

정신 모형이 인터랙티브 시스템 설계에 미치는 영향에 관한 연구

A study on the Effects of Mental model on Interactive system design

한 혁 수*

Hyuksoo Han

요 약 인터랙티브 시스템은 여러 번에 걸친 평가 와 수정을 거쳐서 사용자들에게 보다 편리한 방향으로 설계된다. 인터랙티브 시스템의 인터페이스 설계자들은 사용자들의 만족도와 소프트웨어의 기능성을 평가하기 위해 표본집단을 구성하고 그들로 하여금 시스템을 사용해 보게 한 후에 QUIS (Questionnaire for User Interaction Satisfaction) 와 같은 평가 양식에 따라 사용성을 테스트 해 왔다. 그러나, 이러한 방식으로는 사용자가 시스템에 대해서 사용하기 전에 형성하고 있는 정신모형에 관한 평가가 이루어지지 않았다. 정신 모형은 사용자들의 배경과 경험을 통하여 얻어진 시스템에 대한 기대를 말한다. 만약 인터랙티브 시스템의 인터페이스가 사용자의 정신모형과 일치하도록 만들어진다면, 사용자들은 그들의 휴리스틱(Heuristic)을 이용하여 배우는 시간을 단축할 수 있고, 보다 효율적으로 소프트웨어를 사용할 수 있다. 인터페이스의 설계가 정신 모형에 기반을 두고 이루어지면, 사용자들이 시스템에 대해 갖는 기대를 파악하여 인터페이스를 수정 보완 함으로서 소프트웨어의 사용성을 향상시킬 수 있고, 사용자들에게 보다 친숙한 인터페이스를 제공할 수 있다.

1. 서론

대부분의 컴퓨터 사용자들은 특정 시스템을 처음 사용했을 때의 당혹스러운 기억들을 가지고 있다. 과거 보다는 편하도록 만들어 지고 있기는 하지만 여전히 개선되어야 할 부분들이 많이 있다. 사용자는 특정한 목적을 가지고 시스템을 사용하는데, 사용법이 부담이 되거나 이로 인하여 작업능률이 떨어지면 시스템을 사용하여 얻어지는 소득이 없게 되며 심지어는 사용하지 않는 것 보다는 못하게 된다.

시분할(Time sharing) 시스템이 개발된 이후로 사용자들은 인터랙티브(Interactive)한 방식으로 컴퓨터 시스템을 사용하기 시작했다. 배치(Batch)방식과는 달리, 인터랙티브 시스템에서는 기능성이외에 시스템이 제공하는 인터페이스가 사용자의 작업 능력과 만족도를 좌우하게 되었는데, 하드웨어의 기술적

발전과 더불어 컴퓨터 인터페이스도 급격히 향상되어 왔다.

오늘날, 인터랙티브 시스템은 컴퓨터의 사용 없이 이루어 지던 작업을 더욱 효율적으로 대신하기 위해 개발되어진다. 그러므로, 이러한 시스템들은 사용자들과 그들의 작업 방식에 대한 정확한 지식을 기반으로 이루어 져야 한다.

인터랙티브 시스템의 인터페이스 설계에서 가장 먼저 고려하는 것은 시스템이 가지고 있는 기능들에 사용자가 접근할 수 있도록 방법을 제공하는 것이다. 사용자는 제공된 인터페이스를 통해 자신이 원하는 기능을 제공 받고 의도한 작업을 수행할 수 있다. 인터페이스가 이러한 기능들에 접근하기 힘들게 제공되거나, 어떻게 해야 할지 분명하지 않게 제공되면 인터랙티브 시스템은 실패하게 된다[1].

시스템의 인터페이스 설계는 여러 계층의 사용자를 대상으로 하는 창의적인 작업이기 때문에 한 번의 생각으로 결정되지 않는다. 여러 번에 걸쳐 설계-평가

* 서울 종로구 홍지동 7번지
상명대학교 컴퓨터학부

수정의 과정을 반복하여 만족 할 만한 설계를 이끌어 낸다(2).

인터페이스 설계자는 사용자의 작업 단계들을 예상하고 적합한 모델을 설계한다. 이렇게 설계된 인터페이스가 사용자들의 경험과 직관에 기초하는 정신 모형(Mental Model)에 맞게 설계된다면, 사용자는 배우는 시간도 줄일 수 있고, 사용 과정에서 에러(Error)도 줄일 수 있을 것이다. 본 논문에서는 사용자 인터페이스 설계에 사용자의 정신 모형을 반영할 수 있는 방법에 대해 연구하였다. 정신 모형을 기반으로 하는 인터페이스 설계 모델을 구성하였으며, 기존의 평가 방식과의 차이를 보기 위해 마이크로 소프트웨어사의 Visual Basic 5.0을 대상으로 인터페이스 평가 양식을 만들고 기존의 평가 방식과 비교 분석하였다

2. 인터페이스 설계

사용자 인터페이스는 사용자와 컴퓨터 시스템과의 대화 방법이다. 사용자는 소프트웨어의 구현 방법이나 기술에는 관심이 없고 인터페이스가 제공하는 대로 시스템을 받아 들이고 사용한다. 그러므로, 사용자 인터페이스는 사용자가 하고자 하는 작업만을 위해 컴퓨터가 존재하는 것처럼 설계될 때 가장 효과적이라고 할 수 있다(3).

사용자 인터페이스를 설계하는 것은 분석적이고 창의적인 작업이다. 우수한 사용자 인터페이스는 단 한번의 판단으로 이루어지는 것이 아니라, 그림 1에서와 같이 여러 번에 걸친 평가와 수정을 거쳐 이루어진다(4).

1)시스템이 지원하게 될 작업과 사용자들을 연구하고 분석한다.

개발될 시스템이 지원할 업무와 시스템의 사용자들을 분석하고 연구한다. 컴퓨터 시스템을 사용하지 않고 있는 현재 환경의 불편함을 파악하고, 인터랙티브 시스템을 사용하여 이러한 문제의 해결책을 찾는다.

2)시스템의 기능에 대한 요구사항을 정의 한다.

시스템이 갖추어야 할 기능들을 사용자에게 연결할 수 있는 사용자 인터페이스 요구 사항들을 정의한다.

3)사용자 인터페이스를 설계한다.

사용자 인터페이스의 설계는 소프트웨어 기능 설계와는 다르기 때문에 소프트웨어의 수행과정 보다는 사용자와 시스템과의 인터랙션 방식을 분석하여 요구사항을 기록한다. 인터랙션의 순서는 인지 프로세스

를 나타내고 있기 때문에, 사용자의 작업과정에 대한 개념모형(Conceptual Model)을 정의하여야 한다.

4)프로토타입(Prototype)을 만든다.

인터페이스 설계자가 선택한 인터페이스의 스타일과 작업방법이 나타날 수 있도록 프로토타입을 만든다.

5)프로토타입을 평가하고, 결과에 따라 인터페이스를 수정한다.

인터페이스 설계과정에서 설계된 인터페이스가 사용자들의 요구사항을 충족시키는가를 판단하는 인터페이스 평가는 매우 중요한 역할을 한다.

인터페이스의 평가는 매우 복잡하고 어려운 작업이다. 어떤 제품은 여러 가지 요소가 완전하지 못한 데도 사용자가 별 불편함을 못 느끼도록 만들어지기도 하고, 어떤 제품은, 거의 모든 것이 완벽한데 잘못된 하나의 요소 때문에, 전혀 쓸모가 없이 되어 버린다. 그러므로, 제품의 사용성을 제대로 평가하는 포괄적

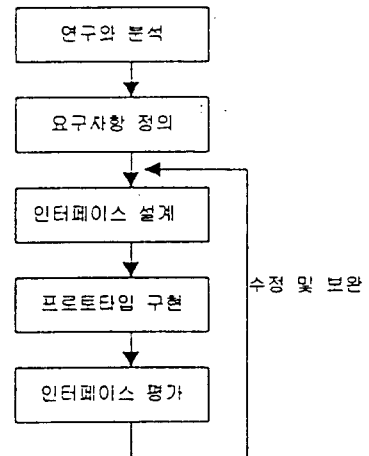


그림 1) 반복 설계

이고, 체계적인 방법은 인터페이스 설계에 매우 중요한 역할을 한다.

6)만족할 때까지 3, 4, 5 번을 반복한다.

시스템의 개발 기간이나 비용 등 제한 사항들을 고려하여 평가와 수정 작업을 수행한다. 완성품이 나온 이후에도 사용자들로부터의 평가 반응을 고려하여 수정을 반복한다.

기존의 평가 방식들은 이러한 사용성을 평가하기 위해 표본 집단을 설정하고 이들로 하여금 시스템을 사용하도록 한 후 사용자들의 만족도를 평가하고, 또는 테스트 작업을 수행하도록 하여 소프트웨어의 기능성을 평가했다.

사용자의 시스템에 대한 정신 모형은 시스템을 배우고, 익숙해지는 과정에 중요한 역할을 한다. 만약, 인터페이스가 사용자들의 정신 모형과 일치하도록 만들어진다면, 사용자는 잘못된 정신모형을 고치는 시간을 줄이고 시스템에 빨리 적응할 수 있다. 정신 모형에 영향을 미치는 요소들은 상당히 여러 가지 측면에서 찾을 수 있겠지만 크게 모든 사용자에게 공통되는 것과, 사용자마다 각각 다른 것들로 구분된다. 모든 사용자에게 공통되는 요소는 인간이 어떻게 받아들이고 어떻게 반응하는가에 대한 본질적인 것이고, 사용자의 지식이나 경험, 물리적 특성, 사용자가 사용했던 도구 등은 사용자마다 각각 다른 요소들이라 할 수 있다.

3. 정신 모형

사용자는 어떤 시스템을 처음 접했을 때 그 시스템에 대해서 자기 나름대로의 이미지를 갖게 된다. 이러한 이미지를 정신 모형이라고 한다. 정신 모형은 "어떤 시스템에 대해 사용자가 가지고 있는 개념적인 이해의 내부적인 표현"이다.

사용자의 시스템에 대한 정신 모형 즉, 사용하고자 하는 시스템에 대한 이미지 형성은 사용자가 그 시스템을 배우고, 익숙해지는 과정에 중요한 역할을 한다. 사용자가 배우는 과정 없이 소프트웨어를 접하게 되면, 나름대로의 정신 모형을 형성하고 시스템을 사용하게 된다. 처음 접했을 때는 그 시스템에 대한 불완전하고, 때로는 잘못된 정신 모형을 가지게 되지만, 점차 시스템에 익숙해짐에 따라 사용자의 정신 모형은 정확해지고, 완전한 모습을 가지게 된다(4, 5).

그러므로, 소프트웨어의 인터페이스가 사용자들의 정신 모형과 일치하도록 만들어진다면, 사용자는 잘못된 정신모형을 고치는 시간을 줄이고 시스템에 빨리 적응할 수 있게 될 것이다.

사용자가 시스템을 사용할 때 정신모형은 사용자의 액션(Action), 시스템의 반응(Response), 그리고 사용자의 작업 목적(Purpose)을 연결하는 원인-결과 연결(Cause-Effect Connection)이라고 생각할 수 있다. 사용자는 특정한 목적을 이루기 위해 시스템의 인터페이스를 통해 액션을 취하고, 시스템은 그 액션에 대해 준비된 반응(Response)을 보인다. 이 반응이 사용자의 기대와 일치 한다면, 그 작업에 대한 정신 모형은 적절한 것이었으며, 그렇지 않으면, 정신 모형을 수정해야 한다.

사람의 정신 모형이 형성되는 특징들을 살펴보면

다음과 같다.

첫째로, 정신 모형은 시스템의 보이는 부분이 어떻게 보이지 않는 부분과 연결되어 있는지에 대한 가치관과 전체를 가지고 있다.

둘째로, 사람들이 시스템을 처음 접하였을 때 개념적 논리보다는 기존에 가지고 있는 경험이나 유사한 정신 모형을 사용하여 문제를 풀려고 시도한다.

셋째는 정신 모형에 대해서 명백한 부정적인 반응이 있을 때만 정신 모형을 수정하고, 문제 해결의 방향을 바꾼다. 그러므로 비효율적이고 잘못된 정신 모형을 가지고 있더라도 명백한 부정적인 반응이 오지 않는다면 사용자가 이 정신 모형을 수정하지 않는다.

넷째로, 정신 모형은 완전하지 않고 또한 정신 모형끼리의 상호 경계가 명확하지 않다. 그래서 비슷한 모형끼리는 혼란을 가지게 되는 것이다.

다섯째로, 정신 모형은 비과학적이며 비효율적일 수 있다는 것이다. 그럼에도 불구하고, 사람들은 조금의 수고가 더 들더라도 정신 모형을 적용하려 한다.

이러한 특성을 살펴보면, 인터페이스 설계 시에 사용자들의 정신 모형은 매우 중요한 요소이며 반드시 반영되어야 한다.

4. 정신 모형의 적용

지금까지의 인터페이스 평가는 주관적인 사용자의 만족도를 평가하기 위해 인터뷰(Interview)나 설문조사등을 하였으며, 평가기준에 대한 객관적인 검증을 위해서는 표본 집단을 추출하여 테스트하는 방법을 취하였다. 이러한 방법의 대표적인 평가 양식은 QUIS(Questionnaire for User Interaction Satisfaction)이다. 이 양식은 Maryland 대학의 Shneiderman 박사 팀이 만든 것으로 가장 많이 채택되고 있다(6). 그림 2는 QUIS의 한 예를 보여 주고 있다.

5.1 시스템 전반에 걸친 용어의 사용	불일치	일치
	1 2 3 4 5 6 7 8 9	NA
5.1.2 작업용어	불일치	일치
	1 2 3 4 5 6 7 8 9	NA
5.1.3 컴퓨터 용어	불일치	일치
	1 2 3 4 5 6 7 8 9	NA
6.3 용어와 명령어를 기억하기	어려웠다	쉬웠다
	1 2 3 4 5 6 7 8 9	NA

그림 2) QUIS 평가 양식

이러한 평가 방법은 사용자의 주관적인 의견을 수렴하는 데는 적합 하지만, 객관적인 자료를 얻기는 어렵다.

기존의 평가 방법들이 시스템을 사용한 후 사용자들의 만족도와 작업 능률만을 평가하는 것인데 비하여, 정신 모형을 기반으로 하는 평가가 추가되면 수정과 보완을 위한 제안에 중점을 두는 시스템의 특성에 맞는 효과적인 평가방법을 제공할 수 있다.

본 연구의 목표는 사용자의 정신 모형을 적용하는 인터페이스 평가 방법을 제공하여 시스템의 특성에 맞는 요소들로 인터페이스들을 평가 할 수 있도록 하고, 평가 결과로 얻어진 자료들을 분석하여 세부적인 수정 제안을 제공할 수 있도록 하는데 있다.

본 논문에서는 정신 모형에 기반을 둔 평가 모델을 제시하여 기존의 문제점들을 보완하고 보다 정확하게 사용성을 테스트 할 수 있게 한다. 정신 모형에 기반을 둔 평가 양식은 사용자들이 시스템을 사용하기 전 단계에서 인터페이스의 구성 요소들을 대상으로 실시된다. 시스템의 자세하고 정확한 부분들에 대하여 사용자들이 가지고 있는 개념을 평가하기 때문에 사용자들로부터의 피드백이 보다 정확하게 수정에 반영될 수 있다.

그림 3에서 보듯이 정신 모형을 기반으로 하는 평가는 기존의 평가와 상반되는 방식이 아니라 상호 보완하는 방식인 만큼 보다 나은 평가와 수정 작업을 위하여 제시하고자 하는 것이다.

5. 실험 및 분석

본 연구에서는 소프트웨어 개발 도구 Visual Basic 5.0을 테스트 시스템으로 선정하여 이를 대상으로 인터랙티브 시스템을 위한 정신 모형을 이용한 평가양식의 필요성을 실험했다.

5.1 실험 대상

표본 집단은 소프트웨어 개발도구를 한 번도 사용해 보지 않은 사용자들로 구성했다. 이들은 프로그램 경력을 기준으로 상, 중, 하로 나누어, 상급 그룹, 중급 그룹, 하급 그룹으로 분류했다. 각 그룹은 2팀씩 6팀이 되도록 하고 각 팀은 4~5명으로 구성했다. 그리고 소프트웨어 전문가들로 5인의 평가단을 구성했다.

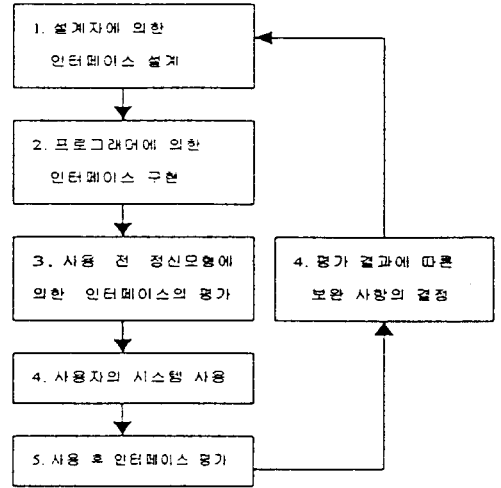


그림 3) 정신 모형에 기반한 인터페이스

5.2 실험 과정

1) QUIS를 이용한 실험

표본 집단 중 한 팀은 Visual Basic 5.0을 사용해 보고 QUIS를 이용하여 평가서를 작성하도록 하였으며, 그 결과 자료를 이용하여 평가단으로 하여금 수정해야 할 사항들에 대해 보고서를 작성하도록 하였다.

2) 정신 모형을 기반으로 하는 평가 양식을 이용한 실험

표본 집단의 다른 한 팀에게는 정신 모형을 기반으로 만들어진 평가 양식에 따라 평가를 실시하도록 하였다. 그리고, 평가단으로 하여금 수정해야 할 사항들에 대해 보고서를 작성하도록 하였다.

5.3 실험 결과

QUIS와 같은 기존의 인터페이스 평가 방식들은 기능성의 전달 과정과 사용자들의 만족도를 중시 여겼기 때문에 사용자가 시스템에 대해 기대하는 요소는 고려하지 않았다. 그러므로 평가로부터 얻어진 결과를 인터페이스 설계를 수정하기 위해 사용하기가 매우 어려웠다.

전체 8 분야로 나누어진 질문 항목들은 "시스템에서 사용된 용어들은 일관성이 있었는가?", "명령어들은 기억하기 쉬웠는가?"와 같이 주로 사용자의 만족도를 측정하는 것들이었다. 이렇게 평가 항목들이 일

반적이고 추상적이라 평가를 수행한 후 얻어진 결론은 추상적인 결론에 도달하는 일이 많았다. 예를 들어 평가 항목이 "용어들은 일관성이 있었는가?" 였고, 그에 대한 평가 결과로 6.5점을 얻었다면, 분석가의 결론은 어떻게 내려져야 하는가? 만약 "10점의 만족을 얻기 위해 용어들을 수정하라"고 결론을 내렸다면, 설계자는 어떻게 수정해야 하는가?

보고서에는 평가 단의 기준에 의해 특정 부분들에 대해 수정이 필요하다는 결론이 내려져 있었다. 어떻게 바꾸어야 하는가에 대해서는 제안이 나타나 있지 않았다.

정신 모형을 기반으로 하는 평가 양식에서는 먼저 아이콘을 제시한 후에 "다음 아이콘은 무엇을 위한 것이라고 생각되는가?" 그리고 메뉴에 대해서도 특정 기능들을 제시하고 "주 메뉴 중 어디에서 이 기능을 찾겠습니까?"라는 질문으로 사용자가 기대하는 것을 살펴 볼 수 있도록 질문을 만들었다.

5.3 정신 모형에 기반한 평가로부터 얻어진 수정 제안

Visual Basic 5.0을 테스트 하고 나서 얻어진 평가 보고서의 일부를 소개 하면, 다음과 같다.

1) 툴바 (Tool Bar)

그 순서나 구성하고 있는 아이콘의 이미지 들은 정신 모형을 다르게 되어 있다. Visual Basic에서 요구하는 프로그래밍 개발 방법의 순서에 따라 각각의 기능의 아이콘들이 배열된 것을 볼 수 있다.

사용자가 자주 쓰는 기능 중에 정렬 기능을 나타내 는 아이콘이 툴바에 있었으면 한다.

2)속성 창(Property Window)

모든 사용자가 시스템이 제공하는 배열을 원하는 것이 아님으로 사용자가 원하는 대로 속성 배열의 순서를 조절할 수 있도록 하자.

3)도구 상자(Tool Box)

작업을 하다보면, 툴 박스의 컨트롤의 추가와 삭제가 빈번하게 일어난다. 그러므로, 맨 처음 프로젝트를 생성하고 폼을 생성한 이후에 특정 폼에 해당하는 툴 박스를 구성하는 위치드를 두는 것을 제안한다.

6. 결론

본 논문에서는 인터랙티브 시스템의 사용자 인터페이스 설계 과정을 살펴보고, 정신 모형을 반영하기 위한 방안을 연구하였다. 그 결과로 인터페이스 설계 과정 중에서 가장 중요한 역할을 하는 인터페이스 평가 단계의 변화를 제안하게 되었다.

기존의 평가 방법들은 사용자의 만족도를 중심으로 평가가 이루어 지고 있으나 평가의 반응을 수정에 반영하기가 힘들었다. 그러나 만약 정신 모형을 바탕으로 하는 평가 방식이 제공되면, 기존의 방식들이 고려하지 않았던 사용자들의 기대 요소에 대한 평가가 이루어 질 수 있으며, 기대에 일치하는 부분과 일치하지 못하는 부분을 분석하여 향상된 설계를 위한 제안을 할 수 있다.

2. 도구 상자(Tool Box) 구성
다음은 도구상자의 구성을 나타내고 있다.

2.1 다음 아이콘들이 어떤 컨트롤들을 나타내는지 예상하여 보시오.

? 알맞은 아이콘이 없을 때

a.	checkbox
b.	label
c.	text box
d.	image
e.	select
f.	radio button
g.	command
h.	frame

그림 4. 정신 모형을 기반으로 하는 평가

인터랙티브 시스템의 설계에 사용자들의 정신 모형을 반영시키면, 실험 결과에서도 볼 수 있듯이 시스템의 기능성과 사용자의 만족도를 중심으로 평가해오던 기존의 방식들과 더불어 사용자들이 시스템에 대해 가지고 있는 기대를 만족시키는 방향으로 인터페이스를 향상시킬 수 있을 것이다.

7. 인용 문헌

- [1] H.Thimbleby, "User Interface Design", Addison Wesley, 1990
- [2] K. Cox and D. Walker, "User Interface Design", Prentice Hall, 1993
- [3] 우치수, 한 혁수 "사용자 인터페이스", 영지 문화사, 1994
- [4] D. J. Mayhew, "Principles and guidelines in software user interface design", Prentice Hall, 1992
- [5] 김 영채, 박권생, "인지 심리학", 박영사, 1991
- [6] B. Shneiderman, "Designing the User Interface", Addison-Wesley Publishing Company, 1992

A Study on the Effects of Mental model on Interface design

Hyuksoo Han

(Computer Science, Sangmyung Unviversity)

Abstract Interactive system is evolved toward convenient product through a number of iteration of prototype and evaluation. User Interface designers of the system hire a sample group of subjects and let them test the system's features using a test model like QUIS(Questionnaire for User Interaction Satisfaction). However, the traditional evaluation methods failed to incorporate the test of mental models of users that are formed by their background and experience before they use the system. The mental model is important because if the system is built conforming to user's mental model, the users easily learn the system just by following their heuristics. If evaluation is performed based on mental model, the designers can find out the expectations that users have about the system. Therefore, they can effectively modify the interface to improve usability and provide friendly interface.