A. Data Portals / Web Apps

1. Is anything pulled from the backend by any other means than a call to a REST HTTP API? [no] (decoupling question)
2. (to be sure) are all graphics and interactive elements assembled on the (Web) client side? [yes]
3. Are all API calls CORS enabled? (can we call them from web Apps hosted in other domains) [yes]
4. Is there a Postman or Swagger page documenting and demonstrating the use of the API methods? [yes]
5. If login to the data/analytic resource is required, is there a OAUTH2 mechanism to generate bearer tokens? [yes]
6. Are there client-side SDKs (say, for JavaScript, R, Python…) facilitating programmatic usage of the data source? [yes]

B. Cloud usage

1. Is your use of the Cloud **zero-idle serverless** ? (the most important question!) [yes/no]
2. If no to (1), are you using a Client environment with fixed costs (like Google Colab) where Cloud resources engaged are zero-idle serverless ? [yes]
3. Are the Cloud services programmatically interoperable via CORS enabled, REST APIs? [yes]
4. Are there SDKs facilitating the use of those APIs (say, for JavaScript, R, Python…)? [yes]
5. Is your development of Cloud hosted data and analytical services mediated by Cloud functions (FaaS)? [yes/no]
6. In the 1st diagram below locate your data infrastructure components. If you rely on analytical enclaves, diagram 2, please confirm that it is operating external data sources as per A1.
7. Is your use of the Cloud overseen and monitored by your institutional IT? [yes]

C. Cloud middle layers

If you are using a Cloud Middle Layer programmatically, such as Box, Google Drive, DropBox and OneDrive, B1-6 are automatically checked at a technical level. Instead they are reduced to the confirmation of B7: is the use of the Cloud as the back-end data layer for client-side applications (checking A) vetted and monitored by CBITT?

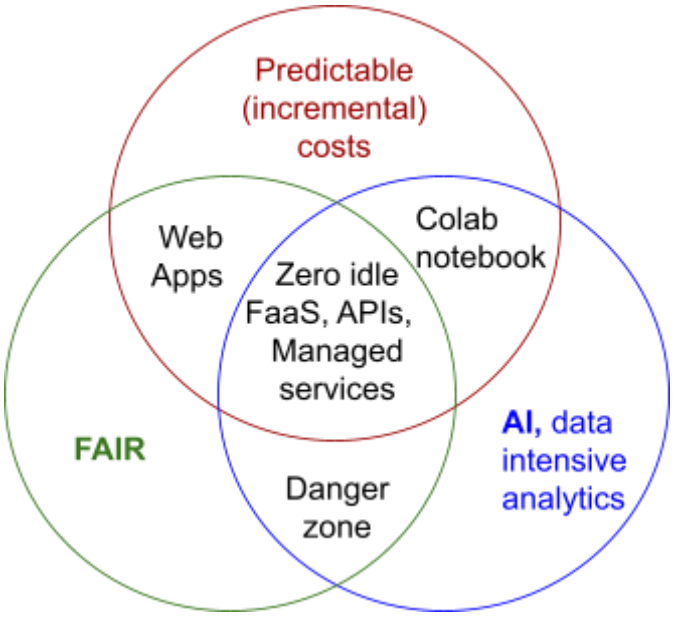

Diagram 1 - Identify where the components of your system fall in this combination of FAIRness (particularly the R, reusability), cost structure, and data intensive analytics.

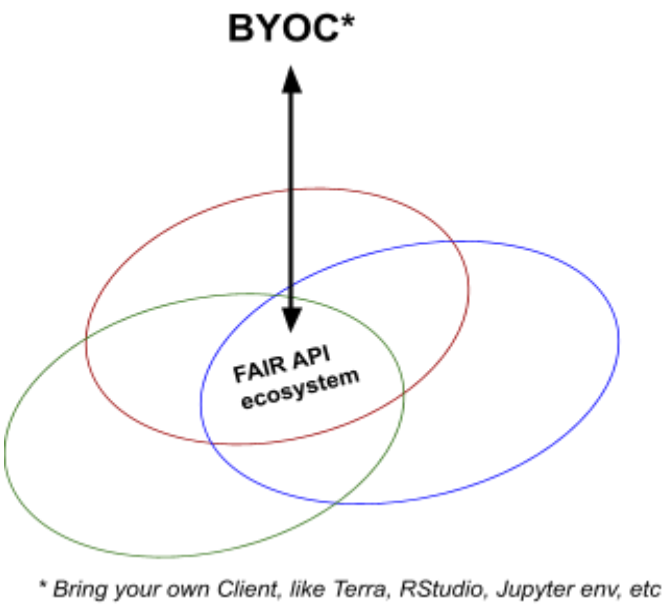

Diagram 2 - is your non-Web client or client component doing anything else than operating stateless HTTP REST APIs (i.e. are there any other I/O arrows)?

Ground Rule for software development separation of concerns

Innovation identifies and validates architecture that **Implementation** should not change (when they do, they need the experimental practices of Innovation). These are distinct groups of people, with some overlap. They are not the same group of people.