

효율적인 방법론 테일러링에 대한 연구-의료업종을 중심으로

최원영, 김순곤
중부대학교 정보과학과
e-mail : cwyo112@naver.com

A Study for Effective Method Tailoring in Construction of Medical Information System

Won-Young Choi, Soon-Gohn Kim
Dept. of Information Science, Joong-Bu University

요 약

국내 의료정보시스템 구축 시 구축 범위의 확대와 단 납기 추세가 심화되어 감에 따라 많은 의료정보시스템 개발기관들은 방법론의 적용에 있어서 개발기관 위주의 무분별한 테일러링을 채택하고 있으며, 이로 인하여 만성적인 납기의 지연과 소프트웨어 품질의 저하라는 심각한 부작용을 나타내고 있다. 이에 본 연구에서는 소프트웨어 품질과 생산성 향상에 기여할 수 있도록 의료정보시스템 구축 환경에 따른 테일러링 규칙을 제시함으로써 실무적인 가이드라인을 제시하고자 한다.

1. 서론

최근 의료정보시스템 구축에 있어서 주된 특징은 구축 대상 범위의 확대와 단 납기 추세라고 할 수 있다. 이로 인해 의료정보시스템 개발 기관들은 짧은 기간 내에 많은 범위의 시스템 개발을 위해서 사실상 기존의 소프트웨어 구축 방법론을 포기하고 있는 실정이다. 하지만 이것은 소프트웨어 개발에 있어서 생산성을 향상시키지도 못하며, 품질에 있어서는 심각한 부작용을 나타내고 있다. 따라서 소프트웨어 구축에 있어서 생산성과 품질을 동시에 향상시키기 위해서는 기존의 소프트웨어 구축 방법론을 의료정보시스템 구축 환경에 맞도록 테일러링(tailoring)할 필요가 있다.

소프트웨어 방법론을 사용할 때 테일러링이 필요하다는 것은 오래 전부터 인식되어오고 있다. 특히 De Marco 는 많은 기업들이 방법론이라는 이름 하에 구체화시키고 정당화시키는데 쏟아 붓는 대단히 유연하지 못한 규칙들의 집합에 대해 격분하면서 테일러링을 위한 시작점으로서만 방법론을 사용할 것을 주장하였다[4]. 또한 Sommerville 등도 어떠한 방법론이든 특별한 사용 환경에 맞게 개조되어야 한다고 하였다[5]. 하지만 이러한 논쟁에도 불구하고 방법론 테일러링에 대한 연구는 많이 수행되지 않고 있다[1].

2. 선행연구

소프트웨어 개발 문헌에 있어서 많은 주제와 이슈로서, 소프트웨어 엔지니어링 방법론의 테일러링은 서로 다른 많은 용어들로 언급되어 오고 있다. 하지만 이러한 상이한 용어들은 상황 기반 방법론 선택(contingency-based method selection)과 방법론 엔지니어링(method engineering)이라는 대단히 중요한 두 가지의

접근 방법 가운데 하나로 분류될 수 있다.

상황 기반 방법론 선택은 소프트웨어 개발 방법론을 보편적으로 적용되는 것으로서 받아들이기 보다는 차라리 팀이 개발 방법론의 넓은 포트폴리오(portfolio)로부터 각기 다른 프로젝트 환경에 맞추기 위해서 방법론을 선택해야 한다는 전제에 기반을 두고 있다. 반면에 방법론 엔지니어링은 라이브러리(library)로부터 적절한 방법론을 선택하지 않고, 현재의 방법론 조각들을 가지고 처음부터 다시 시작하여 새로운 방법론이 구축되어지거나 엔지니어링 되어진다[2].

본 연구는 국내 시스템통합 업체들의 방법론을 검토하여 국내의 의료정보시스템 구축 환경에 적합하도록 방법론의 표준 테일러링 결과를 제시한다. 그런 다음 의료기관의 업력, 개발기관의 프로세스 관리 능력, 전산감리의 수검 필요 여부에 따라 다시 테일러링이 가능하도록 구성하였다. 그리고 기존의 연구[3]가 어느 특정 방법론의 테일러링에 초점을 맞춘 것에 반하여 본 연구는 특정 업종을 대상으로 방법론을 테일러링하고 있다는 점이 기존 연구와의 차별화된 점이라고 생각된다.

3. 연구방법

본 연구는 아래 그림 1에 제시된 바와 같이 6 단계로 수행되었다. 먼저 국내 의료분야 시스템통합 업체들의 방법론 자료와 실제 사이트에 적용된 사례를 수집하여 정성적인 분석을 실시하였다. 이를 토대로 의료정보시스템 구축 상황에 적합한 것으로 판단되는 표준 테일러링 결과를 제시하였으며, 다양한 상황에 따라 적용 가능하도록 테일러링 알고리즘을 제시하였다. 테일러링 방법 적용을 위한 간단한 프로토타입(prototype)이 구현되고, 일정한 수의 의료정보시스템

분야 전문가의 설문조사를 통하여 구축된 테일러링 결과와 방법을 평가하였다. 마지막으로 현장에 적용을 함으로써 테일러링 알고리즘을 평가하였다.

가. 자료수집

국내 의료분야 SI(System Integration, 시스템통합)업체들의 소프트웨어구축 방법론을 분석하기 위하여 업체가 보유한 방법론 자료를 수집하였는데, 의료분야 대형 SI 업체 4 군데 중 자료 수집이 어려웠던 1 군데를 제외하고 3 군데의 자료를 수집하였다. 또한 각 업체들이 실제 사이트에 어떻게 적용하고 있는지를 분석하기 위하여 가장 최근에 적용된 사례를 의료기관의 협조를 구하여 수집하였다.

나. 자료분석

다음으로 각 사의 소프트웨어 구축 방법론에 대해 살펴 보면, A 사 소프트웨어구축 방법론의 경우 반복적 개발방식으로 구성되어 있으며, 중소 규모의 단납기 개발에 적합하도록 간략화되어 있다.

B 사 소프트웨어구축 방법론의 경우 반복·점증적 개발방식에, 정보공학/객체지향/CBD 의 공정과 산출물을 기반으로 구성되어 있다. 또한 핵심 산출물(53 개 내외의 필수 산출물)과 제출 산출물(유지보수용 최소 산출물)에 대한 규정을 제시하였으며, 산출물간의 선후연관도를 제시하고 있는 것을 발견하였다.

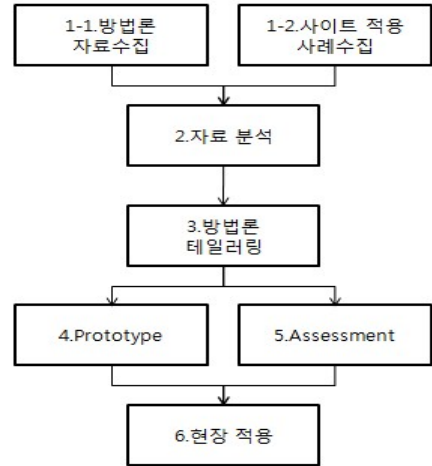
C 사 소프트웨어구축 방법론의 경우 점증적 개발방식에, 마르미 방법론을 기반(CBD 방법론 지향)으로 구성되어 있으며 RUP 와 유사성을 갖추고 있다[7].

다. 테일러링을 위한 표준절차의 정의

국내 의료 분야 3 개 SI 사의 방법론 분석 결과, 테일러링을 위한 표준 절차를 구할 수 있었다. 먼저 개발 단계에 있어서는 기존의 분석 단계가 요구사항정의 단계와 분석 단계로 분리된 것을 발견할 수 있다. 최근 소프트웨어공학의 경향이 요구사항정의를 중요시하는 것을 감안할 때 적절하게 정의된 것으로 판단된다. 또한 설계 단계와 개발 단계가 합쳐져서 상세 설계 및 개발 단계로 통합되었으며, 대개의 경우 3 회 이상 반복적으로 시행되고 있음을 볼 수 있다. 이는 기존의 프로토타입 모형이 설계 단계 초기에 사용자와 한 번 정도의 검토 과정을 거치며 반하여 계속적으로 사용자와의 검토 과정을 거칠 수 있다는 데에서 긍정적인 것으로 판단된다.

다음으로 개발 단계별 주요 액티비티와 활동들을 살펴보면, 각 사의 방법론은 객체지향 방법론을 따르는 경우에도 기존 방법론의 틀을 고수하고 있는 것을 살펴볼 수 있다. 이는 아직 의료정보시스템과 같은 업무 개발에 있어서는 완전한 의미의 객체지향 방법론이 경험상 쉽지 않았음을 반증하는 것이라고 판단된다. 또한 테스트의 중요성에 대한 인식의 변화를 읽을 수가 있는데, 과거에는 시스템 가동 한 두 달 전에 두 세 번 정도의 테스트(통합테스트, 리허설의 구분이 없는)만 수행되는 것이 관례였으나, 최근에는

충분한 기간의 통합테스트 및 리허설 기간을 설정하고 있는 것을 발견할 수 있다. 그 외에도 과거에는 형식적이었던 품질 활동이 이제는 주요 이정표 별로 내부 및 외부 검토 과정을 거치고 있는 것을 발견할 수 있다. 이는 과거에 비해 의료정보시스템의 규모가 커지고, 의료기관 내에서 정보시스템 사용의 중요성이 높아졌으며, 이에 따른 사용자들의 인식 또한 변화되고 있는 것에 기인한 것으로 판단된다.



(그림 1) 연구방법

4. 연구결과

가. 테일러링 규칙 도출

방법론의 표준절차 상에서 각 프로젝트의 환경에 맞게 적절한 테일러링이 가능하도록 테일러링 규칙을 도출하였다.

1) 테일러링 규칙-1

의료정보시스템 구축은 여전히 SI 개발 차원에서 이루어지고 있으나, 최근에는 많은 기업들이 패키지에 대한 커스터마이징 개념으로 접근하는 경우도 많이 발견되고 있다. 이는 과거보다 규모는 커지고 있으나, 납기는 오히려 짧아지고 있는 데에 대한 기업들의 자구책이 반영된 것으로 판단된다. 참고로 이웃 나라인 일본의 경우는 의료정보 시장 자체가 이미 패키지에 대한 커스터마이징 개념으로 전환된 것을 발견할 수 있었다[6]. 따라서 테일러링 규칙 첫 번째로서 패키지 커스터마이징 여부에 따라 테일러링 결과가 달라질 수 있도록 구성하였다.

2) 테일러링 규칙-2

국내의 경우 1987 년 정보시스템 감리 제도가 도입되었으나, 대개 공공기관 위주로 진행이 되었다. 그러다가 2007 년 의무감리제도가 시행되면서부터 의료기관에서도 공공의료기관을 중심으로 정보시스템 감리가 시행되었다[8]. 감리 수행은 대개 인터뷰 또는 문서를 통해 평가가 이루어지다 보니 이에 대한 대응을 위해 몇 가지 산출물들이 추가적으로 필요하게 되었다. 따라서 테일러링 규칙 두 번째로서 감리 대응 필요 여부에 따라 테일러링 결과가 달라질 수 있도록

구성하였다.

3) 테일러링 규칙-3

의료기관의 경우 크게 보면 비슷한 업무들로 구성되어 있지만, 의료기관의 규모와 업력에 따라 사용자들이 요구하는 수준은 천차만별인 것으로 알려져 있다. 즉, 의료기관의 규모와 업력이 크면 클수록 사용자들의 요구 수준은 어려우며, 이에 대한 대응 방안으로는 개발 단계가 더 이상 진행되기 전에 내부 품질 검토 및 사용자들에 의한 검토 과정이 필요하다는 것이다. 그렇지 않으면 개발의 마지막 단계에 가서 수정 및 보완해야 할 물량이 많아질 것이다. 이는 납기를 맞추지 못하는 원인이 될 뿐만 아니라 사용자들로부터 신뢰를 잃어버릴 수도 있는 매우 중요한 것이므로 가급적 개발 단계가 더 이상 진행되기 전에 검토 과정을 거칠 필요가 있다. 따라서 테일러링 규칙 세 번째로서 단계별 검토 강화의 필요 여부에 따라 테일러링 결과가 달라질 수 있도록 구성하였다.

4) 테일러링 규칙-4,6

국내 의료정보 업계의 경우 여전히 구조적 및 정보공학방법론에 근거한 방법론을 채택한 곳이 많다. 몇 년 전 객체지향방법론이 소개될 때 대부분 객체지향 방법론으로 방법론이 개편되었으나 시행상의 어려움으로 인하여 많은 경우 구조적 및 정보공학방법론으로 회귀한 상태이다. 하지만 여전히 객체지향방법론을 근거로 하는 개발기업도 존재하고 있기 때문에 이 둘을 구별해서 테일러링할 필요가 있다. 따라서 테일러링 규칙 네 번째와 여섯 번째로서 객체지향 방법론 적용 여부에 따라 테일러링 결과가 달라질 수 있도록 구성하였다.

5) 테일러링 규칙-5

의료정보시스템을 개발할 때 국내 개발업체들의 경우 대개는 상세설계 및 개발단계를 반복적으로 수행하는 경우가 대부분이다. 이 경우 개발 중인 화면과 각종 보고서는 지속적으로 변화하는 경향이 있기 때문에 이 단계에서 산출물로 만들지 않으며 개발이 완료되고 난 다음 산출물로 만드는 경우가 많다. 하지만 프로젝트에 따라서는 절차적 방법에 따라 프로젝트를 진행하는 경우도 있으며 이 경우에는 해당 단계에서 산출물이 만들어지고, 향후에 수정 및 보완된다. 따라서 테일러링 규칙 다섯 번째로서 절차적 방법의 적용 여부에 따라 테일러링 결과가 달라질 수 있도록 구성하였다(엄밀하게 말하면 테일러링 결과는 동일하나, 산출물 제출 시기가 상이하게 됨).

나. 현장적용

국내 개발기관의 경우 비교적 소프트웨어 품질에 대한 개념이 부족한 것으로 판단된다. 대형 개발기관을 제외하고는 품질관리를 위한 조직과 방법론조차 구비되어 있지 않은 경우가 많았다. 또한 산출물 작성에 매우 소극적인 것을 알 수 있었다. 개발기관들이 대개 반복적 개발을 채택하고 있기는 하였으나,

사양의 변경 시 산출물에 반영하기 보다는 직접 소스 코드에 반영하는 것이 관례화되어 있는 경우가 많았다. 하지만 이는 향후 유지보수에 있어서 많은 어려움을 낳게 되고, 그 결과로서 구축된 시스템의 수명이 짧아지는 원인을 제공하고 있는 것으로 판단된다.

다음에 제시된 표 1 은 본 연구에서 제안된 방법론 테일러링 규칙을 실제 현장에 적용하였을 때 해당 사이트에서의 결과를 나타내고 있다.

<표 1> 테일러링 규칙 적용

규칙	조건	결과	Output
No.1	패키지 커스터마이징?	No	-
No.2	감리대응필요?	No	-
No.3	단계별검토강화?	Yes	검토및시정조치 결과서
No.4	객체지향분석?	No	논리 ERD, 프로세스흐름도
No.5	절차적 방법?	Yes	화면정의서, 보고서정의서
No.6	객체지향설계?	No	DB 트리거명세서, 프로시저명세서

1) 테일러링 규칙 No.1 과의 차이점 분석

해당 사이트는 계약 조건 상 사용자 요구사항의 분석 후 시스템을 구축하는 사이트이므로 규칙 No.1의 적용을 받지 않는다. 따라서 GAP 분석서를 작성할 필요가 없다. 하지만 실제로는 GAP 분석서를 임의의 양식으로 현업 사용자들이 직접 작성하게 하고 그 결과를 받아서 시스템 구축에 반영하고 있었다. 이것은 의료정보시스템 개발 업체들이 사실상 업체들이 보유하고 있는 시스템을 상당수 재활용하고 있다는 것을 나타내며, 현실적으로 계약 조건과는 관계없이 패키지 커스터마이징의 형태로 개발하고 있음을 보여준다고 판단된다. 유사한 사례로서 K1 대학교병원, K 암센터, K2 대학교병원 등에서도 패키지의 커스터마이징 형태로 시스템이 구축되었음을 관계자와의 면담 결과 확인하였다.

2) 테일러링 규칙 No.2 와의 차이점 분석

해당 사이트는 사립대학교병원으로서 감리 수검 대상 기관이 아니므로(공공기관으로서 5억 이상의 개발 사업 시에는 감리 수검을 의무화하고 있음) 규칙 No.2의 적용을 받지 않는다. 따라서 요구사항추적표를 작성할 필요가 없다. 하지만 실제로는 비공식 문서로서 작성을 하기로 하였다. 물론 요구사항의 추적이 쉽지 않은 관계로 요구사항-화면-단위테스트간의 관계만 추적하는 것으로 축소 조정을 하였다. 이것은 향후 시스템 구축이 완료된 후 검수 과정을 염두에 둔 해당 기업의 전략이라고 판단된다. 시스템 구축 후 검수 시에는 현업 사용자의 요구사항이 해당 프로그램에 반영되었는지의 여부가 가장 중요한 만큼, 사용자의 요구사항이 화면(프로그램)으로 구현되고 해당 화면의 테스트 결과를 사용자들이 확인할 수 있다면 비교적 검수를 받기가 쉬워질 것으로 판단된다.

3) 테일러링 규칙 No.3 와의 차이점 분석

해당 사이트는 소프트웨어 구축 단계별 검토의 중요성과 필요성을 인식하고 있었으므로 규칙 No.3 을 적용하여 검토 및 시정조치 결과서를 작성해야 한다. 하지만 실제로는 요구사항정의 단계에서의 검토는 요구사항정의서에 데이터를 갱신하는 방식을 채택하였으며, 분석 단계에서의 검토는 의사결정요청서의 작성으로, 그리고 상세설계 및 개발 단계에서의 검토는 현업 사용자의 검토로 대체하였다. 이것은 개발기관 내에 품질보증 관련 조직과 방법론의 부재에 원인이 있었던 것으로 판단된다. 현재 개발기관의 방법은 사용자 요구사항의 누락 방지에는 도움이 될 수 있으나, 객관적인 시각에서의 품질검토에는 한계가 있을 것으로 판단된다.

4) 테일러링 규칙 No.4 와의 차이점 분석

해당 사이트는 객체지향분석 방법을 채택하고 있지 않으며, 정보공학에 근거한 자체 방법론을 보유하고 있었다. 따라서 데이터 관점에서는 ERD(Entity Relationship Diagram, 개체관계도), 프로세스 관점에서는 프로세스 흐름도의 작성을 비교적 중시하고 있었다. 이 중에서 ERD 의 경우는 논리 및 물리 ERD 모두 작성하는 것을 원칙으로 하고 있었다. 해당 기업은 자체의 의료정보시스템을 보유하고 있었던 관계로 분석 단계가 끝나면 먼저 테이블을 수정한 다음 역공학에 의하여 물리 ERD 를 만들어 내고 있었다 (CASE(Computer Aided Software Engineering) 도구 활용).

5) 테일러링 규칙 No.5 와의 차이점 분석

해당 사이트는 절차적 방법론을 채택하고 있었으므로 규칙 No.5 를 적용하여 화면정의서와 보고서정의서를 작성하고 있었다. 다만 보고서정의서는 작성되지 않았으나 화면정의서로 대체하여 작성하고 있으므로 별 차이는 없다고 판단된다. 그리고 화면정의서는 기능 명세를 포함할 수 있도록 작성함으로써 분석 단계에서 작성을 생략하였던 기능명세서를 포함하고 있었다.

6) 테일러링 규칙 No.6 과의 차이점 분석

해당 사이트는 객체지향설계 방법을 채택하고 있지 않으며, 정보공학에 근거한 자체 방법론을 보유하고 있었으므로 규칙 No.6 를 적용하여 DB 트리거명세서, 프로시저/함수명세서를 작성해야 한다. 하지만 작성이 생략되어 있는데, 이는 상세설계 및 개발 단계에서 반복적 개발 방식을 채택하는 기업들의 공통적인 특성으로 파악된다. 이 단계에서는 개발과 사용자 검토가 반복적으로 이루어지는 과정에서 지속적인 수정 및 보완 사항이 발생하게 되므로 해당 단계에서 문서화시키는 것이 업무적으로도 부담이 되는 것으로 판단된다. 따라서 프로그램 개발과 검토가 종료된 후 사후에 문서를 작성하는 경우도 많은 것으로 보여진다. 그리고 한 가지 특성은 규칙 상 작성되지 않아도 될 배치(Batch)프로그램 명세서의 작성을 살펴 볼 수 있다. 이는 외래 및 입원 청구집계, 급여계산, 고장자

산 감가상각처리 등의 주요 배치성기능에 적용되고 있는데, 화면정의서 또는 프로시저명세서로 대체될 수 있는 부분이라고 판단된다.

7) 기타 차이점 분석

변경요청서, 단위테스트결과서, 데이터전환결과서의 경우는 표준 적용 항목으로 정의되어 있으나 해당 기관에서 채택되지 않았는데, 이는 개발기관의 프로세스 관리 능력과 관계되는 것으로 판단된다. 그리고 통합테스트결과서, 리허설결과서의 경우는 대개 발주 기관에서 테스트 결과를 정리하는 경우가 많으므로 표준 적용 항목으로 정의되어 있지 않으나 해당 기관에서는 채택되어 있는데, 이는 반대로 발주기관의 관리 능력과 관계되는 것으로 판단된다.

5. 결론

국내 의료정보시스템 구축 시 품질과 생산성 향상에 기여하고자 테일러링 규칙을 도출해 내고 현장에 적용한 결과 실제 개발자들이 작업하고 있는 것과 차이점이 크지 않았다. 다만 테일러링에 대한 원칙 없이 개발자 편의대로 작업이 이루어지는 것이 문제였다. 본 연구는 무분별한 테일러링이 이루어지고 있는 현장에 실무적인 가이드라인을 제시할 수 있을 것으로 기대된다.

참고문헌

- [1] Aydin, M. and Harmsen, F., et al., "An agile information systems development method in use", Turk. J. Electron. Eng. 12, 2, 2004, pp.127-138
- [2] Brinkkemper, S., "Method engineering: Engineering of information systems development methods and tools", Inform. Softw. Tech. 38, 4, 1996, pp.275-280
- [3] Conboy, K., Fitzgerald, B., "Method and Developer Characteristics for Effective Agile Method Tailoring: A Study of XP Expert Opinion", ACM Transactions on Software Engineering and Methodology, Vol. 20, No. 1, Article 2, 2010
- [4] De Marco, T., "Controlling Software Projects: Management Measurement and Estimation", Prentice-Hall, Englewood Cliffs, NJ., 1982, pp.13
- [5] Sommerville, I. and Ransom, J. B., "An industrial experiment in requirements engineering process assessment and improvement", ACM Transactions on Software Engineering and Methodology, Vol.14, No.1, Article33, 2005, pp.93
- [6] (株)麻生情報システム, (株) I C M ジャパン, "電子カルテシステム及び看護支援システムプロジェクト計画書", 株式会社麻生 飯塚病院, 2011
- [7] (주)GNSK, (주)최앤어소시에이츠, "2010 년도 SW 품질 및 경쟁력 제고를 위한 SW 공학기술 현장 적용 지원사업 중·대형 의료기관 소프트웨어 구축 프로세스 정립 최종보고서", 정보통신산업진흥원, 2010
- [8] 한국정보사회진흥원, "정보시스템 감리원 기본교육", 한국정보사회진흥원, 2008