

PLDI'17 일지

이준영

이 글은 제가 PLDI'17 (스페인 바르셀로나, June 17 - June 25) 에 참가하면서 겪은 일 & 느낀 점들을 기록한 것입니다.

금요일 - 토요일 (6월 17 - 18일)

학회는 스페인 바르셀로나에서 열렸는데, 인천에서 도시로 가기 위해서는 도하공항을 한 번 경유해야 했다. 비행기 중에 직항이 있기는 했으나 날짜가 PLDI일정에 맞지 않았다. 금요일 저녁까지 있었던 일정을 마무리 하고 서울역에서 AREX를 타고 인천공항으로 이동했다. 비행기의 이륙 시간은 오전 1시 20분이었고, 인천공항에서 도하공항까지의 비행 시간은 대략 10시간 정도였다. 도하 공항에서는 3시간 동안 머물렀는데, (비행기 안도 그랬지만) 도착하고 나서도 스크립트 외우느라 조금 정신이 없었다.

오후 1시 반에 바르셀로나 공항에 도착했다. 바르셀로나는 꽤 더웠는데, 후에 만난 분 설명에 따르면 최근 며칠은 바르셀로나에서 정말로 시원한 기간 중에 하나였다고 한다.

숙소로 이동하기 위해선 첫째로 A1 공항 버스를 타고, 에스파냐 광장으로 이동한 다음 지역 버스를 타야 했다. 에스파냐 광장으로 이동한 다음 점심을 먹을 계획이었다. 원래는 용주가 생각해 둔 어떤 맛집을 찾아가려고 했었는데 에스파냐 광장에 포장마차 같은 곳이 보이길래 현지 음식도 먹어볼 겸 거기서 점심을 한 번 질러 보았다. 포장마차 메뉴 옆에 사진이 붙어 있었는데, 가장 맛있어 보이고 && (내 생각에) 가장 실패하지 않을 것 같은 메뉴를 적당히 시켜 보았다. 5유로짜리 메뉴가 있었고 (참고로 1유로는 1200원 정도이다) 10유로 짜리 메뉴가 있었는데, 난 10유로짜리 메뉴 하나 + 생맥주 하나를 주문했다. 나에게 나온 것은 훈제 고기 + 소세지 + 알 수 없는 과일을 구운 것이었는데 나쁘진 않았다.



공항에 있던 '화물을 얼마나 오래 기다려야 하셨나요?' 설문조사 기계. 인터페이스의 간결함이 마음에 들었다. (스페인과는 별로 상관이 없는 사진이다)



에스파냐 광장(사진이 잘 나온 것). 배경에 있는 구조물들은 보이는 것보다 멀리 있고 덩치도 크다.

토요일 낮 3시 즈음 Castellnou hotel에 도착했다. 작은 호텔이었는데, 굉장히 쾌적했다. 냉장고 안에 생수가 있었는데 우린 이것이 공짜가 아닐까 라고 추측했다. (이 추측은 틀렸고 우린 퇴실할 때 20~30유로 가량을 지불해야 했다) 고대하던 샤워를 한 다음 노트북을 조금 쳐다보다가, 시차때문에 바로 잠들었다. 바르셀로나에 낮 4시는 한국으로 밤 11시이다. 밤 늦게 새벽 2시 경에 일어나서 다시 스크립트를 보면서 연습을 했다.

일요일 (6월 19일)

아침 일찍 학회장으로 가서 등록을 했다. 학회가 열리는 곳은 <Universitat Politècnica de Catalunya> 이었다. 지도는 아래와 같았다. 알록달록하게 색칠된 건물들이 학회가 열린 곳이었다.



캠퍼스 사진. 참고로 위 사진은 영어가 통하지 않는 택시 드라이버에게 집으로 돌아가고 싶다고 표현할 때 요긴하게 사용이 되었다.

이번 PLDI 는 다른 PL 학회와 같이 열렸고, 발표도 한 번에 여러 곳에서 진행이 되었다. 건물들이 분산되어 있어서 원하는 세션에 참여하거나 점심식사 장소로 이동하기 위해선 많이 걸어야 했다.

Journey to Find Bugs in JavaScript Web Applications in the Wild

아침 8시 반에 SOAP 학회 발표를 들으러 갔다. 카이스트 PLRG 연구실의 류교수님께서 키노트 발표를 해주셨는데, 연구실에서 개발하고 있는 JavaScript 분석기를 소개해주고 그와 관련된 이슈들을 강의해 주셨다.

강의를 시작하시면서 JavaScript는 (웹 사이트뿐만이 아닌) 광범위한 곳에서 사용되는 언어이며, 그 예시로 스마트 TV에서 에러가 나서 "undefined" 키워드가 하단에 가득 차 있는 사진을 보여주셨다. 재밌는 것은 해당 사진은 연구실에 있는 TV를 직접 찍으신 것이라 한다. 또한 JavaScript는 언어 자체의 기능이 많기 때문에 분석기의 존재가 프로그램 개발의 효율에 큰 도움을 줄 수 있다고 한다. 예를 들어 JavaScript에선 객체의 attribute가 dynamic하게 추가 및 삭제가 가능한데, 프로그램 분석 기술을 사용하면 이와 관련된 오류를 효과적으로 잡아줄 수 있다는 것이다.

기존의 JavaScript 분석기들은 아직 충분히 강력하지가 않으며 잘못된 보고를 하는 경우 또한 종종 발생한다. JavaScript에는 두 가지 (혹은 더 있을 수도 있으나) equality operator 가 존재하는데 바로 "=="와 "==="이다. "=="는 abstract equality comparison 으로서 implicit하게 casting한 다음 비교할 수 있다. "==="는 strict equality comparison 으로서 casting하지 않고 비교를 한다고 한다.

기존의 분석기 중 하나인 JS Lint는 "=="의 사용을 "==="로 바꾸는 것을 권고하는데, 이 때문에 위키피디아 페이지에 introduce 된 버그가 있었다고 한다. 분석 결과 "=="를 써야 하는 케이스가 있었는데, "==="가 되어 있었다는 것이다. 이를 Wikipedia에 클레임했고, 해당 페이지를 수정했으나, 나중에 보니깐 또 다시 == 로 바뀌어 있었다고 한다.

JavaScript 는 다른 언어와 결합해서 자주 사용될 수 있으며, 그 예 중 하나는 안드로이드의 하이브리드 애플리케이션이라고 한다. Java와 JavaScript간에 통신을 위해서는 API를 사용해야 하는데, 존재하지 않는(혹은 타입에 맞지 않는) 멤버에 접근하려고 시도하더라도 이는 자바의 컴파일 에러에 잡히지도 않고 기존의 JavaScript 분석기에도 잡히지 않는다. 이렇게 여러 언어에 걸친 분석에 대해서도 잘 해결해 주려고 하는 시도가 있었다고 한다.

그 이외에 여러 이슈에 대해서도 이야기를 해 주셨다.

왜 사람들은 JavaScript를 쓸까? 아마 그 이유는 배우기 쉽고 (JavaScript를 가르치는 곳이 많음) 프로그램을 실행한 결과가 눈에 잘 보이고 (웹브라우저로 잘 확인할 수 있음) 라이브러리 또한 많이 제공되고 있기 때문이 아닐까 한다. 언어의 '아름다움'은 사실 빨리 애플리케이션을 개발해야 하는 사람 입장에선 그다지 중요한 것이 아닐지도 모른다.

한동안 JavaScript 가 계속해서 쓰일 것이라면, JavaScript를 대체할 새로운 언어를 개발하는 것보다는 언어를 쓰는 사람이 겪는 어려움을 assist해주는 툴을 빠르게 개발하는 것이 세상에 주는 도움이 더 클 수도 있겠구나 라는 생각이 들었다. DeepCode란 프로젝트 또한 JavaScript를 대상으로 하고 있는데, 이것을 쓰면 주어진 자바스크립트 코드 내의 변수에 사람이 이해할 수 있는 이름을 자동으로 부여해준다고 한다. 조금 다른 이야기이긴 하지만 Facebook에서 legacy php code를 사용하는 이야기를 들어본 적이 있는데, php보다 더 괜찮은 언어로 개발을 진행하고 싶었으나 이미 php로 개발된 코드가 너무 많고 또 php를 할 줄 아는 사람이 너무 많아서 실패했다고 한다.

그리고 사람들이 JavaScript를 쓰는 이유가 위와 같다면, 향후에 JavaScript나 php처럼 널리 쓰이는 새로운 언어를 만드려면 어떻게 해야하는지 또한 대충 추측할 수 있을 것이다. 만약 program synthesis의 세상이

온다면 스펙을 선언할 때 쓰는 언어가 우후죽순처럼 새로 나올 것이고 그 때에도 위를(배우기 쉽고, 눈에 잘 보이고, 라이브러리 많고)을 만족하는 언어가 결국엔 사람들에게 많이 쓰이지 않을까.

키노트 다음 강의를 하나 듣고 나서 밖으로 나와 발표 준비를 했다.

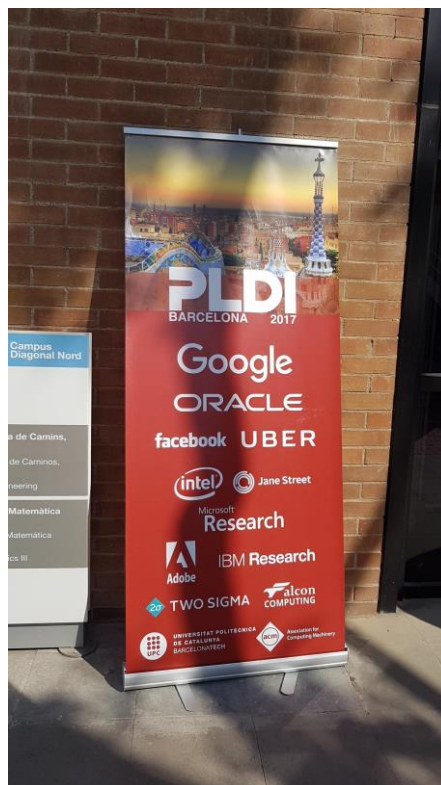
한 가지 기억에 남는 것은 바르셀로나는 해가 늦게 지는 도시였다는 것이다. 밤 9시까지도 여전히 하늘이 밝았던 기억이 난다.

월요일 (6월 20일)

오전에 학회장에 와서 Invited Speaker인 Marín Abadi의 <TensorFlow considered as a programming language> 토크를 아주 잠깐 들었다. 개인적으로 꼭 듣고 싶던 토크였는데, 스크립트를 보느라 뒷 부분만 잠깐 들었다. 후에 유튜브에 올라오면 다시 봐야겠다고 다짐했었는데 이 토크는 유튜브에 올라오지 않아서 결국 후회가 남게 되었다.

공저자 중 한 명인 Nuno P. Lopes 를 만나 인사하고, 나는 스크립트를 연습했다.

저녁은 교수님이랑 같이 학교 근처에서 먹었는데, 교수님으로부터 현재 딥러닝이 뜨면서 실리콘밸리에서 컴파일러를 잘 하는 사람들에 대한 수요가 있다는 소식을 들었다. 왜냐하면 결국엔 러닝 속도가 중요하기 때문이라고 한다. 저녁을 먹고 ppt 및 스크립트 리뷰를 했다.



학교 내에 세워져 있던 PLDI 패널. 놀랍게도 ORACLE이 있었다.

아마 SQL 처리하는데 JIT가 쓰여서가 아닐까 하고 추측해 본다.

(지훈이형: Oracle JAVA가 있기 때문이라고 하셨는데 이것이 더 major한 이유인 것 같다)



매달려 있는 풍선을 보니 학부때 ACM-ICPC 를 준비하던 기억이 났다.

화요일 (6월 21일)

오전 5시경에 일어나서 스크립트 연습을 했다. 아침에 학회장소에서 교수님을 뵈어서 1분 pitch talk준비를 했다.

Taming Undefined Behavior in LLVM

당일 발표 일정은 첫째로 오전에 1분 pitch talk가 있었고, 둘째로 오후 4시 45분에 25분짜리 본 발표가 있었다.

발표 이전엔 pitch talk 내용을 까먹진 않겠지.. 하면서 긴가민가 했었는데, 그게 현실로 일어나 버렸다. 1분 pitch talk를 하는데, 앞에 두 문장 정도 말하고 나서 머리가 새하얘지는 경험을 했다. 그 뒤에는 준비한 것을 모두 말하지 못하고 대충 얼버무린 다음에 내려와 버렸다. 너무나도 황당한 경험이었다.

하는 수 없이 본 발표때는 노트북을 보고 발표를 진행하였다. 지도교수님 및 Nuno앞에서 노트북을 보고 발표를 하는 맹연습을 했는데, 처음엔 발표가 너무 빨랐다는 평이 있었으나 연습을 거듭하면서 적절한 속도로 수렴했던 것 같다.

다행히 본 발표는 성공적으로 마쳐질 수 있었으나.. 다음에는 1분 pitch talk때와 같은 이런 실수를 하지 말아야 겠다는 생각이 들었다. 질답 시간이 되어서야 방 안에 사람이 정말 많았구나 라는 것을 알 수 있었다. 완벽하진 않았어도 사람들의 질문에는 적당히 어떻게든 대답할 수 있었다.

저녁 식사

저녁에는 구글에서 제공하는 저녁 자리에 참가했다. 이동할 때 교수님, Nuno 및 마이크로소프트에서 일하시는 어떤 분과 같이 지하철을 타고 움직였다. 해변가에 있는 어떤 호텔의 옥상에서 뷔페 형식으로 열렸었는데, ISMM 에 논문을 낸 친구와 많은 이야기를 했다. 그 친구가 연구한 내용은 Erlang언어에 기능을 추가하면서 garbage collection 기능을 유지시키는 것이었다. Erlang은 language-level에서 메모리의 separation(표현이 적합한 지 모르겠다)을 보장할 수 있다. 각 쓰레드가 보는 메모리를 완전히 구분할 수 있는데, 이것을 활용하면 garbage collection이 더 이득을 볼 수 있다. 하지만 특정 기능을 추가하면 이것이 깨질 수가 있는데, 이것을 어떻게 잘 추가했는지에 대한 이야기이다.

재미있는 것은 이것을 하면서 Azul system이야기가 나왔는데, Azul system은 내 논문의 공저자 중 한 명이 일하고 있는 곳이다. 이 회사에서 하는 일은 Java의 concurrent garbage collection 를 구현하는 것이다. 이 친구는 성능 평가를 할 때 이 회사의 제품과 성능 비교를 했다는 것이다. 내가 Azul system에 공저자가 있다, 아마 Azul system이 LLVM에 garbage collection 구현해서 쓸 것이다 하고 이야기하니까 자기는 그 사실을 한 달 전에야 알았다고 한다.

예전에 한 때 LLVM의 garbage collection을 공부할 때가 기억이 났다. Garbage collection 을 IR단위에서 추상화 시켜서 표현할 때에는 대단히 재미있는 이슈들이 등장한다. 예를 들어, 포인터를 정수로 바꾸는 연산은 GC가 있는 상황에선 잘 정의가 되지 않는데, GC가 메모리 블록을 다른 곳으로 옮겨버릴 수 있기 때문이다.(사실 포인터 - 정수 간 캐스팅은 구지 GC가 없어도 C에서든 LLVM IR 에서든 잘 정의가 안 된다. 자세한 이야기는 나중에..)

많은 이야기를 나누고 대략 밤 10시쯤 호텔에서 출발했는데, 집으로 돌아가는 길이 잘 기억이 나지 않아 택시를 잡아서 이동을 했다. 이 때 위에 있는 학회 장소 지도를 노트북으로 펼쳐서 보여주었는데 성공적이었다.

수요일 (6월 22일)

발표가 끝나면서 여유가 생겨 여러 발표를 재미있게 들을 수 있었다. 아래는 내가 들었던 발표 중 두 개의 내용을 요약한 것이다.

Probabilistic Programming and Inference Compilation, or, How I Learned to Stop Worrying and Love Deep Networks

오전에 옥스퍼드 대학교의 Frank Wood라는 분의 특을 들었다. Probabilistic programming이 무엇인지는 잘 몰랐으나, 한 가지 관련이 있을 것 같았던 경험은 학부때 머신 러닝 수업 숙제를 구현했던 것이었다.

그 당시 머신러닝 숙제를 하면서 (여러 기계학습 알고리즘을) C++로 한 번 작성해보고 싶었다. Matlab으로도 작성해 볼 수 있었을 것이지만, 무언가 C++의 욕증함과 tricky함을 ML숙제를 구현하면서도 한 번 느껴보고 싶었다. 행렬 곱셈 및 eigenvector 구하는 연산은 OpenCV에 있는 라이브러리들을 이용해 구현했으나, 내가 수식을 섬세하게 구현해야 하는 것도 많았던 것으로 기억한다. (예: gaussian distribution이 계산 중간에 등장할 경우. 어려운 일은 아니었지만 실수할 여지가 있었다)

그리고 이 talk의 도입부에선 probabilistic programming을 소개하면서 이 때의 불편함 및 verbose함이

probabilistic programming이란 것으로 잘 해결이 된다는 이야기를 했던 것 같다. 예를 들어 이 언어를 사용하면 probabilistic programming에서 디버깅이 쉬워진다고 한다. 실제로 내가 ML 과제를 할 땐 디버깅이 쉽지가 않았던 기억이 있는데, 그 이유는 내가 수식을 잘못 적었다고 해서 그것이 즉각적인 에러로 나타내지진 않았기 때문이다. 만약 확률과 관련된 assertion을 수식꼴로 정의할 수 있다면 이런 종류의 디버깅이 쉬워졌을 것이다.

도입부에서 Frank Wood가 Probabilistic Programming이 ML보단 PL 분야에 오히려 더 큰 영향을 주고 있는 것 같고 이야기를 했다(근데 그러면 안 된다고 생각한다.) 특에서는 Probabilistic Programming가 ML + STATS(statistics community) + PL (compilers, semantics, transformations) 이렇게 세 가지 분야에 걸쳐져 있다고 소개를 했다.

Probabilistic program이 가지고 있는 두 가지 독특한(그리고 중요한) 기능들은 다음과 같다.

(1) Ability to draw values at random from distribution

(2) The ability to condition values of variables in a program via observations.

사실 (1)의 경우는 일반 program도 라이브러리를 쓰면 가능한 기능인데, 아무래도 라이브러리 수준이 아닌 language 수준에서 이 기능이 있다는 의미인 듯 하다.

(2)의 경우 probabilistic program만의 독특한 기능이다. 이런 류의 계산을 probabilistic inference라 하고 이 때 찾는 확률 함수를 posterior라고 한다. 주로 이 posterior 를 찾는 것이 기계학습에서 하는 일이라고 한다.

하지만 여기서 challenge가 발생하는데 바로 probabilistic programming의 표현력이다. Probabilistic program이 표현할 수 있는 것이 수학적인 것에 머무르지 않고 일반적인 프로그램이 수행할 수 있는 결과까지 포섭할 수 있어야 그 용도가 무궁무진하게 늘어날 수 있다는 것이다.

그 예제로 물리 시뮬레이터를 사용해야 결과를 알 수 있는 어떤 함수를 보여주었다. 어떤 함수가 나타내는 확률을 공을 던졌을 때 어떤 바구니 안에 들어갈 확률로 정의해보자. 만약 공의 시작점과 바구니 사이에 막대기를 둔다면 공이 바구니에 들어갈 확률은 달라질 것이다. 이 시점에서 probabilistic programming으로 풀고 싶은 것은, 공이 바구니 안에 들어갈 확률이 20%가 되도록 막대기를 잘 배치하는 것이다. 이 과정에서 해당 언어로 작성한 program이 확률과 상관 없는 외부 라이브러리 (2D 물리 시뮬레이터)까지 끌어다 쓸 수 있다면 분명히 많은 도움이 될 것이다.

(분량도 많고 일부 이해가 되지 않은 것도 있어서, 뒷 부분은 생략)

Probabilistic programming이 무엇인지 대충 알게 된 것만으로도 얻어간 것은 있다고 생각한다. 개인적으로 이 언어가 deep learning에 쓰일 수 있는것인지 하는 궁금증이 들었다. 후에 좀 더 이해를 해보고 싶다.

Responsive Parallel Computation: Bridging Competitive and Cooperative Threading

이 talk에서는 thread를 competitive하게도 스케줄링 하고 cooperative 하게도 스케줄링 하고 싶을 때 이것을 언어 레벨에서 어떻게 명시할 수 있는지에 대해 다루었다. Cooperative scheduling이 필요한 경우는 성능을 위해 특정 thread 에게 자원을 몰아주어야 할 때이고, competitive scheduling이 필요한 경우는 thread 수가 많아서 일일이 자원에 대한 접근 순서를 주기 힘들 때이다. 일반적으로 thread를 여럿 생성하면은 OS는 thread들을 competitive하게 schedule한다.

아이디어 자체는 정말 간단했는데, 바로 쓰레드를 생성할 때 foreground로 돌릴지 background로 돌릴지를 언어 레벨에서 명시해주는 것이었다. 그러면 한 가지 발생하는 challenge는 foreground thread가 background thread의 결과에 의존할 경우 foreground thread가 오랫동안 잠들게 되는 것인데, 이 talk 에서 제시하는 방법은 type checking을 이용해 이런 케이스를 처음부터 막는 것이다. 정확히 어떻게 막는지는 잘 이해하지 못했다.

실험을 위해 이것을 parallel version of SML에 구현했다고 한다.



점심 식사 장소. 거대한 하얀색 튜브이다.

오후 1시 경에 점심 식사를 먹었다. 뷔페 형식이었는데, 처음 먹을 땐 나쁘지 않았는데 비슷한 메뉴가 여러 번 나오다 보니까 조금 질렸던 것 같다. 점심 먹으면서 Microsoft Redmond에서 온 나이가 많아 보이는 분을 만났는데, 대단히 성품이 온화하신 분이였다. 그 분이 이야기하기로는, computer science 학회에서 지구 온난화를 고려해서 다른 학회와 장소 및 학회기간을 맞추고 있다고 한다. 근데 사실 이것이 지구 온난화에 얼마나 영향을 미칠까..? 그 분은 의문이라 하셨고 나도 잘 모르겠다고 이야기했다.

Integrated IoT Programming with Selective Abstraction

3시 반에 포항공대 코어랩 연구실의 아는 후배가 LCTES 학회에서 하는 발표를 들었다. 주제는 여러 디바이스에 들어갈 프로그램을 C++ 레벨에서 추상화를 잘 시키는 방법이었는데, 앞 부분 내용을 듣지는 못했지만, (만약 옳게 이해한 것이라면) 구현량이 상당했겠다는 생각이 들었다. 디바이스의 종류를 클래스로 정의하고 실제 디바이스 한 개를 객체 인스턴스 하나로 정의하는 것이 중요한 포인트이었던 것 같은데, large-scale IoT 애플리케이션을 만들 때 괜찮겠다는 생각이 들었다. 예를 들어 스마트 팩토리를 짓는다고

하면은 디바이스의 개수가 무지막지하게 많을 것이고, 이들이 서로 상호작용하는 과정 또한 대단히 복잡할 것인데, 스마트 팩토리가 동작하는 과정 자체를 하나의 프로그램으로 표현한 뒤에 그 프로그램을 디바이스들로 바로 incorporate할 수 있다면 스마트 팩토리의 구현이 많이 쉬워질 것이라는 생각이 들었다.(생각해보니 연구실에서 하는 PALS 프로젝트와 무언가 비슷한 바가 있다) 그렇다면 디버깅을 할 때에 발생하는 이슈는 없을까? 만약 이슈가 있다면 이것을 해결하는 것 또한 재미있을 것이다.

저녁은 포항공대 코어랩과 Furiosa 대표님과 함께 먹었다. Furiosa 대표님은 대구과학고 선배이셨고, GeorgiaTech에서 공부하셨다가 한국에 돌아와 스타트업을 하고 계신다고 한다. Google의 TPU와 같이 딥러닝에 특화된 칩을 만드는 것이 목표라고 하셨다. 샹그리아와 타파스를 먹었는데 정말 맛있었다.

목 - 금요일 (6월 23 - 24일) 및 맺음글

목요일에는 바르셀로나 투어를 했었고, 금요일에는 이런 저런 미루어 두었던 일을 했다. 호텔에서 파는 주문음식 중에 피자가 있었는데, 방에 들어올 때부터 꼭 먹어보고 싶었다. 그것을 금요일 저녁에 드디어 시켜먹어 보았다. 가격이 싸고 양이 많아서 좋았다.

이번 학회에서 한 가지 아쉬웠던 점은 여행 비용이다. 만약 비행기표를 일~이주일만 더 일찍 예약했다면 적어도 10만원 정도는 아낄 수 있었을 것이다. 하루에 사용한 돈이 5만원을 잘 넘지 않았으니, 10만원은 꽤 큰 돈이다. 그 이외에 호텔에 있는 물을 마셔서 돈을 3만원 정도 사용했었는데, 대형마트에서 물을 사먹으면 많이 저렴하다고 한다.

그리고 그것보다 더 아쉬웠던 것은 바로 내 발표준비 때문에 다른 세션에 거의 참가하지 못했다는 것이다. 이론적인 것을 다루는 발표 또한 들어보고 싶었는데 수요일엔 그런 특이 없어서 결국 듣지 못했다. 만약 다음에 학회에 참가한다면 학회에 오기 전에 미리 준비를 더 해둘 것이다. 이것이 가능하려면 평소에도 영어로 말하고 듣는 것에 무리가 없어야 할 것이다.