

WoT and HTTP Open API

Introduction to Internet and Web



부산대학교 정보·의생명 공학대학
정보컴퓨터공학부



Internet of Things

❖ What is the IoT (Internet of Things) ?

- [What is the Internet of Things ?](#) - YouTube
- [사물인터넷/Internet of Things](#) – Wikipedia

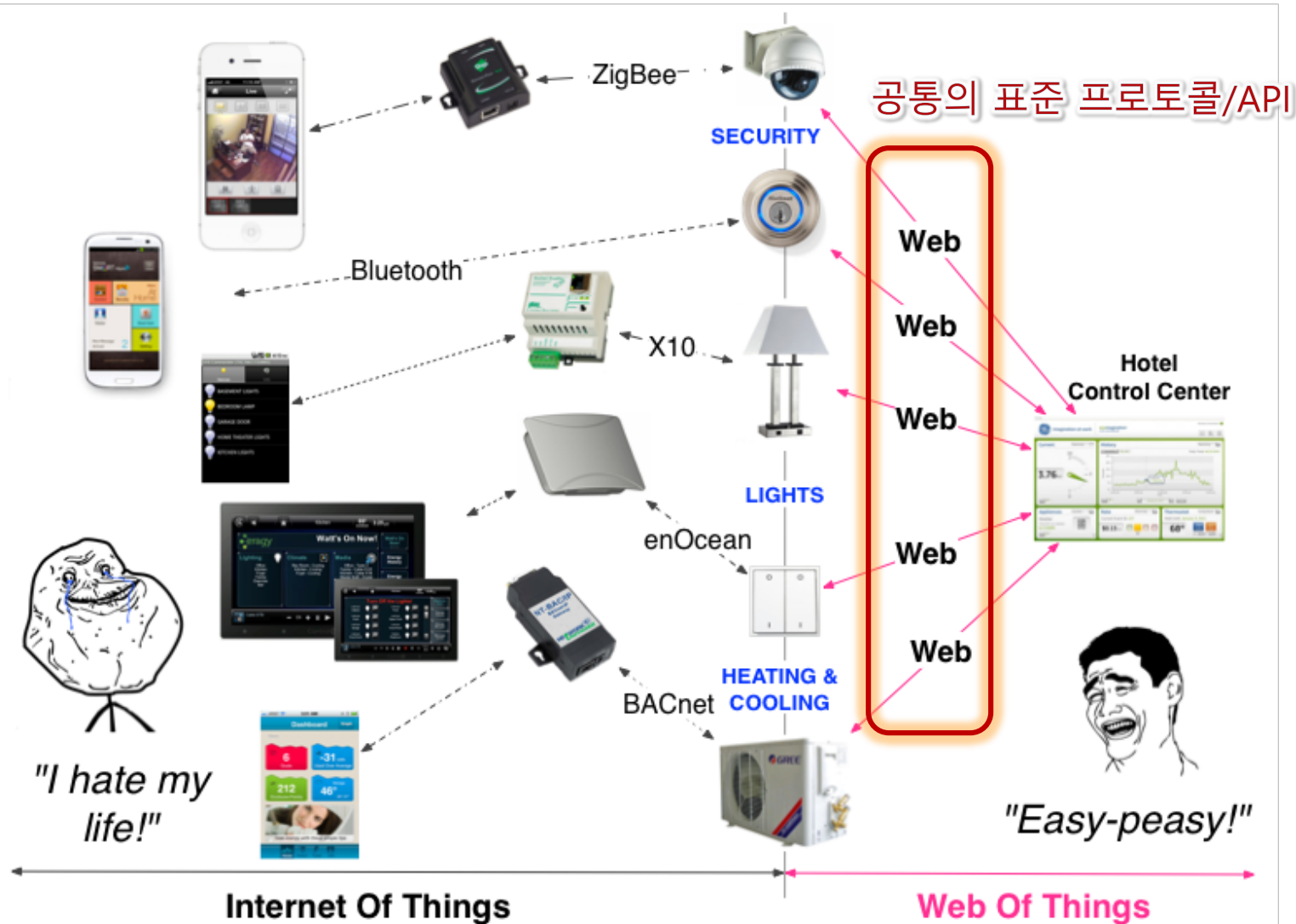


Internet of Things

❖ How can we realize IoT vision/services effectively and quickly ?

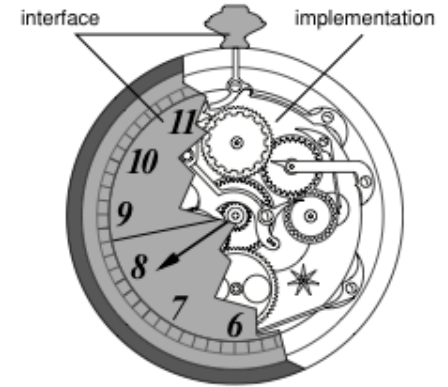
- 문제점 : 사물인터넷에 쓰이는 서로 다른 다양한 프로토콜과 기술들
- 해결책 : 공통의 표준 프로토콜과 개발 방법론
- IoT를 위한 새로운 표준 프로토콜과 방법론의 개발 ?
- 우리는 이미 성공적인 프로토콜과 방법론을 가지고 있음 → **HTTP/Web Standard**
- [Introduction to Web of Things](#) - YouTube
- [Web of Things](#) – Wikipedia
 - 참고 : 위 참고 자료에서는 IoT를 Network 계층의 기술로, WoT를 Application 계층의 기술로 구분하여 설명하고 있으나 모든 연구/기술자들이 이러한 시각에 동의하는 것은 아님.
WoT를 IoT의 일부로 해석하는 시각도 상당히 많음.

Web of Things



Source: Building the Web of Things: book.webofthings.io
Creative Commons Attribution 4.0

API

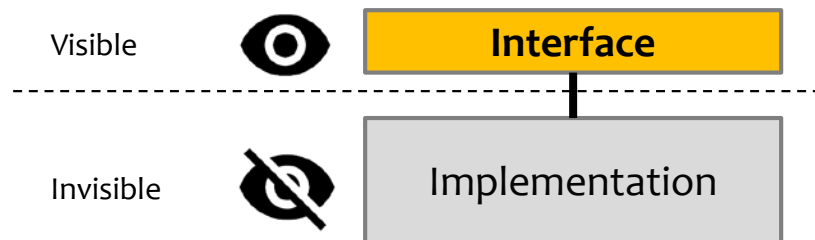


❖ What is an API ?

- API : Application Programming Interface
- [What is an API ?](#) – YouTube (#2)

❖ Separation of **Interface** from **Implementation**

- 기능/서비스를 제공하는 객체는 Interface와 Implementation으로 구성
- 기능/서비스를 이용하기 위해서는 Interface에 대한 이해만 있으면 충분함
- 예) 자동차와 운전
 - 가솔린, 디젤, 하이브리드, 순수 전기차 등 자동차 엔진들의 동작 원리와 차이점에 대해 이해하지 못해도 우리는 운전 핸들, 가속/브레이크 페달의 기능과 조작 방법만 이해하면 자동차를 운전할 수 있음
 - 자동차 엔진 : Implementation; 핸들, 가속/브레이크 페달 : Interface
- **Abstraction** in Computer Science : Separation of Interface from Implementation
 - Interface와 Implementation을 구분하는 것, Implementation에 독립적(Independent)인 Interface를 설계하는 것, 구현의 세부 내용을 숨기는 것 (Hiding the details of Implementation) S/W Engineering의 기본.



Interface vs. Implementation

가솔린 엔진과
디젤 엔진 작동 원리의 차이

가솔린 엔진		디젤 엔진
▶ 가솔린+공기 흡입	흡입	▶ 공기 흡입
▶ 가솔린과 공기를 10분의 1로 압축	압축	▶ 흡입한 공기를 20분의 1로 압축
▶ 압축한 가솔린과 공기를 전기불꽃으로 점화	팽창	▶ 압축된 공기에 경유를 분사해 자연 발화
▶ 연소가스 배출	배기	▶ 연소가스 배출



Website vs. Web Services



A website has a user interface or GUI .	A web service doesn't have a user interface
Websites are meant for use by humans .	Web services are meant for other applications to be interacted with over the internet.
Websites are accessed by using their GUI components – buttons, text boxes, forms, etc.	Web services are accessed by HTTP methods / Web API – GET, POST, PUT, DELETE, etc.
Websites are cross-platform as they require tweaking to operate on different browsers, operating systems, etc.	Web services are platform independent as they use open protocols
E.g. naver.com is a website that has a collection of related web pages containing tutorials.	E.g. Google maps API is a web service that can be used by websites to display Maps by passing coordinates to it.

Open API / Web API

❖ Open API / Public API

- A **publicly available** API that provides developers with programmatic access to a proprietary software application or web service – [Open API](#) from Wikipedia.

❖ Web API

- An API for either a web server or a web browser - [Web API](#) from Wikipedia.

❖ Examples

- 본 수업에서 실습해 볼 Open Web API 들
 - 대한민국 정부 공공데이터 API : www.data.go.kr
 - KAKAO, NAVER Search API
- Philips HUE API

