



8 định luật tiến hoá phần mềm của Lehman

Search

[advanced search](#)

[Forums](#) » [Học tập](#) » [Thảo luận về CNTT](#)

You need to be logged in to post in the forums. If you do not have an account, please [sign up](#) first.

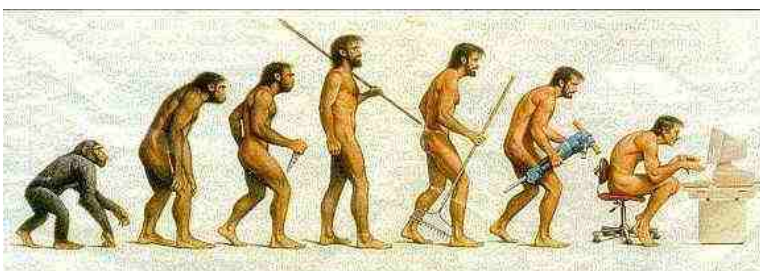
18. June 2008, 16:58:10



DVDdoc

Banned user

8 định luật tiến hoá phần mềm của Lehman



Có sinh sản là có tiến hoá. Sinh vật có F1 F2, phần cứng có model1 model2, phần mềm có version1 version2. Sinh vật tiến hoá theo định luật Darwin, Mendel. Phần cứng tiến hoá theo định luật Moore, nano. Phần mềm tiến hoá theo định luật gì?

Giáo sư Lehman đã nghiên cứu vấn đề này từ tận những năm 1970 đến nay! Bài viết này giới thiệu 8 định luật ông phát hiện qua nghiên cứu rất nhiều phần mềm mã đóng. Phần mềm mã đóng có lịch sử từ khoảng 1950, mã mở, outsourcing mới chỉ ra đời mười năm nay. Do đó để áp dụng cho 2 loại này, có lẽ phải mở rộng thêm 8 định luật này hoặc đưa ra những định luật khác chẳng?

Định luật 1: Continuing Change Phải chỉnh sửa phần mềm liên tục, nếu không mức độ hài lòng của khách hàng ngày càng giảm.

Có 2 lí do, liên quan đến lí thuyết về feedback control system của môn điều khiển học, vì rõ ràng khách hàng là người điều khiển lập trình viên:

- * Khả năng diễn đạt của khách hàng có hạn chế, họ cần x, nhưng lại nói thành y, lập trình viên nghe thành z
- * Nhu cầu của khách hàng thay đổi theo thời gian

Như vậy phần mềm phải tiến hoá liên tục nếu không muốn bị khách hàng xoá khỏi máy.

Định luật 2: Increasing Complexity

Khi tiến hoá, độ phức tạp của phần mềm luôn tăng, nếu không bỏ công sức để làm giảm nó xuống.

Đây chỉ là hệ quả của định luật 2 của môn nhiệt động học bao trùm mọi thứ trong vũ trụ, nói rằng entropy luôn tăng. Điều này có nghĩa chương trình ngày càng béo ra, cấu trúc ngày càng xấu tệ, cần phải refactor.

Định luật 3: Large Program Evolution

Các yếu tố liên quan đến quá trình tiến hoá (ý thích của khách hàng...) tuân theo qui luật phân phối xác suất chuẩn.

Định luật 4: Invariant Work-Rate

Độ ổn định là chỉ số quan trọng trong hệ thống điều khiển. Để đảm bảo độ tiến hoá ổn định, nhân sự cần ổn định qua thời gian.

Ai từng đọc quyển The mythical man-month đều biết: tăng thêm người vào team càng làm cho project đã chậm càng chậm hơn.

Định luật 5: Conservation of Familiarity

Ở mỗi phiên bản mới, phiên bản này chỉ thành công nếu những người liên quan (lập trình viên, nhân viên bán hàng, người dùng...) hiểu rõ sự khác biệt của phiên bản mới so với phiên bản trước. Do đó, phải bảo toàn hệ số góc của đường phát triển. Thay đổi nhanh quá sẽ bà con theo không kịp{#emotions_dlg.tongue_out}.

Định luật 6: Continuing Growth

Phải thêm tính năng vào phần mềm, nếu không mức độ hài lòng của khách hàng ngày càng giảm.

Có vẻ giống định luật 1, vì 2 cái nói về 2 hiện tượng khác nhau nhưng không phải không liên quan. Định luật 1 liên quan đến lý thuyết bất định Heisenberg của môn cơ học lượng tử: khách hàng không biết trước để có thể trình bày đầy đủ và chính xác mọi yêu cầu.

Ở định luật 6 thì ngược lại, khách hàng biết rõ họ cần 100 tính năng, nhưng do điều kiện tài chính, thời gian, tay nghề của lập trình viên... họ phải cắt bớt số tính năng xuống còn 60 để version 1 có thể hoàn thành kịp thời hạn. Sau đó, qua thời gian họ sẽ yêu cầu thêm tính năng còn thiếu vào version 2, 3.

Định luật 7: Declining Quality

Chất lượng của phần mềm càng ngày càng giảm nếu không được bảo trì và thay đổi cho phù hợp với điều kiện thực tế.

Theo thời gian mọi thứ đều tốt lên, nên về mặt tương quan, cái nào không tiến hoá sẽ tự động được coi là kém chất lượng. Ví dụ thời bao cấp mỗi tháng có nửa kí thịt thì được là có chất lượng sống cao, nhưng cũng nửa kí thịt đó (không thay đổi) thì hiện nay được coi là đói nghèo.

Định luật 8: Feedback System

Để có thể sửa chữa cải tiến, phải coi qui trình phát triển phần mềm là hệ thống điều khiển kiểu feedback.

Tham khảo

* Laws of Software Evolution Revisited (1999)

 <http://www.doc.ic.ac.uk/~mml/feast2/papers/pdf/556.pdf>

Bài viết lấy từ trang này:  <http://cntt.tv/nodes/show/393>

Tác giả: Đào Thanh Ngọc:  <http://cntt.tv/users/31>

Forum jump:

Go

Copyright © 2001 - 2013 Opera Softw are. All rights reserved.

[Help/FAQ](#) | [API](#) | [Privacy](#) | [Terms](#) | [Opera Brow ser](#) | [Extensions](#) | [Developers](#) | [New s Portal](#)

English