

컨텍스트 인지 시스템을 위한 요구사항 도출 및 명세화 방법

(A Study of Requirements Elicitation and Specification for Context-Aware Systems)

최 종 명 [†]

(Jongmyung Choi)

요 약 컨텍스트 인지 시스템에서 컨텍스트는 매우 중요한 요소이지만, 기존 요구공학은 컨텍스트를 도출하고 명세화하는 방법을 지원하지 못하기 때문에 이를 지원할 수 있는 연구가 필요하다. 본 논문은 컨텍스트 인지 시스템의 요구사항을 효과적으로 도출하기 위한 방법과 명세화 방법을 제안한다. 논문에서 제안하는 요구사항 도출 방법은 6단계로 구성된 점진적이고 반복적인 프로세스로서 비즈니스 로직을 위한 요구사항을 먼저 파악하고, 이를 기반으로 컨텍스트 로직을 위한 요구사항을 파악하고, 컨텍스트 모델링과 서브시스템 식별 등의 작업을 수행한다. 요구사항 명세화 방법으로는 컨텍스트 인지 유스케이스 다이어그램, 컨텍스트의 개념을 표현할 수 있는 컨텍스트 다이어그램, 컨텍스트 타입에 영향을 받는 유스케이스를 표현하는 컨텍스트타입-유스케이스 의존 다이어그램을 제안한다. 논문에서는 또한 제안한 방법을 적용한 시스템에 대한 사례연구를 소개하고, 이를 정성적으로 평가한 내용을 제시한다. 본 연구는 이해관계자가 컨텍스트 인지 시스템의 요구사항을 효과적으로 파악하고, 이를 명확히 기술하며, 이해할 수 있도록 도움을 줄 수 있다.

키워드 : 컨텍스트 인지 시스템, 요구사항 도출, 요구사항 명세화

Abstract Even though context is the most important feature in context-aware systems, the existing requirements engineering cannot support methodology for elicitation and specification of contexts. In this paper, we propose a requirements elicitation method and a requirements specification method for context-aware systems. Our requirements elicitation method is a 6-stepped, incremental, and iterative process. At the beginning steps in the process, we identify the requirements for business logic. Afterwards, we gather the requirements for context logic, model contexts, and identify subsystems. For requirements specification, we suggest a context-aware use case diagram, a context diagram for context modeling, and a context-type-use-case-dependency diagram for the traceability of use cases on the change of context types. We also introduce a case study that we apply our approaches to a real system, and a qualitative evaluation of our approaches. Our study will help stakeholders to efficiently elicit requirements for context-aware systems and to specify them clearly.

Key words : Context-aware System, Requirements Elicitation, Requirements Specification

1. 서 론

유비쿼터스 컴퓨팅 환경에서 컨텍스트 인지 시스템(context-aware system)은 매우 중요한 요소이기 때문에 이를 효과적으로 개발하기 위한 소프트웨어 공학 관련 연구들이 활발하게 진행되고 있다. 예를 들어, 컨텍스트 모델링[1-4], 아키텍처 패턴[5], 디자인 패턴[6,7], 각종 라이브러리[8-10], 구현 시스템[11-13] 등의 연구들이 수행되고 있다. 그러나 소프트웨어 공학에서 요구공학이 상당히 중요한 위치를 차지하고 있고, 컨텍스트 인지 시스템에서 컨텍스트를 효과적으로 파악하기 위한

[†] 장 희 원 : 목포대학교 컴퓨터공학 교수

jmchoi@mokpo.ac.kr

논문접수 : 2007년 8월 21일

심사완료 : 2008년 5월 22일

Copyright©2008 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저술물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지 : 시스템 및 이론 제35권 제8호(2008.8)

방법이 필요함에도 불구하고 이와 관련된 연구는 상대적으로 적었다. 기존 요구공학은 컨텍스트 인지 시스템에 적용할 때 다음과 같은 문제점을 가지고 있다. 첫째로 기존 요구사항 도출 방법은 컨텍스트, 컨텍스트를 구성하는 요소, 요구사항과 컨텍스트의 관계, 컨텍스트가 시스템에 미치는 영향 등을 파악하기 위한 절차나 방법을 제시하지 못한다. 둘째 기존 요구사항 도출 방법은 컨텍스트 인지 시스템의 다양한 사용 시나리오를 요구사항에 반영하기 위한 방법과 절차를 지원하지 못한다. 셋째 기존 UML은 컨텍스트 개념, 컨텍스트에 따라 시스템의 기능이 달라지는 것을 표현할 수 없다. 따라서 시스템 이해관계자들(stakeholders)이 쉽게 시스템을 이해할 수 있기 위해서는 컨텍스트 개념을 포함한 다이어그램들이 필요하다.

기존 요구공학의 문제점을 해결하기 위해서 본 논문에서는 컨텍스트 인지 시스템을 위한 요구사항 도출(elicitation) 방법과 명세화(specification) 방법을 제안한다. 본 논문에서 제안하는 요구사항 도출 방법은 6단계로 구성된 반복적인 프로세스이고, 점진적으로 요구사항을 파악하는 방법이다. 첫 번째 단계에서는 시스템에 대한 기본적인 개념을 정립하고, 두 번째 단계에서는 비즈니스 유스케이스와 비기능적 요구사항을 파악한다. 세 번째 단계에서는 컨텍스트 인지 서비스의 사용 시나리오를 파악하고, 사용자의 단말 플랫폼을 결정한다. 또한 비즈니스 유스케이스들 중에서 컨텍스트 인지 서비스 기능을 추가할 수 있는 것들을 파악한다. 네 번째 단계는 컨텍스트를 모델링하고, 컨텍스트 속성과 컨텍스트 추론 규칙을 파악하고 결정한다. 다섯 번째 단계에서는 컨텍스트 타입별로 의존관계에 있는 유스케이스들을 파악하고, 문서화한다. 여섯 번째 단계에서는 서브시스템들을 파악한다.

컨텍스트 인지 시스템의 요구사항은 이해관계자가 쉽고, 정확하게 이해할 수 있도록 문서화되어야 한다. 본 논문에서는 컨텍스트의 특성을 효과적으로 기술할 수 있도록 UML을 확장해서 컨텍스트 인지 유스케이스 다이어그램, 컨텍스트 다이어그램, 컨텍스트타입-유스케이스 의존 다이어그램을 제안한다. 컨텍스트 인지 유스케이스 다이어그램은 유스케이스와 컨텍스트의 관계를 보여주기 때문에 컨텍스트에 따른 기능적인 요구사항을 효과적으로 파악할 수 있다. 컨텍스트 다이어그램은 클래스 다이어그램과 상태를 결합한 형태의 다이어그램으로서 컨텍스트를 보다 명확히 이해할 수 있도록 컨텍스트, 컨텍스트 타입, 컨텍스트의 변화, 컨텍스트 속성 등에 대한 정보를 기술할 수 있다. 컨텍스트타입-유스케이스 의존 다이어그램은 컨텍스트에 영향을 받는 유스케이스들을 컨텍스트별로 표현함으로써 컨텍스트 변경

에 따른 유스케이스의 변경을 추적할 수 있게 한다.

본 논문은 8개 장으로 구성되어 있다. 2장에서는 본 연구와 관련이 있는 기존 연구들을 소개하고, 본 연구의 차이점을 기술한다. 3장에서는 컨텍스트를 측정 가능한 형태로 새로 정의하고, 컨텍스트 인지 서비스를 5개 형태로 분류한다. 4장에서는 컨텍스트 인지 시스템의 요구사항 도출 방법을 제안하고, 5장에서는 도출된 요구사항을 표현하기 위해 UML을 확장한 다이어그램들을 소개한다. 6장에서는 논문에서 제안한 방법에 따라 요구사항을 도출하고, 이를 명세화하는 사례연구를 소개한다. 7장에서는 본 논문에서 제안한 방법에 대한 정성적인 평가를 기술하고, 마지막으로 8장에서는 결론을 밝힌다.

2. 관련연구

본 연구와 관련된 기존 연구들은 크게 두 가지 형태로 분류할 수 있다. 첫 번째는 컨텍스트 인지 시스템을 위한 요구공학에 관한 연구들이고, 두 번째는 컨텍스트 인지 시스템의 명세화 방법에 관한 연구들이다.

2.1 컨텍스트 인지 시스템 요구공학 관련 연구

컨텍스트 인지 시스템의 요구사항 도출에 관련된 연구들은 본 논문의 선행연구[14,15], Dan Hong[16], Naoyasu[17], Hannes[18], Anthony[19] 등의 연구가 있다. 선행연구[14,15]에서는 컨텍스트 인지 시스템에서 컨텍스트의 중요성 및 요구사항을 파악하기 위한 방법을 소개한다. 그러나 선행연구에서는 요구사항 도출 절차와 명세화 방법의 구체성이 떨어진다.

Dan Hong[16]은 컨텍스트를 3개의 범주(컴퓨팅, 사용자, 물리적 컨텍스트)로 분류하고, 요구사항 추출을 위한 메타 모델을 제시한다. 또한 컨텍스트 인지 시스템의 요구사항을 추출하기 위한 가이드라인으로 요구사항 추출 절차를 제시한다. Dan Hong의 연구는 메타 모델과 절차를 제시한다는 측면에서 의의가 있지만, 구체적인 내용이 부족해서 실질적으로 적용하기 어렵다는 문제점을 가지고 있다.

Naoyasu[17]는 컨텍스트 인지 시스템에 대해서 시스템과 컨텍스트를 분리하고, 컨텍스트를 Aspect로 표현해서 VDM으로 정형화하는 방법을 제시한다. 이처럼 시스템과 컨텍스트를 분리하고, 시스템을 먼저 찾고, 나중에 컨텍스트와 결합하는 방법은 본 논문과 유사하다. 본 논문과 차이점은 Naoyasu의 연구는 VDM을 통한 정형화에 관점을 맞춘 것에 비해 본 논문은 요구사항 파악이라는 소프트웨어 공학에 관점을 맞춘 것이 차이점이다.

Hannes[18]은 컨텍스트 인지 시스템에서 상황을 파악하는 것이 매우 중요하기 때문에 시스템에서 매우 중요한 상황을 체계적으로 찾을 수 있기 위한 방법으로 컨텍스트 주도 유스케이스 작성 절차를 소개한다. Hannes

는 컨텍스트 정보와 부목표(sub-goal)를 연결하는 유스 케이스 컨텍스트 매트릭스를 사용해서 컨텍스트를 찾는다. 본 논문과 차이점으로는 Hannes의 연구는 컨텍스트를 찾는 방법과 절차에 대해서는 기술하지 않는다는 점이다.

Anthony[19]는 모바일 장비를 사용하는 컨텍스트 인지 시스템 환경에서 반영 기반(reflection-based) 요구공학 프레임워크를 제시하는데, 이 연구의 주된 내용은 실행 시에 컨텍스트가 변경되면, 요구사항도 역시 변경되기 때문에 반영을 기반으로 반복적인 요구사항 파악 프레임워크를 사용해야 한다는 점이다. Anthony의 연구는 컨텍스트와 목적(goal)을 기반으로 반영을 지원할 수 있는 요구공학 프레임워크를 제안한다는 점에서 의의가 있다. 그러나 요구사항 추출에 관련된 구체적인 방법이나 절차를 제시하지 않고 있다.

2.2 컨텍스트 인지 시스템 명세화 관련 연구

컨텍스트 인지 시스템의 명세화에 관한 연구들은 요구사항을 명세화하기 위한 연구들은 거의 없고, 컨텍스트 모델링을 그래픽 노테이션을 이용해서 표현한 연구들이 있다. 이러한 연구의 대표적인 것은 UML을 확장한 Sheng[20], Object-Role-Modeling(ORM)을 확장한 Karen Henriksen[21]의 연구가 있다. 이 연구는 컨텍스트 구성 요소의 관계, 의존성 등을 그래픽으로 기술할 수 있다. 컨텍스트 모델링을 표현하기 위한 방법은 본 연구의 컨텍스트 다이어그램과 관계가 있지만, 시스템의 요구사항을 거의 표현할 수 없다는 점에서 본 연구와 차이점을 가지고 있다.

3. 컨텍스트와 컨텍스트 인지 서비스의 정의 및 분류

3.1 컨텍스트 정의

컨텍스트 인지 시스템은 전통적인 시스템의 비즈니스 기능에 컨텍스트 관련 기능을 추가한 것이다[14,15]. 예를 들어, 전통적인 미디어 플레이어는 멀티미디어를 재생시키는 기능만을 갖지만, 컨텍스트 인지 미디어 플레이어[22]는 미디어 재생 기능은 물론 주위 환경을 인식하고, 주변에 있는 데스크톱 컴퓨터에 저장된 음악을 재생할 수 있다. 이처럼 컨텍스트 관련 기능은 비즈니스 기능을 바탕으로 주위 환경을 인식함으로써 비즈니스 기능을 강화해주는 역할을 한다. 따라서 컨텍스트 인지 시스템은 그림 1과 같이 비즈니스 로직을 바탕으로 하고, 컨텍스트 로직은 사용자 인터페이스처럼 비즈니스 로직과 외부를 연결하는 구조를 갖는다. 그림 1에서 user 액터는 사용자를 의미하고, env. 액터는 시스템이 인식하는 외부 환경을 의미한다. 컨텍스트 로직과 비즈니스 로직은 필요한 정보를 관리하기 위해서 각각 컨텍

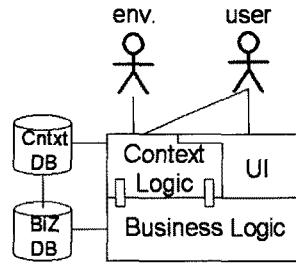


그림 1 컨텍스트 인지 시스템 레이어

스트 데이터베이스(Cntxt DB)와 비즈니스 데이터베이스(Biz DB)를 가질 수 있다.

컨텍스트 인지 시스템은 전통적인 시스템과 달리 컨텍스트 로직의 중요성이 강조되기 때문에 요구사항을 효과적으로 파악하기 위해서는 컨텍스트에 대한 정의와 특징을 파악할 필요가 있다. 현재까지 컨텍스트에 대한 정의는 매우 다양하다. 예를 들어, Schilit[23]은 컨텍스트를 위치, 사람 혹은 사물의 식별자, 사물에 대한 변화로 보았고, Brown[24]은 컨텍스트를 사용자의 컴퓨터가 인식하는 사용자의 환경 정보(위치, 사용자, 시간, 온도 등)라고 정의하고 있으며, Dey[25]는 컨텍스트를 엔티티의 상황을 특징짓기 위해 사용되는 정보라고 정의한다.

기존 컨텍스트에 대한 정의는 나름대로 장점을 가지고 있지만, 소프트웨어 공학에서 필요로 하는 추상성과 측정 가능성이 떨어진다. 따라서 본 연구에서는 컨텍스트를 추상적이면서도 측정 가능한 형태로 정의하고, 이를 바탕으로 시스템 요구사항을 분석한다. 컨텍스트에 대한 정의는 다양하지만, 기존 연구들의 공통된 결과는 컨텍스트가 정보로 구성된다는 것이다[13,23-25]. 이러한 사실에 기초하여 본 연구에서는 컨텍스트를 구성하는 가장 기본적인 정보를 컨텍스트 속성(context attribute)이라 부르고, 정의 1과 같이 정의한다.

정의 1. 컨텍스트 속성

컨텍스트 속성 a_i 는 컨텍스트 인지 시스템의 서비스에 영향을 주는 단일 개념의 상황정보를 의미한다. 즉, 컨텍스트 속성의 값은 센서로부터 측정되거나 시스템 내부에 존재하는 값이다. □

컨텍스트 속성은 센서로부터 측정되는 외부 정보와 그렇지 않은 내부 정보로 분류할 수 있다. 예를 들어, 온도, 사용자 식별번호, 시간 등은 대표적인 컨텍스트 속성이고, 온도와 사용자 식별번호는 각각 온도센서와 RFID 리더 등의 서브시스템을 통해서 인식할 수 있는 외부 정보이다. 반면에 시간은 별도의 서브시스템 없이 시스템 내부에서 파악할 수 있는 내부 정보이다. 그러나 가상 센서[26]의 개념을 도입하면, 내부 정보도 가상 센서라는 가상 서브시스템을 통해서 인식할 수 있다. 따라

서 컨텍스트 속성을 외부 혹은 내부에 관계없이 일관된 방법으로 관리할 수 있고, 컨텍스트 속성과 센서 시스템은 1:1 관계를 갖는다.

컨텍스트 속성 값들은 컨텍스트를 결정한다. 또한 시스템 관점에서 볼 때 컨텍스트는 시스템의 서비스에 영향을 줄 수 있는 어떤 상황을 의미하는 추상적인 개념이다. 즉, 동일한 컨텍스트라도 이를 측정하는 센서의 종류에 따라 시스템에서 사용하는 컨텍스트 속성은 달라진다 [4,22]. 예를 들어, “사람들이 대화하는 상황”을 인식하는 시스템을 가정해보면, A라는 시스템은 대화 상태를 파악하기 위해서 음성을 인식하는 센서를 사용하고, B라는 시스템은 카메라를 이용할 수 있다. 이처럼 A 시스템과 B 시스템은 모두 동일한 컨텍스트를 사용하지만, 사용하는 센서가 다르기 때문에 컨텍스트 속성들도 달라진다. 즉, 컨텍스트는 추상적인 성격이 강하고, 이를 컨텍스트 속성이라는 구체적인 값들로 구현될 수 있다. 따라서 컨텍스트는 정의 2와 같이 정의할 수 있다.

정의 2. 컨텍스트

컨텍스트 c 는 컨텍스트 인지 서비스를 결정하고, 호출할 수 있는 추상화된 정보이다. 컨텍스트는 컨텍스트 속성 값들의 유한한 튜플, $c = \langle a_0, a_1, \dots \rangle$ 로 구현된다.

대부분 컨텍스트 인지 시스템은 라이프사이클 동안에 여러 개의 컨텍스트를 갖고, 시스템은 현재 컨텍스트가 무엇인지를 보고 서비스를 결정한다. 컨텍스트 인지 시스템에서 라이프사이클동안 가질 수 있는 컨텍스트들은 시스템 개발에서 중요한 의미가 있기 때문에 이 집합을 컨텍스트 집합이라고 부르고, 정의 3과 같이 정의한다.

정의 3. 컨텍스트 집합

컨텍스트 집합은 컨텍스트 인지 시스템에서 사용하는 모든 컨텍스트들의 집합이다.

시스템의 컨텍스트 집합 중에서 어떤 컨텍스트들은 서로 유사한 개념을 가지고 있다. 예를 들어, “달리다”, “걷다” 등은 사용자의 행동과 관련이 있는 유사한 개념이고, 시스템 관점에서 사용자 상태를 의미한다. 이처럼 컨텍스트의 공통된 개념을 추상화한 것을 컨텍스트 타입이라고 하고, 정의 4와 같이 정의한다.

정의 4. 컨텍스트 타입

컨텍스트 타입, t 는 컨텍스트의 공통된 개념을 추상화한 개념으로 컨텍스트의 데이터 타입으로 볼 수 있다.

컨텍스트 인지 시스템은 기존의 데스크톱에서 실행되는 시스템과는 달리 여러 개의 서브시스템으로 구성된 분산 시스템이다. 따라서 시스템의 기능과 모듈에 따라서 실행되는 플랫폼이 다양하다. 또한 일부 시스템은 동일한 기능을 여러 개의 플랫폼에서 지원하는 경우도 있

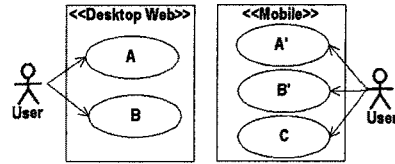


그림 2 시스템 컨텍스트와 유스케이스

다. 예를 들어 사용자가 데스크톱에서 응용프로그램으로 사용할 수 있는 기능을 웹이나 모바일 플랫폼에서 사용할 수도 있다. 이처럼 동일한 기능이 다른 플랫폼에서 실행되는 경우에 플랫폼 역시 컨텍스트로 볼 수 있다. 플랫폼처럼 시스템 전체에 영향을 미치는 컨텍스트를 본 연구에서는 시스템 컨텍스트라고 한다. 그림 2는 개념적으로 동일한 유스케이스가 플랫폼이 달라짐에 따라 변형되거나 새로운 유스케이스가 추가될 수 있다는 것을 보여준다. 그림 2에서 시스템 범위를 나타내는 사각형이 플랫폼이다. <<Desktop Web>> 플랫폼에서 사용자는 A와 B라는 유스케이스를 사용하고, <<Mobile>> 플랫폼에서 사용자는 A', B', C라는 유스케이스를 사용할 수 있다.

3.2 컨텍스트 인지 서비스 분류

컨텍스트 인지 시스템의 가장 큰 특징은 컨텍스트에 따라 자동적으로 실행되거나 서비스 내용이 달라질 수 있는 컨텍스트 인지 서비스이다. 따라서 컨텍스트 인지 시스템에서 요구사항을 파악하기 위해서 컨텍스트 인지 서비스의 유형과 특징을 알아볼 필요가 있다. 특히, 컨텍스트 인지 서비스는 시스템의 사용 시나리오 혹은 사용 패턴을 결정하기 때문에 요구사항 단계에서 파악해야 할 중요한 특성이다.

컨텍스트의 사용 방법 혹은 유형에 따라 컨텍스트 인지 서비스를 5가지 형태로 분류할 수 있다. 다음에 소개하는 컨텍스트 서비스 유형은 배타적이지 않고, 서로 결합될 수 있다. 예를 들어, A라는 컨텍스트 인지 서비스는 컨텍스트가 변경됨에 따라 자동적으로 실행되면서, 컨텍스트 정보의 값에 따라 여러 형태의 서비스를 제공할 수도 있다.

- 컨텍스트 리코딩(context recording) : 현재 컨텍스트에 대한 정보를 저장한다. 예를 들어, Nicky[27]는 일상생활을 비디오/오디오로 기록하면서 동시에 몸에 부착된 센서를 이용해서 컨텍스트를 추출하는 것에 대해서 연구하였다.
- 트리거(trigger) : 컨텍스트가 지정된 조건을 만족시키는 경우에 서비스가 자동적으로 실행된다. 예를 들어, 사용자가 집에 들어오면, 자동적으로 현관의 불이 켜지는 것은 간단한 형태의 트리거 서비스이다.
- 브랜치(branch) : 현재 컨텍스트에 따라 서비스 내용

과 형태가 결정된다. 예를 들어, Bravo[28]가 제시한 시스템은 강의실에서 사용자가 칠판에 접근하는 경우에 사용자의 역할(강사 혹은 학생)에 따라 프레젠테이션에 보이는 내용이 달라진다. 즉, 강사인 경우에는 강의 내용을 보여주고, 학생인 경우에는 학생의 숙제를 화면에 보여준다.

- 자원검색(resource scanning) : 사용가능한 자원들을 찾고, 이를 활용할 수 있는 서비스가 제공된다. 예를 들어 냉장고가 음식 재료들을 인식해서 사용자에게 적합한 음식의 요리법을 소개해주는 것도 있다[11].
- 팔로우미(follow-me) : 팔로우미 서비스[29]는 사용자의 이동에 따라 서비스가 이동할 수 있다. 따라서 팔로우미 서비스는 이동성과 자원 검색이라는 2가지 기능을 동시에 지원하는 서비스이다.

컨텍스트 인지 서비스의 유형을 분류하는 것은 시스템 요구사항 파악에서 다음과 같은 3가지 역할을 한다. 첫째로 서비스의 5개 유형은 컨텍스트 인지 서비스의 패턴 역할을 한다. 따라서 서비스 유형의 이름은 이해관계자 사이의 의사소통을 도와주고, 공통된 개념을 갖는데 도움을 준다. 둘째로 5개의 서비스 유형은 비즈니스 유스케이스가 컨텍스트 인지 서비스로 전환될 수 있는지 여부를 파악할 수 있는 기준을 제공한다. 셋째로 서비스 유형은 각 서비스별로 일반적인 요구사항을 파악할 수 있도록 해준다. 표 1은 각 서비스 유형별로 필요로 하는 일반적인 요구사항을 보여준다. F는 기능적 요구사항을 의미하고, N은 비기능적 요구사항을 의미한다.

4. 컨텍스트 인지 시스템을 위한 요구사항 도출

4.1 요구사항 도출 프로세스 개요

컨텍스트 인지 시스템의 요구사항 도출은 기존 시스

템에 비해 어렵다. 이것은 그림 1에서 볼 수 있듯이 컨텍스트 인지 시스템에서는 비즈니스 로직과 컨텍스트 로직이 복잡하게 얽혀 있고, 컨텍스트를 정확히 파악하기 어렵기[18] 때문이다. 시스템의 복잡성을 줄이기 위해서 본 연구에서는 그림 1의 시스템 레이어에 따라 비즈니스 로직과 컨텍스트 로직을 분리하는 접근방법을 사용한다. 이러한 분리는 시스템 복잡성을 줄일 수 있고, 요구사항 도출에 다양한 접근방법이 가능하게 한다. 즉, 비즈니스 로직의 요구사항은 유스케이스를 통한 기존의 요구공학 방법을 적용하고, 컨텍스트 로직의 요구사항은 컨텍스트 서비스와 컨텍스트를 중심으로 파악하는 방법을 적용한다.

컨텍스트 인지 시스템의 요구사항 도출은 유스케이스, 컨텍스트, 컨텍스트 속성, 서브시스템 등을 단계적으로 파악하는 단계적이고 반복적인 절차를 따른다. 또한 중요한 요구사항부터 파악하는 점진적 절차를 따른다. 그림 3은 컨텍스트 인지 시스템의 요구사항을 도출하는 절차를 보여준다. 요구사항 도출을 위한 각 단계의 구체적인 내용은 4.2절에서 다룬다.

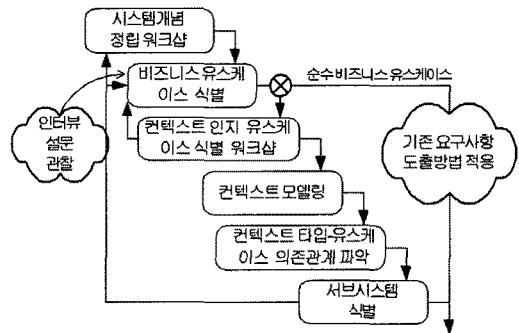


그림 3 컨텍스트 인지 시스템 요구사항 도출 절차

표 1 컨텍스트 인지 서비스 유형과 일반적인 요구사항

F: 기능적 요구사항, N: 비기능적 요구사항

서비스 유형	일반적인 요구사항
컨텍스트 리코딩	F: 컨텍스트를 효과적으로 추출, 저장, 검색할 수 있어야 한다.
트리거	F: 컨텍스트를 정확히 인식할 수 있어야 한다.
브랜치	N: 비즈니스 로직과 컨텍스트 로직의 결합도가 낮아야 한다.
자원검색	F: 주위 환경을 효과적으로 인식할 수 있어야 한다. F: 동적으로 서비스를 구성할 수 있어야 한다. N: 확장성이 높아야 한다. N: 보안성이 있어야 한다.
팔로우미	F: 주위 환경을 효과적으로 인식할 수 있어야 한다. F: 컨텍스트를 효과적으로 저장 및 관리할 수 있어야 한다. F: 동적으로 서비스를 구성할 수 있어야 한다. N: 확장성이 높아야 한다. N: 이동성을 지원해야 한다. N: 보안성이 있어야 한다.

4.2 요구사항 도출 프로세스

4.2.1 단계 1 : 시스템 개념 정립 워크샵

시스템 요구사항을 파악하기 위해서 첫 번째로 수행해야 하는 작업은 시스템의 개략적인 개념을 파악하는 것이다. 시스템에 대한 정의는 주요 이해관계자들(고객, 사용자, 분석가 등)이 참석한 워크샵을 통해서 결정한다. 이 단계에서 수행해야 하는 주요 작업은 다음과 같다.

1. 시스템 목표(goal) 파악 : 시스템의 가장 중요한 목표가 무엇인지 파악한다.
2. 시스템에 대한 개념 정립 : 시스템에 대한 개요 및 개념을 정의하고, 이해관계자들이 시스템에 대한 공통된 개념을 갖도록 한다.
3. 이해관계자 파악 : 시스템에 관련된 추가적인 이해관계자가 있는지 파악한다.

4.2.2 단계 2 : 비즈니스 유스케이스를 통한 요구사항 식별
컨텍스트 인지 서비스는 비즈니스 로직을 기반으로 하기 때문에 컨텍스트 인지 서비스에 대한 요구사항을 파악하기에 앞서 비즈니스 유스케이스 파악을 통한 기능적 요구사항을 도출해야 한다. 따라서 이 단계에서는 이해관계자와 인터뷰, 질의서, 관찰 등의 방법을 통해 기능적 요구사항과 비기능적 요구사항을 파악한다.

1. 비즈니스 유스케이스 파악 : 이해관계자에 대해서 인터뷰, 질의서, 관찰 등의 방법을 통해서 시스템의 비즈니스 유스케이스를 파악하고, 이를 문서화한다.
2. 비기능적 요구사항 파악 : 이해관계자에 대해서 인터뷰, 질의서, 관찰 등의 방법을 통해서 비기능적 요구사항, 제약사항, 컨텍스트와 컨텍스트 속성의 후보가 될 수 있는 정보들을 수집하고, 문서화한다.

4.2.3 단계 3 : 컨텍스트 인지 서비스 식별 브레인스토밍 워크샵

컨텍스트 인지 서비스는 새로운 형태의 서비스이고, 기존 데스크톱 시스템과는 달리 사용자가 일상생활에서 시스템을 사용할 수 있기 때문에 매우 다양한 형태의 사용 시나리오가 존재한다. 따라서 시스템을 어떤 형태로 사용할 것인지를 결정하는 것은 많은 창의성을 필요로 한다. 시스템의 사용 시나리오가 중요한 이유는 이것이 시스템 구조와 컨텍스트에 큰 영향을 끼칠 수 있기 때문이다. 따라서 브레인스토밍을 통해 컨텍스트 인지 시스템의 다양한 사용 시나리오를 파악하는 것이 매우 중요하다. 이 단계에 구체적으로 수행해야 하는 작업은 다음과 같다.

1. 시스템 사용 시나리오 파악 : 이해관계자들이 브레인스토밍 방법을 통해 “단계 2”에서 파악한 기능적/비기능적 요구사항을 만족시키면서 시스템을 효과적으로 사용하는 시나리오를 파악한다. 이때 기술적인 관점을 제외하고, 시스템을 사용하는 창의적인 방법에

초점을 맞춘다. 또한 관심을 가져야할 비기능적 요구사항은 사용자 관점의 요구사항이다. 사용자 관점의 비기능적 요구사항은 시스템 사용 시나리오에 영향을 주고, 이것은 다시 컨텍스트에 영향을 준다.

2. 사용자 플랫폼 결정 : “단계 2”에서 파악한 비기능적 요구사항과 시스템 사용 시나리오를 분석해서 사용자 플랫폼을 결정한다. 사용자 플랫폼은 사용자가 서비스를 받기 위해서 사용하는 서브시스템을 의미한다. 예를 들어, 모바일 단말기, 데스크톱, 웹 플랫폼 등은 대표적인 사용자 플랫폼이다. 사용자 플랫폼 역시 많은 창의성을 요구하기 때문에 브레인스토밍 방법을 통해서 다양한 아이디어를 수집하여야 하고, 플랫폼을 결정하기 위해서는 요구사항의 만족 여부 및 장단점을 점수화한 매트릭스를 이용한다.
3. 컨텍스트 후보 파악 : 시스템 사용 시나리오에서 사용될 수 있는 컨텍스트 후보 및 컨텍스트 속성 후보에 대한 정보들을 파악하고, 이를 문서화한다.
4. 비즈니스 유스케이스에서 컨텍스트 인지 서비스와 관련된 내용 파악 : 시스템 사용 시나리오를 바탕으로 비즈니스 유스케이스에서 컨텍스트의 영향을 받을 수 있는 것들을 파악하고, 이를 문서화한다. 이때 비즈니스 유스케이스가 5개의 컨텍스트 인지 서비스 유형에 해당되는지 여부를 파악해서 해당 유스케이스가 컨텍스트 인지 서비스인지를 판단한다. 또한 컨텍스트 인지 서비스 유형에 따라 각 유형의 일반적인 요구사항을 고려할 대상으로 문서화한다.
5. 컨텍스트 인지 유스케이스 문서화 : 비즈니스 유스케이스 다이어그램에서 컨텍스트 인지 기능을 갖는 유스케이스에 서비스 유형과 서비스에 영향을 미치는 컨텍스트 후보 및 컨텍스트 속성 후보를 기술한다.

4.2.4 단계 4 : 컨텍스트 식별 및 모델링

이 단계에서는 단계 2와 3에서 식별한 컨텍스트 및 컨텍스트 속성 후보들을 분류하고, 이를 바탕으로 컨텍스트를 모델링한다. 이 단계에서 수행하는 작업은 다음과 같다.

1. 모든 가능한 컨텍스트 관련 정보를 수집하고 목록화 : 단계 2와 3에서 파악한 컨텍스트 후보와 컨텍스트 속성 후보 정보를 수집한다.
2. 컨텍스트 관련 정보들을 유사성에 따라 그룹화 : 컨텍스트 관련 정보를 유사성에 따라 그룹화 혹은 계층적으로 분류한다.
3. 컨텍스트 모델 구성 : 분류된 컨텍스트 관련 정보를 바탕으로 가능한 형태의 컨텍스트 모델들을 작성하고, 각 컨텍스트에 적합한 컨텍스트 속성들을 기술한다. 가능한 컨텍스트 모델 중에서 시스템의 요구사항에 가장 적합한 모델을 선택해서, 시스템의 컨텍스트

모델로 결정한다.

4. 컨텍스트 속성에서 컨텍스트로 추론 규칙 구축 : 컨텍스트 속성으로부터 컨텍스트를 유추하기 위한 추론 규칙을 찾는다.
5. 컨텍스트 문서화 : 컨텍스트, 컨텍스트 속성, 추론 규칙 등을 문서화한다.

4.2.5 단계 5 : 컨텍스트 타입과 유스케이스의 의존관계 파악

컨텍스트 인지 서비스 A의 내용은 컨텍스트 X에 의해서 결정된다. 이를 다르게 표현하면, "서비스 A에 해당되는 유스케이스 u(A)은 컨텍스트 X의 컨텍스트 타입인 t(X)에 의존적이다." 따라서 컨텍스트와 서비스의 관계는 컨텍스트 타입과 유스케이스의 의존관계로 표현할 수 있다. 이러한 의존관계는 시스템 설계에서 필요할 뿐만 아니라, 요구사항에서 컨텍스트의 변화에 따른 유스케이스의 변화를 추적하기 위해서도 필요하다. 의존관계를 파악하기 위해서 다음과 같은 작업들을 수행한다.

1. 컨텍스트와 유스케이스의 관계 파악 : 컨텍스트에 따라 내용이 달라지는 유스케이스를 파악한다.
2. 컨텍스트 타입과 유스케이스 의존 다이어그램 작성 : 컨텍스트에 영향을 받는 유스케이스들을 컨텍스트 타입별로 분류하고, 컨텍스트 타입-유스케이스 의존 다이어그램을 작성한다.

4.2.6 단계 6 : 서브시스템 식별 및 시스템 아키텍처 파악

컨텍스트 속성과 센서 시스템은 1:1 매칭이 이루어지기 때문에 컨텍스트 속성을 분석하면, 필요한 센서 서브시스템들을 파악할 수 있다. 또한 이러한 서브시스템들을 이용해서 기초적인 시스템 아키텍처를 파악할 수 있다. 이 단계에서 수행해야 할 구체적인 작업 내용은 다음과 같다.

1. 서브시스템 식별 : 컨텍스트 속성을 인식할 수 있는 센서 디바이스 혹은 서브시스템을 식별한다. 컨텍스트 속성 하나에 한 개의 센서와 매치된다. 일부 컨텍

스트 속성은 외부 센서 시스템과 매칭되고, 다른 일부는 시스템 내부 정보이기 때문에 가상 센서로 매치된다.

2. 시스템 아키텍처 파악 : 파악된 서브시스템들을 이용해서 기초적인 시스템의 아키텍처를 파악한다.

4.2.7 반복

컨텍스트 인지 시스템의 요구사항 도출은 UP(Unified Process)에 적용될 수 있도록, 점진적이며, 반복적으로 진행된다. 즉, 요구사항의 중요도와 위험성에 따라 점진적으로 작업이 이루어지고, 궁극적으로는 설계, 구현, 테스트의 결과가 요구사항에 반영될 수 있다.

5. 컨텍스트 인지 시스템 요구사항 명세화

5.1 컨텍스트 인지 시스템을 위한 UML 확장

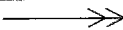



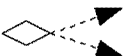
요구사항을 효과적으로 명세화할 수 있는 방법은 매우 중요하다. 만약, 4장의 요구사항 도출 절차를 통해 요구사항을 도출하였다도 이를 명확히 표현하지 못하면 이해관계자들 사이의 의사소통과 협력은 원활히 이루어질 수 없다. 현재 UML이 소프트웨어 시스템 개발에서 표준적으로 사용되기 때문에 본 연구에서는 UML을 확장해서 컨텍스트 인지 시스템의 요구사항을 기술하는 방법을 제안한다. 표 2는 새로운 다이어그램에서 사용할 수 있는 심벌들이다.

5.2 컨텍스트 인지 유스케이스 다이어그램

유스케이스 다이어그램은 시스템의 기능적인 요구사항을 파악하기 위해서 널리 사용하는 UML 다이어그램이다. 컨텍스트 인지 시스템에서는 기능적인 요구사항과 컨텍스트가 밀접한 관계를 갖고 있기 때문에 유스케이스 다이어그램에 컨텍스트 개념을 추가할 필요가 있다.

컨텍스트 인지 유스케이스 다이어그램은 기존 유스케이스 다이어그램에 컨텍스트, 컨텍스트 속성, 컨텍스트 인지 서비스 유형 등의 개념을 추가한 것이다. 유스케이스 외부의 사각형은 시스템 레벨 컨텍스트를 의미하고, 컨텍스트 속성을 인식할 수 있는 외부 센서와 컨텍스트

표 2 컨텍스트 인지 시스템을 위해 추가한 심벌

심벌	설명	다이어그램
	컨텍스트 정보의 연관관계	유스케이스 다이어그램
	컨텍스트 전이	컨텍스트 다이어그램
	인스턴스 관계	컨텍스트 다이어그램
	컨텍스트 속성(센서 시스템)	유스케이스 다이어그램
	컨텍스트에 따른 서비스 변경(브랜치 서비스)	유스케이스 다이어그램

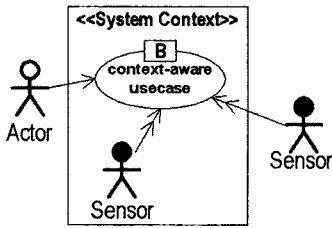


그림 4 컨텍스트 인지 유스케이스 다이어그램

속성으로 사용되는 내부 상태 정보는 검은색 액터 모양으로 표현한다. 컨텍스트 인지 서비스에 해당되는 유스케이스에는 컨텍스트 인지 서비스 유형을 표시함으로써 이해관계자가 쉽게 이해할 수 있도록 한다.

그림 4는 컨텍스트 인지 유스케이스 다이어그램의 예이다. 유스케이스에 표시된 마크 "B"는 컨텍스트 인지 서비스 유형이 브랜치(Branch) 타입임을 의미한다. 그림에서 유스케이스의 바운더리 외부에 있는 Sensor는 센서 서비스시스템을 의미하고, 내부에 있는 Sensor는 가상센서를 의미한다.

5.3 컨텍스트 다이어그램

컨텍스트와 컨텍스트 속성은 시스템을 이해하는데 매우 중요하다. 따라서 컨텍스트를 표현할 수 있는 UML 다이어그램이 필요하고, 이것은 클래스 다이어그램을 변형해서 표현할 수 있다. 컨텍스트 다이어그램은 컨텍스트, 컨텍스트 속성, 컨텍스트 타입, 컨텍스트 전환 등의 개념을 표현할 수 있다.

그림 5는 UserState라는 컨텍스트 타입, 컨텍스트, 컨텍스트 속성의 관계를 보여주는 컨텍스트 다이어그램이다. 그림에서 시스템은 Normal과 Busy라는 컨텍스트를 갖고, 조건에 따라 컨텍스트 변경이 발생한다. Normal 컨텍스트는 Sensor A로 구현될 수도 있고, Sensor B로 구현될 수도 있으며, Sensor A로 구현되는 경우에 컨텍스트 속성은 C와 D를 사용한다.

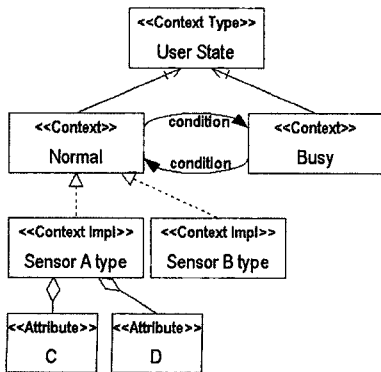


그림 5 컨텍스트 다이어그램

5.4 컨텍스트타입-유스케이스 의존 다이어그램

컨텍스트타입-유스케이스 의존 다이어그램은 컨텍스트 타입에 영향을 받는 유스케이스들을 의존관계 그래프 형태로 기술한다. 이는 컨텍스트에 변화가 있는 경우에 영향을 받는 유스케이스를 추적하는 경우에 매우 효과적이다. 그림 6은 컨텍스트타입-유스케이스 의존 다이어그램의 예를 보여준다. 그림에서 유스케이스 1, 2, 3은 컨텍스트 타입 A에 영향을 받고, 유스케이스 1과 2는 컨텍스트 타입 B의 영향을 받는다.

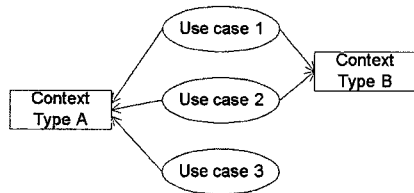


그림 6 컨텍스트타입-유스케이스 의존 다이어그램

6. 사례연구 : 컨텍스트 인지 전시물 안내 시스템

이 장에서는 컨텍스트 인지 시스템의 요구사항 도출을 위한 사례연구로 컨텍스트를 활용한 전시물 안내 시스템[30,31]의 요구사항을 파악해보도록 한다. 전시물 안내 시스템은 기본적으로 다음과 같은 시스템이다.

시스템 개요

전시물 안내 시스템은 사용자에게 전시물의 정보를 효과적으로 제공할 수 있어야 한다.

본 논문에서 제안한 방법을 따르는 경우에 첫 번째로 해야 할 일은 시스템의 개념을 파악하고, 이해관계자를 파악하는 것이다. 이해관계자는 워크샵을 통해서 시스템의 목표가 "전시회에서 전시물에 대한 정보를 사용자에게 제공"하는 것이라는 것을 파악할 수 있다. 이해관계자는 관람객(사용자), 고객(전시회 관계자 혹은 박물관), 시스템 관리자, 시스템 분석가 등으로 파악할 수 있다.

두 번째 단계는 시스템의 비즈니스 유스케이스를 식별하는 것이다. 시스템 분석가는 이해관계자의 인터뷰를 통해서 유스케이스를 파악한다. 이 시스템에서 파악할 수 있는 유스케이스로는 "(F1)전시물 정보 제공", "(F2) 사용자 등록", "(F3)방문한 전시물 기록 보기" 등이 있다. 파악한 기능적 요구사항은 그림 7과 같이 유스케이스 다이어그램으로 작성할 수 있다.

이 단계에서 수행해야 할 또 다른 작업은 비기능적 요구사항, 설계 제약사항 등을 파악하는 것이다. 이해관계자와 인터뷰를 통해서 시스템 구축에서 "(N1)저비용", "(N2)관람자의 언어와 지식수준을 고려한 정보제공", 고급적 관람객이 이해하기 쉬운 "(N3)멀티미디어 정보 제

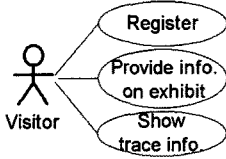


그림 7 비즈니스 유스케이스 예

공"이라는 비기능적 요구사항을 파악할 수 있다.

세 번째 단계에서는 워크샵을 통해서 지금까지 파악한 요구사항을 만족시킬 수 있는 시스템 사용 시나리오를 파악한다. 요구사항 F1과 N2를 만족시키기 위해서는 개인화된 안내 정보를 제공할 수 있어야 하고, 또한 N3를 만족시키기 위해서는 멀티미디어 디스플레이를 이용할 수 있어야 한다. 비기능적 요구사항 N2와 N3는 시스템 사용 시나리오와 컨텍스트에 영향을 준다. 이에 따라 다양한 시나리오를 검토하고, 평가해서 다음과 같은 시스템 사용 시나리오를 결정한다.

시스템 사용 시나리오

관객은 휴대용 단말기를 소지하고 있으며, 전시물로 접근하는 경우에 전시물에 대한 안내 정보가 단말기로 멀티미디어 형태로 제공된다. 이때 안내 정보는 관객의 언어와 지식수준을 고려해서 결정한다.

이 단계에서는 또한 시스템의 컨텍스트/컨텍스트 속성 후보에 대한 정보를 파악해야 한다. 이 시스템에서 파악할 수 있는 컨텍스트 속성 후보로는 사용자 위치(혹은 전시물ID), 관람객의 언어, 관람객의 지식수준 등이다. 비즈니스 유스케이스에서 컨텍스트에 영향을 받는 것은 F1이고, F1의 서비스 유형은 브랜치 및 트리거 타입이다. 그림 8은 이 단계에서 파악한 컨텍스트 인지 서비스를 바탕으로 작성한 컨텍스트 인지 유스케이스 다이어그램이다.

네 번째 단계는 컨텍스트 모델링 단계이다. 그림 9는 전시물 안내 시스템을 위한 컨텍스트 모델을 보여준다.

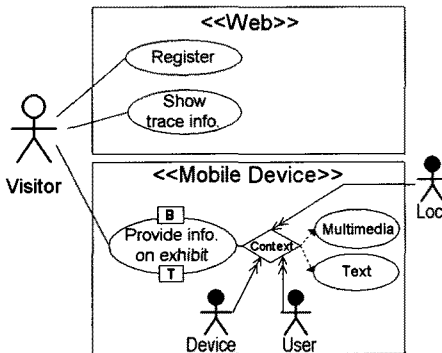


그림 8 컨텍스트 인지 유스케이스 다이어그램 예

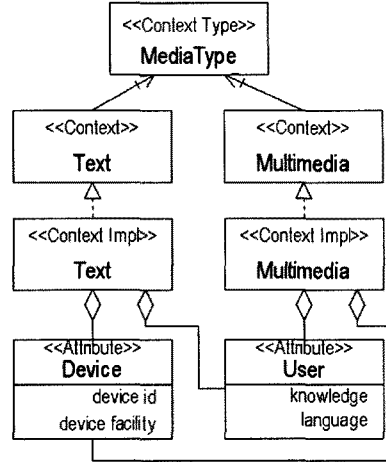


그림 9 컨텍스트 다이어그램 예

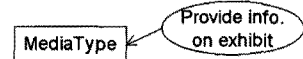


그림 10 컨텍스트타입-유스케이스 의존 다이어그램 예

이 시스템은 Text와 Multimedia라는 2개의 컨텍스트를 갖고, 컨텍스트 속성으로 단말기(Device)에 대한 정보, 사용자(User) 정보를 갖는다. 2개의 컨텍스트는 MediaType이라는 컨텍스트 타입의 인스턴스이다.

다섯 번째는 컨텍스트타입-유스케이스 의존관계를 파악하는 단계이다. 전시물 안내 시스템의 경우에 컨텍스트타입-유스케이스 의존관계는 그림 10과 같이 파악할 수 있다. 즉, MediaType에 따라 전시물 정보를 제공하는 유스케이스가 영향을 받는다.

여섯 번째는 서브시스템을 식별하는 단계이다. 컨텍스트 속성과 사용 시나리오를 이용해서 그림 11과 같은 기초적인 시스템 아키텍처를 파악할 수 있다. 전시물 안내 시스템의 경우에 서비스를 제공하는 서버, 사용자를 위한 모바일 단말기, 사용자 위치를 파악하기 위한 위치 센서, 사용자 정보로 구성된다. 사용자 정보는 내부의

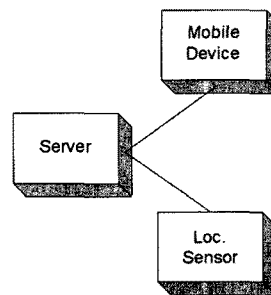


그림 11 서브시스템 다이어그램 예

가상 센서를 통해서 파악하기 때문에 서브시스템 다이어그램에는 기술하지 않는다. 이 단계에서 파악한 시스템 아키텍처는 비기능적 요구사항의 특성들을 파악해 향후 점진적으로 발전시켜야 한다.

전시물 안내 시스템에 대한 요구사항 도출은 반복적으로 수행되면서 요구사항이 점차 구체적이며, 완전한 형태로 발전한다. 또한 컨텍스트에 무관한 요구사항들에 대해서 기존 요구사항 도출 방법을 결합해서 적용한다. 즉, 기존의 유스케이스 도출 방법, 비기능적인 요구사항 도출 방법들을 활용한다.

7. 평가

본 연구에서 제시한 컨텍스트 인지 시스템을 위한 요구사항 도출 방법과 요구사항 명세 방법은 6장에서 소개한 사례연구[30,31]에 직접 적용해 봄으로써 평가를 수행할 수 있었다. 그러나 이 요구사항 도출 방법과 명세화 방법을 아직 많은 시스템에 적용하지는 못하였기 때문에 개발 기간 혹은 비용 등에 대한 정량적인 평가는 아직 얻을 수 없었고, 다른 관련연구들과 비교해볼 수 있는 정성적인 평가들을 얻을 수 있었다.

본 연구 방법은 요구공학에서 필요로 하는 요구사항 파악 절차, 파악 방법, 명세화 방법을 지원하기 때문에 컨텍스트 인지 시스템의 요구사항 도출에서 발생하는 시행착오를 줄일 수 있었다. 현재까지 컨텍스트 인지 시스템을 위한 요구공학 연구들은 요구사항 파악 절차는 지원하지만, 구체적으로 요구사항을 파악하는 방법이나 요구사항을 명세화하는 방법은 지원하지 않는다. 표 3은 본 연구와 관련연구들을 비교한 것이다.

컨텍스트는 소프트웨어 아키텍처[5], 시스템 디자인[6,7] 등 시스템 전체에 미치는 영향이 크기 때문에 효과적으로 파악하는 것이 매우 중요하다. 그러나 컨텍스트 인지 시스템에서 컨텍스트를 찾는 것은 매우 어려운 작업[14,18]으로 알려져 있다. 본 연구 방법을 적용하는 경우에 요구사항 도출 단계에서 컨텍스트와 시스템 개발에 영향을 주는 정보 수집을 체계적으로 진행할 수 있다. 표 4는 컨텍스트와 기타 정보 수집을 지원하는 것을 비교한 내용이다.

표 4에서 제시한 5개 항목은 컨텍스트 인지 시스템의 아키텍처와 설계에 영향을 미치는 주요 5개 산출물에 대해서 파악한 것이다. 그림 12는 5개 주요 항목의 산출물이 생성되는 과정과 다음 단계로 데이터가 전달되는 과정을 보여준다. 그림에서 음영으로 표시한 것은 본 논문에서 주로 다룬 절차 단계이고, 번호는 표 4에서 기술한 것과 일치하는 산출물을 의미한다. 컨텍스트 속성, 시스템 서비스 시나리오는 시스템 아키텍처에 영향을 미치고, 컨텍스트 및 컨텍스트 인지 서비스는 시스템 설계에 영향을 미치는 것을 알 수 있다.

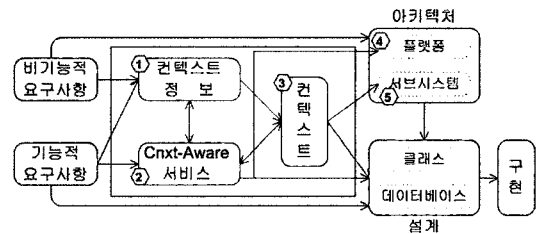


그림 12 컨텍스트 인지 시스템 요구사항 도출 단계에서 데이터 흐름

표 3 요구사항 도출 연구들의 지원성 비교

기술요소 관련연구	요구사항 파악 절차	요구사항 파악 방법	명세화 방법
본 논문의 방법	○	○	○
Dan Hong [16]	○	×	×
Naoyasu [17]	○	△	△
Hannes [18]	○	△	×
Anthony [19]	△	×	×

○: 지원, △: 일부 지원, ×: 지원안함

표 4 컨텍스트 관련 산출물 획득을 위한 방법 및 절차 지원성 비교

산출물 번호	항목	본 연구	기존 방법
①	요구사항을 컨텍스트에 반영하는 방법 및 절차	○	×
②	컨텍스트 인지 서비스 파악 방법 및 절차	○	×
③	컨텍스트 파악 방법 및 절차 컨텍스트와 유스케이스의 관계성 파악 방법 및 절차	○	×
④	사용자 플랫폼 파악 방법 및 절차	○	×
⑤	서브시스템 파악 방법 및 절차	○	×

○: 지원, △: 일부 지원, ×: 지원안함

표 5 요구사항 명세화 방법 지원성 비교

○: 지원, △: 일부 지원, ×: 지원안함

기술요소 관련연구	유스케이스 지원	컨텍스트 모델링	컨텍스트 추적성
본 논문의 방법	○	○	○
ORM 확장[21]	×	○	×
UML 확장[20]	×	○	×
Naoyasu [17]	×	△	×

사례연구 개발 과정에서 컨텍스트 인지 유스케이스 다이어그램은 컨텍스트 인지 서비스에 대한 이해를 높일 수 있었고, 이해관계자 사이에 의사소통을 수월하게 할 수 있었다. 즉, 기존에 주석에 의존해서 기술했던 컨텍스트 관련 정보들을 정형화된 방법으로 기술하기 때문에 정보의 형태가 더욱 명확해졌고, 구체적인 내용을 파악할 수 있게 되었다. 또한 본 연구에서 제공하는 컨텍스트 다이어그램은 컨텍스트에 대한 정보를 구체적으로 표현함으로써 컨텍스트에 대한 이해도를 높일 수 있게 하였고, 또한 설계 단계에서 클래스 다이어그램으로 변환해서 재사용할 수 있었다. 마지막으로 컨텍스트타입-유스케이스 의존 다이어그램은 사례연구 시스템 개발에서 많이 사용하지는 않았지만, 향후 요구사항에서 컨텍스트의 변화가 많은 경우에 유용할 것이다.

표 5는 본 연구의 요구사항 명세화 방법과 기존 연구의 명세화 방법의 차이점을 보여준다. 기존 연구들이 컨텍스트 모델링에 관련된 명세화 방법은 제공하지만, 컨텍스트 인지 서비스의 이해를 돕기 위한 유스케이스와 컨텍스트의 변화에 따른 유스케이스의 변화를 추적할 수 있는 명세화 방법은 지원하지 못하고 있다.

본 연구 방법은 정성적인 평가에서는 위에서 기술한 바와 같이 기존 연구들에 비해서 우수하다. 그러나 향후에 개발 기간, 개발 비용, 요구사항의 정확도 등에 대한 정량적인 평가를 수행할 필요가 있다.

8. 결론

유비쿼터스 컴퓨팅의 가장 중요한 특징 중의 하나는 사용자가 시스템의 복잡한 내부를 모르더라도 사용자의 의도에 맞는 최적의 기능을 제공하는 컨텍스트 인지 서비스이다. 컨텍스트 인지 시스템은 외부 환경을 인식해야 하고, 컨텍스트의 개념을 포함하기 때문에 기존 시스템에 비해 복잡하며, 시스템 사용에 있어서 창의성을 필요로 한다. 기존 시스템에서는 없던 특징들(컨텍스트, 멀티플랫폼의 분산 환경, 다양한 형태의 사용 시나리오 등) 때문에 기존 요구사항 도출 방법을 컨텍스트 인지 시스템에 그대로 적용하기 어렵다. 이러한 문제점을 해결하기 위해서 본 논문에서는 컨텍스트 인지 시스템을

위한 요구사항 도출 방법과 명세화 방법을 제안하였다.

본 논문에서 제안하는 요구사항 도출 방법은 하향식의 점진적이며 반복적인 프로세스를 갖는다. 컨텍스트 인지 시스템은 비즈니스 로직 기반의 시스템에 컨텍스트 인지 서비스를 추가한 것이기 때문에 비즈니스 유스케이스를 먼저 파악하고, 컨텍스트 인지 서비스에 관련된 유스케이스는 나중에 파악하는 방법을 사용한다. 따라서 요구사항 도출 절차는 시스템 개념 정립, 비즈니스 유스케이스 파악, 사용자 플랫폼 및 컨텍스트 인지 유스케이스 파악, 컨텍스트 모델링, 컨텍스트 타입과 유스케이스의 의존관계 파악, 서브시스템 파악의 단계를 따른다. 본 요구사항 도출 방법의 특징은 컨텍스트 파악 및 모델링 절차, 이해관계자의 요구사항을 컨텍스트 및 컨텍스트 인지 서비스에 포함시킬 수 있는 방법과 절차, 시스템의 사용 시나리오의 창의성을 요구사항에 포함시킬 수 있는 방법 등을 제공한다는 점이다.

본 논문에서는 또한 컨텍스트 인지 시스템의 요구사항을 효과적으로 기술할 수 있도록, UML을 확장한 컨텍스트 인지 유스케이스 다이어그램, 컨텍스트 다이어그램, 컨텍스트타입-유스케이스 의존 다이어그램을 제안한다. 컨텍스트 인지 유스케이스 다이어그램은 유스케이스와 컨텍스트, 컨텍스트 속성의 관계를 보여줌으로써 시스템의 컨텍스트 및 기능적 요구사항을 효과적으로 표현할 수 있다. 컨텍스트 다이어그램은 시스템 이해에 필요한 컨텍스트 개념을 컨텍스트, 컨텍스트 타입, 컨텍스트 속성의 관계를 통해서 보여준다. 컨텍스트타입-유스케이스 의존 다이어그램은 컨텍스트에 의해 영향을 받는 유스케이스들을 보여줌으로써 시스템 설계에 도움을 주고, 컨텍스트의 변경 시에 유스케이스의 추적을 가능하게 한다.

본 논문에서 소개한 방법은 컨텍스트 인지 시스템의 요구사항을 효과적으로 도출할 수 있도록 함으로써 시스템 개발에서 시행착오를 줄여주고, 이해관계자들이 의사소통을 원활히 할 수 있도록 도움을 줄 것이다.

참고 문헌

[1] Nicholas A. Bradley and Mark D. Dunop, "Towards

- a Multidisciplinary Model of Context to Support Context-Aware Computing," *Human-Computer Interaction*, Vol.20, pp. 403-446, 2005.
- [2] Anind K. Dey, Gregory D. Abowd, and Daniel Salber, "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications," *Human-Computer Interaction*, Vol.16, No.2, pp. 97-166, 2001.
- [3] Strang T. and Linnhoff-Popien C., "A context modeling survey," *UbiComp*, 2004.
- [4] Karen Henriksen, Jadwiga Indulska, and Andry Rakotonirainy, "Modeling Context Information in Pervasive Computing Systems," *Conf. on Pervasive Computing*, Springer-Verlag, LNCS 2414, pp. 167-180, 2002.
- [5] Wolfgang Beer, et al. "Modeling Context-Aware Behavior by Interpreted ECA Rules," Springer-Verlag, *LNCS(Mobile and Ubiquitous Computing)*, pp. 1064-1973, 2004.
- [6] Gustavo Rossi, Silvia Gordillo, and Fernando Lyardet, "Design Patterns for Context-Aware Adaptation"
- [7] James A. Landay and Gaetano Borriello, "Design Patterns for Ubiquitous Computing," *IEEE Computer*, pp. 93-95, Aug., 2003.
- [8] Daniel Salber, Anind K. Dey, Gregory D. Abowd, "The Context Toolkit: Aiding the Development of Context-Enabled Applications," *Proc. of CHI*, pp. 15-20, May 1999.
- [9] Manuel Roman, Christopher Hess, Renato Cerqueira, Anand Ranganat, Roy H. Campbell, Klara Nahrstedt, "Gaia: A Middleware Infrastructure to Enable Active Spaces," *IEEE Pervasive Computing*, pp. 74-83, Oct.-Dec., 2002.
- [10] Tao Gu, Hung Keng Pung, Da Qing Zhang, "A Service-Oriented Middleware for Building Context-Aware Services," *Elsevier Journal of Network and Computer Applications (JNCA)*, Vol.28, Issue 1, pp. 1-18, Jan., 2005.
- [11] M. Levinson, "All-in-One Appliance - THE REFRIGERATOR," *CIO Magazine*, Feb., 2003.
- [12] Keith Cheverst, et al., "Developing a Context-aware Electronic Tourist Guide: Some Issues and Experiences," *Proc. of CHI*, 2000.
- [13] Miguel A. Munoz, et al. "Context-Aware Mobile Communication in Hospitals," *Computer*, Vol.36, No.9, pp. 38-46, Sep., 2003.
- [14] Jongmyung Choi, "Context-driven Requirements Analysis," *Proc. of ICCSA*, Springer-Verlag, LNCS, 2007.
- [15] 최중명, "컨텍스트 인지 시스템을 위한 요구공학 모델", *디지털산업정보학회논문지*, 6월, 2007.
- [16] Dan Hong, Dickson K.W. Chiu, and Vincent Y. Shen, "Requirements Elicitation for the Design of Context-aware Applications in a Ubiquitous Environment," *Proc. of ICEC*, Aug., 2005.
- [17] Naoyasu Ubayashi and Shin Nakajima, "Context-aware Feature-Oriented Modeling with an Aspect Extension of VDM," *Proc. of SAC*, Mar., 2007.
- [18] Hannes Omasreister and Eduard Metzker, "A Context-Driven Use Case Creation Process for Specifying Automotive Driver Assistance Systems," *Proc. of Int'l Requirements Engineering Conf.*, 2004.
- [19] Anthony Finkelstein and Andrea Savigni, "A Framework for Requirements Engineering for Context-Aware Services," *Proc. of International Workshop From Software Requirements to Architectures*, 2001.
- [20] Sheng Q. Z. and Benatallah B., "ContextUML: A UML-Based Modeling Language for Model-Driven Development of Context-Aware Web Services," *Proc. of ICMB*, 2005.
- [21] Hendriksen K., Indulska J., and Rakotonirainy A., "Generating context management infrastructure from high-level context models," *MSM*, pp. 1-6, 2003.
- [22] Seng Loke, *Context-Aware Pervasive Systems*, Auerbach Pub., 2007.
- [23] Schilit B. and Theimer M., "Disseminating Active Map Information to Mobile Hosts," *IEEE Network*, Vol.8, No.5, pp. 22-32, 1994.
- [24] Brown P.J., "The Stick-E Document: A Framework for Creating Context-Aware Applications," *Conf. on Electronic Documents, Document Manipulation, and Document Dissemination*, pp. 259-272, 1996.
- [25] Anind K. Dey, Gregory D. Abowd, "Towards a Better Understanding of Context and Context-Awareness," *Workshop on The What, Who, Where, When, and How of Context-Awareness in CHI'00*, 2000.
- [26] Matthias Baldauf, Schahram Dustdar, and Florian Rosenberg, "A survey on context-aware systems," *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. 2, No.4, pp. 263-277, 2007.
- [27] Nicky Kern, et al., "Context Annotation for a Live Life Recording," *Pervasive 2004 Workshop on Memory and Sharing of Experiences*, 2004.
- [28] Bravo J., Hervas R, Chavira G, and Nava S., "Modeling Contexts by RFID-Sensor Fusion," *Proc. of Pervasive Computing and Communications Workshops*, pp. 30-34, 2006.
- [29] Bill N. Schilit, Norman Adams, and Roy Want, "Context-Aware Computing Applications," *Proc. of the Workshop on Mobile Computing Systems and Applications*, pp. 85-90, 1994.
- [30] Jongmyung Choi and Jong-bae Moon, "MyGuide: A Mobile Context-aware Exhibit Guide System," *Proc. of ICCSA*, Springer-Verlag, LNCS, 2008.
- [31] Jongmyung Choi and Hyun-Joo Moon, "Software Engineering Issues in Developing a Context-aware Exhibition Guide System," *Proc. of 9th SNPD*, 2008.

**최 중 명**

2004년 3월~현재 국립목포대학교 정보
공학부 컴퓨터공학 교수. 2003년 8월 송
실대학교 컴퓨터학과(공학박사). 1996년
8월 송실대학교 전자계산학과(공학석사).
1992년 2월 송실대학교 전자계산학과(공
학사). 관심분야는 컨텍스트-인지 시스
템, 유비쿼터스 컴퓨팅, 프로그래밍 언어, 소프트웨어공학