

## 최소단위 서비스를 이용한 U-City 통합플랫폼 내에서의 상황 처리 시나리오 개발 방법론 및 개발 도구 구현\*

송훈구\*\* · 김무정\*\* · 현기홍\*\* · 이후석\*\*

Implementation of the Development Tool and Methodology to  
Handling the Event Process in a U-City Integrated Platform by  
Using the Minimum Unit Service\*

Hun-Gu Song\*\* · Moo-Jung Kim\*\* · Ki-Hong Hyeon\*\* · Hoo-Seok Lee\*\*

### ■ Abstract ■

USM(Unit Service Management System) is the development tool and methodology to handling the event process in a U-City integrated platform by using the minimum unit services. USM can be one of the SOA development methodology. Minimum unit service has a minimum business logic that can be executed with input and output parameters. Minimum unit service consists of three parts : service profile, service input and output parameters and service execution information. USM provides two types of the execution method. One is module execution and the other is web service execution. The development of the event sinario by the USM development methodology can reduce the cost and duration of the u-service development by raising the rate of reusing minimum unit service.

Keyword : USM, Minimum Unit Service, Development Methodology, SOA, U-City, Integrated Platform

## 1. 서 론

현재 추진되고 있는 U-City 구축 사업은 필연적으로 U-City 시스템을 통합 운영할 수 있는 통합 운영 센터와 이를 운영할 수 있는 근간이 되는 통합 플랫폼이 필수적으로 필요하게 된다.

이에 국토해양부에서는 U-City 통합운영센터 운영체계 개발이라는 과제를 통해 국가 표준의 통합 플랫폼 개발 연구 과제를 진행하게 되었고, 이 통합 플랫폼 상에서 SOA 개념을 적용한 상황 이벤트 처리를 위한 시나리오 설계/개발 및 실행을 위한 USM(Unit Service Management System)을 개발하게 되었다.

현재까지 많은 SOA 개발 방법론이 제안되었지만 실제 적용할 수 있는 정도의 SOA 방법론은 거의 없는 실정이다. 현재까지 제안되고 있는 주요 SOA 개발 방법론으로는 SODA(Service Oriented Development of Application)[7], SOAD(Service Oriented Analysis and Design)[8], SOUP(Service Oriented Unified Process)[6], SOMA(Service Oriented Modeling and Architecture)[5], 그리고 SOMO(Service Oriented Modeling using Ontology)[3]등이 있지만, 거의 대부분이 서비스의 식별까지만 정의 되고 있고 실제 구현에 필요한 데이터 모델이나, 개발도구는 제공되지 않고 있는 실정이다.

본 논문에서는 SOA 개발 방법론을 적용한 USM의 개념 및 장점과 이를 이용한 서비스 개발 방법론에 대하여 알아보고, USM을 이용한 시나리오 개발 예제를 통해 향후 U-City 구축시 복합 상황 처리의 가능성 및 효용성을 파악해 보고자 한다.

## 2. USM 개념

USM은 최소 단위 서비스<sup>1)</sup> 관리를 통한 서비스 개발로 최소 단위 서비스를 도출하고 이를 관리하

여 U-City 구축시 재사용성을 획기적으로 높여서 개발 기간 및 비용을 절감할 수 있는 개발 방법론과 개발 도구를 제공한다.

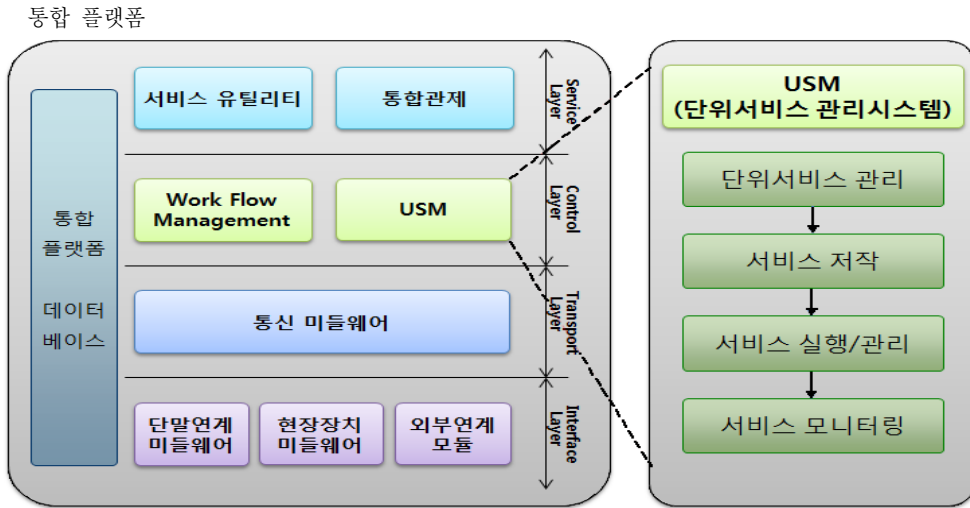
기존 통합 플랫폼 개발의 경우 각 U-City별 구축사에 따라 플랫폼 및 서비스 개발이 이루어져 도시별 연동이 어렵고 구축 비용이 증가할 수 밖에 없는 구조로 이루어지고 있다. 이를 해결하기 위해서는 개발되는 서비스가 단위화 및 정의되고 이를 조합하여 사용할 수 있는 구조로 개발 되는 것이 바람직하다. 이를 위해 기존 서비스들을 구성하는 모든 단계를 최소화하고 유의미한 최소 단위 서비스로 쪼개어 개발되고 관리되어야 한다.

USM을 이용한 서비스 구축은 이를 위해 USM 개발 방법론과 최소 단위 서비스를 등록, 관리, 운영할 수 있는 USM 개발/운영 도구를 제공한다. 향후 U-City 상황 처리 시나리오가 USM을 통해서 개발이 된다면 이들 최소 단위 서비스들이 축적되고 배포될 수 있는 환경이 만들어 질 수 있을 것이다.

[그림 1]은 U-City 통합 운영센터 운영체계 개발 연구의 전체 아키텍처 구조이다. 여기서 USM의 역할은 통신 미들웨어에서 전달 받은 단순/복합 상황 이벤트 정보를 수신하여 이에 대응하는 상황 처리 시나리오를 수행하는 역할을 한다. 이때 상황 처리 시나리오는 USM의 서비스 개발도구를 이용하여 최소 단위 서비스 형태로 개발되어야 하며, 개발 방법은 USM 개발 방법론을 적용하여 개발되어야 한다.

USM은 크게 4가지의 모듈로 이루어져 있다. 단위 서비스 관리 모듈은 최소 단위 서비스를 등록하고 관리하는 모듈로 도출된 모든 최소 단위 서비스가 관리 대상이 된다. 서비스 저작 모듈은 등록된 최소단위 서비스들을 연계하여 서비스 시나리오를 구성하는 모듈로, 서비스 저작도구를 제공한다. 서비스 실행/관리 모듈은 개발된 서비스를 실행하고 실행중인 프로세스의 상태를 조회하거나 중지하는 등의 관리 역할을 담당한다. 서비스 모니터링 모듈은 현재 및 과거의 서비스 실행현황

1) 본 논문에서 최소 단위 서비스(Minimum Unit Service)는 단위 서비스와 혼용하여 사용 된다.



[그림 1] 통합 플랫폼 아키텍처

이력 조회 및 통계를 조회하여 서비스 실행의 문제점을 파악하고 시스템 부하를 모니터링하여 신규 서비스 개발 시 참조할 수 있도록 지원 하는 모듈이다.

USM의 핵심 개념인 최소 단위 서비스는 [그림

2]에서 보는 바와 같이 세 가지 구성요소를 가지게 된다. 첫째, 서비스의 내용을 구성하는 서비스 프로파일은 서비스명, 카테고리, 서비스 설명, 책임부서, 책임자, 서비스 요구수준, 키워드 등의 정보를 포함한다. 둘째로 서비스 입출력 인자 정보는 입력

■ 최소단위 서비스 구조

입력인자			단위서비스		출력인자		
인자명	타입	필수	서비스명	<단위 서비스명>	인자명	타입	필수
<인자명>	<데이터타입>	<필수여부>	카테고리	<대중소 카테고리 입력>	<인자명>	<데이터타입>	<필수여부>
화재코드	String	Y	설명	<단위 서비스가 제공하는 기능 및 사용조건 및 방법들을 상세히 기술>	화재코드	String	Y
화재위치	GisPos	Y	책임부서	<서비스 관리 책임부서>	화재위치	GisPos	Y
화재종류	String	N	책임자	<서비스 관리 책임자>	화재종류	String	N
화재건물	String	N	요구수준	<단위 서비스가 제공하는 요구수준을 정량적으로 기록>	화재건물	String	N
추가정보	String	N	키워드	<단위 서비스 검색을 위한 키워드 추가 정의>	추가정보	String	N
			업무타입	<사람/시스템 처리 업무>			
			실행타입	<웹 서비스/모듈실행>			
			실행위치	<실제 실행 로직 위치>			
			화면정의	<사람 업무인 경우 커스텀마이즈된 화면 위치를 지정>			

[그림 2] 최소 단위 서비스 구성 요소

및 출력 인자의 인자명, 데이터 타입, 필수 여부, 인자 순번 등의 정보로 구성된다. 세 번째로 서비스 실행정보는 업무타입, 실행타입, 실행위치 정보, 화면정의 정보 등으로 구성되어 있다.

단위 서비스들의 조합으로 개발되는 서비스의 구성 요소는 [그림 3]과 같이 구성되어 있다. 단위 서비스와 마찬가지로 서비스 프로파일과 서비스 입출력 인자 정보로 구성되며 여기에 추가적으로 서비스 구성정보를 담는 서비스 프로세스와 서비스 실행 역할 정의 정보를 가지는 서비스 참여자 정보로 구성된다.

앞에서 살펴본 USM의 통합 플랫폼 내에서의 역할과 USM의 개략적인 기능, 그리고 최소단위 서비스 및 서비스의 구성요소를 이해하고 이를 실제 서비스 개발에 적용할 수 있도록 다음 장에서는 USM 개발 방법론에 대하여 자세히 알아보도록 한다.

### 3. USM 개발 방법론

USM 개발 방법론은 대우정보시스템의 CBD 개발 방법론과 SOA 개발 방법론을 접목시켜 새로이 제안하는 최소 단위 서비스 개발 방법론이다.

USM 개발 방법론은 크게 절차서, 산출물 템플릿, 개발 지침서, sample로 이루어져 있으며 각각은 개발 단계별로 세분화되어 구성되어 있다. 절차서는 USM 개발 방법론 절차서, 절차 관계도, 절차 및 산출물 관계, 역할 정의서, 역할 절차 관계도로 이루어져 있다.

#### 3.1 방법론 개요

USM 개발 방법론은 U-City 통합 플랫폼의 핵심인 서비스를 개발하고 운용할 수 있는 서비스 개발 및 운용에 필요한 개발 방법론으로 개발 프로세스 및 개발 툴을 포함하고, 산출물 및 지침서를 제공한다. 또한 개발 툴인 USM은 개발의 효율성 및 재 사용성을 극대화 하고 서비스 운용 서버를 제공하는 종합적인 개발 플랫폼이다. 서비스 개발자는 개발하려는 서비스를 정의하고 이를 기능별로 구분하여 최소 단위 기능으로 재사용 가능하게 동작하도록 최소 단위 서비스를 식별하고 이를 레퍼지토리에 등록을 한 후에 서비스 개발시 최소 단위 서비스의 조합으로 서비스가 이루어 지도록 구성하여야 한다. 따라서 서비스 개발자는

#### ■ 서비스 구조

입력인자			단위서비스		출력인자		
인자명	타입	필수	서비스명	<단위 서비스명>	인자명	타입	필수
<인자명>	<데이터타입>	<필수여부>	카테고리	<대중소 카테고리 입력>	<인자명>	<데이터타입>	<필수여부>
화재코드	String	Y	설명	<단위 서비스가 제공하는 기능 및 사용조건 및 방법들을 상세히 기술>	화재코드	String	Y
화재위치	GisPos	Y	책임부서	<서비스 관리 책임부서>	화재위치	GisPos	Y
화재종류	String	N	책임자	<서비스 관리 책임자>	화재종류	String	N
화재건물	String	N	요구수준	<단위 서비스가 제공하는 요구수준을 정량적으로 기록>	화재건물	String	N
추가정보	String	N	키워드	<단위 서비스 검색을 위한 키워드 추가 정의>	추가정보	String	N

서비스 구성	서비스 프로세스	<등록된 단위 서비스를 검색하여 서비스 프로세스를 등록>
	서비스 참여자 정보	<사람 업무인 단위 서비스는 업무 담당자를 지정>

[그림 3] 서비스의 구성 요소

아래 서비스 개발 방법론을 준수해서 USM에서 동작하는 서비스를 구현하여야 한다.

### 3.2 방법론 특징

USM 개발 방법론은 SOA 기반의 서비스 개발 방법론으로 아래와 같은 특징을 가진다.

- 서비스 구현을 단위 서비스 개념으로 분할하여 조합하는 아키텍처이다.
- 서비스 개발을 공유와 재사용이 가능한 단위 서비스의 집합으로 구현한 디자인 패러다임이다.
- Loosely coupling한 관계를 가진 서비스 제공자와 서비스 요청자로 구성된 Interface-Oriented Architecture이다.
- 서비스의 통합성 및 이질적인 환경에서의 통합의 당위성 대두된다.
- Component의 재사용에 한계(기존 분산 Component 기술의 한계-벤더, 플랫폼, 언어에 종속)로 필요성이 증대 되고 있다.
- 환경 변화에 신속하게 대응하기 위한 서비스의 재 사용성 극대화가 필요하다.
- 사용자가 정의하는 비즈니스 프로세스 중심적인 관점에서 서비스 구현을 시도할 수 있다.

### 3.3 방법론 구성 요소

USM 방법론을 구성하는 요소들을 앞에서도 간략히 살펴보았지만 여기서 다시 자세히 살펴본다. 먼저 최소 단위 서비스 구성 요소는 다음과 같다.

단위 서비스 프로파일은 단위 서비스가 제공하는 서비스 내용을 상세히 기술한 서비스 정의를 말한다. 서비스 내용은 단위 서비스의 용도 및 사용 방법, 입출력 인자 설명, 사용 가능 역할 등을 포함해야 하며, 서비스 설명에 기술한다. 또한 서비스 설명과 별도로 단위 서비스 키워드를 제공하여 단위 서비스 검색 시 활용할 수 있도록 필수적으로 입력해 주어야 한다. 단위 서비스는 성능이나 비 기능적 요구사항을 정의하거나 서비스의 보장 수준을 정의하여야 한다. 예를 들어 로그인 서비스의

경우 24시간 이용 가능해야 하며 사용자 인증은 1초 이내로 수행 되어야 한다 라는 서비스 수준을 정의할 수 있다.

단위 서비스 입출력 인자 정보는 모든 단위 서비스는 호출을 위한 입력 인자 정보와 단위 서비스가 제공하는 출력 인자 정보를 정의하여야 한다. 입출력 인자는 입력인자, 출력인자, 입출력인자 세 가지 타입으로 나누어 등록한다. 인자로 등록을 하기 위해서는 인자의 특성과 용도를 고려하여 명명을 하고 인자의 데이터 타입 및 길이를 정의하여 변수 마스터에 등록이 되어야 한다. 등록된 입출력 인자에 의해서 USM은 서비스 실행 시 실행 모드를 자동 생성하여 결과를 전달할 수 있게 된다.

단위 서비스 실행 정보 : 단위 서비스는 사람이 수행하는 업무와 시스템이 수행하는 업무로 크게 구분된다. 시스템이 수행하는 업무의 경우 웹서비스 혹은 로컬 모듈실행으로 나뉘게 되며, 두 경우 모두 서비스 실행에 대한 위치정보를 포함하여야 한다. 웹서비스의 경우 wsdl 위치 및 서비스명이 포함되며, 모듈 실행의 경우 package명 및 function명이 포함되어야 한다. 단위 서비스 정의 시 등록하는 항목 중에서 업무타입, 실행타입, 실행 위치정보, 커스텀 화면정의 항목이 이에 해당한다.

다음으로 단위 서비스의 조합으로 개발되어 지는 서비스의 구성 요소는 서비스 프로파일, 서비스 프로세스, 서비스 참여자 정보로 이루어진다.

서비스 프로파일은 서비스가 제공하는 서비스 내용을 상세히 기술한 서비스 정의를 말한다. 서비스 내용은 서비스의 용도 및 사용 방법, 입출력 인자 설명, 사용 가능 역할 등을 포함해야 하며, 서비스 설명에 기술한다. 서비스 설명과 별도로 서비스 키워드를 제공하여 서비스 검색 시 활용할 수 있도록 필수적으로 입력해 주어야 한다.

서비스 프로세스는 등록된 단위 서비스를 이용하여 서비스 플로어(flow)를 생성하는 단계이다. USM은 서비스 플로어 작성을 위한 서비스 디자이너를 제공한다. 서비스 플로어는 조건분기, 병렬 실행, 순차실행 등 제어 단위를 이용하여 단위서

비스의 흐름을 제어할 수 있다. 이전 단위 서비스의 출력 인자 중에서 이후 단위 서비스 입력 인자로 전달 될 인자를 연결시켜 인자 값을 전달할 수 있다. 인자 전달 중 간단한 매핑 로직을 추가 할 수 있다.

서비스 참여자 정보는 사람이 처리하는 단위 서비스인 경우 서비스를 처리할 수 있는 권한이 있는 참여자 정보를 지정한다. 참여자는 부서나 역할 또는 사용자를 직접 지정할 수도 있다.

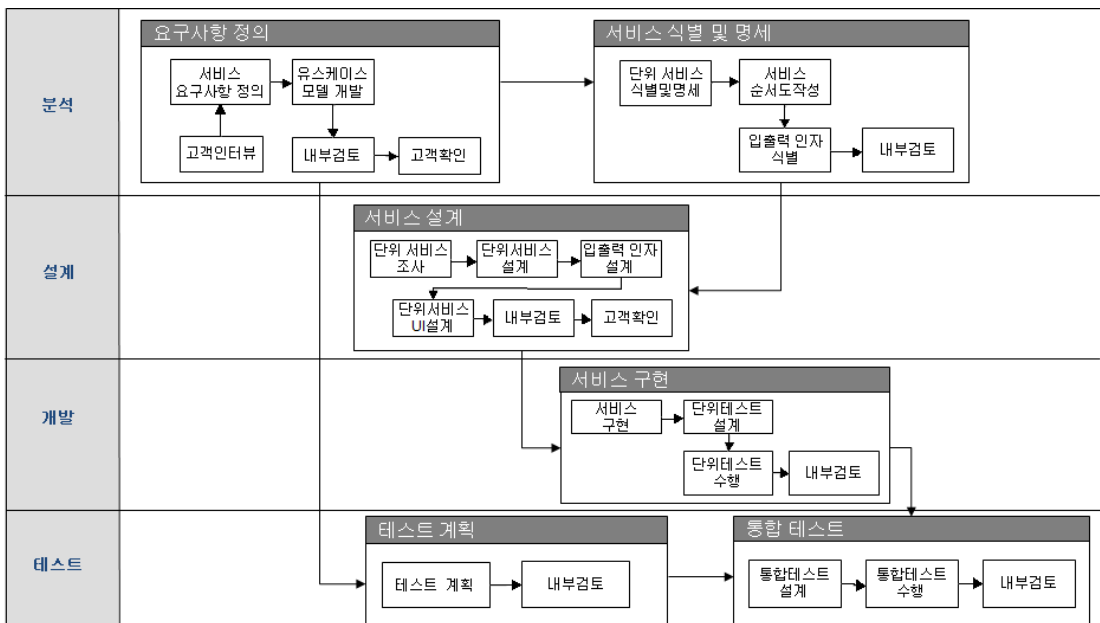
### 3.4 방법론 절차 관계도

USM 개발 방법론이 U-City 서비스 개발을 위한 사용되기 위해서는 아래의 <표 1>과 같은 환경 및 제약사항을 고려하여 분석 및 설계 작업을 수행하여야 한다.

[그림 4]는 USM 개발 방법론의 절차 관계도를 보여준다. 먼저 분석 단계에서 요구사항 정의, 서비스 식별 및 명세 작업이 있다. 요구사항 정의는

<표 1> 분석/설계 단계 고려사항

항목	고려사항
공공정보 서비스	전체 적용할 공공정보 서비스들의 서비스 목적, 서비스 대상, 기대효과를 고려하여 유기적으로 구성되도록 요구사항을 파악한다. 기타 추가적인 서비스의 추가 또는 선정된 서비스에 대한 삭제 요구시 변경평가회의를 거쳐 서비스 변경 요구에 대한 접수 여부를 결정한다.
공공정보통신망	지원 가능한 공공정보 통신망 인프라를 고려한다. QoS, 이동성 지원 요구사항, 광대역 통신망과 Connection Point, Network Manageability/Controllability 등을 고려하여야 한다.
통합운영센터	결정된 통합 운영센터의 주요 기능과 구조를 파악하고 운영센터의 주요 기능을 수행할 상황관 시스템, 출입통제시스템, FMS 등의 기타 시스템을 고려하여야 한다.
통합운영솔루션	통합플랫폼, 통합DB, GIS솔루션 등 통합운영 솔루션이 제공하는 공통 기능 및 소프트웨어에 대한 내용을 고려하여야 한다.



[그림 4] USM 개발 방법론 절차 관계도

기존의 방법론과 유사하지만 U-City 구축 사업에서는 <표 1>과 같은 사항을 유념하여 요구사항을 파악하여야 한다.

분석 단계의 세부 작업 단계로 고객 인터뷰, 서비스 요구사항 정의, 유스케이스 모델 개발, 내부 검토, 고객 확인으로 구성된다. 요구사항 정의에서도 출되는 산출물은 인터뷰 질의서, 인터뷰 결과서, 서비스 요구사항 정의서, 유스케이스 모델 정의서 등이 있다.

서비스 식별 및 명세 작업은 요구사항 정의 단계에서 도출된 유스케이스 별로 세부 기능을 제공할 수 있는 서비스를 구성하는 단계이다. 이를 위하여 각 단위 서비스 들이 최소 단위 서비스로 구성이 될 수 있도록 최대한 상세히 규모와 기능을 분류하여 서비스 순서도를 작성한 후 서비스 입출력 인자를 결정한다. 세부 작업으로는 단위 서비스 식별 및 명세, 서비스 순서도 작성, 입출력 인자 식별, 내부 검토로 이루어져 있으며, 주요 산출물로는 서비스 분해도, 서비스 순서도, 서비스 인자 명세서 등이 있다.

설계 단계의 서비스 설계는 이전 단계에서 도출된 서비스 분해도 및 서비스 순서도의 단위 서비스 들을 정형화하고, 기존에 개발된 기능의 존재 여부를 서비스 레퍼지토리에서 검색하여 존재하는 경우 최대한 재사용하고, 신규 단위 서비스인 경우 단위 서비스를 설계하여 서비스 정의서를 작성하는 단계이다. 세부 작업으로는 단위 서비스 조사, 단위 서비스 설계, 입출력 인자 설계, 단위 서비스 UI 설계, 내부 검토, 고객 확인으로 이루어져 있으며, 주요 산출물은 단위 서비스 조사서, 서비스 정의서, 변수 마스터 명세서, 단위 서비스 마스터, 서비스 입출력 인자 매핑, 사용자 화면 설계서 등이 있다.

개발 단계의 서비스 구현 작업은 서비스 설계 및 데이터 베이스 설계를 바탕으로 USM 시스템 상에 단위 서비스 정보 및 입출력 인자 정보를 등록하고, 해당 단위 서비스의 UI나 비즈니스 로직을 개발하는 단계이다. 그리고 등록된 최소 단위 서비스를 이용하여 서비스를 작성하고 이를 서버에 배

포하여 실행 가능한 상태로 등록한다. 상세 작업으로 서비스 구현, 단위 테스트 설계, 단위 테스트 수행, 내부 검토로 이루어져 있으며, 주요 산출물로는 모듈 소스코드 목록, 서비스 정의 파일 목록, 단위 테스트 설계서, 단위 테스트 결과서 등이 있다.

테스트 단계에서는 통합 테스트를 계획하고 수행하여 테스트 결과를 바탕으로 서비스가 운영 가능한 상태를 보장하는 단계로 테스트 계획, 통합 테스트 설계, 수행, 내부 검토로 이루어져 있으며, 주요 산출물로는 테스트 수행 계획서, 통합 테스트 설계서, 통합 테스트 결과서 등이 있다.

아래의 <표 2>는 USM 개발 방법론의 주요 산출물 리스트를 보여준다.

<표 2> 개발 산출물 리스트

활동영역	프로덕트(Product)
분석-AN00	인터뷰 질의서 인터뷰 결과서 서비스요구사항 정의서 유스케이스모델 정의서 서비스 분해도 서비스 순서도 서비스 인자 명세서
설계-DS00	단위 서비스 조사서 서비스 정의서 변수 마스터 명세서 단위 서비스 마스터 서비스 입출력 인자 매핑 사용자 화면 설계서
개발-DV00	모듈 소스코드 목록 서비스 정의 파일 목록 단위 테스트 설계서 단위 테스트 결과서
테스트-TE00	테스트 수행 계획서 통합 테스트 설계서 통합 테스트 결과서

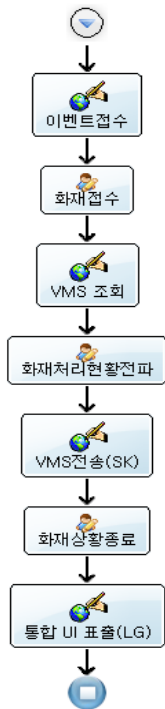
#### 4. USM을 이용한 개발 예

U-City 통합 운영센터 운영체계 개발 과제는 개발이 완료되어 테스트베드 구축작업이 진행 중이다. 따라서 USM을 이용한 상황처리 시나리오의 실제적인 적용 사례는 없는 상황이다. 이에 통합

플랫폼의 통합 테스트용으로 개발된 적용 예제를 바탕으로 USM을 통한 서비스 개발 절차를 알아보도록 한다.

#### 4.1 차량화재 시나리오 구현

[그림 5]은 차량화재 발생시 상황을 접수하고 전파하는 시나리오를 구성한 화재상황 시나리오 구성도이다. 도출된 단위서비스는 7개로 통신 미들웨어에서 이벤트를 접수하는 이벤트 접수, 관제자가 화재 상황을 인지하고 이를 접수하는 화재 접수, 화재 위치와 연계하여 상황 전파를 할 수 있는 VMS 리스트를 조회하는 VMS 조회 서비스, 조회된 VMS 리스트 중에서 표출할 VMS를 선택하고 상황 전파 메시지를 입력하고 요청하는 화재 처리 현황 전파 서비스, 단말연계 미들웨어에 VMS 호출 정보를 연계하는 VMS 전송 서비스, 화재 상황 종료시 상황 종료료를 처리하는 화재상황 종료



[그림 5] 차량 화재 시나리오

서비스, 마지막으로 통합 관제 UI에 화재 상황 종료를 알려주는 통합 UI 표출 단위 서비스로 구성되어 있다.

[그림 7]과 같이 각각의 최소 단위 서비스의 역할을 정의하는 단위 서비스 마스터를 작성한다. 단위 서비스 마스터에는 영문명, 한글명, 업무유형, 서비스 요구수준, 서비스 실행 위치 정보, 화면 정의, 서비스 상세 설명들이 포함되어 있다. 다음으로 [그림 8]과 같이 서비스 구성 정보 및 서비스 입출력 인자를 포함하는 서비스 정의서를 작성한다. 서비스 정의서까지 설계가 모두 이루어지면 도출된 최소 단위 서비스들을 USM 시스템에 등록을 하게 된다. [그림 6]은 USM의 최소 단위 서비스 등록 화면을 보여준다. [그림 9]은 최소 단위 서비스의 입출력 인자 등록 화면이다. 변수 마스터에 등록된 인자를 입력, 출력, 입출력 인자로 구분해서 등록을 해주게 된다.



[그림 6] 최소 단위 서비스 등록 화면

입출력 인자까지 등록된 최소 단위 서비스는 이를 이용하여 서비스 시나리오를 만드는데 이용될 수 있다. [그림 10]은 시나리오를 구성하는 서비스 디자이너 구동 화면을 보여준다.

서비스 시나리오 구성이 완료되고 배포된 서비스는 이벤트 수신시 대응하여 실행이 되게 된다. 이벤트가 접수되면 접수 담당자의 화면에 새 업무 리스트로 나타나게 되고, 사람이 처리하는 업무인



단위 서비스 영문명	단위 서비스 한글명	업무 유형	서비스 수준	실행정보	화면정보	상세내용
EventReceipt	이벤트접수	시스템	통합테스트용	module.us.exec. ReceiveEvent. receiveUnit03		통신 미들웨어에서 수신 받은 이벤트 정보를 접수 하는 서비스이다. 모든 이벤트에 대한 서비스 실행의 첫 단위 서비스로 등록되어야 한다. 입출력 변수로는 이벤트코드, 접수일시, 이벤트명, 이벤트 등급, 이벤트상태, 발생위치X, 발생위치Y, 발생위치 Z, 서비스정의 ID 인스턴스 ID, 발생일시 상세내용, 발생장소로 등록된다.
HreceiptFire	화재접수	사람	통합테스트용		/usm/Receive FireEvent	화재 이벤트 접수처리, 화재유형, 레벨, 발생위치를 전송받아 화재이벤트를 접수하고, 이를 화면에 표현하여 담당자가 확인 후 이후처리를 시작할 수 있도록 하는 단위 서비스이다
SlistVMS	VMS조회	시스템	통합테스트용	module.us.info. FacilityService. listVMS		GIS위치 정보 및 이벤트 종류별로 VMS 리스트를 검색한다. 입력인자 : 발생위치, 사건코드 출력인자 : 발생위치, 사건코드, VMS 리스트 개발자 : songhu, 관리부서 : 시설과
Hrequest MinVMS	화재처리 현황진과	사람	통합테스트용		/usm/Request MinVMS	조회된 VMS 정보를 보여주고, 이중 선택해서 상황진과 요청을 처리한다. 입력 : VMS 리스트 출력 : VMS리스트, VMS 메시지 개발자 : songhu, 관리부서 : 시설과
SsendVMSsk	VMS 전송(SK)	시스템	통합테스트용	module.us.exec. SendVMS.vms SendSK		SK에서 제공하는 VMS 전송 서비스를 호출하여 VMS 전송한다. 입력값으로 VMS ID 리스트, 전송한 문자를 받아들이고, 출력으로 전송 결과를 내보낸다.
HcloseFire EventTest	화재 상황종료	사람	통합테스트용		/usm/Fire Close03	소방차 출동후 화재가 모두 진압된 후, 화재 상황을 종료하는 단위 서비스이다. 화면에 화재에 대한 정보를 보여주고 담당자가 종료처리를 하면 서비스가 종료된다.
SviewTuiLG	통합 UI 표출(LG)	시스템	통합테스트용	/module/us/show/ ShowWindow. showTULG		LG에서 제공하는 통합 UI 표출 서비스를 호출하여 통합 UI에 상황종료 표출 요청을 전송한다.

[그림 7] 단위 서비스 마스터

경우 담당자의 화면에 업무별 처리 화면이 나타나게 된다. 시스템이 처리하는 단위 서비스인 경우 시스템 내부적으로 비즈니스 로직이 실행된 후 다음 단계의 단위 서비스가 처리되게 된다. [그림 11]은 내 업무 리스트를 보여주며, [그림 12]은 화재상황 종료 업무 처리 화면을 보여준다.

#### 4.2 최소 단위 서비스 재활용 방안

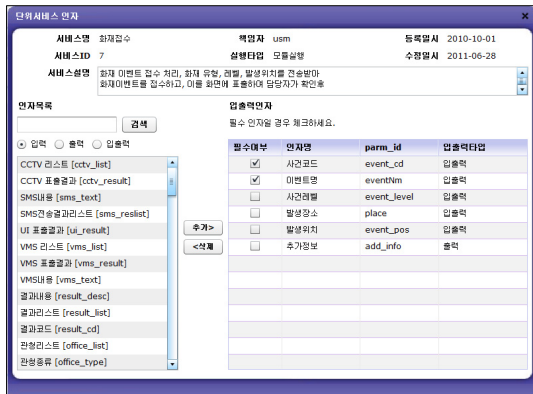
지금까지 USM 개발 방법론을 이용한 서비스 개발 절차 및 개발된 서비스가 운영되는 절차를 알아보았다. USM 개발 방법론 및 USM을 이용하여 개발된 최소 단위 서비스들은 모두 서비스 레퍼지토리에 등록되어 관리되어 지게 된다. 또한 한번 개발된 최소 단위 서비스는 쉽게 다른 U-City 구축 사업에 재사용될 수 있다. 이를 위해 USM 개발 도구는 최소 단위 서비스를 추출하고 등록하는 기능을 제공하고 있다.

서비스 구축에서의 재활용 율을 정량적으로 딱히 결정할 수는 없지만, 아래와 같은 정도의 재활용율 산정식을 생각해 볼 수 있다. 하나의 서비스 시나리오가 총 T개의 개념적인<sup>2)</sup> 최소 단위 서비스로 등록되어 있을 때 이중에 기존에 개발되어 등록되어 있는 최소 단위 서비스를 N개라 하고, 이를 그대로 이용하여 사용하는 경우 추가 공수가 10%<sup>3)</sup>정도로 가정하면, USM을 사용하지 않는 경우 개발 공수와 사용할 때의 개발공수는 아래의

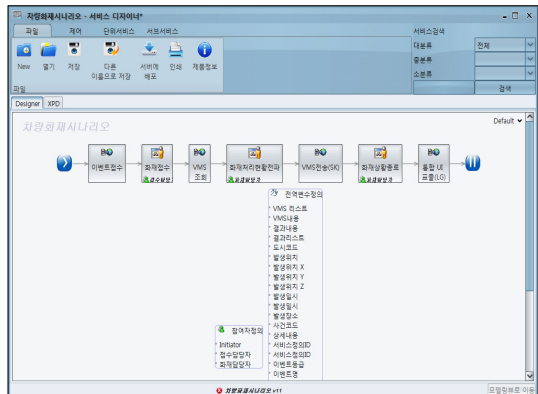
- 2) 개념적인 최소 단위 서비스는 실제 USM 상에 최소 단위 서비스로 등록 할 수는 없으나, 구현 규모 단위가 최소 단위 서비스와 유사한 정도의 비즈니스 로직 구현 단위를 말한다.
- 3) 추가 공수는 최소 단위 서비스를 USM에 등록 관리하고 이를 사용하는 공수로 전체 비즈니스 로직 개발 공수와의 비율을 10%로 가정하였다. 복잡한 로직일수록 추가 공수 비율은 낮아지게 된다. 10%는 이해를 돕기 위하여 표시한 숫자이고 해당 비율은 재활용 한계 비율과 연관되어 의미를 갖게 된다.

서비스 정의				
서비스 명(한글)	화재차량 시나리오		책임부서	개발부
서비스 명(영문)	VehicleFireTo1		책임자	운영자
서비스 설명	통합플랫폼 통합 시연을 위한 시나리오이다. 화재차량 이벤트를 통신미들웨어로부터 수신하여 USM 서비스를 실행한다. 상황진과가 필요한 경우 단말연계 미들웨어를 통해 VMS에 상황 알림 메시지를 표출한다. 화재 상황 종료시 VMS 및 통합 UI에 상황 종료를 전파한다.			
분류	방법/관계 > 관리 > 관리지원			
키워드	통합 플랫폼 통합 시연 시나리오 차량화재 이벤트 수신 USM 차량화재 서비스 실행 상황전과 필요 단말연계 미들웨어 VMS 상황 알림 메시지 표출 화재 상황 종료 통합 UI 상황 종료 전파			
단위서비스 정의				
단위서비스 명(한글)	이벤트 접수		업무유형	시스템
단위서비스 명(영문)	EventReceipt		실행유형	모듈
설명	이벤트 상황 애플레이터에서 수신받은 이벤트 정보를 접수하고 서비스를 시작한다.			
실행정보	module.us.exec.ReceiveEvent.receiveUnit			
화면정보				
입력인자	영문명	한글명	타입	
	제1인자	svclId	서비스정의ID	Number
	제2인자	instId	인스턴스ID	Number
	제3인자	ctyCd	도시코드	String
	제4인자	evtCd	이벤트코드	String
	제5인자	rcvDt	접수일시	Date
	제6인자	eventNm	이벤트명	String
	제7인자	eventGrd	이벤트등급	String
	제8인자	eventState	이벤트상태	String
	제9인자	placeCoordX	발생위치X	Double
	제10인자	placeCoordY	발생위치Y	Double
	제11인자	placeCoordZ	발생위치Z	Double
	제12인자	place	발생장소	String

[그림 8] 서비스 정의서



[그림 9] 단위 서비스 입출력 인자 등록 화면



[그림 10] 서비스 저작도구 화면

분류	업무명	사자자명	시작일	완료예정일	종료일	우선순위	상태
방범/경차 > 감시 > 자력감시	방범자상황중료	이도민	2011-09-27	2011-09-28			정상
방범/경차 > 감시 > 자력감시	화재발달 접수	이도민	2011-07-21	2011-07-15	2011-07-21		완료
교통 > 안전 > 차량안전	화재상황중료	이도민	2011-07-24	2011-07-15			정상
교통 > 안전 > 차량안전	화재발달 접수	이도민	2011-06-27	2011-06-28	2011-06-25		완료
교통 > 안전 > 차량안전	화재발달 접수	이도민	2011-06-25	2011-06-26			완료
교통 > 안전 > 차량안전	화재발달 접수	nobody	2011-06-25	2011-06-26			정상
교통 > 안전 > 차량안전	화재발달 접수	nobody	2011-06-25	2011-06-26			정상
교통 > 안전 > 차량안전	화재발달 접수	이도민	2011-06-25	2011-06-26			정상
교통 > 안전 > 차량안전	화재상황중료	이도민	2011-06-24	2011-06-25			완료

[그림 11] 내업무 목록 화면

이벤트코드	TRF00108
화재레벨	0
화재장소	X 좌표 : 37.569206 Y 좌표 : 126.979783
화재건물	정계전
추가정보	
처리결과	
종료의견	

[그림 12] 화재상황 종료 업무처리 화면

식과 같다.

$$\begin{aligned}
 Ha &= u \times T \\
 Hb &= 0.1 \times u \times N + (1 + 0.1) \times u \times (T - N) \\
 &= u \times (1.1T - N)
 \end{aligned}$$

- \*  $Ha$  : USM을 사용하지 않은 경우 개발 공수
- \*  $Hb$  : USM을 사용한 경우 개발 공수
- \*  $u$  : 1최소 단위 서비스 개발 공수
- \*  $T$  : 총 개발단위수
- \*  $N$  : 재사용 가능 단위수

$$\begin{aligned}
 \text{재활용 효율 } \gamma &= 100 \times \frac{Ha - Hb}{Ha} \\
 &= 100 \times \frac{uT - u(1.1T - N)}{uT} \\
 &= 100 \left( \frac{N}{T} - 0.1 \right)
 \end{aligned}$$

이는 재활용 한계 비율인 10% 이상의 최소 단위 서비스만 재활용 되어도 재활용 효율이 나타난다는 말이 되며, 만약 절반의 최소 단위 서비스를 재활용할 경우 재활용 율은 40%가 되고 이는 시나리오 개발 공수의 40%를 절감할 수 있다는 뜻이 된다.

## 5. 결 론

지금까지 U-City 통합 플랫폼 내에서의 USM의 역할 및 개념을 알아보고, SOA 개발 방법론의 일종인 USM 개발 방법론을 제안하였다. 또한 USM 개발 방법론에 의해 개발된 시나리오 예제를 통해 USM의 적용 가능성을 알아보았다.

앞에서 살펴본 바와 같이 최소 단위 서비스 도출 및 USM 개발 방법론을 통해서 통합 플랫폼 내에서의 상황 처리 시나리오를 개발 및 운영하게 되면, 이미 개발된 시나리오들을 다른 U-City 구축 사업에서 재사용할 수 있어 개발 기간 및 비용을 획기적으로 절감할 수 있다. 또한 각 U-City를 추진하는 지자체들 간의 상황처리 시나리오 표준이 자연스럽게 통일되어 업무처리 프로세스 자체의 표준화를 이룰 수 있는 효과도 기대할 수 있을 것이다.

향후 모든 U-City 구축 사업에서 USM을 통한 상황처리 시나리오가 개발이 되고 적용이 되면 향후 개발된 각각의 최소 단위 서비스들이 국가적으로 관리되고 재사용 가능한 최소 단위 서비스 표준 레퍼지토리의 구축도 이루어 질 수 있을 것이다.

## 참 고 문 헌

- [1] 조위덕, 『U-SERVICE DESIGN』, 진한엠앤

- 비, 2009.
- [2] 구지희, 김복환, 김은형, 정대교, 정우영, 정태웅, 최규태, 최창수, 한인교, 『유비쿼터스 도시, U-City 총론』, 시그마 프레스, 2009.
- [3] 장효선, 박세권, 류승완, 신동천, “SOA를 위한 온톨로지 기반의 서비스 도출 방법론”, 『한국 IT서비스학회지』, 제10권, 제2호(2011), pp.309-327.
- [4] 최고봉, 박세권, 류승완, “서비스 온톨로지 기반 SOA 개발 방법론”, 『한국IT서비스학회지』, 제9권, 제2호(2010), pp.193-203.
- [5] Arsanjani, A., “Service-oriented modeling and architecture”, IBM developerworks, 2004.
- [6] Mittal, K., “Servicee Oriented Unified Process (SOUP)”, IBM Journal, <http://www.kunalmittal.com/html/soup.html>, 2005.
- [7] Nigam, S., Service Oriented Development of Applications (SODA) in Sybase Workspace, Sybase Inc. whitepaper, 2005.
- [8] Zimmermann, O., P. Krogdahl, and C. Gee, “Elements of service-oriented analysis and design”, IBM developerworks, 2004.

## ◆ 저 자 소 개 ◆

**송 훈 구 (songhu@disc.co.kr)**

포항공대 기계공학과(학사, 1996)를 졸업 후 현재는 대우정보시스템 재직 중이며, 관심분야는 BPM, U-City 등이다.

**김 무 정 (moojung@disc.co.kr)**

숭실대학교 전산학과(학사, 1998)를 졸업 후 현재는 대우정보시스템 재직 중이며, 관심분야는 U-City, RFID 등이다.

**현 기 홍 (pmhyeon@paran.com)**

건국대학교 전자공학(학사, 1992)을 졸업 후 현재는 대우정보시스템 재직 중이며, 관심분야는 정보처리 및 정보통신, U-City 등이다.

**이 후 석 (nasilin@disc.co.kr)**

단국대학교 컴퓨터공학(학사, 2002)을 졸업 후 현재 대우정보시스템 재직 중이며, 관심분야는 U-City, RFID 등이다.