# Chương 2: Ứng dụng luồng công việc

*Phần mở đầu của khóa luận này đã bàn đến sự phát triển mạnh mẽ cũng như sự ứng dụng ngày một rộng rãi trong hầu hết các lĩnh vực của Công nghệ Thông tin, kèm theo đó là nhu cầu mỗi lúc một cấp thiết hơn đối với các ứng dụng quản lý đa năng, linh động và thật sự mạnh mẽ. Đó là các yêu cầu về các mức độ quản lý từ tổng quát đến chi tiết, về phương thức quản lý trực tiếp đến quản lý từ xa,về khả năng đưa ra các thông tin quan trọng hỗ trợ tối đa quá trình dự báo, cảnh báo cũng như giải quyết các rủi ro v.v… Và quan trọng hơn hết, một ứng dụng quản lý cấp cao phải thật sự linh động đối với sự thay đổi quy trình nghiệp vụ bên trong…*

*Phần này tiếp tục giới thiệu các khái niệm liên quan đến mô hình luồng công việc, và những vấn đề tồn tại. Tiếp sau đó là các đặc điểm yêu cầu quan trọng đối với một ứng dụng quản lý luồng công việc tổng quát dưới góc nhìn của người sử dụng. Phần cuối sẽ trình bày về các vấn đề tồn tại cũng như các yêu cầu đối với một ứng dụng quản lý luồng công việc tổng quát.*

1. Tổng quan về mô hình luồng công việc.
   1. Luồng công việc

Vào những năm tám mươi của thế kỷ hai mươi, khái niệm Luồng công việc (*Workflow*) mới được sử dụng lần đầu tiên trong ngành công nghệ phần mềm [[[1]](#footnote-2)]. Tuy thế, sự xuất hiện của Luồng công việc và Mô hình luồng công việc trong cuộc sống của con người hẳn đã từ rất lâu. Giữa vô vàn các hoạt động của mình trong cuộc sống, có nhiều công việc mà con người lặp đi lặp lại nhiều lần. Một hoặc một số các công việc có thể được lặp lại theo chu kỳ nhiều giờ, nhiều ngày, nhiều tháng v.v… hoặc xảy ra bất kỳ lúc nào. Dựa vào kinh nghiệm đúc kết được, con người vạch ra các bước phải làm và dần dần hoàn thiện chúng cho các lần thực hiện sau. Nói khác hơn, dựa vào kinh nghiệm và sự hiểu biết của mình, con người có thể vạch ra các kế hoạch, các bước phải làm; xác định thành phần con người, công cụ cần có để thực hiện công việc nhằm đạt được mục đích nhất định. Các bước cần thực hiện trong các kế hoạch đó có mối quan hệ với nhau và tạo thành luồng công việc.

*Luồng công việc là một tập có thứ tự các công việc tuân theo các quy tắc nhất định, được định sẵn sao cho việc thực hiện theo đó sẽ đạt được các mục đích công việc đã đề ra. [[[2]](#footnote-3)]*

Khái niệm về luồng công việc là một khái niệm tương đối. Cùng tham gia vào một hệ thống trong cùng một môi trường, nhưng những đối tượng với vai trò khác nhau sẽ có các luồng công việc khác nhau. Hình vẽ dưới đây là ví dụ về các luồng công việc trong hệ thống tổ chức thi tuyển sinh đại học tại Việt Nam. Dưới góc nhìn của các thí sinh, luồng công việc thi tuyển sinh gồm 3 việc chính phải làm. Thực hiện luồng công việc này, các thí sinh đạt được mục đích là dự thi đại học thành công. Dưới góc nhìn của đơn vị tổ chức tuyển sinh thì lại khác, luồng công việc tổ chức thi tuyển sinh gồm có 9 công việc chính phải làm, thực hiện luồng công việc này sẽ đạt được mục đích là tổ chức thành công một kỳ tuyển sinh đại học.



Luồng công việc Thi tuyển sinh đại học dưới góc nhìn của thí sinh.



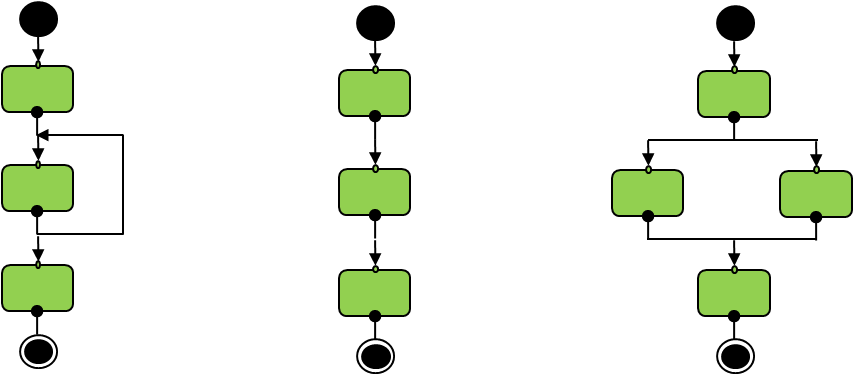
Luồng công việc Tổ chức tuyển sinh đại học dưới góc nhìn của đơn vị tổ chức tuyển sinh

* 1. Mô hình luồng công việc

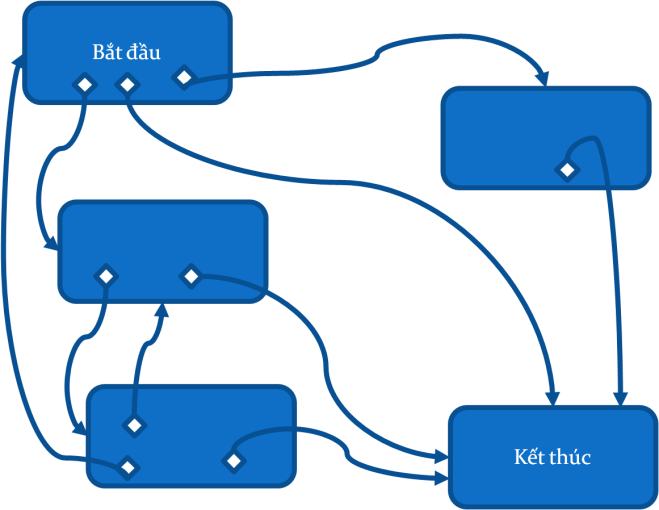
*Mô hình luồng công việc là một dạng biểu diễn luồng công việc thành sơ đồ.* [[3]](#footnote-4)

Việc biểu diễn một LCV thành mô hình luồng công việc gọi là *Mô hình hóa luồng công việc.* Một công việc thực tế sẽ mô hình hóa thành một thành phần xử lý của MH LCV, và quy tắc thực hiện công việc trong thực tế sẽ được mô hình hóa thành các quan hệ logic giữa các thành phần xử lý đó.

Có hai loại MH LCV phổ biến nhất đó là MH LCV tuần tự (Sequential Workflow) và MH LCV máy trạng thái (State machine Workflow). Với loại tuần tự, các công việc sẽ được biểu diễn liên tiếp nhau. Một công việc chỉ được thực hiện khi các công việc phía trước đó đã hoàn thành. Ngược lại, với MH LCV máy trạng thái, hệ thống sẽ được mô hình hóa thành một tập các trạng thái cụ thể. Tại mỗi trạng thái có thể có một tập các sự kiện được phép xảy ra. Mỗi một sự kiện xảy ra sẽ dẫn tới việc chuyển đổi hệ thống từ trạng thái này sang trạng thái khác. Ngoài ra còn có một loại MH LCV khác ít phổ biến hơn đó là MH LCV dựa trên tập luật (Rules-based Workflow). Đây là một sự kết hợp giữa MH LCV tuần tự và một tập các luật, các quy tắc chuyên dùng để mô phỏng các LCV trong các quy trình nghiệp vụ phức tạp.



Các mô hình LCV tuần tự



MH LCV máy trạng thái

Một luồng công việc có một trạng thái bắt đầu và một trạng thái kết thúc nhất định. Trạng thái bắt đầu là điểm khởi đầu mặc định của một quá trình thực thi quy trình. Còn trạng thái kết thúc thì tùy vào loại mô hình luồng công việc mà có những điều kiện khác nhau để đạt được đến các trạng thái này. Ví dụ, với luồng công việc Tuần tự thì trạng thái kết thúc sẽ được đạt đến khi tất cả các công việc thành phần đều được thực hiện xong; còn với luồng công việc thuộc loại Máy trạng thái thì có thể kết thúc bất cứ khi nào xảy ra sự kiện dẫn đến trạng thái này.

* 1. Sự thực thi các luồng công việc.

Ứng với một quá trình thực thi LCV là một thể hiện của LCV đó. Một LCV có thể có nhiều thể hiện khác nhau do quá trình thực thi khác nhau. Mỗi LCV khi thực thi sẽ phải tuân thủ các quy tắc nhất định. Đó là các quy định về quy trình nghiệp vụ thực tế của tổ chức, doanh nghiệp tạo ra và thực thi LCV đó như các ràng buộc về sự tương tác với người thực hiện công việc, ràng buộc về thời gian thực thi, các điều kiện cho phép xảy ra sự kiện chuyển đổi trạng thái v.v… Ngoài ra, từng công việc thành phần bên trong một LCV cũng có các ràng buộc riêng của nó trong quá trình thực thi.

Ví dụ, công việc “Nhận đăng ký thi” trong ví dụ về LCV Tổ chức tuyển sinh đại học ở phần trước có thể có một số ràng buộc như: bắt đầu từ ngày t1 và kết thúc vào ngày t2; chỉ cho phép đăng ký khi các giấy tờ đều hợp lệ, thí sinh đăng ký phải tốt nghiệp trung học phổ thông trước đó, v.v…

2. Tầm quan trọng của LCV và MH LCV

3. Mô hình hóa luồng công việc và vấn đề tồn tại

Việc mô hình hóa luồng công việc bằng các ngôn ngữ mô hình hóa là nhằm mục đíc­­­­­­­­h phục vụ cho việc đưa luồng công việc vào trong ứng dụng hỗ trợ việc thực thi quy trình nghiệp vụ của các doanh nghiệp, hay các hệ quản trị luồng công việc. Tuy nhiên, các hệ quản trị luồng công việc thật sự giúp ích rất nhiều cho doanh nghiệp dẫn đến một vấn nạn, đó là việc ngày càng nhiều doanh nghiệp có nhu cầu đưa luồng công việc vào trong ứng dụng hỗ trợ dẫn tới việc phát sinh nhiều loại ứng dụng hỗ trợ xây dựng khác nhau. Các ứng dụng này được xây dựng bởi nhiều tổ chức khác nhau nên sẽ có nhiều quy cách mô hình hóa riêng biệt đặc trưng. Điều này dẫn đến sự khó khăn cho các doanh nghiệp khi muốn thay đổi một ứng dụng hỗ trợ này sang một ứng dụng hỗ trợ khác, bởi họ phải thay đổi toàn bộ mô hình đã được thiết kế và thực thi, dẫn đến việc ngừng sử dụng các ứng dụng này cho đến khi thay đổi bằng 1 ứng dụng khác phù hợp hơn.

Nhu cầu thay đổi xảy ra khi ứng dụng trở nên cũ, không còn đáp ứng nhu cầu ngày càng phát triển của doanh nghiệp, hoặc ứng dụng không có khả năng đáp ứng hết các yêu cầu của doanh nghiệp khiến họ phải nghĩ đến việc thay đổi một hệ quản trị luồng công việc khác.

Đặc biệt là việc đưa vào trong hệ quản trị luồng công việc mới các ứng dụng hỗ trợ khác nhau đáp ứng các nhu cầu khác nhau của doanh nghiệp. Đây là điều hiển nhiên bởi trên thực tế, khó có hệ quản trị nào đáp ứng hết các nhu cầu đa dạng của doanh nghiệp, nhất là trong việc xử lý các quy tắc kinh doanh hay các quy luật bản chất bên trong do doanh nghiệp định ra. Nhưng các ứng dụng không tương thích hoặc không hỗ trợ cùng một định dạng mô hình hóa dẫn đến sự bế tắc trong việc sử dụng nhiều giải pháp hỗ trợ, khiến các doanh nghiệp không thỏa mãn được hết các nhu cầu của họ.

Chính những điều kiện này đã phát sinh nhu cầu xây dựng một tiêu chuẩn quốc tế cho việc xây dựng các hệ quản trị luồng công việc và các ngôn ngữ mô hình hóa luồng công việc. WfMC (Workflow Management Coalition) ra đời nhằm mục đích này.

4. *Workflow Management Coalition* (WfMC)

Workflow Management Coalition là tổ chức thế giới (gọi tắt là WfMC) được thành lập nhằm mục đích quy định ra các tiêu chuẩn cho việc tự động hóa luồng công việc

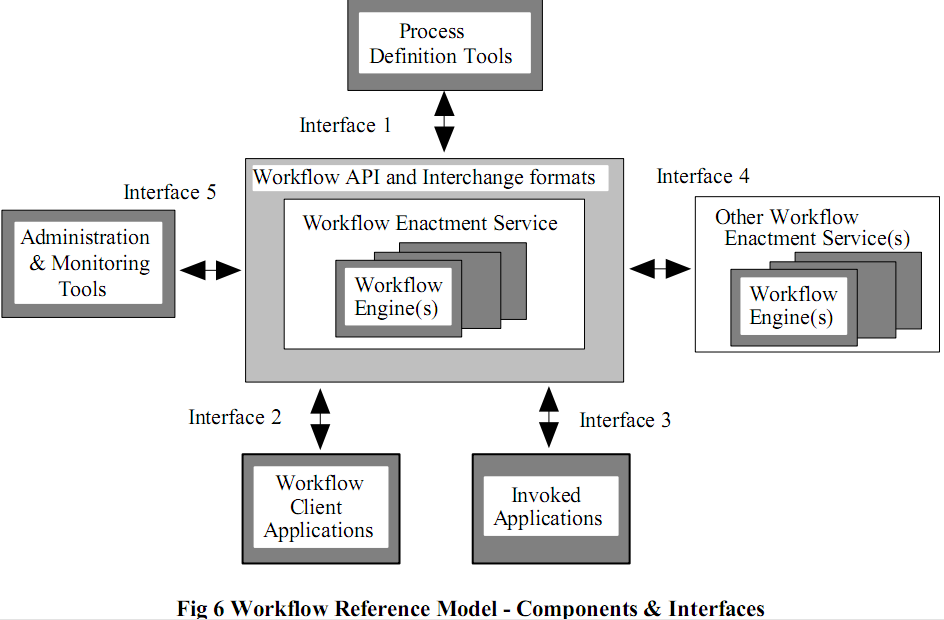
Được thành lập vào tháng 8 năm 1993, hiện nay WfMC đã có hơn 200 thành viên đến từ các ngành công nghiệp và các khu nghiên cứu khác nhau. Nhiệm vụ của tổ chức WfMC là tập trung vào việc xác định các phạm vi chức năng quản lý luồng công việc phổ biến, từ đó phát triển các chức năng này và bổ sung 1 cách thích hợp cho các ứng dụng hỗ trợ và các hệ quản trị luồng công việc.

Cho đến nay, WfMC đã đưa ra mô hình tham chiếu chuẩn cho luồng công việc và liên tục cải tiến, đồng thời phát triển các ngôn ngữ chuẩn cho việc mô hình hóa luồng công việc sử dụng trong các ứng dụng tự động hóa và các hệ quản trị luồng công việc.

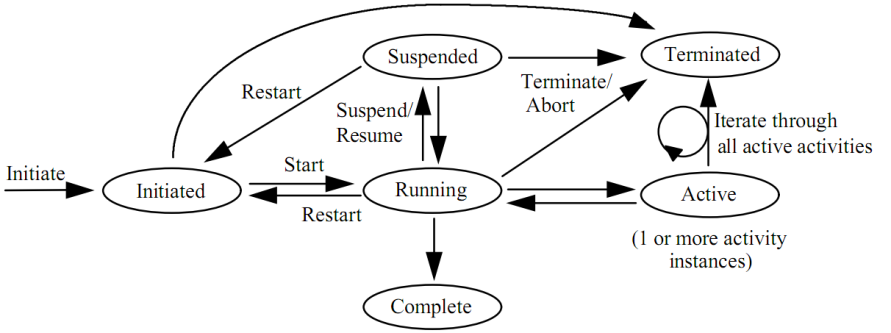
4.1. Mô hình tham chiếu luồng công việc (*Workflow Reference Model - WfRM*)

WfRM phát triển cấu trúc tổng quát của một ứng dụng luồng công việc bằng cách sử dụng các giao diện (interface) cho phép sản phẩm tương tác với nhau theo nhiều cấp độ. Tất cả những hệ thống luồng công việc chứa đựng nhiều thành phần khác nhau được định nghĩa theo nhiều cách đồng thời cùng một thành phần nhưng với các sản phẩm khác nhau sẽ được thể hiện khác nhau.

Hình bên dưới mô tả những thành phần và giao thức quan trọng bên trong kiến trúc luồng công việc.

[[4]](#footnote-5)

* *Workflow Enactment Services (WES):* 
  + Định nghĩa: là một dịch vụ chứa một hay nhiều *engine* để tạo ra, quản lý và thực thi những thể hiện luồng công việc (*workflow instane*
  + ). Những ứng dụng bên ngoài tương tác với dịch vụ này thông qua API của luồng công việc, gọi là WAPI[[5]](#footnote-6). Tiến trình và các tác vụ được ngăn cách với nhau một cách logic. Chính sự ngăn cách này tạo nên một khả năng ứng dụng rộng rãi đối với từng loại nghiệp vụ khác nhau hoặc giữa các ứng dụng với nhau. Những tài nguyên bên ngoài được truyền vào WES bằng 1 trong 2 giao thức sau:
    - Giao thức cho các ứng dụng phía người dùng (client): trình điều khiển danh sách công việc sẽ chịu trách nhiệm cho việc chọn lựa và thực thi những công việc. Việc khởi tạo những ứng dụng cũng nằm trong sự quản lý của trình điều khiển này.
    - Giao thức cho các ứng dụng được gọi thực thi: sẽ làm cho *Workflow Engine* có khả năng khởi động các ứng dụng chịu trách nhiệm một tác vụ cụ thể nào đó, có thể đó là một ứng dụng về phía server. Nó được gọi thực thi thông qua *Interface Worklist* nhằm đem lại sự linh động hơn cho điều phối những tiến trình của người dùng.
  + Chức năng:
    - Cung cấp môi trường thực thi để việc khởi tạo và khởi động các tiến trình xảy ra.
    - Sử dụng *Management Engine* của luồng công việc, chịu trách nhiệm trong việc thông dịch và khởi động các tiến trình.
    - Tương tác với những tài nguyên cần thiết để thực thi các tác vụ khác nhau.
* *Workflow Engine (WE):*
  + Định nghĩa: là một dịch vụ hoặc là một engine cung cấp môi trường để một thể hiện luồng công việc thực thi. Một WES có thể chứa nhiều WE.
  + Chức năng:
    - Thông dịch sự khởi tạo của một tiến trình.
    - Quản lý các thực thể của một tiến trình bao gồm: tạo ra, khởi động, tạm ngưng, kết thúc v.v…
    - Đăng ký và kết thúc một tiến trình tham gia nhất định.
    - Nhận diện những công việc cụ thể để người dùng hoặc các giao thức để người dùng có thể can thiệp được.
    - Bảo trì dữ liệu của các *Workflow Control*, các luồng công việc liên quan và truyền tải dữ liệu đến hoặc đi từ những ứng dụng của người dùng.
    - Gọi thực thi các ứng dụng bên ngoài và kết nối những dữ liệu liên quan.
* *Homogeneous & Heterogeneous Workflow Enactment Services*:
  + Homogeneous WES: bao gồm một hay nhiều WE cung cấp môi trường thực thi các tiến trình của luồng công việc với các thuộc tính được định nghĩa sẵn.
  + Heterogeneous WES: bao gồm 2 hay nhiều dịch vụ khác nhau với một chuẩn mực nhất định về khả năng tương tác ở một mức độ nhất định.
  + Chức năng:
    - Hỗ trợ những tiến trình trong việc định nghĩa các đối tượng và thuộc tính.
    - Hỗ trợ việc vận chuyển dữ liệu liên quan.
    - Hỗ trợ tiến trình, tiểu tiến trình hoặc các tác vụ giữa các WE khác nhau.
    - Hỗ trợ những chức năng quản trị và điều khiển.
  + Tiến trình và các trạng thái chuyển của các tác vụ:
    - WES được xem như một cái máy chuyển đổi trạng thái, nơi mà những tiến trình riêng rẽ hoặc các thể hiện của các tác vụ chuyển đổi trạng thái khi có những sự kiện bên ngoài tác động hoặc điều khiển những quyết định của WE.
    - Mô hình chuyển đổi trạng thái:

[[6]](#footnote-7)

* + - * Initiated – thực thể của các tiến trình được khởi tạo bao gồm những trạng thái của các tiến trình liên quan và những dữ liệu liên quan. Tuy nhiên ở giai đoạn này các tiến trình chưa hoàn toàn đầy đủ thông tin về các điều kiện để phát tín hiện thực thi.
      * Running – thực thể của các tiến trình bắt đầu khởi tạo.
      * Active - một hoặc nhiều các tác vụ được bắt đầu.
      * Suspended – một thực tể tiến trình dừng hoạt động và không có bất cứ một tác nào được thực hiện cho đến khi tiến trình trở lại trạng thái Running.
      * Completed –thực thể tiến trình có được đầy đủ thông tin để hoàn tất. Thực thể bị hủy.
      * Terminated – sự thực thi của các thực thể tiến trình dừng trước khi hoàn tất, tất cả các những hoạt động như những ghi nhận lỗi sẽ được thông báo và thực thể tiến trình bị hủy.
* *Process Definition Tools*: Là các công cụ khác nhau được sử dụng để phân tích, mô hình hóa, mô tả và ghi nhận một tiến trình của một nghiệp vụ nào đó. Mục đích cuối cùng từ việc mô hình hóa tiến trình và thiết kế tác vụ chính là định nghĩa tiến trình được thông dịch lúc thực thi bời WE bên trong WES.

4.2. Các ngôn ngữ mô hình hóa luồng công việc:

WfMC đã đưa ra tiêu chuẩn nhằm thống nhất định dạng ngôn ngữ mô hình hóa chung cho các ứng dụng tự động hóa luồng công việc, nhằm giúp các doanh nghiệp thay đổi hoặc kết hợp sử dụng các phần mềm quản lý luồng công việc khác nhau 1 cách thống nhất, dễ dàng, không phải xây dựng lại khi thay đổi hay thêm phần mềm khác vào hệ thống.

Hiện nay, có 2 chuẩn đã được WfMC đề nghị là XPDL và Wf-XML.

*4.2.1. XML Process Definition Language (XPDL):*

XPDL là 1 trong 2 định dạng chuẩn được WfMC xem xét và đề nghị. Mục đích của XPDL là trao đổi các định nghĩa và thể hiện của các tiến trình nghiệp vụ (*Business Process*[[7]](#footnote-8)), luồng công việc giữa các sản phẩm LCV khác nhau, chẳng hạn như giữa công cụ mô hình hóa và hệ quản trị LCV. XPDL định nghĩa 1 lược đồ xml (*XML schema*) nhằm xác định phần khai báo của LCV/ Tiến trình nghiệp vụ.

Hiện nay XPDL được xem là định dạng file tốt nhất cho việc trao đổi sơ đồ BPMN (*Business Process Modelling Notation*[[8]](#footnote-9)), cũng như sơ đồ LCV. Nó được thiết kế đặc biệt để có thể lưu trữ tất cả các tình trạng của 1 sơ đồ BPMN. XPDL chứa các đối tưởng để lưu trữ thông tin đồ họa, như vị trí X,Y của node, cũng như các tình trạng thực thi, dùng để chạy 1 thể hiện LCV. Điều này giúp phân biệt XPDL với BPEL (*Business Process Execution Language[[9]](#footnote-10)*), chỉ tập trung vào tình trạng thực thi của tiến trình. BPEL không chứa các đối tượng diễn tả thông tin đồ họa của sơ đồ LCV.

Cho đến hiện nay, trên thế giớn đã có khoảng hơn 80 sản phẩm, ứng dụng sử dụng XPDL được xây dựng trên cả nền Java, Microsoft.Net Framework và Linux[[10]](#footnote-11).

*Ví dụ:* 1 file mô tả 1 workflow sử dụng XPDL 2.0 có thể download tại trang web sau: http://wfmc.org/Download-document/XPDL-Sample-Workflow-Schema.html

*4.2.2. Wf-XML:*

Wf-XML là 1 định dạng file tuân theo chuẩn BPM (*Business Process Management*) được phát triển bởi WfMC.

Wf-XML được thiết kế và thực thi như 1 phần mở rộng cho giao thức ASAP (*OASIS Asynchronous Service Access Protocol*) - 1 giao thức đã được chuẩn hóa cung cấp các dịch vụ bất đồng bộ, nghĩa là cung cấp cách thức để các chương trình bắt đầu, theo dõi sự thay đổi trạng thái của các chương trình hay dịch vụ khác thực thi trong khoản thời gian dài. ASAP cung cấp cho người dùng chức năng giám sát dịch vụ đang thực thi, đồng thời thông báo cho người dùng sự thay đổi trạng thái của nó. Wf-XML đã mở rộng chức năng này từ ASAP bằng cách cung cấp thêm 1 dịch vụ mạng cho phép gửi và nhận chương trình hoặc định nghĩa của dịch vụ được cung cấp. 1 Engine có tính năng này sẽ có thể cung cấp 1 dịch vụ hoạt động trong khoảng thời gian dài, có thể được lập trình bằng cách cho phép cài đặt thêm các định nghĩa luồng công việc.

Wf-XML cung cấp 1 phương thức chuẩn hóa cho 1 engine BPM[[11]](#footnote-12) để gọi 1 thể hiện LCV trong 1 engine khác, đồng thời đợi cho thể hiện LCV đó hoàn tất. Vì công cụ chỉnh sửa LCV và công cụ thực thi LCV có thể được sản xuất từ nhiều nhà phát triển khác nhau, nên cần có 1 phương thức chung để trao đổi giữa các công cụ đó. Với phương thức Wf-XML cung cấp (chuẩn hóa việc trao đổi định nghĩa LCV giữa các công cụ thiết kế và engine thực thi), người dùng có thể kết hợp chính xác các *Process Definition Tools[[12]](#footnote-13)* tốt nhất với *Process Execution Engine*[[13]](#footnote-14) tương ứng theo nhu cầu.

Wf-XML được nghiên cứu từ khoảng năm 1997 với tên gọi là SWAP (Simple Workflow *Access Protocol*) bởi các nhà phát triển như Netscape, Oracle.... Tiếp theo đó, WfMC đã tiếp bước và liên tục phát triển, cải tiến, đầu tiên là Wf-XML 1.0 và Wf-XML 1.1. Wf-XML đã được ra đời và đưa vào sử dụng trong một số sản phẩm thương mại. Phiên bản hiện nay là Wf-XML 2.0[[14]](#footnote-15) và đang được tiếp tục nghiên cứu, phát triển. Tuy nhiên, các sản phẩm xây dựng với Wf-XML 2.0 không tương thích ngược được với các sản phẩm sử dụng Wf-XML 1.1.

4.3. Các loại LCV

Chuẩn WfMC định ra các loại LCV dựa trên quy tắc hoạt động của LCV và loại nghiệp vụ kinh tế đang được đề cập. Bao gồm: Production, Administrative, Collaborative, và Ad-Hoc[[15]](#footnote-16)

*4.3.1. Production:* Quản lý 1 số lượng lớn các tác vụ tương tự nhau, nhằm tối ưu hóa năng suất nghiệp vụ. Cách thức hoạt động của LCV loại này là tự động hóa, nghĩa là các tác vụ bên trong LCV được thực hiện một cách tự động, con người chỉ tác động lên các công việc không nằm trong tiến trình đã được định nghĩa sẵn, tức là các ngoại lệ (*exceptions*). Như vậy, trong loại LCV này, thời gian và độ phức tạp của các sự kiện cần sự tương tác với con người được giảm thiểu... Việc tối ưu hóa nhằm đạt chất lượng và độ chính xác cao trong loại LCV này có thể đạt được bằng cách thi hành các tác vụ có tính lặp lại cao theo cùng 1 phương pháp 1 cách liên tục. Ứng dụng của LCV thuộc loại *Production* là để quản lý các tiến trình có độ phức tạp cao, đặc biệt có thể kết hợp chặt chẽ với những hệ thống đang tồn tại. Tuy nhiên, xu hướng hiện nay của việc sử dụng loại LCV này là nhúng các thành phần chức năng trong LCV vào trong các ứng dụng lớn dưới vai trò như các *Rules Engine*. Điều này dẫn đến việc phân chia bên trong loại LCV này thành 2 loại nhỏ: *Autonomous Workflow Engines* và *Embed Workflow*. Trong đó, sự khác nhau giữa 2 loại này ở chỗ, *Autonomous Workflow* bản thân nó không cần thêm các phần mềm bổ sung, còn *Embed Workflow* cần phải được gắn vào 1 hệ thống nào đó, chẳng hạn như hệ thống ERP[[16]](#footnote-17).

*4.3.1.2. Administrative*: Dễ dàng xác định LCV. Thông thường sẽ có có rất nhiều thể hiện LCV cùng thực thi đồng thời, và chúng cần sử dụng 1 lượng lớn nhân viên. Thể hiện LCV luôn dc tạo ra từ trong chương trình, và nếu chương trình đó quá phức tạp, thì họ chỉ cần sử dụng chương trình khác. Điều này có nghĩa là loại LCV này rất linh hoạt trong việc sử dụng các chương trình quản lý LCV. Như vậy, tính linh hoạt ở đây quan trọng hơn năng suất, và những hệ thống theo dạng này xử lý các trường hợp mỗi giờ với cường độ thấp hơn từ 1 đến 2 lần so với các hệ thống *Production Workflow*.

*4.3.1.3. Collaborative:* Tập trung vào các hoạt động làm việc nhóm. Các nhóm cùng hoạt động với nhau để xây dựng 1 mục tiêu chung, từ những nhóm nhỏ, hướng đề tài, đến những nhóm người khác nhau có cùng 1 mục tiêu chung... Hiệu quả của việc sử dụng mô hình LCV này để là hỗ trợ làm việc nhóm, hiện nay được xem như 1 yếu tố quan trọng trong sự thành công của các doanh nghiệp. Lợi ích của việc sử dụng Internet và www hổ trợ liên lạc nhóm giữa các doanh nghiệp cũng là 1 thành công thực tế trong hầu hết các tổ chức. Định nghĩa LCV ở đây không cứng nhắc mà có thể thường xuyên được thay đổi, Thỉnh thoảng người ta gọi các chương trình theo loại này là *Groupware[[17]](#footnote-18)*. Dĩ nhiên là có rất nhiều loại Groupware không được xem như 1 ứng dụng LCV loại Collaborative, chẳng hạn như *Bulletin Boards[[18]](#footnote-19)* hay *videoconference[[19]](#footnote-20)*.

*4.3.1.4. Ad-Hoc:* Cho phép người dùng tạo ra và sửa đổi định nghĩa LCV nhanh chóng và dể dàng để đáp ứng các trường hợp phát sinh. Như thế, Ad-Hoc có thể có rất nhiều định nghĩa LCV. Ad-hoc Workflow tối đa hóa tính linh hoạt trong các lĩnh vực mà bảo mật không phải là vấn đề chính yếu. Nghĩa là, chẳng hạn với các LCV loại Production thì Tổ chức, doanh nghiệp là người sở hữu LCV, còn ở Ad-Hoc thì các người sử dụng ứng dụng có thể có tiến trình riêng của họ.

5. Yêu cầu của một ứng dụng quản lý luồng công việc tổng quát

*Tóm lại,...*

1. <http://en.wikipedia.org/wiki/Workflow> [↑](#footnote-ref-2)
2. SHI Meilin, YANG Guangxin, XIANG Yong, WU Shangguang, Workflow Management Systems: A Survey, Department of Computer Science, Tsinghua University, Beijing, P.R.China 100084, 1998 [↑](#footnote-ref-3)
3. Từ đây về sau, chúng tôi dùng từ viết tắt LCV để thay cho cụm từ “Luồng Công việc” và MH LCV để thay cho cụm từ “Mô hình luồng công việc”. [↑](#footnote-ref-4)
4. David Hollingsworth, The Workflow Reference Model, Workflow Management Coalition Specification, 1995, trang 20 [↑](#footnote-ref-5)
5. Workflow Application Programming Interface & Interchange: thư viện chứa các API và các hàm mở rộng tùy theo chức năng của các giao thức [↑](#footnote-ref-6)
6. David Hollingsworth, The Workflow Reference Model, Workflow Management Coalition Specification, 1995, trang 23 [↑](#footnote-ref-7)
7. Xem chi tiết ở khóa luận tốt nghiệp "Tìm hiểu và ứng dụng Windows Workflow Foundation để hỗ trợ các quy trình nghiệp vụ" , Khoa CNTT, trường ĐHKHTN tp HCM, năm 2009 [↑](#footnote-ref-8)
8. là dạng biểu diễn đồ họa nhằm xác định Tiến trình nghiệp vụ trong LCV [↑](#footnote-ref-9)
9. là dạng rút gọn của WS-BPEL - Web Service Business Process Execution Language - một chuẩn ngôn ngữ thực thi tiến trình xác định các tương tác với các dịch vụ web [↑](#footnote-ref-10)
10. Xem phụ lục: Danh sách các sản phẩm, ứng dụng sử dụng XPDL [↑](#footnote-ref-11)
11. Xem chi tiết ở http://en.wikipedia.org/wiki/Business\_process\_management [↑](#footnote-ref-12)
12. Xem lại mục 4.1 [↑](#footnote-ref-13)
13. Các engine thực thi tiến trình/LCV [↑](#footnote-ref-14)
14. Xem phụ lục: XML Schema của Wf-XML 2.0 [↑](#footnote-ref-15)
15. Charles Plesums, Introduction to Workflow, Computer Sciences Corporation, Financial Services Group [↑](#footnote-ref-16)
16. Hệ thống hoạch định tài nguyên doanh nghiệp - một hệ thống được sử dụng trong các tổ chức nhằm mục đích hoạch định tài nguyên trong tổ chức đó. Đó có thể là một doanh nghiệp, hay bất kì loại tổ chức nào khác... [↑](#footnote-ref-17)
17. Chương trình hỗ trợ làm việc nhóm [↑](#footnote-ref-18)
18. Bảng thông báo, thông báo các thông tin mới đến các thành viên trong nhóm [↑](#footnote-ref-19)
19. Chương trình hỗ trợ họp nhóm bằng Video, hỗ trợ làm việc từ xa với nhau, khi các nhân viên không đang cùng ở chung một nơi, chẳng hạn như khi đang ở các quốc gia khác nhau. [↑](#footnote-ref-20)