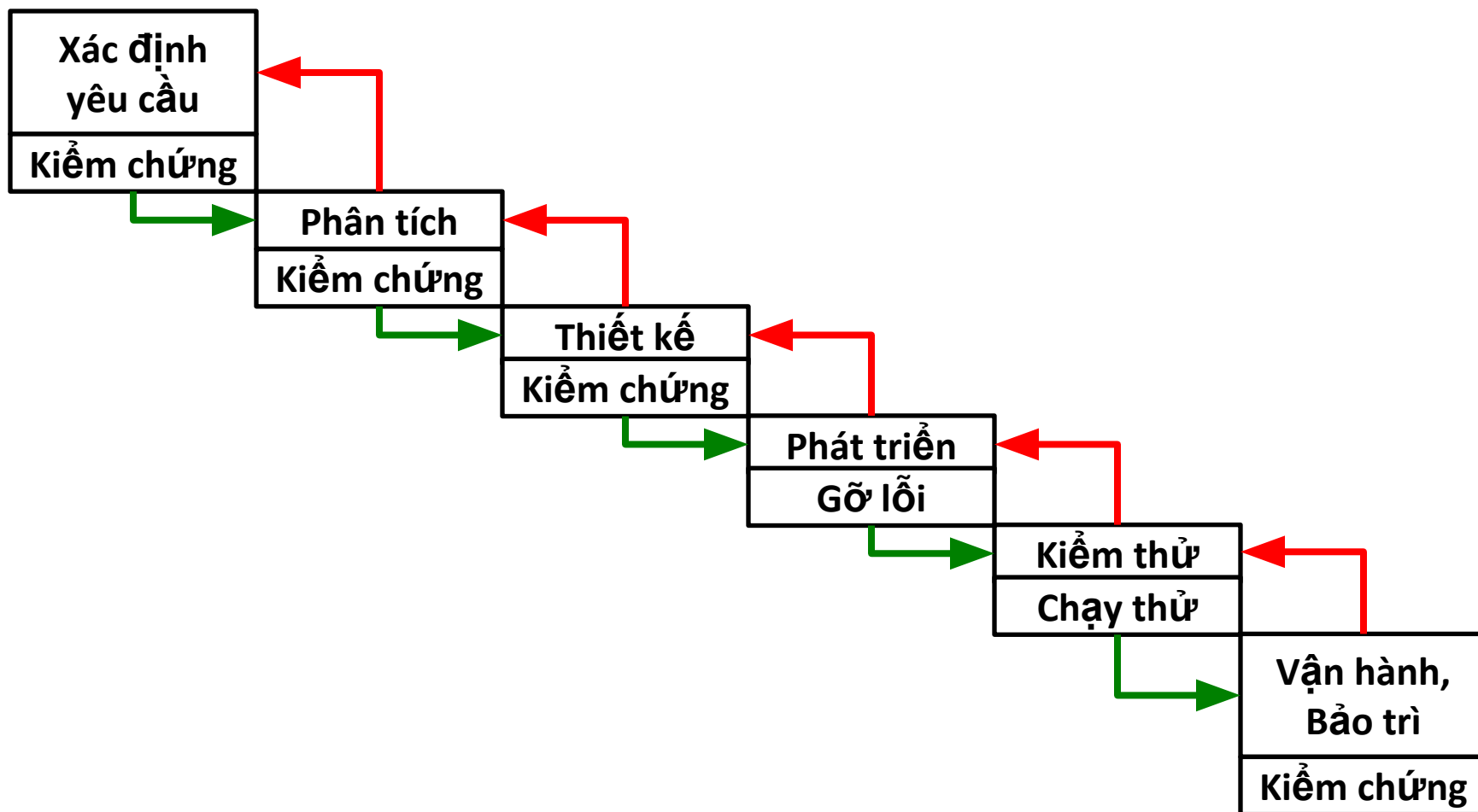
- Vòng đời (Life cycle) của phần mềm là gì?
 - Vòng đời của 1 phần mềm được tính từ khi phần



- Triển khai (Implementation)
- Bảo trì (Maintenance)

Vòng đời của PM

- Mô hình vòng đời phần mềm của Boehm:



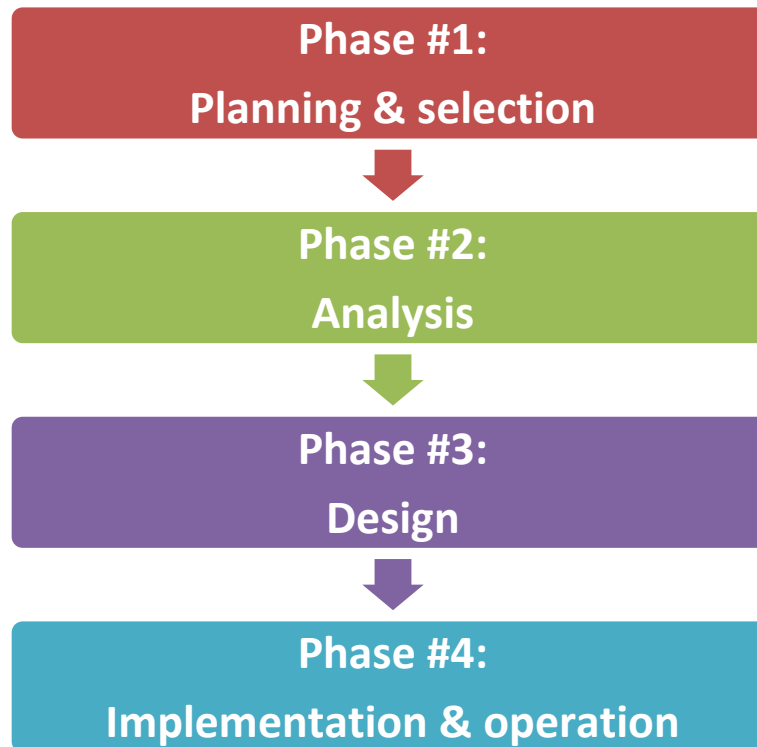
Vòng đời của PM

- Các đặc trưng của vòng đời phần mềm:
 - Pha phân tích, thiết kế có vai trò quyết định đến chất lượng phần mềm, chiếm phần lớn thời gian và công sức trong quy trình phần mềm
 - Pha thiết kế, phát triển từ trên xuống (top-down) còn pha kiểm thử từ dưới lên (bottom-up)
 - Trước khi chuyển sang pha kế tiếp, cần đảm bảo pha hiện tại đã được kiểm thử và không còn lỗi
 - Cần có cơ chế kiểm tra để đảm bảo mỗi pha không gây lỗi cho pha sau

Quy trình PM

Quy trình PM

- Quy trình phần mềm chính là vòng đời phần mềm được biểu diễn, mô hình hóa, chi tiết hóa. Cách biểu diễn khác nhau tùy mỗi người, mỗi quy chuẩn:



Các quy trình phần mềm

1

Những đặc trưng của PM

2

Quy trình PM

3

Các mô hình phát triển PM

4

Đặc tính chất lượng PM

Các quy trình phần mềm

1

Những đặc trưng của PM

2

Quy trình PM

3

Các mô hình phát triển PM

4

Đặc tính chất lượng PM

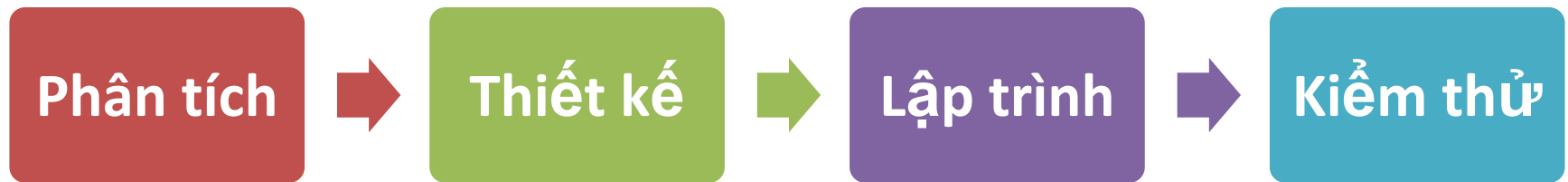
Các mô hình phát triển PM

- Các loại mô hình phát triển phần mềm (SDLC – Software Development Life Cycle):

- Mô hình tuyến tính
- **Mô hình thác nước**
- Mô hình bản mẫu
- Mô hình tăng trưởng
- **Mô hình xoắn ốc**
- **Mô hình chữ V**
- ...

Các mô hình phát triển PM

- Mô hình tuyến tính là mô hình phát triển PM sơ khai nhất, bao gồm 4 giai đoạn:



- Điểm yếu của mô hình tuyến tính:

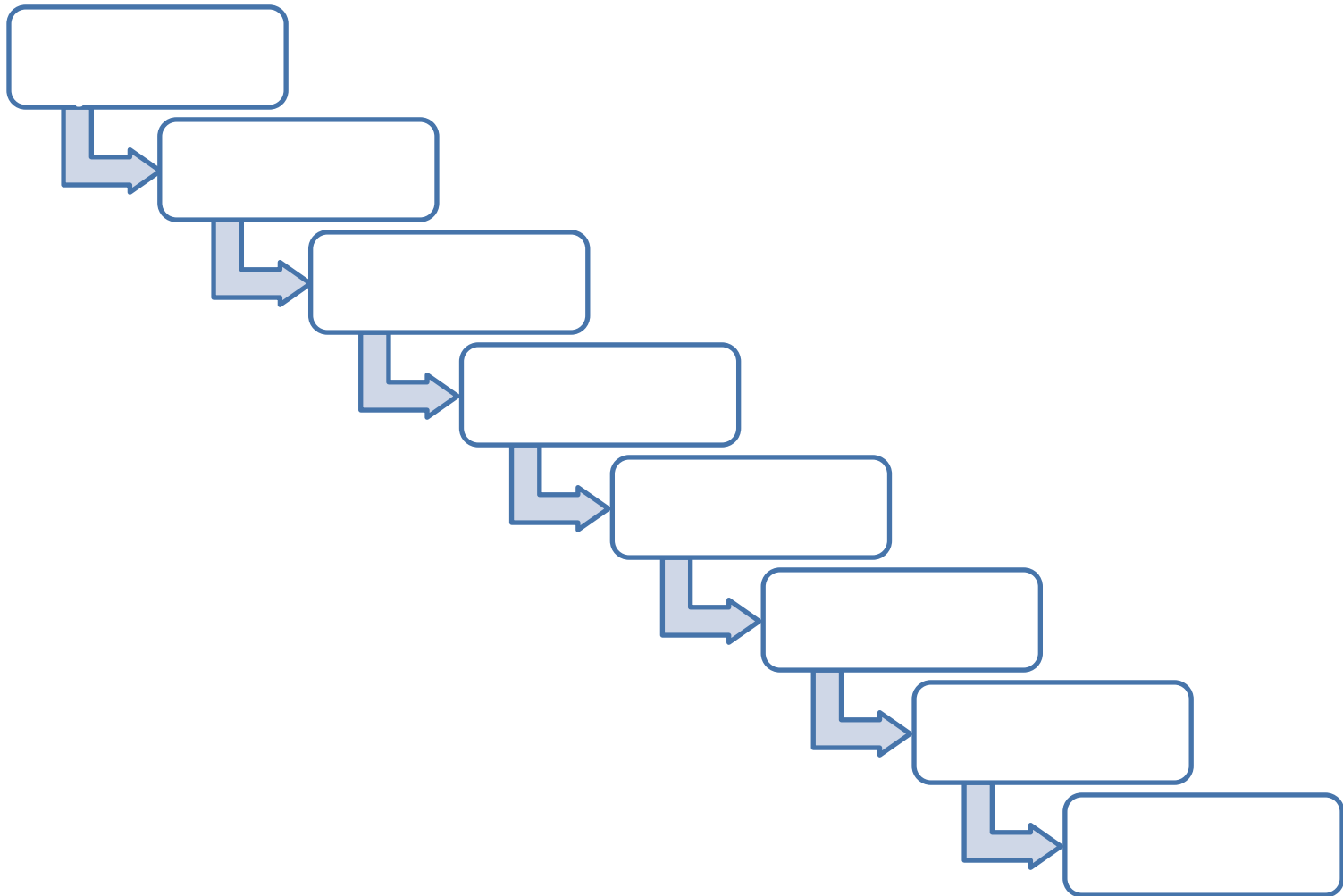
- Thường các dự án ít khi tuân theo dòng tuần tự của mô hình mà thường có lặp lại
- Khách hàng ít khi tuyên bố khi nào hết yêu cầu
- Thời gian chờ sản phẩm lâu
- Không có kiểm thử trong quá trình phát triển mà chỉ kiểm thử cuối quy trình

Mô hình thác nước

- Mô hình thác nước (Waterfall model) là 1 mô hình phát triển phần mềm mà trong đó quy trình trông giống như 1 dòng chảy, với các pha được thực hiện theo trật tự nghiêm ngặt và không có sự quay lui hay nhảy vượt pha.
- Do Winston W. Royce đề xuất năm 1970, thay thế cho phương pháp “code and fix”.

Các mô hình phát triển PM

Mô hình thác nước



Mô hình thác nước

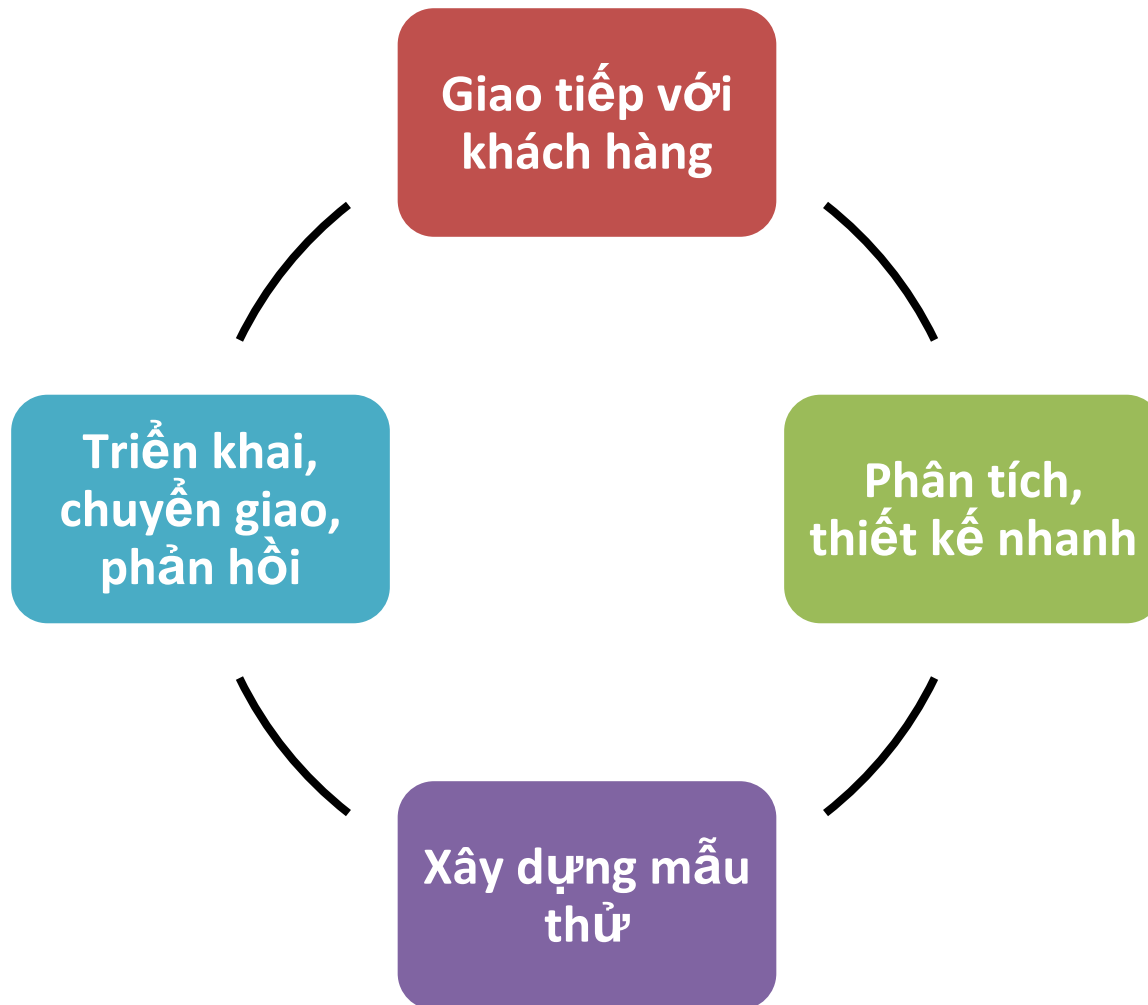
- Ưu điểm:
 - Pha kế tiếp chỉ bắt đầu khi pha hiện tại hoàn thành và không còn lỗi
 - Người dùng cuối và khách hàng biết rõ quy trình
- Nhược điểm:
 - Không linh hoạt
 - Không có quay lui, không kiểm thử giữa quy trình
 - Phải đặc tả hệ thống hoàn chỉnh ngay từ đầu
- Tuy nhiên, mô hình này phản ánh đúng thực tế công nghệ nên có thể xem đây là mô hình phổ biến nhất

Mô hình bản mẫu

- Mô hình bản mẫu (Prototype model) là 1 mô hình phát triển phần mềm mà trong đó phần mềm được tạo ra thông qua các bản mẫu. Các bản mẫu này sẽ được lấy ý kiến khách hàng và có thể được chỉnh sửa, phát triển tiếp hoặc loại bỏ đến khi có phần mềm hoàn chỉnh.

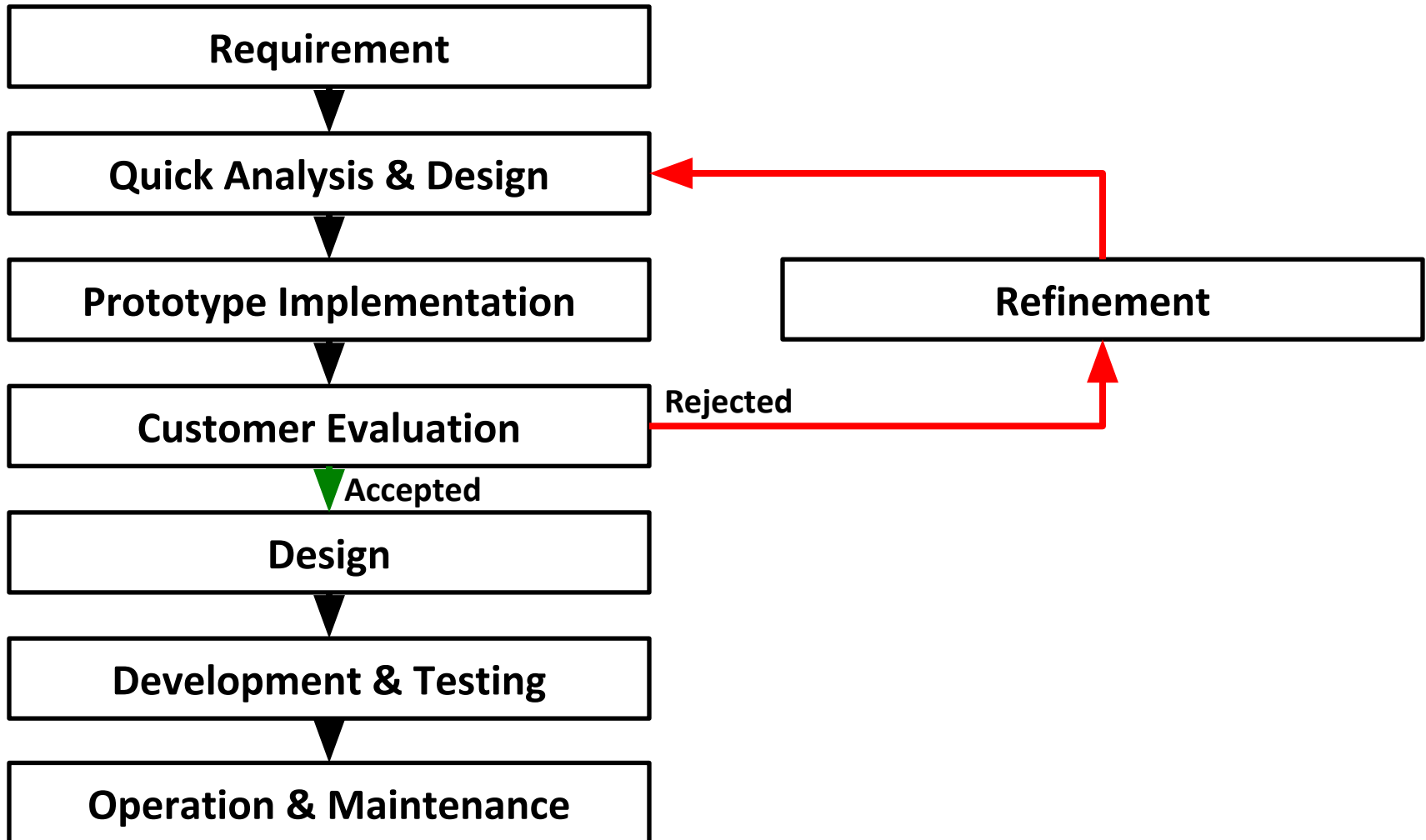
Các mô hình phát triển PM

Mô hình bản mẫu



Các mô hình phát triển PM

Mô hình bản mẫu



Mô hình bản mẫu

- Ưu điểm:

- Khách hàng tương tác sớm với hệ thống
- Khách hàng cùng làm việc với nhà phát triển
- Linh hoạt, mềm dẻo

- Nhược điểm:

- Nhanh nhưng kém chất lượng
- Khó bảo trì
- Bị khách hàng hối thúc
- Nhà phát triển dễ rơi vào tình trạng “code and fix”

Mô hình tăng trưởng

- Mô hình tăng trưởng (Incremental model) là 1 mô hình phát triển phần mềm mà trong đó sản phẩm lõi với các yêu cầu cơ bản nhất của hệ thống được phát triển trước, các chức năng khác sẽ được bổ sung sau. Quy trình này sẽ được lặp lại để hoàn thiện dần hệ thống.
- Do D. R. Graham đề xuất năm 1988, kết hợp giữa mô hình tuần tự và tính lặp của mô hình bản mẫu.

Các mô hình phát triển PM

Mô hình tăng trưởng

Phân tích
thiết kế

Lập trình

Kiểm thử

Chuyển
giao #1

Phân tích
thiết kế

Lập trình

Kiểm thử

Chuyển
giao #2

Phân tích
thiết kế

Lập trình

Kiểm thử

Chuyển
giao #3

Mô hình tăng trưởng

- Ưu điểm:

- Chức năng chính, chức năng có độ rủi ro cao sẽ được thực hiện trước
- Sau mỗi vòng đều có thể chuyển giao
- Giảm thiểu rủi ro cho toàn bộ dự án

- Nhược điểm:

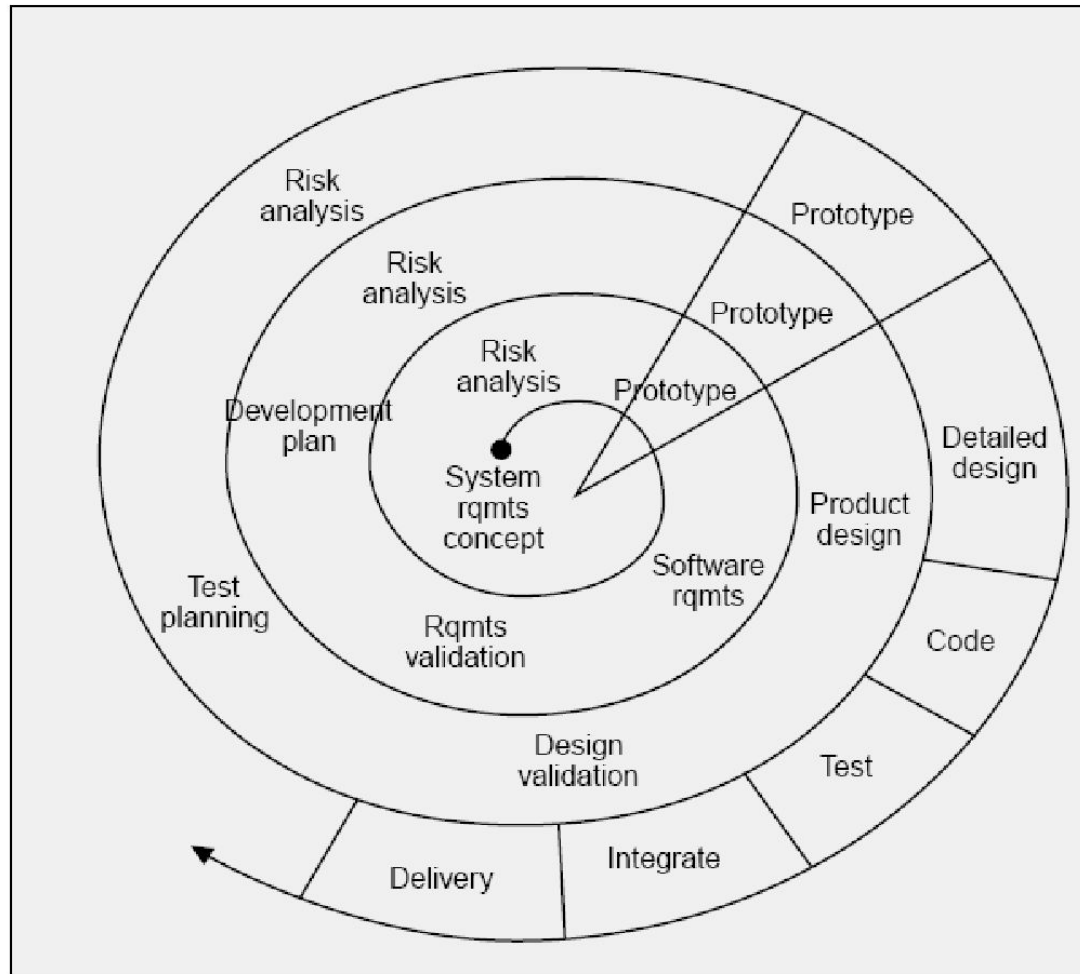
- Phải xác định chức năng đầy đủ và hoàn chỉnh trước khi qua vòng sau
- Khó áp dụng cho các hệ thống không phân chia rõ ràng các thành phần, module

Mô hình xoắn ốc

- Mô hình xoắn ốc (Spiral model) là 1 mô hình phát triển phần mềm mà trong đó quy trình phát triển được thực hiện qua nhiều lần lặp, mỗi lần lặp trải qua 4 pha:
 - Xác định mục tiêu, các phương án và ràng buộc
 - Đánh giá các phương án
 - Thiết kế và tạo lập 1 nguyên mẫu
 - Thử nghiệm nguyên mẫu
- Do Barry Boehm đề xuất năm 1988 để khắc phục các nhược điểm của mô hình thác nước.

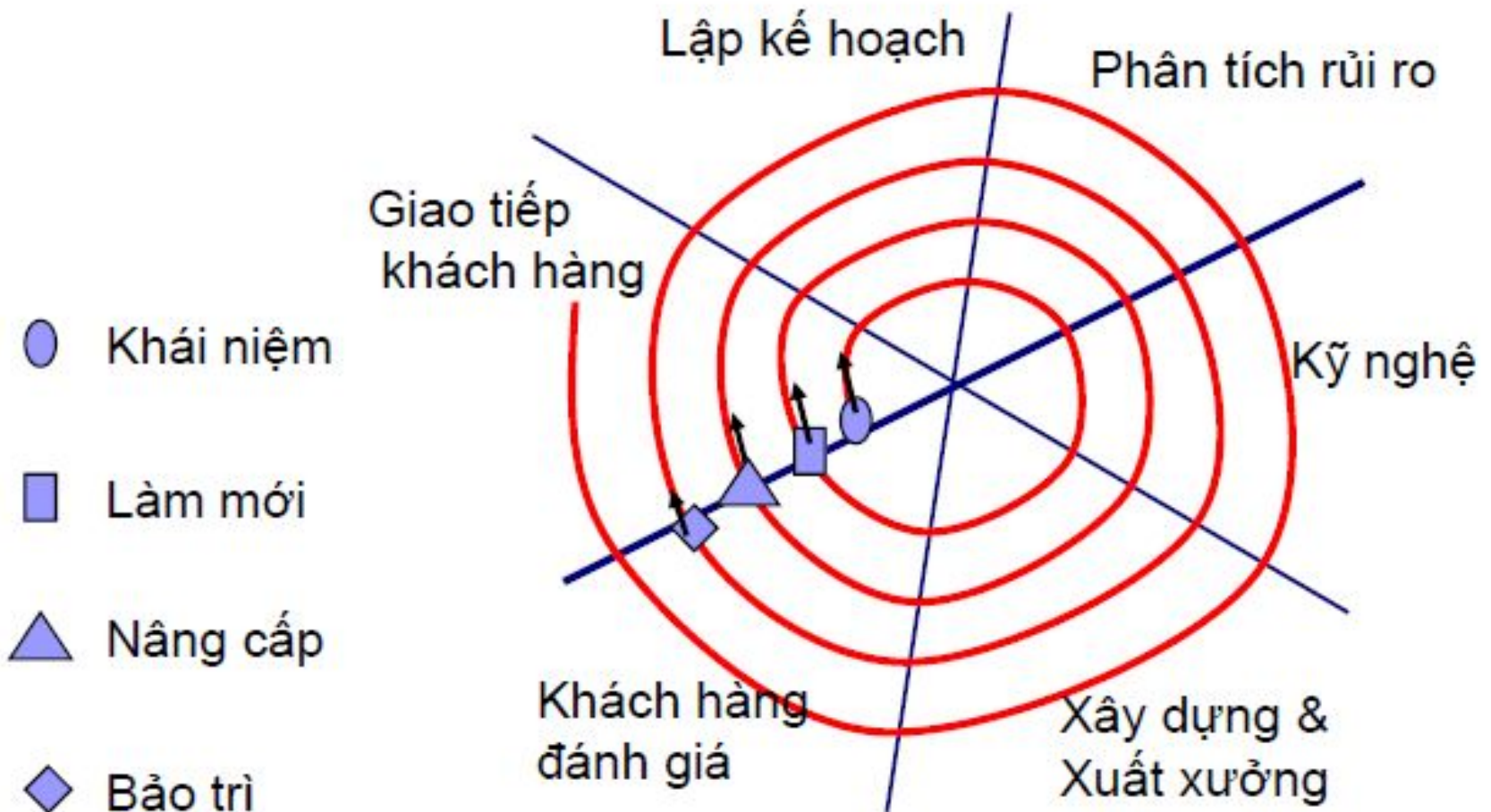
Các mô hình phát triển PM

Mô hình xoắn ốc



Các mô hình phát triển PM

Mô hình xoắn ốc



Mô hình xoắn ốc

- Ưu điểm:

- Tạo bản mẫu sớm
- Cho phép khách hàng tham gia vào quy trình phát triển phần mềm
- Ưu tiên thực hiện trước các chức năng quan trọng

- Nhược điểm:

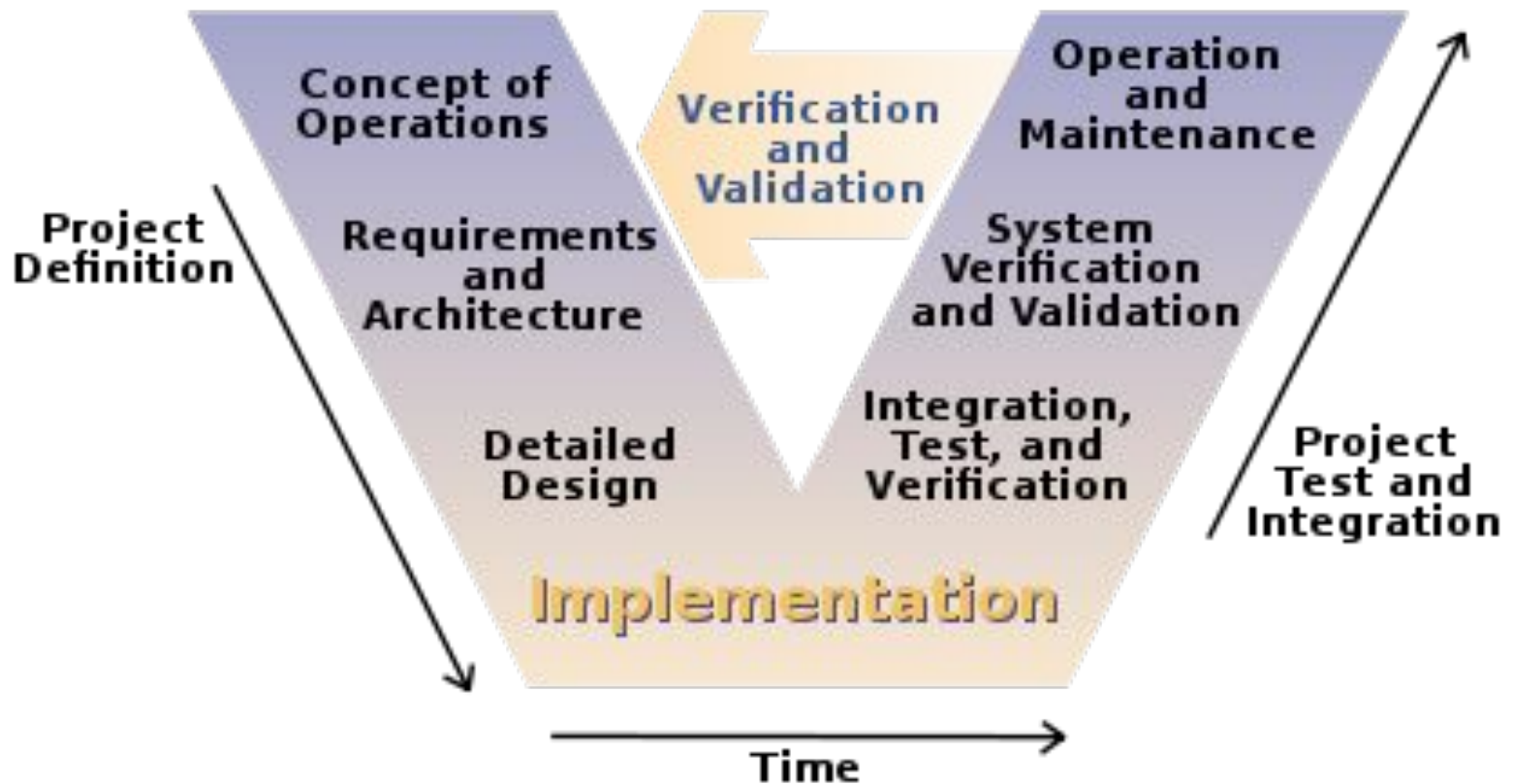
- Việc đánh giá rủi ro tốn nhiều chi phí
- Phức tạp, khó quản lí
- Khó thuyết phục khách hàng

Mô hình chữ V

- Mô hình chữ V (V model) là 1 mô hình phát triển phần mềm được mở rộng từ mô hình thác nước, nhưng thay vì đi xuống theo 1 hướng tuyến tính, quy trình sẽ được đảo chiều tại pha lập trình (Implementation) để thể hiện mối tương quan giữa việc phân tích thiết kế và việc kiểm thử hệ thống.

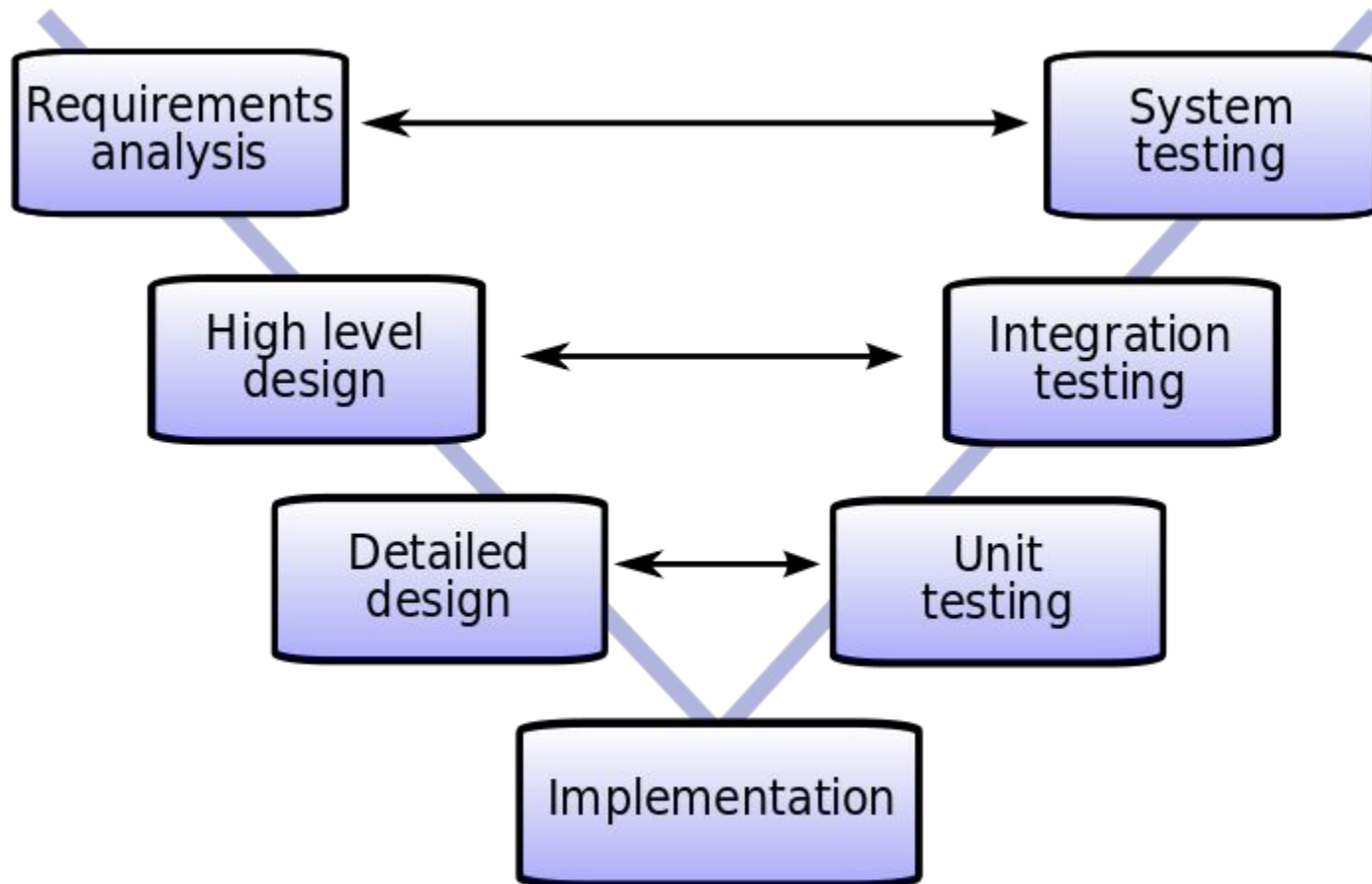
Các mô hình phát triển PM

Mô hình chữ V



Các mô hình phát triển PM

Mô hình chữ V



Mô hình chữ V

- Ưu điểm:

- Đơn giản, dễ hiểu
- Tại 1 thời điểm chỉ có 1 pha được thực hiện
- Thể hiện rõ mối liên quan giữa phân tích, thiết kế và kiểm thử

- Nhược điểm:

- Không có quay lui, không kiểm thử giữa quy trình
- Không phù hợp với các hệ thống lớn hoặc các phần mềm phát triển trong thời gian dài
- Phần mềm hoàn chỉnh chỉ có ở cuối quy trình

Các quy trình phần mềm

1

Những đặc trưng của PM

2

Quy trình PM

3

Các mô hình phát triển PM

4

Đặc tính chất lượng PM

Các quy trình phần mềm

1

Những đặc trưng của PM

2

Quy trình PM

3

Các mô hình phát triển PM

4

Đặc tính chất lượng PM

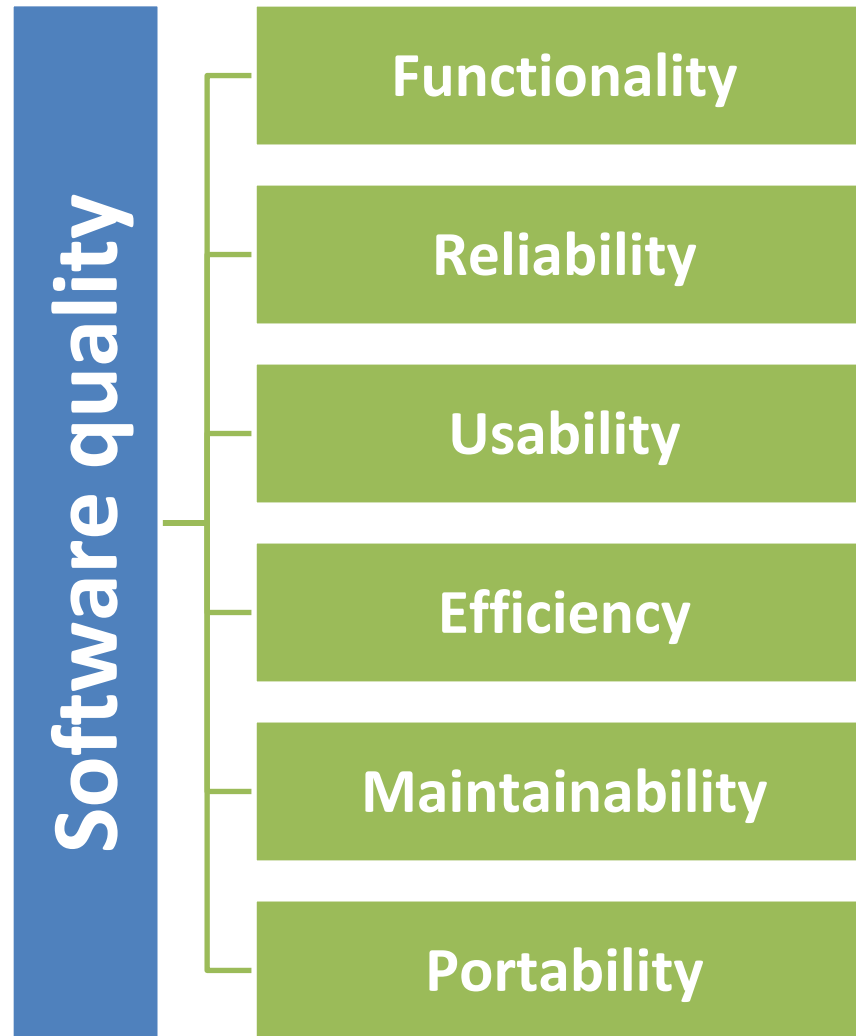
Đặc tính chất lượng PM

- Các tiêu chuẩn quốc tế quy định đặc tính chất lượng phần mềm:

- ISO, ACM, IEEE, DOD...
- DOD-STD-2167A
- DOD-STD-498A
- ISO 9000
- ISO 9300
- **ISO 12207**
- ...

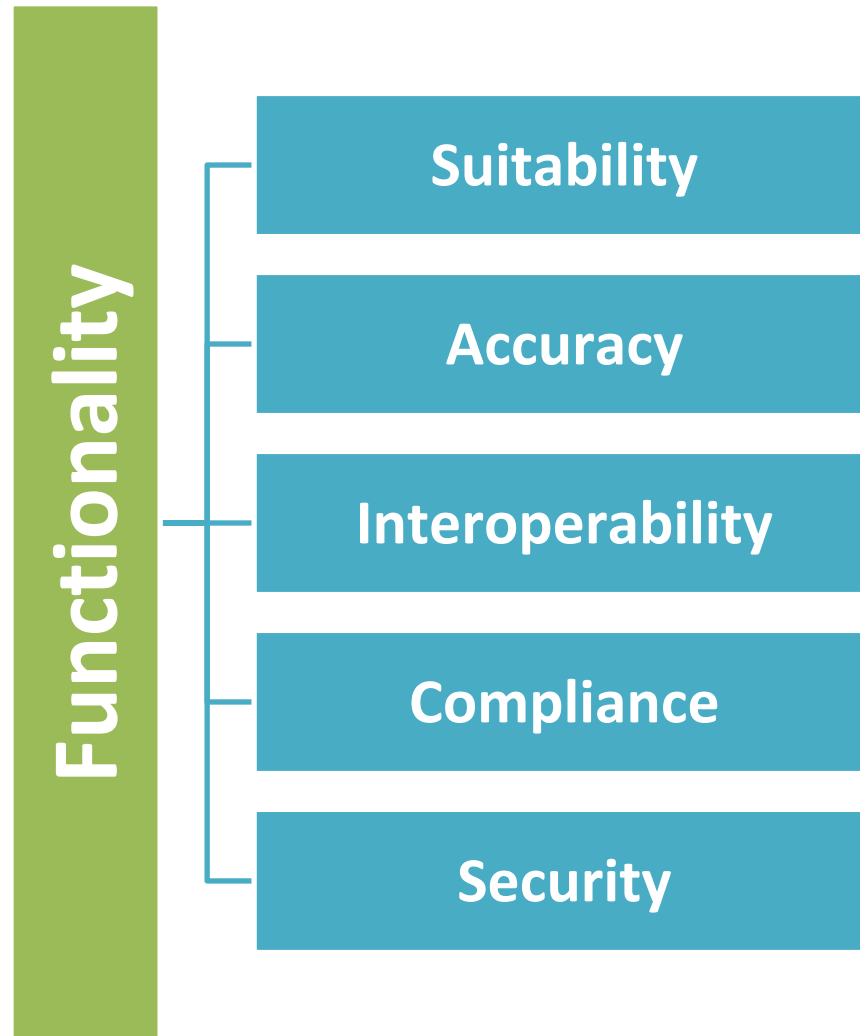
Đặc tính chất lượng PM

- Đặc tính chất lượng phần mềm theo ISO/IEC 9126:



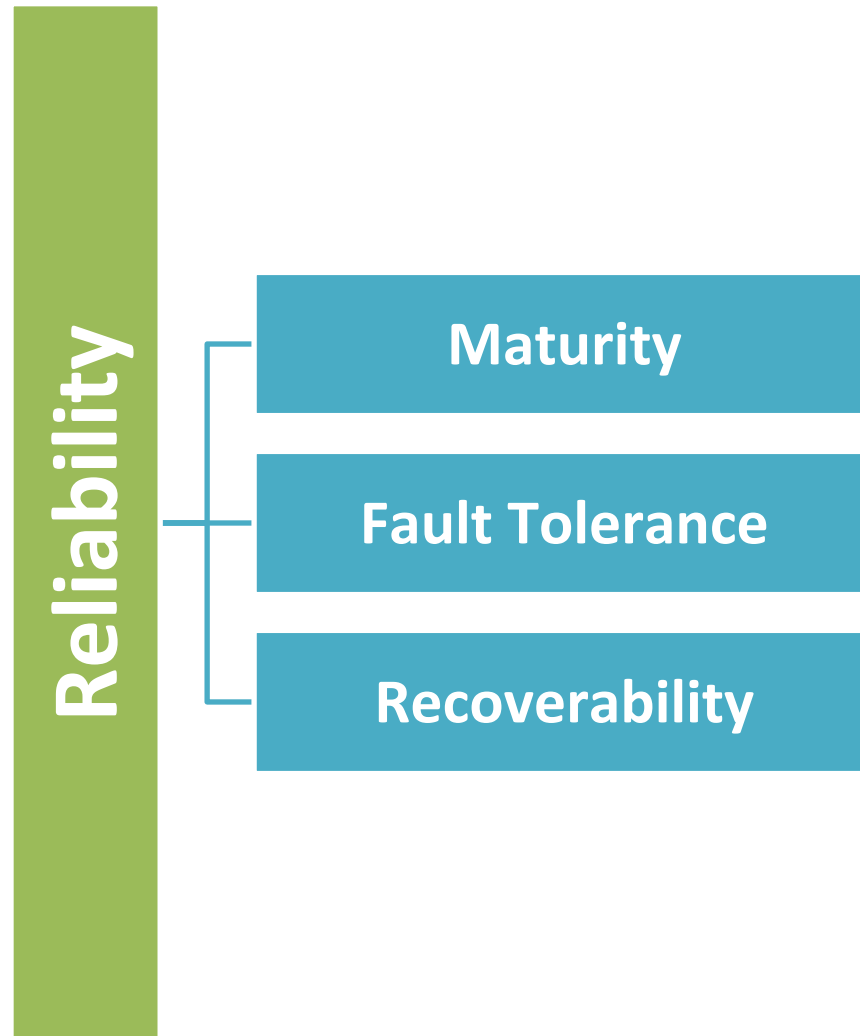
Đặc tính chất lượng PM

- Đặc tính chất lượng phần mềm theo ISO/IEC 9126:



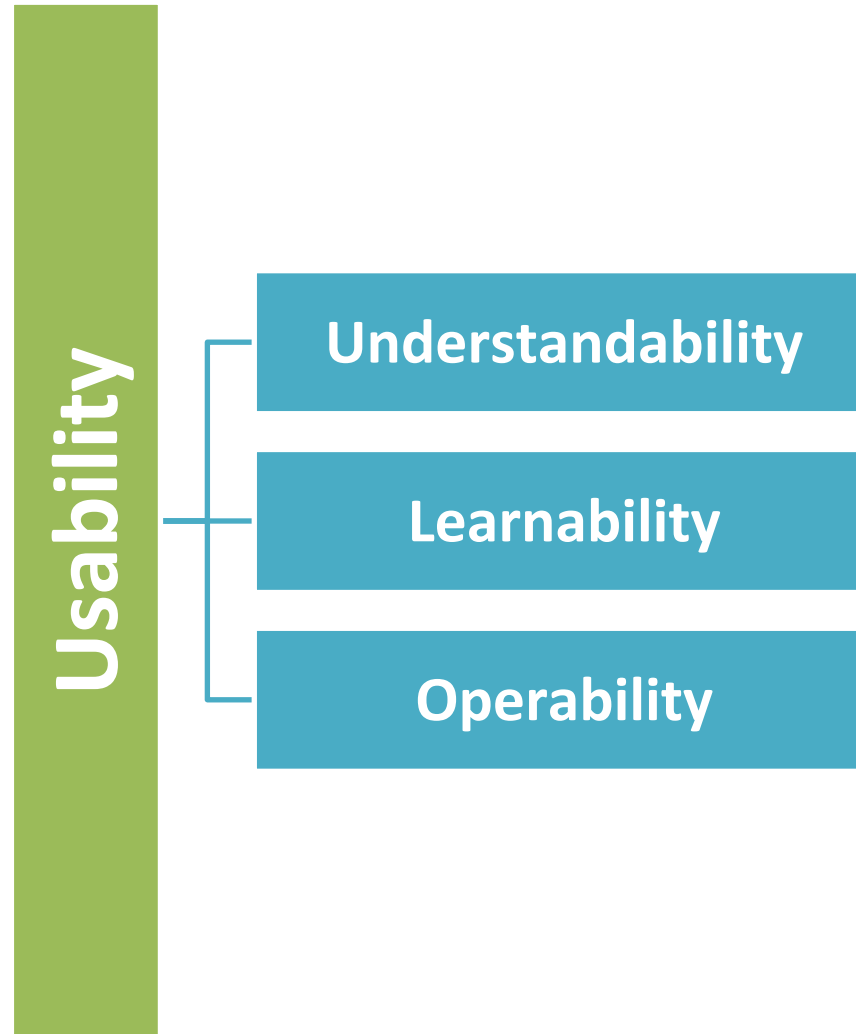
Đặc tính chất lượng PM

- Đặc tính chất lượng phần mềm theo ISO/IEC 9126:



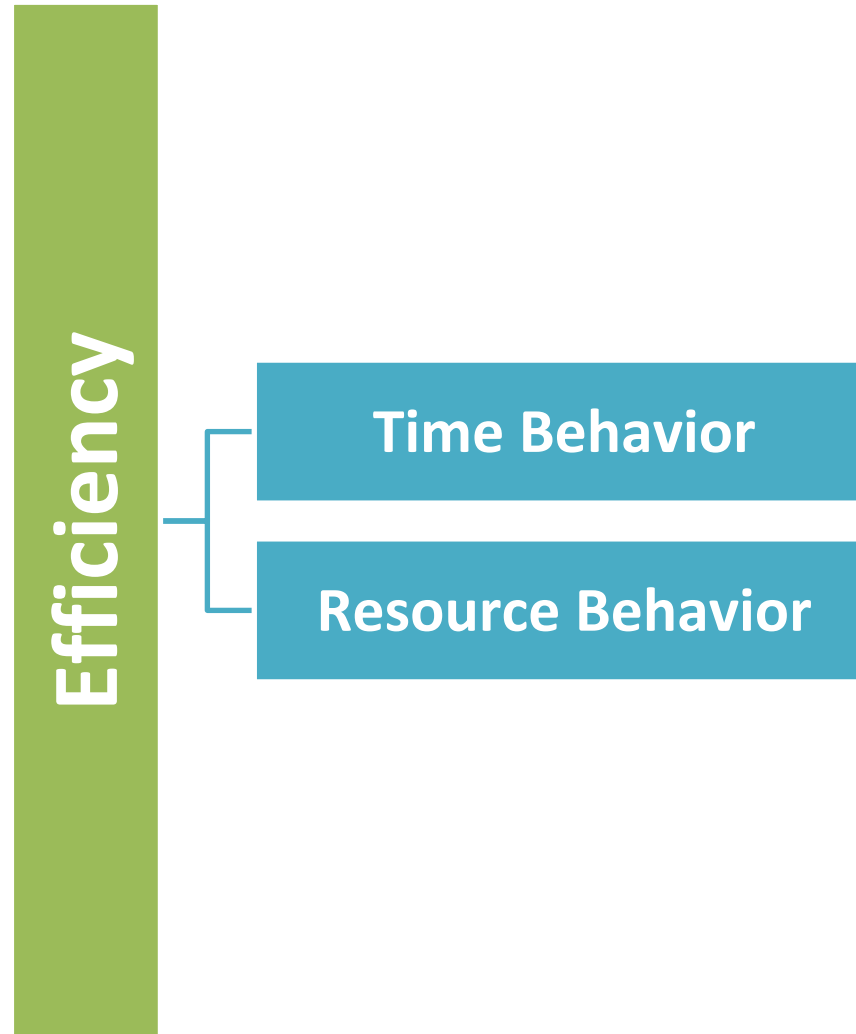
Đặc tính chất lượng PM

- Đặc tính chất lượng phần mềm theo ISO/IEC 9126:



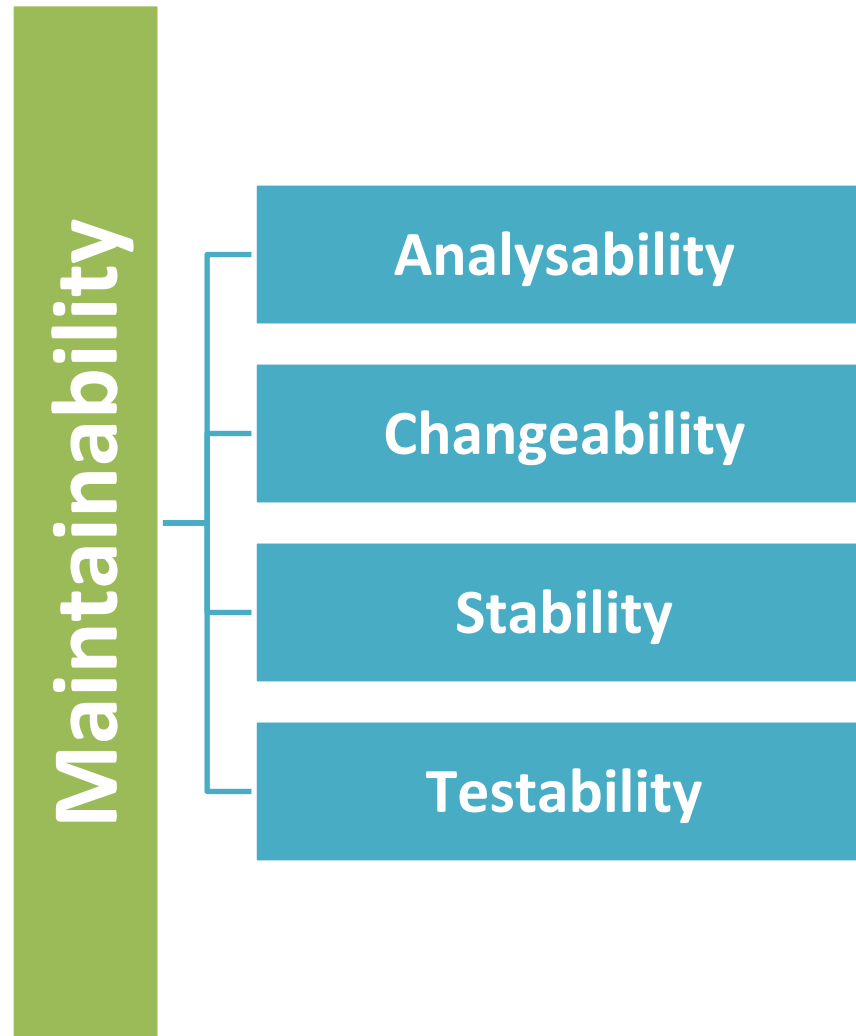
Đặc tính chất lượng PM

- Đặc tính chất lượng phần mềm theo ISO/IEC 9126:



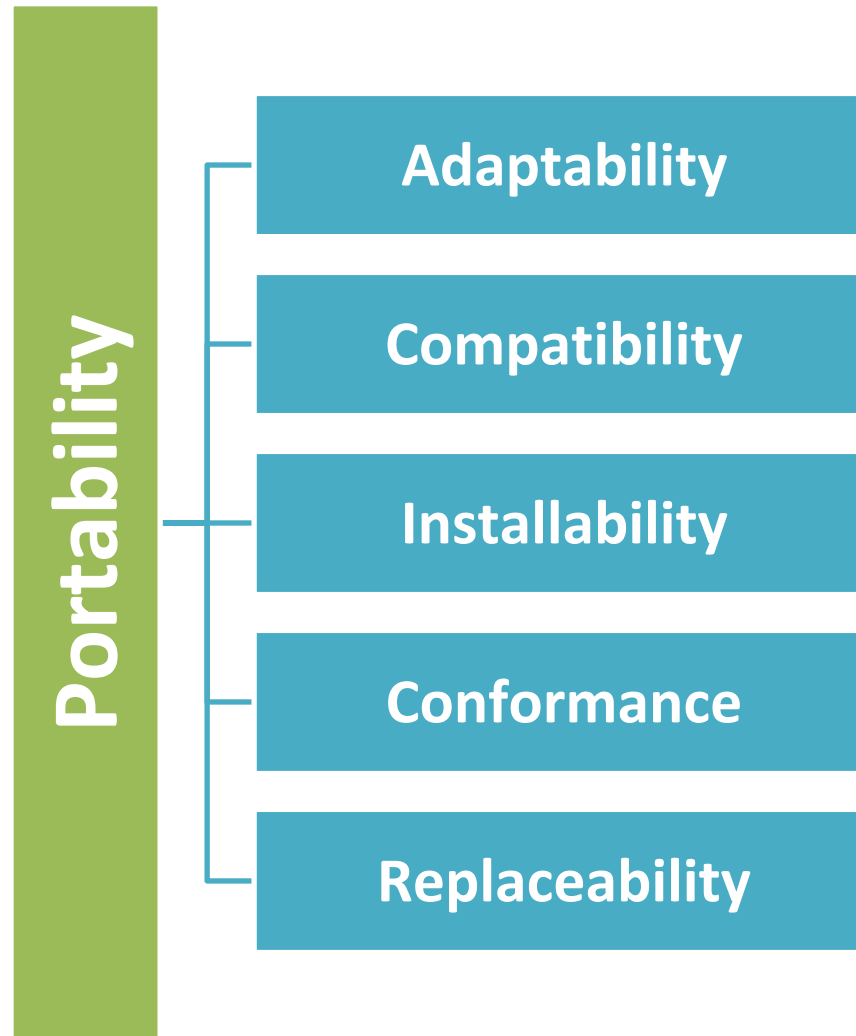
Đặc tính chất lượng PM

- Đặc tính chất lượng phần mềm theo ISO/IEC 9126:



Đặc tính chất lượng PM

- Đặc tính chất lượng phần mềm theo ISO/IEC 9126:



Công nghệ phần mềm

2

Các quy trình
phần mềm