Data APIs

<https://www.youtube.com/watch?v=rXxeRf1rv04>

Proceed to 31:30

* Data APIs are not the same as Application APIs
* Challenges
  + Own Pools of truth: If you are creating APIs to front a source of truth, others might use those APIs to create their own source of truth (duplicate pools of data)
    - Solution:
      * SLAs that support the throughput in such a way that you can run reports on it but cannot wholesale duplicate the data
  + Data ownership
    - Sometimes you get into the issue of trying to monetize customer or supplier data without rights
  + Data Ownership
    - If you share data how can you make sure it is not copied or resold? (How to make sure it remains your data?)
    - VRM = Visual rights management
    - VRM or DRM for digital data – There is a new proposal out there
* APIs in itself is not valuable – it is the reduction in
* Data-as-a-Service using Layer 7 Data Lens
  + Roles
    - Service Provider
    - Service User
  + Layer 7
    - Portal – human
    - Gateway (PEP)
  + Layer 7 Data Lens Flow
    - Data setup
      * Service provider goes into the portal, essentially specifies where the data is residing, clicks through something called data explorer, selects what kind of data he wants to share and clicks a button
      * Self-service
    - Under the hood
      * Generate what needs to access the data
      * Generates a REST APIs on top of it
    - Data publish
      * Those APIs are published as Provider defined data lens
    - Data consumption
      * Consumer can log in to the portal, view the data that has been shared with him and then create view (customer defined data lenses) on top of it
      * Creates/generates APIs on top of that
      * Self-service
      * Can share with downstream customers or providers

<https://www.oreilly.com/ideas/data-apis-design-and-visual-storytelling>

**Tips to successfully create an API for data publication**

If you decide to make your data accessible through an API, here are some key points to help you succeed:

1. Technology (platform) does not really matter too much. This is the beauty of the Web: all that matters is that you output your data in an open format that is developer friendly (i.e. JSON, XML or CSV).
2. JSON is more readable than XML, which is easier to parse than CSV. This might sound weird, as XML is actually more a document format than a data format, but in practice, XML is very verbose and does a poor job at representing the structure, especially when you have numeric data types. JSON is the better choice, provided you have plug-ins to doll it up.
3. APIs can have different formats (if you feel the need), but JSONP is probably the most important. The main reason is that it allows for cross-domain query. For instance, our Out of School interactive is pulling live data from UNESCO’s API, which is only possible because the API offers a JSONP format (P=Procedure call).
4. Be mindful of performance. In the case of the Out of School visualization, the UNESCO data API is queried live by the browser, which means that the API needs to support a potentially high number of queries. A side effect of that is that the visualization’s load time is directly dependent on the speed of the API.

Check out <https://dev.socrata.com/> for Open Data