

Context? Context!

David Kim, Danny Park,
and Brad Park

2002-03-16

*'convergence 그 자체만으
로는 고객에게 가치가 될
수 없다는 부분에는 동의가
되지만 구체적인 Context
의 실천 방법은 무엇이 있
겠는가?'*

지난번 유무선 '통합포탈은 없다'가 나간 후 davidndanny는 많은 질문을 받았다. 대부분의 질문이 'convergence 그 자체만으로는 고객에게 가치가 될 수 없다는 부분에는 동의가 되지만 구체적인 Context의 실천 방법은 무엇이 있겠는가?'라는 것이었다.

지난 칼럼에서는 convergence 환경에서 고객이 느끼는 벨류는 context를 통해 만들어 질 것이라고 지적했던 것이 다였기 때문에 오늘은 왜 지금 'Context'에 대해 새삼스럽게 주의를 더 기울여야 한다고 davidndanny가 말하고 있는지 좀 더 자세하게 이야기 하면서 다소 실험적이지만 상상력을 자극하는 예를 소개하려 한다.

David Kim

davidkim@davidndanny.com

Danny Park

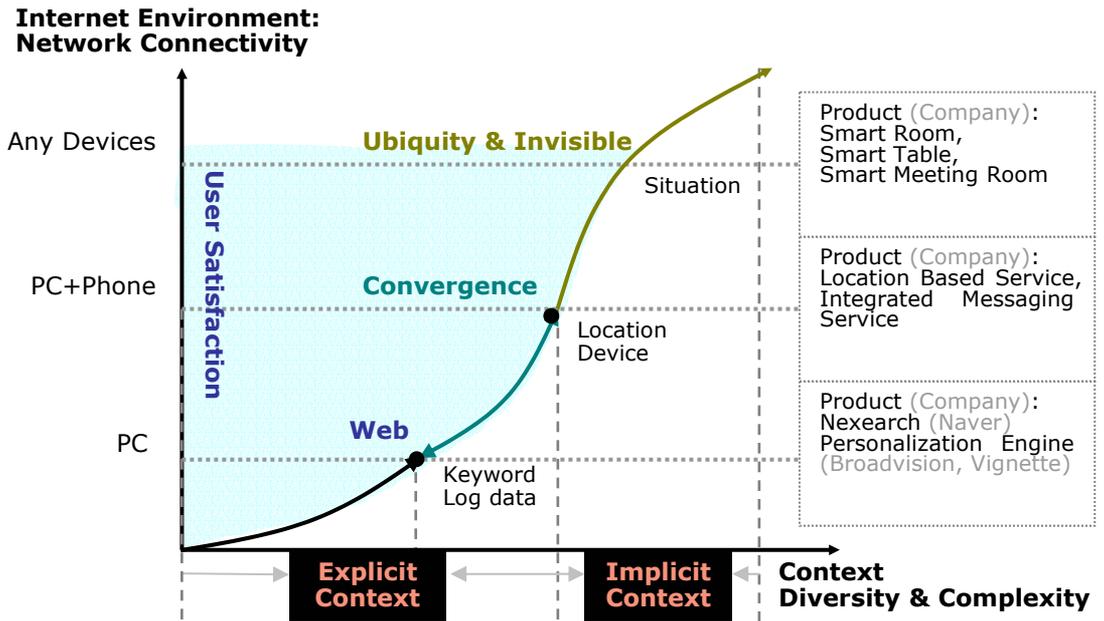
danny@davidndanny.com

Brad Park

brady@oztek.co.kr

인터넷 환경의 발전과 context

<Internet Environment와 Context Information의 진화>



<Source: 2002 davidndanny.com>

위의 그래프는 첫째, PC중심의 웹기반 인터넷환경에서는 이용자들의 Context(정확히는 Explicit한 Context)를 이해하는데 필요한 정보들이 키워드나 로그데이터와 같은 deterministic한 것들이 대부분이었지만 '유무선 통합' 인터넷환경에서는 이용자들의 Location이나 사용하고 있는 Device 정보(예전에는 Implicit Context에 해당되었던 정보)등과 같은 이용자가 어떤 상황에 있는 지를 좀 더 심도 있게 이해할 수 있게 해주는 Context를 확보하게 됨에 따라 Location Based Service나 Integrated Messaging Service등이 가능하게 되고 유저들의 인터넷사용만족도 역시 훨씬 커지고 있음을 보여주고 있다.

지난 주의 '유무선 포탈은 없다'의 표<Innovation of Portal's Value Proposition>를 보면 기존의 웹기반 인터넷에서는 User Input정보인 ID, Password, Credit Card번호 등의 정보와 Cookie 값을 통한 경로 정보가 사용할 수 있는 정보의 대부분이었고 유저들이 방문하고 있는 페이지 등에 대한 속성정보도 충분하지 않았기 때문에 웹기반 인터넷에서는 이용자 Context를 충분히 이해한 유저가 만족할 만한 서비스가 지속적으로 발전해가기 어려움이 있었다.

'Ubiquitous Internet의 시대'로 진화하게 되면서 인터넷서비스 사업자들은 인터넷에 다양한 디바이스를 통해 들어와 있는 이용자와 Embedded Internet Server를 통해 인터넷에 올라와 있는 이용자 주위의 환경(다양한 공간과 기기들)이 만들어 내는 Implicit Context를 활용한 스마트서비스를 제공해야 유저들의 보다 큰 만족을 만들어 낼 수 있음을 보여주고 있다."

두 번째로, 전기나 전자 프로세서 기술이 성숙해감에 따라 모든 기기들 속에 들어가서 우리들의 눈에서 사라졌던 것처럼 인터넷 역시도 기기 속으로 Embedded 되어 우리가 있다고 느끼지 못한 채 인터넷을 쓰는 'Ubiquitous Internet의 시대'로 진화하게 됨을 보여주고 있으며 인터넷서비스 사업자들은 인터넷에 다양한 디바이스를 통해 들어오는 이용자와 Embedded Internet Server를 통해 인터넷에 올라와 있는 이용자 주위의 환경(다양한 공간과 기기들)과 관련한 Implicit Context를 활용한 스마트서비스를 개발해서 제공해야 유저들의 만족을 보다 크게 만들어 낼 수 있음을 보여주고 있다.

이 그래프에서 나타난 Context Aware Service가 "빌게이츠가 꿈꾸는 집" 정도의 스마트 환경으로 짧은 시간 안에 실현될 것이라고는 davidndanny도 생각하지 않는다.

그럼에도 불구하고 이렇게 Context Aware Service를 강조하는 것은 무료로만 경쟁하던 웹기반 인터넷과는 달리 이용자 context에 기반한 서비스들은 고객들의 주머니를 확실하게 열어 줄만한 차별화된 포인트가 있다는데 가장 큰 이유가 있다.

거기다가 Convergence 환경이 열리고 있는 요즘의 상황에서 앞으로 나타날 거대 유무선통합포탈사업자들은 이용자와 새로운 수준의 Billing Relationship을 가지고 사업을 시작할 것이기 때문에 이런 Billing Relation을 최대한 활용하기 위해서도 Context Aware Service가 중요함을 davidndanny는 자꾸만 강조하는 것이다.

그렇다면 Context는 무엇인가?

Context란 무엇인가?

'Context Aware Service'를 정의한다면, '유저가 입력한 정보와 상황정보들이 결합되어 사용자가 처한 Situation에 맞게 유저가 원하는 수준으로 조정되어 제공되는 Smart 서비스'

인간은 매우 효과적으로 커뮤니케이션 하고 있다고 말할 수 있는 이유는 우리의 대화가 언어의 교환에만 의존한 것이 아니라 대화를 나누고 있는 당사자를 둘러싼 상황(물리적 상황, 정서적 상황, 역사적 상황 등)에 대한 정보(implicit situational information or Context)를 적극적으로 활용하기 때문에 의도적이지 않았다면 실수를 최소화할 수 있기 때문이다. 만약 인간이 이런 상황정보를 이용하지 않거나 의도적으로 일부 정보를 무시하면 커뮤니케이션에는 심각한 왜곡이 일어날 수 있다.

영화관에서는 가능한 전화를 끄고 혹은 전화를 받게 되더라도 정말 작은 소리로 전화통화를 하려 하는 것은 우리 인간의 감지객체가 "내게 전화가 왔다"라는 "따라서 전화를 받아야 한다"라는 이벤트와 대상객체인 영화관이 가진 상황정보인 "영화상영 중이며 여러 사람이 있고 그들은 방해 받고 싶지 않다"등을 활용하여 여러 선택 가능한 Context Aware Action중에 조용하게 말하는 것을 선택했다라고 말할 수 있는데 이런 것이 바로 우리들이 무의식 중에 활용하는 Context의 예이다.

불행하게도 인간과 컴퓨터가 대화할 때는 이런 상황정보를 사용할 수 없다. 바로 이런 점을 보완해서 컴퓨터가 Context를 이해하게 만듦으로써 인간과 컴퓨터 간의 커뮤니케이션을 더욱 효과적으로 만들고, 사용성이 더욱 뛰어난 컴퓨팅을 가능하게 하는 것을 목적으로 하는 분야가 "Context Aware Computing"이다.

Georgia Tech의 미래 컴퓨팅 연구소는 Context의 정의를 다음과 같이 내리고 있다.

'Environmental information or context covers information that is part of an application's operating environment and that can be sensed by the application. This typically includes the location, identity, activity and state of people, groups and objects.

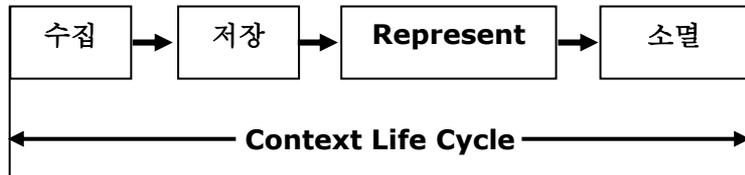
Context may also be related to places or the computing environment. Places such as buildings and rooms can be fitted with sensors that provide measurements of physical variables such as temperature or lighting. Finally, an application may sense its software and hardware environment to detect, for example, the capability of nearby resources'.

이러한 정의를 기반으로 'Context Aware Service'를 정의한다면, '유저가 입력한 정보와 상황정보들이 결합되어 사용자가 처한 Situation에 맞게 유저가 원하는 수준으로 조정되어 제공되는 Smart 서비스' 라고 정의 할 수 있겠다.

Context Aware Service 방법론

Context Aware Service를 실제로 구성해가는 방법론을 살펴보면 크게 4 스텝들이 있다고 볼 수 있으며 그 4스텝은 Context 정보의 수집, Context 정보의 저장, Context Represent, 그리고 Context의 소멸이다. 이 전체 스텝을 Context Life Cycle이라 부른다.

<Context Aware Service의 구성 요인>



"Context 정보의 수집, Context 정보의 저장, Context Represent, 그리고 Context의 소멸이다. 이 전체 스텝을 Context Life Cycle이라 부른다."

1.수집 단계: 앞서 기술한 context의 정의를 다시 살펴보면 'application에 의해 감지된 작동환경'이라는 내용을 볼 수 있다. 즉 context의 수집은 사용자와 맞닿아 있는 device 혹은 application에서 얻어 낼 수 있는 사용자의 작동환경 정보를 모으는 것이라 할 수 있다. 이러한 수집 과정에는 사용자로부터 얻어 낸 정보에 application 혹은 device 자신이 덧붙일 수 있는 정보를 첨삭하여 의미 있는 완전한 하나의 정보형태로 가공하는 것도 포함된다.

따라서 단순한 sensing이상의 역할이 수행되는 과정이라고 할 수 있다. 예를 들어 GPS와 같은 장치가 sense할 수 있는 것은 위도, 경도의 위치정보 혹은 이동 벡터정보이다. 여기에 location과 관련되는 GIS의 정보를 합쳐서 사용자가 병원에 있다거나 도로를 주행 중이라는 context 정보를 얻어 내는 것이다.

다시 말해서 '수집단계'는 sensor로부터 얻어 낸 정보와 그 sensor와 연관되는 object (객체)와의 상관관계 분석을 통한 정보취합의 단계라 정의된다.

"Context의 저장은 사용자의 status 등의 environmental information과 대상 객체 연관도의 다차원 matrix 형태로 구성되며 이러한 정보로 거대한 network을 형성하게 된다."

2.저장 단계: context 정보의 저장은 중앙집중식의 저장형태로 구성되어야 한다. 즉 여러 경로로 유입된 사용자의 context 정보를 각 수 집단말과는 분리해서 저장하게 된다. 이는 사용자의 context 정보의 독립성을 유지하고 저장된 context를 서로 다른 단말에서도 사용할 수 있도록 하기 위함이다.

Context의 저장은 사용자의 status 등의 environmental information과 대상 객체연관도의 다차원 matrix 형태로 구성되며 이러한 정보로 거대한 network을 형성하게 된다.

3.Represent 단계: Context의 representation 단계는 실제 정보나 서비스가 제공되는 단말에서 사용자의 context에 맞는 서비스가 제공될 때 저장된 context 정보에서 현재 사용자와의 접촉 단말이 최적의 선택을 할 수 있도록 기반 context 정보를 제공해 주는 단계이다.

이러한 과정은 사용자의 explicit한 요구에 응답하는 '사용자 주도형'이 될 수 있고, 시스템이 먼저 반응하는 '시스템 주도형'이 될 수도 있다. 사용자 주도형과 시스템 주도형의 차이는 일반 포탈 서비스에서 banner와 PIMS 혹은 reminder의 차이와 같다.

배너와 같은 것은 시스템 주도형의 초기 형태인데 반해 PIMS나 reminder는 사용자 주도형의 초기 형태라 할 수 있다. 기반 context 정보의 제공은 통합되어 저장된 context들 중에서 하나 혹은 n개를 골라내는 search 문제이다.

4.소멸 단계: context는 한번 생성된 것이 영속적으로 사용자의 주변에 존재하는 것이 아니고 context의 속성에 따라, 시간에 따라, 혹은 다른 context에 따라 소멸되거나 다른 형태로 변화하게 된다. 사용자가 끊임없이 사고하고, 행동함에 따라 주어진 상황이 바뀌는 것은 당연하다고 할 수 있다.

이러한 소멸과 관련한 context의 종류는 다음과 같은 3가지의 종류가 있다.영속성 context, 소멸형 context, 변화형 context로 나눌 수 있다. 영속성 context로 가장 대표적인 것이 사용자 자체의 inherent한 속성 혹은 고유속성이라고 할 수 있다. 생년월일과 같은 값이 대표적인 것으로 사용자 profile정보라고도 지칭될 수 있다.

소멸형 context는 시간 혹은 타 context에 따라 자체가 소멸하는 context로서 수집된 context가 일정 조건이 되거나 하면 소멸하게 되는 context이다. 예를 들어 병원에 입원하고 있는 context는 퇴원을 하면서 사라지게 되고 대신 병원입원 경력이 영속성 context로 남게 되는 것이다. 변화형 context는 수집된 context가 자체적 혹은 다른 context로 인해 계속 변화하게 되는 것으로 위치정보 같은 것이 대표적인 예라고 할 수 있다.

Context Aware Service의 예

Context는 context를 갖는 사람과 같은 대상객체와 이를 보고 감지할 수 있는 감지객체를 갖게 된다. 그리고 감지객체에서 감지하는 대상객체의 context는 확연하게 감지할 수 있는 context 정보와 그렇지 않고 감지객체의 문제 혹은 고유한 속성으로 인해 감지할 때 확실해 질 수 없는 context 정보가 있다. 이것을 각각 Explicit context와 Implicit context라고 한다.

Explicit context는 수집된 context의 속성을 확연히 알 수 있는 context를 의미하고 Implicit context는 수집된 context에 대한 속성을 확연히 알 수 없는 context이다. explicit와 implicit를 구분하는 요소는 context 자체의 속성과도 관련이 있지만, 감지 device와도 관련이 있게 된다. 예를 들어 위치정보의 경우 GPS 장치가 일반화되기 이전에는 알 수 없는 막연한 context였지만 GPS가 mobile device에 장착되어 일반화 된다면 이는 explicit context가 된다.

이러한 경우 사용자가 병원에 있다라는 정보는 explicit context가 되고, 어떤 목적으로 왔는지는 여전이 implicit한 context가 되게 된다. 만약 예약관리 시스템과 연동이 되고 이 정보를 context로서 사용할 수 있다면 대상객체가 병원에 가 있는 의도 자체도 explicit하게 되어 처리할 수 있게 된다.

Ubiquitous Internet이 Internet의 궁극적인 지향점이 되고 그런 시대가 온다면 사용자의 많은 context가 explicit하게 처리될 수 있었음을 의미하고, explicit context에 기반한 implicit context의 사용도 점점 정확도를 갖게 될 것이다. Explicit context는 그 속성상 "확실한"(deterministic) 속성을 갖는다 따라서 이런 explicit context에 기반 한 서비스는 rule과 같이 선행조건과 후행행동을 명확하게 매칭하여 정의하고 진행할 수 있다.

그러나 implicit context의 사용은 자칫 서비스의 quality를 떨어뜨릴 수 있는 위험이 있는 반면 새로운 경험을 사용자에게 제시할 수 있는 가능성도 열게 된다. 예를 들어 Amazon.com의 "이 책을 산 사람들이 많이 사는 다른 책"의 개념과 같은 초기적인 implicit context 사용은 사용자에게 다른 각도의 효용성을 제시해 줄 수 있다. 즉 implicit context는 위험한 선형적 rule의 사용이 아닌 pattern 분석과 같은 형태에서 확률형태의 추론을 하는 형태까지의 모양이 가능하리라 생각한다.

그러나 확연한 결정형이 아닌 "추천"형으로 사용자에게 제시하게 될 것이다. 여기서 Georgia Tech의 미래 컴퓨팅 연구소의 Daniel Salber, Anind K. Dey and Gregory D. Abowd의 Cyber Minder를 context aware service의 예로 한번 살펴보자.

"Explicit context는 수집된 context의 속성을 확연히 알 수 있는 context를 의미하고 Implicit context는 수집된 context에 대한 속성을 확연히 알 수 없는 context이다."

1. Cyber Minder-Explicit Context 단계

Explicit Context를 활용한 Context Aware Application의 예로 Cyber Minder를 소개한다. Cyber Minder의 특성을 보면 아래와 같다.

- Use of rich context for specifying reminders, beyond simple time and location and for proactively determining when to deliver them;
- Ability for users and third parties to submit reminders;
- Ability to create reminders using a variety of input devices;
- Ability to receive reminders using a variety of devices, appropriate to the user's situation;
- Use of reminders that include both a signal that something is to be remembered and a full description of what is to be remembered;

< Example Reminders >

Situation	Natural language description	CyberMinder description
Time	9:45 am	Expiration field: 9:45 am
Location	Forecast is for rain and bob is leaving home	City=Atlanta, WeatherForecast=Rain Username=Bob, Location=Bob's front door
Co-location	Sally and colleague are co-located	Username=sally, Location=*1 Username=Bob, Location=*1
Complex #1	Stock price of X is over \$50, Bob is alone and has free time	stockName=X, Stockprice>50 username=Bob, Location=*1 Location=*1, OccupantSize=1 Username=Bob, Freetime>30
Complex #2	Sally is in her office has some free time, and her friend is not busy	Username=Sally, Location=Sally's office Username=Sally, Freetime=60 Username=Tom, ActivityLevel=low

<Source: Future Computing Environments (FCE) Group>

2. Implicit Context 단계

여기서 한발 더 나아간 것이 **implicit context** 단계인데 앞서 예로 든 **Amazon.com**의 예에서도 볼 수 있는 것으로 **social filtering** (혹은 **collaborative filtering**)과 같은 초보적인 형태가 서비스로 구성 되어 현재 제공되어 있다. 이 서비스의 구조를 보면 **A**라는 책에 관심이 있다라는 **context**를 알게 되면 이 서비스는 그 책과 관련 있는 책을 제공하는 (정확히는 추천하는) 구조를 갖는다.

그러면 여기에서 서비스의 감지객체가 과연 사용자의 **context** 만을 보고 있는 것인가? 하는 질문에 도달하게 된다. 답은 감지객체가 사용자의 **context**뿐만 아니라 대상객체의 본질을 설명하는 속성도 보고 있어야 한다는 것이다. 그래야 **implicit context**를 통한 서비스가 가능해지는 것이다.

"즉 사용자의 **explicit context**를 통해 **implicit context**에서 사용 가능한 **context**를 뽑아내고 이를 통해 서비스를 제공할 수 있게 된다. 이러한 서비스는 대상객체인 영화관과 관련한 **context** 수집과 밀접한 연관을 가지고 있음을 알 수 있다."

즉, 내가 2시에 병원에 간다는 약속을 입력해 놓았다면 대상객체의 본질이 필요 없는 것이지만, 병원에 갈 시간을 알려주는 것은 **Explicit**한 단계이고 **Implicit**한 단계가 되려면 병원에 가려는 목적을 추론해 내야 하는 것이다. 이것은 병원이라는 대상객체의 속성을 이해해야 하는 것이다.

자, 이번에 또 다른 설명이다. 현 상황에서 **location** 정보를 **explicit**하게 사용할 수 있다고 하자 그러면 어떤 **implicit context**를 사용할 수 있는지를 예로 들어보면 사용자가 영화를 하나 예약했고 이 **context**는 이미 저장되어 있다.

그리고 지금 그 시간이 다가오고 있다면 일차적으로 예약의 **alarm**을 제공할 수 있다. 또한 지금 이동 중이라면 **navigation system**에 목적지가 영화관으로 된 상태에서 가장 빠른 길을 표시해 줄 수 있고 주차장의 상황과 예상 도착시간을 사용자의 개입 없이 모두 알려줄 수 있다.

즉 사용자의 **explicit context**를 통해 **implicit context**에서 사용 가능한 **context**를 뽑아내고 이를 통해 서비스를 제공할 수 있게 된다. 이러한 서비스는 대상객체인 영화관과 관련한 **context** 수집과 밀접한 연관을 가지고 있음을 알 수 있다.

지금까지 **Context Aware Service**에 대해 살펴 보았다. 국내에서도 학계에서는 오랫동안 이미 **context**와 관련한 의미 있는 리서치들이 많이 진행되고 있는 것으로 듣고 있다.

"정보 기기들이 많아지면 서비스의 차별 점을 둘 수 있는 부분은 **Technology Issue**보다도 **Context**가 될 것으로 본다."

Davidndanny이번 글을 준비하면서 이 분야에서 선구적인 연구를 지속적으로 진행해 오고 계신 연세대학의 김진우 교수님(**Associate Professor, Human Computer Interaction School of Business**)을 만나 뵈 자리에서 "정보 기기들이 많아지면 서비스의 차별 점을 둘 수 있는 부분은 **Technology Issue**보다도 **Context**가 될 것으로 본다."라는 말씀을 들었다.

요즘 유무선 통합논의가 한창인 이때에 비교적 정보가전 **HW**쪽이 강한 한국의 상황을 고려할 때 기기들이 풍성해지는 만큼 서비스의 차별화를 이끌기 위해 **Context**에 사업자들이 더욱 많은 관심을 기울여야 한다는 말씀으로 생각해 볼 수 있겠다.

한국은 '모바일 인터넷 사용인구'와 '광대역 사용자 기반'에서 세계 최고의 환경을 가지고 있다고 할 수 있다. 이런 풍성한 사업환경에서 사업자들이 깊게 고려해야 할 점들이 무엇일지 **Davidndanny**가 생각하는 몇 가지 결론들을 소개하면서 글을 맺고자 한다.

Conclusion

유무선 통합서비스환경은 곧 고객들의 **Context**를 이전 보다 더욱 잘 이해할 수 있는 환경을 만들어 준다는 것이고 이것은 사업자들에게 사용자들의 **Context**를 더욱 깊이 이해한 킬러서비스를 내놓아야 한다는 의무를 지우고 있는 것

1. 2002년 현재 우리가 보고 있는 유무선 통합 과정은 인터넷이 **Ubiquitous Internet**으로 발전해 가는 과정 중의 한 관문이다. 이러한 발전이 사업자들에게 요청하는 바는 유무선 통합서비스환경은 곧 고객들의 **Context**를 이전 보다 더욱 잘 이해할 수 있는 환경을 만들어 준다는 것이고 이것은 사업자들에게 사용자들의 **Context**를 더욱 깊이 이해한 킬러서비스를 내놓아야 한다는 의무를 지우고 있는 것이라고 이해해야 한다.

2. **Context Aware Service**는 제대로 제공하기도 쉽지 않을 뿐만 아니라 실제로 상당한 수준의 기술적인 세련됨을 요구한다. 하지만 이런 서비스가 제대로 진행된다면 결과는 상당히 엄청난 것이다. 아직은 **Visionary**들이 말하는 이야기에 지나지 않는다고 생각하는가?

한국에서도 이미 잘 알려진 기술기반이 좋은 몇몇 인터넷 업체들은 이런 **Context Aware Service**를 장기적인 계획 하에 실행에 들어갔다. 만약 당신의 회사에서 위에서 한 이야기들을 한번도 듣지 않았다면 이미 당신의 회사는 상당히 가까운 미래의 경쟁에서 밀리고 있다는 것을 의미한다.

3. 기술 역량이 관건이다. 웹을 통한 콘텐츠 서비스가 주를 이룰 때와 비교할 때 **Context Aware Service**는 많은 기술적인 도전을 요구한다. 서비스를 기획하는 기획자들에게도 더욱 세련된 기획 테크닉들을 요구한다.

실상 상당히 큰 규모의 투자가 필요한 서비스이기 때문에 초기 시나리오 구성에서부터 프로토타입을 만들어 실수 없는 서비스를 만들어 가는데 기술 역량확보는 필수적이다.