



**Design Document
Technical Overview
India Urban Data Exchange
(IUDX) version 1.0**

Table of Contents

Table of Contents.....	1
1. Introduction and Background	2
2. Benefits of IUDX.....	3
3. Architecture Overview	4
4. Catalogue Server Overview	6
5. Authorization Server	7
6. Resource Server	8
7. Interaction Flow Examples	9
8. IUDX Design and Implementation Considerations	11
9. Conclusion	12
10. Appendix 1: Data Descriptor example	13

1. Introduction and Background

Cities around the world have learned that they possess a new valuable asset, namely the data generated by their various departments and agencies. Some of the data consists of streams of IOT data from installed sensors (e.g., Air Quality, Traffic, etc), some of the data is demographic or geographical, some may be from municipal tax or property records, some from legal documents or registrations, and some may be historical data from archival sources. Each set of data has its own security and privacy consideration, as well as commercial, monetary or subscription aspects which must be observed. Cities such as Copenhagen¹, Columbus² and Manchester³ have taken ownership of their data assets by creating data exchanges, which are software platforms that allow controlled sharing of data by providing common ways of accessing and representing the data. An important idea behind a data exchange is that data silos are actually not a bad thing, as each silo often represents a domain-optimized service that performs that function very well. Instead of breaking silos or moving data en masse into a central repository, this approach chooses to interconnect the disparate and distributed entities through a common data exchange. This provides a way for accessing data in a unified, common format, allowing for sharing of data between different departments in a city, as well as opening up data for third party developers to create innovative new applications and citizen services. In addition, there is an opportunity for third party providers of data, or third-party providers of data analytics or data annotation, to participate in what becomes a data marketplace.

In late 2018, the Smart City Mission⁴ in the Government of India, came together with the Indian Institute of Science, Bangalore to jointly develop and deploy the India Urban Data Exchange⁵ (IUDX). IUDX is now deployed as a production cloud service in the cities of Surat, Varanasi and Pune and will shortly be deployed in 7 more. The intent is to make IUDX broadly available across all Indian cities in the next couple of years, with an ecosystem of collaborating partners. The development of IUDX as described in (iudx.org.in) is collaborative with contributors from various organizations.

IUDX will be completely open source, based on an underlying framework of open APIs, data models, and the security, privacy and accounting mechanisms that will facilitate, easy and efficient exchange of data among disparate urban data silos. IUDX draws on ideas and, where feasible, code, from best-of-breed global projects such as Fiware⁶ (fiware.org). However, we are heavily skewed towards the Indian ecosystem, cultural

¹ <https://cphsolutionslab.dk/en/news/learnings-from-the-city-data-exchange>

² <https://datasmart.ash.harvard.edu/news/article/how-columbus-accomplishing-its-smart-city-vision>

³ <https://www.manchester.gov.uk/smartercity>

⁴ <http://smartcities.gov.in/content/>

⁵ <https://www.iudx.org.in/>

⁶ www.fiware.org

norms, city nomenclature, payment and identification systems, etc. and therefore our solution is particularly suitable for Indian Smart Cities.

IUDX will facilitate secure, authenticated and managed exchange of data amongst various data platforms, 3rd party authenticated & authorized applications and other data sources, data producers and consumers, both within a city to begin with and scaled up across cities eventually at a national level, in a uniform & seamless way. The platform provides full control to the data owners as to what data to expose and to whom. Built-in accounting mechanisms enable connect with payment gateways which will form the foundations for a data marketplace. The whole platform is developer friendly, via definitions of open APIs (application program interfaces) and data schema templates (formats for interpreting data), so that a whole new application ecosystem gets created.

2. Benefits of IUDX

It is expected that all stakeholders in the Indian Smart City ecosystem will gain from the initiative. Within each of the Cities, citizens and the community will benefit through the availability of better, more innovative, and cheaper applications and services. The Cities themselves will benefit by the reduced development cost and faster development times enabled by a standard platform, together with the ability to choose vendors freely and avoid vendor lock-in.

They will see new source of revenue through the unlocking of data assets, and will unleash innovation from entrepreneurs and community, without any cost to themselves. Industry will benefit enormously through the improved ability to find skills and rapidly ramp up projects. They will also see reduced development expense enabled by a standardized and open-source platform, and be able to focus on innovation and differentiated value rather than design basic platform software. Start-ups, in particular, will benefit from the decrease in heterogeneity IUDX provides. Third party sources of data (such as private apartment complexes) will have a new opportunity to share and monetize their assets. Academic institutions and research labs will be able to conduct research and experiments in real world settings.

The focus is to enable data exchange between various city departments, between governments & citizens and governments & private sector within a city. It will directly address the issues that inhibit sharing & extraction of maximum value from the City's data. The initiative will essentially create a unified single-point data market place for various smart cities ecosystem stakeholders. This will help cities with new revenue sources and create a fertile environment for innovation.

IUDX compliant applications will be able to use APIs to pull data from any of the underlying data platforms and using the publisher APIs to push data to any of the applications behind the individual platforms. Standardized APIs and data schema

templates, will enable an IUDX compliant application to work in a city without needing any modification. Additionally, the standardized publisher APIs along with common data schemas, will enable vendor neutrality for IoT devices. In IUDX, there is a clear definition of data ownership and sharing mechanism, under the control of the data owner. The platform itself is only a vehicle for the data owners to more effectively manage and share the data as per their discretion/policies.

The IUDX project will not just focus on technology. In fact, the objective of the project is to create a data space that includes technology, policy, best practices, together with a thriving eco-system of application developers.

3. Architecture Overview

IUDX addresses two key stakeholders: Data providers and Data consumers. A data provider is the provider/custodian of the data resources which are being made available through the data exchange. A data consumer consumes the data resource available through the data exchange for the purpose of application/solution development. A data provider has the full authority to control access and provide authorizations to consumers wishing to access data for resources it controls. IUDX strives for enabling seamless, secure access of data to the consumers while respecting access control policies set by the provider.

Key concerns of the data providers are as follows:

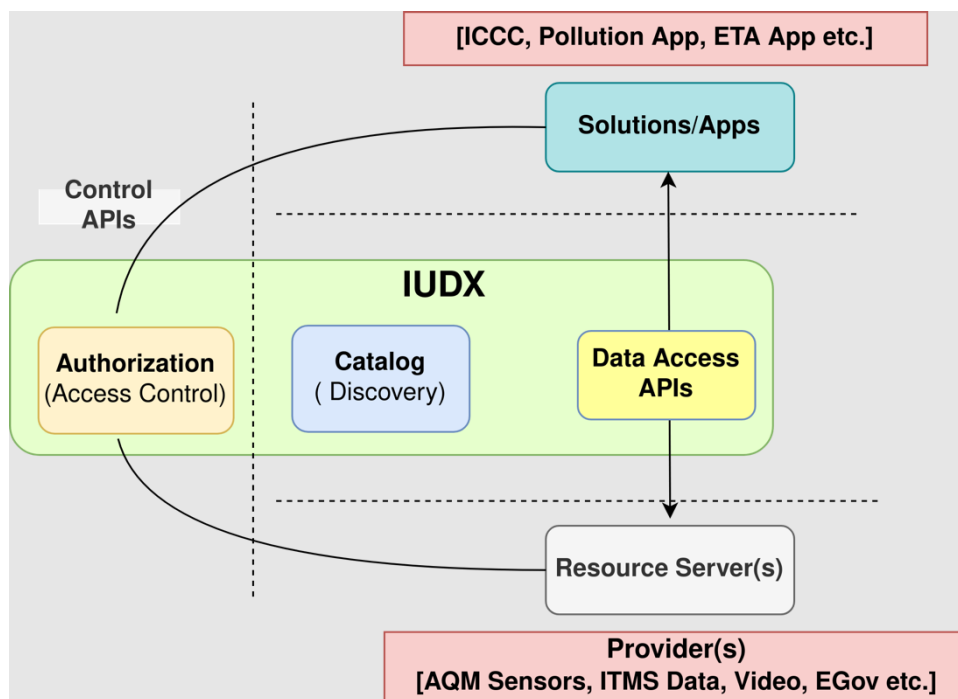
- Be able to control access to its data resources by authorized consumers only.
- Be able to onboard data into the exchange in a secure fashion.
- Be able to provide and manage meta-information about its data resources in order to improve discoverability, data usage and data inter-operability. In short, to make its data resource more attractive to wide spectrum of data consumers.
- Be able to interact with the data exchange in an easy and secure fashion.

Key concerns of data consumers are as follows:

- Be able to discover data resources of interest.
- Be able to access the relevant data from the available data resources in a secure fashion.
- Be able to get access tokens in case authorization is required by the provider of a resource.
- Be able to understand the data from a resource to be able to use it easily in application development.
- Be able to get additional context (meta-information) for the data. For example, to be able to use data from different resources, to be able to connect correlate data from different sources etc.
- Be able to interact with the data exchange in an easy and secure fashion.

IUDX addresses the above concerns by implementing the interfaces and open APIs as described in the 'Unified Data Exchange Architecture' specifications⁷. IUDX is an open data exchange platform and consists of three main components:

- Catalogue service provided by a Catalogue server
- Authorization and Authentication services provided by an Authorization server
- Data access and data ingestion services provided by one or more Resource Servers



Data resources managed by a data Provider are hosted on one or more Resource Servers and a data consumer can access a data resource via open and standard data access APIs. Resource servers also provide publication services to enable data providers to ingest data from their respective data resources.

A data consumer can discover the data resources relevant to its application using the search APIs provided by the catalogue service. The catalogue hosts information (e.g., data formats, units, type of the resource, etc.) for each data resource. This information is registered and managed completely by the provider of the given resource using open management APIs provided by the catalogue service. The meta-information, which is both human and machine understandable, enables the consumer to understand data and get additional context required for intelligent usage of this data.

The Authorization server provides management APIs to register and modify access control policies associated with a protected data resource. The authorization server

⁷ Reference for the spec

also provides services to get access tokens in case the data consumer needs to access a protected resource. Using token validation services provided by authorization server, a resource server can ensure compliance to the data access policy set by the provider.

We now give a brief overview of each of the above component.

4. Catalogue Server Overview

The main functions of IUDX catalogue service are:

- Search and discovery of data resources
- Provide unambiguous description of data from a given resource thereby leading to Improved data understandability and inter-operability
- Provide additional context for a given resource that may improve its usage in applications

IUDX catalogue is a store of meta-information associated with the data resources available with the data exchange. Some illustrative meta-information attributes associated with a resource are: associated provider details, resource server details where the data can be accessed, location of a sensor device, descriptions associated with the resource, data-sheets if applicable, information whether the resource is public or private etc. The meta-information for a given resource is provided and managed by the data provider of the associated resource. The catalogue uses concepts of linked data to provide semantic contexts for the attributes describing meta-information which leads to improved machine readability, interpretability, operational interoperability and enables vocabulary reuse from other data-model stores and taxonomies. The catalogue items are stored using JSON-LD⁸ format which is a lightweight linked-data format based on existing and popular JSON format and is easily readable and writable by humans. Further, each catalogue item adheres to a schema that specifies a set of mandatory attributes and may contain additional custom attributes. A catalogue item may be related to another catalogue item by providing explicit references to one another.

One of the key meta-information objects is the 'data descriptor' object which unambiguously describes each attribute contained in the data from the associated resource. It may contain description of data types, units, value constraints, text descriptions, semantic context etc. for these attributes. Use of JSON-LD enables linking each attribute to a vocabulary thus providing semantic understanding for these attributes contained in the data.

⁸ <https://json-ld.org/>

The IUDX catalogue services, built on top of the meta-information store, provide APIs for:

- Search and discovery
 - Text search, Geo-spatial search, Attribute search and Relationship search
- Management
 - Create, read, update and delete meta-information objects

5. Authorization Server

The IUDX Authorization Server allows data sharing while respecting ownership, privacy and compliance requirements. The IUDX authorization server ensures that only the authorized person can get tokens to access the private/protected data.

The main functions of the authorization service are:

- Resource access authorization to grant access to access-controlled resources
- Resource access policy management to manage policies which specify access rules
- Authentication and registration services

To access a protected data resource, the consumer needs to present a valid authorization token to the resource server. A token can be obtained by using token access APIs provided by the authorization server. Any access token request for a resource by a consumer will be checked against the existing access control policies. These policies are set and managed by the provider of the resource. If the policy decision is a success, an access token is provided to the consumer using which a data access request can be made. The access token is also logged to ensure auditability of consent grant. If no policy exists or no decision can be made, the provider and the consumer may negotiate, outside the scope of the IUDX, to agree to consent terms after which an access policy can be set by the provider. The authorization service also provides token introspection APIs using which a resource server can check the validity of a token presented by the consumer at the time of data access.

IUDX specifications do not mandate any policy authoring language. In general, a policy should have a simple structure and should be able to specify: 'who' can access, 'what' resource and for 'how long' etc.

IUDX mandates the use of X.509 certificates for authentication. A mutual TLS based authentication is used in all interactions with the authorization server. Using the registration process, a consumer and a provider obtain the digital certificates. Note that the entities need not authenticate themselves while interacting with a resource

servers or the catalogue server where all interactions with protected resources are secured using access tokens.

The authorization service provides APIs for:

- Request an access token for a given resource
- Manage access policies for a given resource
- List current access policies for a given resource
- Validate an access token
- Revoke an access token
- Audit access tokens

To summarize, the authorization service enables asynchronous authorization grant and enables data providers and application developers to target a consistent set of APIs for authoring policies and accessing data for protected data resources.

6. Resource Server

The resource server constitutes the data plane for IUDX and provides data access for the resources available with the exchange. It serves data to data consumers in compliance with the access policy requirements set by the provider of the resource. For this compliance, the resource server must implement a token introspection interface with the authorization server. For any access on a protected resource, the resource server requires the consumer to present an access token which it validates using token introspection APIs before serving the data.

Apart from data access APIs, AMQP⁹ subscriptions are also supported for streaming data sources. In future, support for call back subscriptions may also be provided. Data is always served over secure and encrypted channels using TLS channels.

The resource server may also provide data ingestion endpoints to allow data to be published into the resource server by data providers.

IUDX resource access APIs are harmonized with ETSI NGSI-LD Specifications¹⁰. The data access service provides APIs for:

- Search and count
 - Get data using spatial, temporal, attribute and complex searches
- Subscriptions
 - Get access to streaming data using AMQP

⁹ <https://www.rabbitmq.com/amqp-0-9-1-reference.html>

¹⁰ https://www.etsi.org/deliver/etsi_gs/CIM/001_099/009/01.01.01_60/gs_CIM009v010101p.pdf

Note that by design one or more resource servers can exist in an IUDX system. Further an entity other than IUDX service provider, defined as the entity which is deploying and maintaining the catalogue server and authorization server, can provide for a resource server as long as the resources hosted by it are onboarded on the catalogue, data access complies with the sharing policies set by the provider and the data is served according to the IUDX data access APIs.

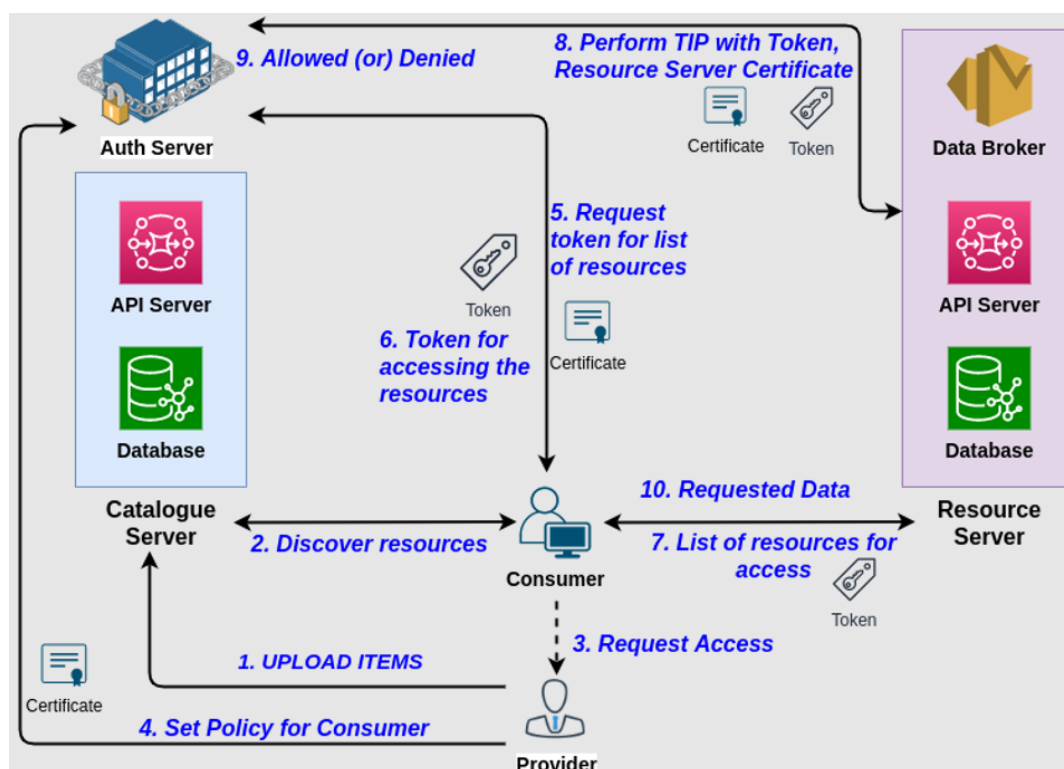
7. Interaction Flow Examples

In this section, we illustrate end to end interactions made by data consumer and data provider with the various components of IUDX system.

We start by describing the interactions for the purpose of data consumption for a given resource. We assume that the resource has already been onboarded on catalogue and the data from the resource is already available with the resource server.

In order to understand the flow better, we have a simple flow diagram below explaining the steps involved.

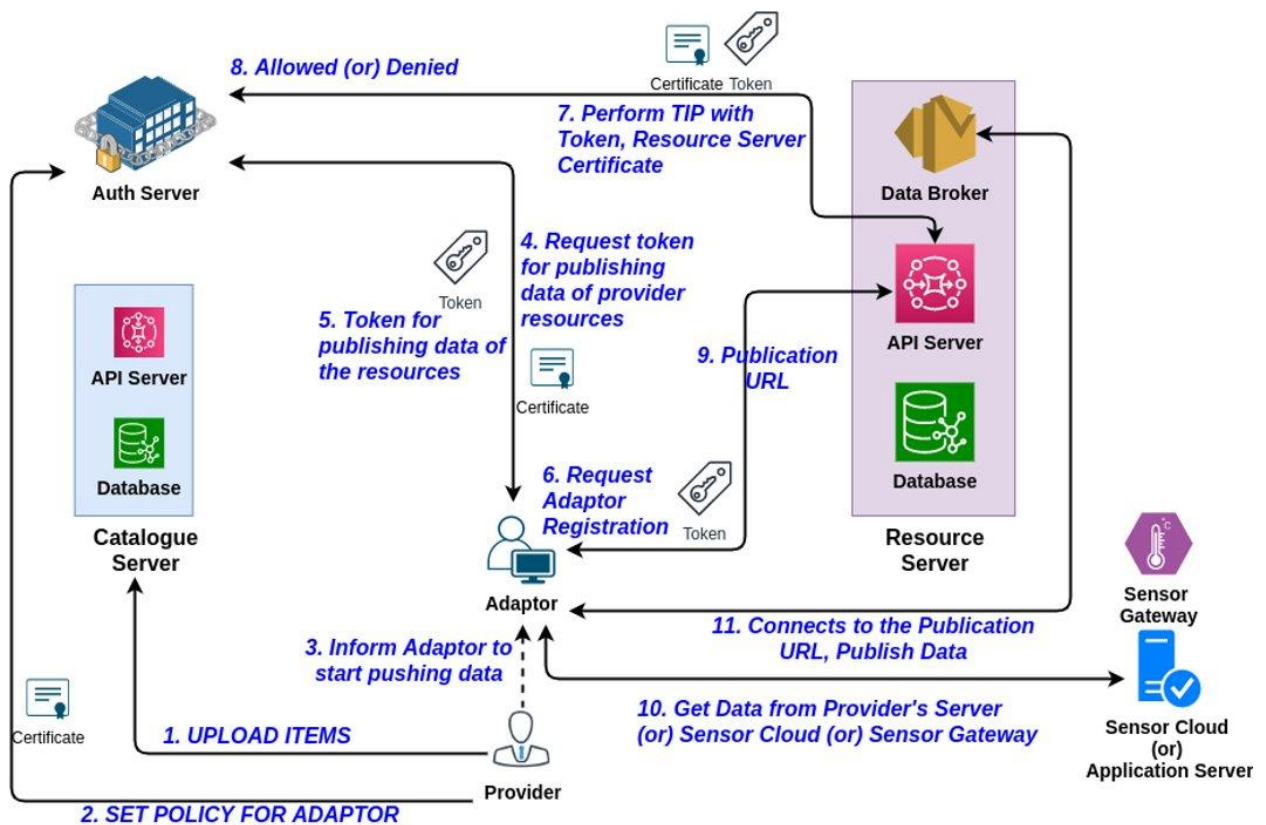
Consumer Interaction



The consumer uses the catalogue service (Catalogue) to discover the resources of interest. For the purpose of this illustration let us assume that the resource is an access-controlled or private resource. The consumer may need to request an access

with the provider which is out of band negotiation between the consumer and the provider. Once the consent is negotiated, the provider sets the access policy and the consumer can request for an access token with the Authorization server. With the Access token, and the ID of the resource obtained from the catalogue, the consumer can access the data from the Resource Server using the data access APIs. Note that for interactions with Authorization server, the provider and the consumer must present the digital certificates which can be obtained during the registration process.

Data Ingestion Interaction



In order to enable data discovery, the provider advertises the resources owned in the Catalogue. On creating a catalogue entry for a resource, the provider gets an ID with which an adaptor can publish data into the IUDX system. In order to publish data, the adaptor requires an Access token. The adaptor can get an Access token by providing the digital certificate obtained during registration along with the ID of the resource in the catalogue. On obtaining the Access token, the adaptor can register with the Resource Server and start publishing data.

8. IUDX Design and Implementation Considerations

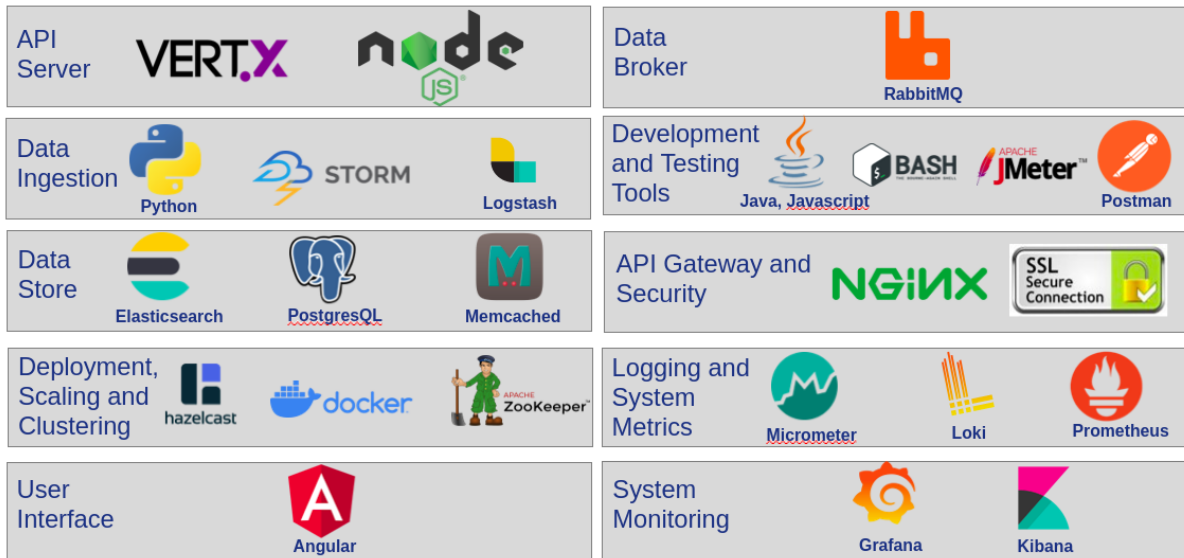
IUDX adheres to the following design and implementation principles:

- **Open APIs and data models:** Use of open and standard APIs and data models
- **Consent Driven:** Allows sharing of data only if an explicit consent is provided by the data provider
- **User Centric:** Simplifies interactions for consumers and providers
- **Secure by design:** Security considerations are part of the design right from the start and all the best practices are followed
- **Open source:** Uses leading tools, technologies from the open-source industry
- **Cloud deployable:** Designed for cloud deployment and utilize the state-of-the-art cloud infrastructure
- **Scalable and Elastic by design:** Upfront considerations for scalable and elastic designs for all the software components
- **Service oriented:** Incorporate service-oriented designs which can be scaled up/down without affecting the other components

In terms of security, the following best practices have been followed in the development of all the IUDX components:

- TLS based secure communications between all entities and components of IUDX
- Services are exposed only as per the access control policies
- Digital certificate-based authentications for all interactions with authorization server: issuing tokens, managing policies etc.
- Communication between IUDX components is not exposed to public IP network
- Using API gateway for authenticating, controlling and analysing API traffic
- Explicit validation for HTTP requests to debar invalid requests
- Limited port exposure for all the public services
- Rate limiting provisions for preventing DDOS attacks for open resources

IUDX is an open-source software platform. All the software components of the platform are based on industry leading open-source components and frameworks. The figure below shows the components used in the IUDX platform development stack.



IUDX is designed to be a cloud ready system where all the components are designed as a stand-alone containerized service. This leads to a highly scalable and available system. The API server components are designed to follow the Service Mesh Architecture which helps us to scale a specific service without scaling the entire system. This in turn helps us in reducing compute, storage and cost requirements. Apart from containerization, clustering strategies are a part of design for some systems, such as database and data brokers, to ensure horizontal scaling as well as high availability.

9. Conclusion

To summarize, IUDX is a data exchange service that

- Hosts and manages meta-information about resources and facilitates their discovery (catalogue server)
- Facilitates exchange of data using common data models (resource server)
- Facilitates authentication, authorization, accounting mechanisms (auth server)

It is available as a production quality open-source distribution and is deployed as a cloud base service for Indian cities. It is currently live in three Indian cities, Surat, Varanasi and Pune, with the expectation of many more cities over the next year.

10. Appendix 1: Data Descriptor example

Data Descriptor: Key Meta-info Object

```
{
  "CARBON_DIOXIDE": 485.09,
  "AQI": 61.0,
  "DATE": "2020-07-20T16:34:59"
}
```

```
{
  "CO2": 0.5129,
  "AQI": 55,
  "time": "2020-07-20T16:34:59"
}
```

What does each attribute mean ?
 How to interpret the values from each attribute ?
 How to relate data from two different sources ?

```
{
  "@context": {
    "iudx": "https://voc.iudx.org.in/",
    "CARBON_DIOXIDE": "iudx:c02",
    "AQI": "iudx:airQualityIndex",
    "DATE": "iudx:observationDateTime",
  },
  "CARBON_DIOXIDE": {
    "unitCode": "qudt-unit:PPM",
    "schema": "iudx:Number"
  },
  "DATETIME": {
    "schema": "iudx:DateTime"
  }
  .....
}
```

```
{
  "@context": {
    "iudx": "https://voc.iudx.org.in/",
    "CO2": "iudx:c02",
    "AQI": "iudx:airQualityIndex",
    "time": "iudx:observationDateTime",
  },
  "CO2": {
    "unitCode": "qudt-unit:PPB",
    "unitText": "Parts per billion",
    "schema": "iudx:Number"
  },
  "time": {
    "schema": "iudx:DateTime"
  }
  .....
}
```

JSON-LD

Data Descriptor

- Associated with each data resource
- Stored in Catalogue
- JSON-LD Document
- Provides "Context"
- Provides additional info like units, schemas etc.





THANK YOU

