

# From REST to GraphQL

Seungjae Lee @huisseoul

# REpresentational State Transfer

- 2000년, Roy Fielding(HTTP 1.x 명세 주요 기여자)이 처음 제시
  - HTTP 1.1 명세와 함께 고안되었기 때문에 상호간 호환성 용이
- 프레임웍, 라이브러리, 언어가 아닌 일종의 아키텍처 방법론 혹은 규칙
- 이 방법론을 API 구현에 접목하면 흔히 RESTful API 라고 부름
- Using HTTP Methods: GET, POST, PUT, DELETE

# REST

- 아키텍처 원칙
  - Client-Server
    - 서로간 역할이 명확히 분리, 의존성 낮아 자유도 높음
  - Stateless
    - 상태(세션, 쿠키 등)를 저장하지 않고 들어오는 요청에 대해서만 처리
  - Cacheable
    - HTTP Header: LastModified, Expire E-Tag ...
  - Layerd System
    - 계층적 구조로 porxy, API gateway 등을 끼워넣기 용이
  - Uniform Interface
    - 설명해 주실분...?

# GraphQL

- Facebook에서 2012년부터 내부적으로 사용해 오다가 2015년에 공개
- API 구현에 쓰이는 일종의 쿼리 언어
  - 혹은 쿼리를 실행하기 위한 server-side runtime
- Using HTTP Methods: GET, POST

# GraphQL

- 아키텍처 원칙
  - Hierarchical
    - 클라이언트가 데이터 요구사항을 자연스럽게 파악 가능
  - Product-centric
    - 제품 스펙의 구현 방향을 클라이언트 개발자가 원하는 방향으로 구축할수 있음
  - Strong-typing
    - 타이핑이 갖는 장점들 포용 (내구성 등)
  - Client-specified queries
    - 클라이언트가 요청한것 외에 다른걸 신경쓸 필요가 없음
  - Introspective
    - 설명해 주실분...?

# References

## REST

- [https://en.wikipedia.org/wiki/Representational\\_state\\_transfer](https://en.wikipedia.org/wiki/Representational_state_transfer)
- [https://www.ics.uci.edu/~fielding/pubs/dissertation/rest\\_arch\\_style.htm](https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm)
- <https://stackoverflow.com/questions/4663927/what-is-rest-slightly-confused>
- <http://www.apiacademy.co/how-to-http-caching-for-restful-hypermedia-apis/>
- <http://meetup.toast.com/posts/92>

# References

## GraphQL

- <http://facebook.github.io/graphql/>
- <http://graphql.org/>

# References

- <https://www.sitepoint.com/rest-2-0-graphql/>
- <https://0x2a.sh/from-rest-to-graphql-b4e95e94c26b>
- <https://dev-blog.apollodata.com/graphql-vs-rest-5d425123e34b>
- <https://philsturgeon.uk/api/2017/01/24/graphql-vs-rest-overview/>
- <https://edgecoders.com/restful-apis-vs-graphql-apis-by-example-51cb3d64809a>
- <http://nordicapis.com/is-graphql-the-end-of-rest-style-apis/>



**감사합니다.**

fin.