Blue Brain Nexus Demo

2022-01-11

Agenda

Overview of Blue Brain Nexus platform

Data transformation to achieve consistent data representation

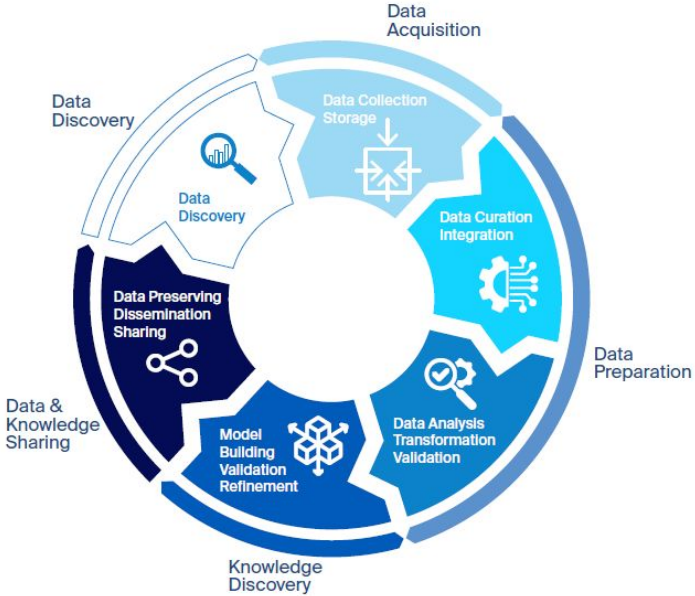
Data sharing

Querying data

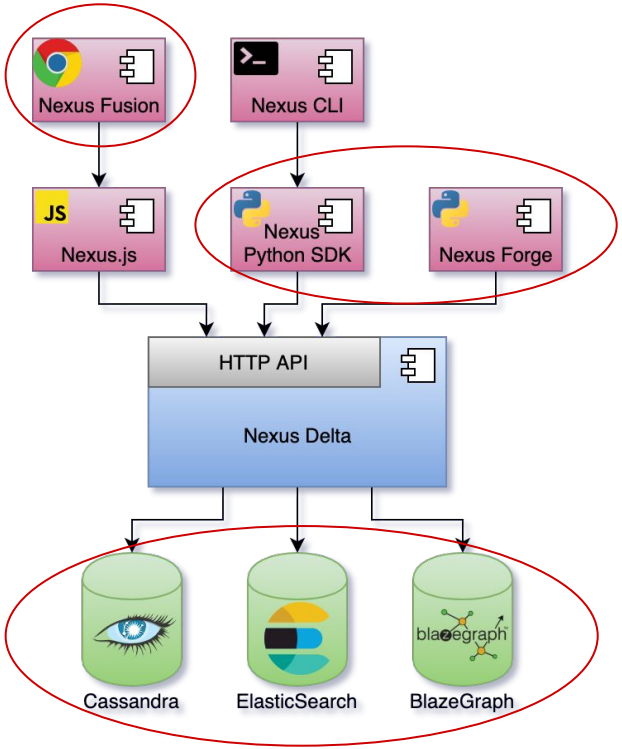
Overview of Platform

User capabilities provided by BBN products

Blue Brain Nexus (BBN) is an ecosystem containing the components on the right. BBN helps us support analysis for individual patients as well as population-level (aggregate) analysis.



High-level BBN Architecture



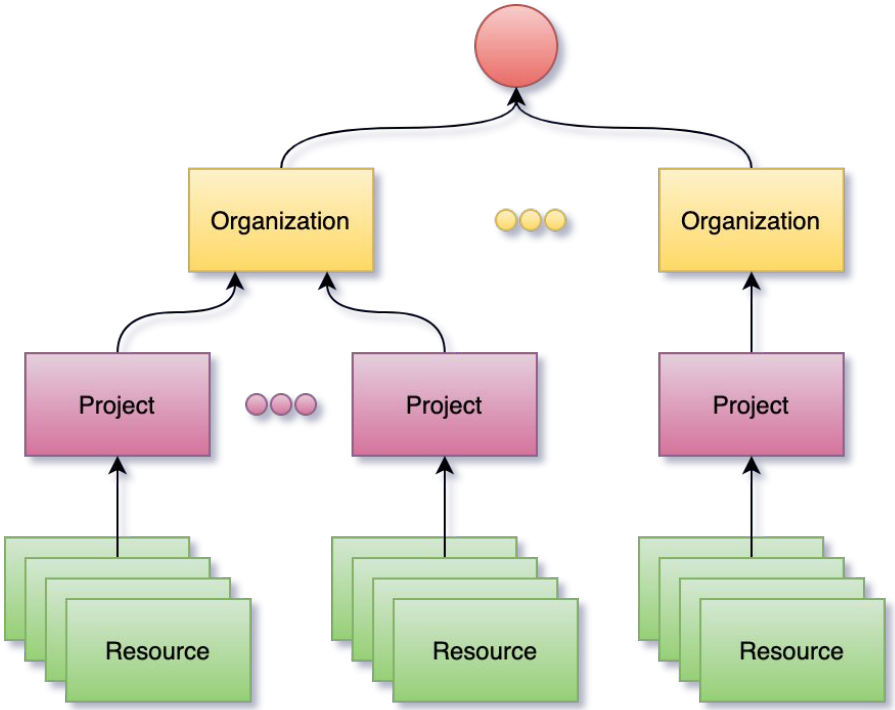
Web interface for users to interact with the Knowledge Graph

Python SDK's and libraries to make it easier to interact with Blue Brain Nexus

Nexus Delta is a scalable and secure service to store the data. Apache Cassandra acts as the primary event store. Data is then automatically indexed into default and custom Elasticsearch and BlazeGraph views.

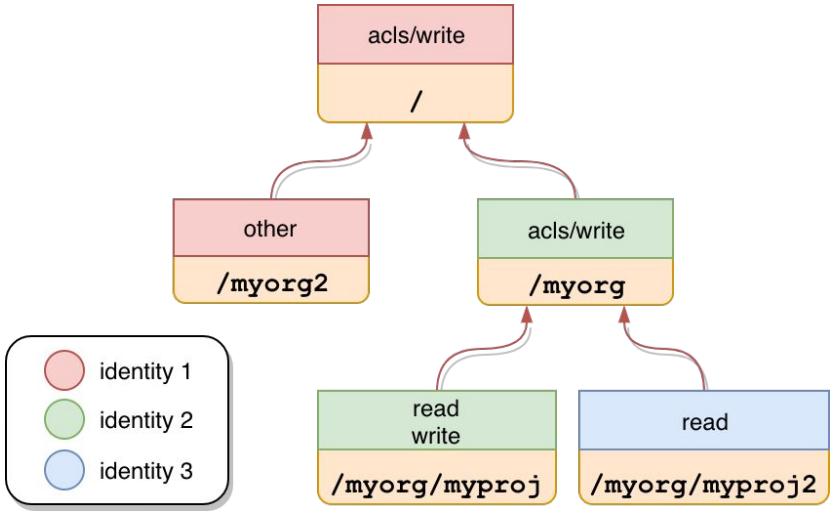
ETL Process - Consistent Data Representation

Project Structure



User data is represented as sub-resources to projects which in turn are sub-resources of organizations. Organization and project resources provide logical grouping and isolation allowing for variation in configuration and access control policies.

Access Control Lists



An ACL defines the applications' data access restriction using the following three parameters:

- permission: the value used to limit a client (user, group) access to resources.
- identity: a client identity reference, e.g. a certain user, a group, an anonymous user or someone who is authenticated to a certain realm.
- path: the location where to apply the restrictions. Examples of paths are: /, /myorg or /myorg/myproject

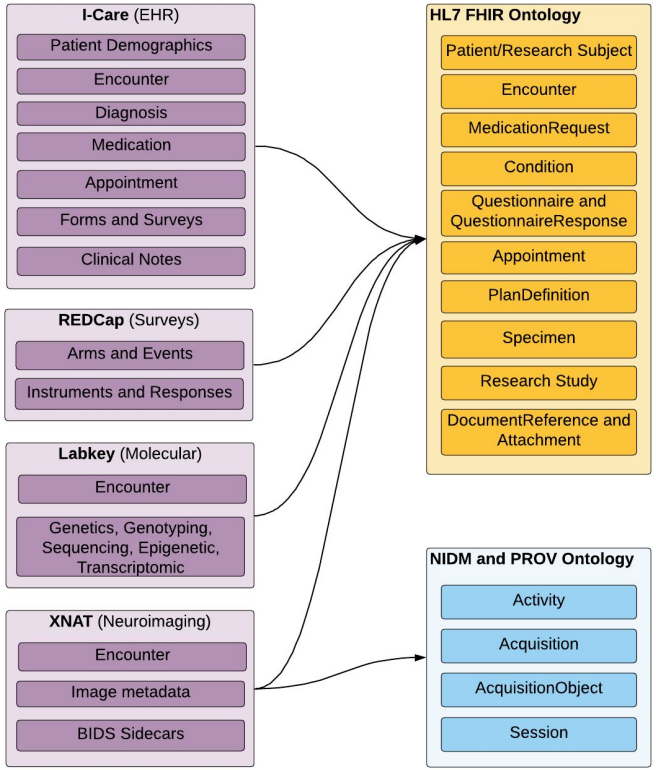
PUT {{deployment}}/acls/{{org}}/{{project}}?rev=1

Params Auth Headers (10) Body Pre-req. Tests Settings

raw JSON Beautify

```
{
    "acl": [
        {
            "identity": {
                "subject": "service-account-nexus-sa",
                "realm": "serviceaccounts"
            },
            "permissions": [
                "schemas/write",
                "views/write",
                "files/write",
                "permissions/write",
                "acls/write",
                "realms/write",
                "projects/read",
                "acls/read",
                "organizations/create",
                "organizations/write",
                "resources/write",
                "realms/read",
                "projects/create",
                "permissions/read",
                "resources/read",
                "organizations/read",
                "resolvers/write",
                "events/read",
                "views/query",
                "projects/write"
            ]
        },
```

Data Modeling - ontology selection and mapping



Criteria:

- Prominent ontologies with good coverage for data sources
- Well maintained by an established team
- Roadmap aligned with our plans
- Easy to understand documentation
- Aligned with internal understanding of data

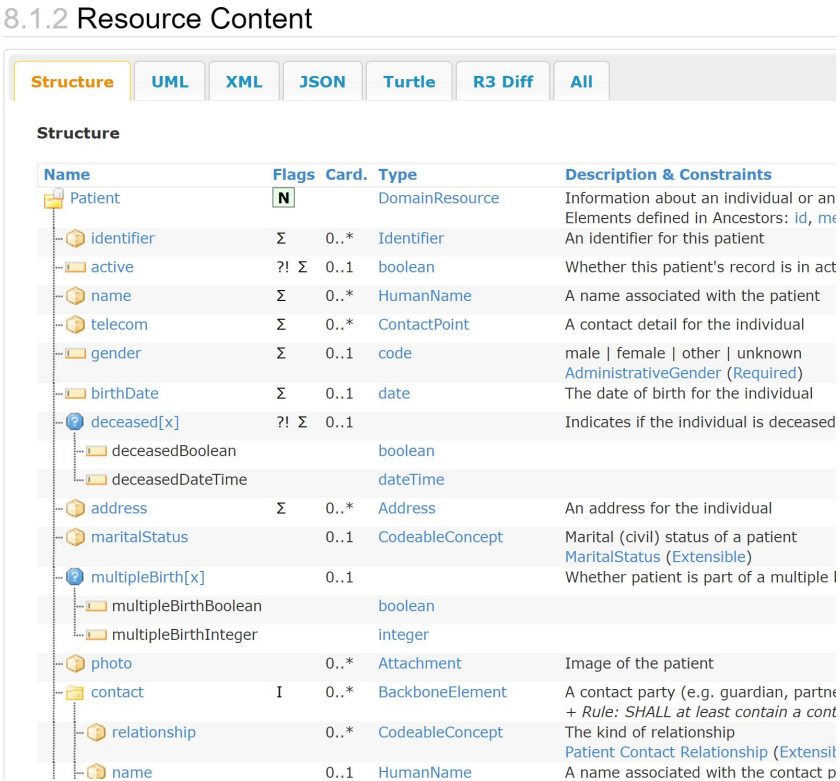
Ontology Links:

- FHIR
- NIDM
- BIDS
- PROV-O
-

Modeling notes:

- Knowledge graph contains metadata about files and points to file location (images, attachments, notes, etc.)

Data Model Example - FHIR Patient Representation



8.1.2 Resource Content

Structure | UML | XML | JSON | Turtle | R3 Diff | All

Structure

Name	Flags	Card.	Type	Description & Constraints
Patient	N		DomainResource	Information about an individual or an Elements defined in Ancestors: id, me
identifier	Σ	0..*	Identifier	An identifier for this patient
active	?! Σ	0..1	boolean	Whether this patient's record is in act
name	Σ	0..*	HumanName	A name associated with the patient
telecom	Σ	0..*	ContactPoint	A contact detail for the individual
gender	Σ	0..1	code	male | female | other | unknown AdministrativeGender (Required)
birthDate	Σ	0..1	date	The date of birth for the individual
deceased[x]	?! Σ	0..1		Indicates if the individual is deceased
deceasedBoolean			boolean	
deceasedDateTime			dateTime	
address	Σ	0..*	Address	An address for the individual
maritalStatus		0..1	CodeableConcept	Marital (civil) status of a patient MaritalStatus (Extensible)
multipleBirth[x]		0..1		Whether patient is part of a multiple l
multipleBirthBoolean			boolean	
multipleBirthInteger			integer	
photo		0..*	Attachment	Image of the patient
contact	I	0..*	BackboneElement	A contact party (e.g. guardian, partne + Rule: SHALL at least contain a cont
relationship		0..*	CodeableConcept	The kind of relationship Patient Contact Relationship (Extensil
name		0..1	HumanName	A name associated with the contact p

Link: https://www.hl7.org/fhir/patient.html

Well maintained specification for describing the common entities in healthcare.

Entities can also be extended and new entities can be described and shared through: https://simplifier.net/

Benefits of a graph model is that the logical model and physical model are the same. Therefore, the technical team, healthcare professionals, and business teams can communicate using a consistent vocabulary and understanding of subject domains.

ETL - Patient data transformation

Currently, there is no library that automatically converts Cerner EHR data to FHIR JSON-LD representation. Therefore, transformation process is manual.

EHR Field Name	EHR Value	FHIR Patient
PatientID	123456789	Patient.identifier
Gender	Female	Patient.gender
PostalCode	A1A1A1	Patient.contact.address

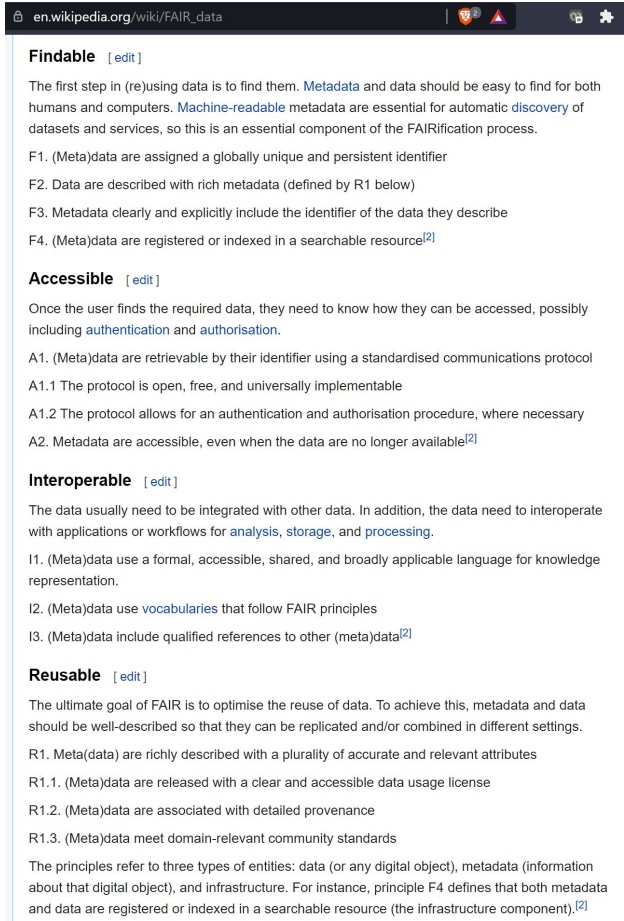
ETL Demo - Loading data to Blue Brain Nexus

Adherence to FAIR data principles.

Resource assigned a UUID

Vocabulary maintained and link can be followed to documentation

Provenance information automatically generated



Validation for FHIR data

Constraint type	Example
Value type	Patient address is of type fhir:Address
Cardinality	Patient name has exactly one first name
Value range	Age is between 0-120
String-based	Patient identifier is exactly nine characters long
Property-pair	Patient birthDate <= deceasedDate
Logical	Patient first name or given name have a value.
Shape-based	Patient address meets all the constraints defined for type fhir:Address.

Typically during data integration, validation rules are embedded in the code.

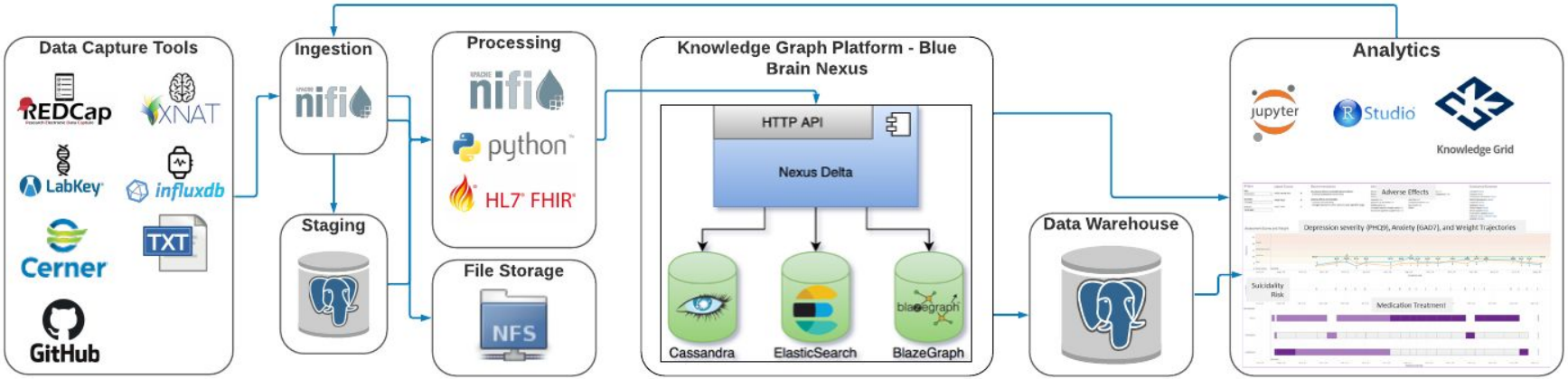
In Nexus, we can define resources that can be used to validate that resources meet a set of data quality conditions.

The benefit of defining the resources in Nexus is that the rules can be modified by users without modifying the backend code.

The table on the left describes the rules that can be defined in Nexus.

```
ex:PatientShape
        a sh:NodeShape ;
        sh:targetClass fhir:Patient ;
        sh:property [
                sh:path fhir:firstName ;
                sh:maxCount 1 ;
                sh:datatype xsd:string ;
        ] .
```

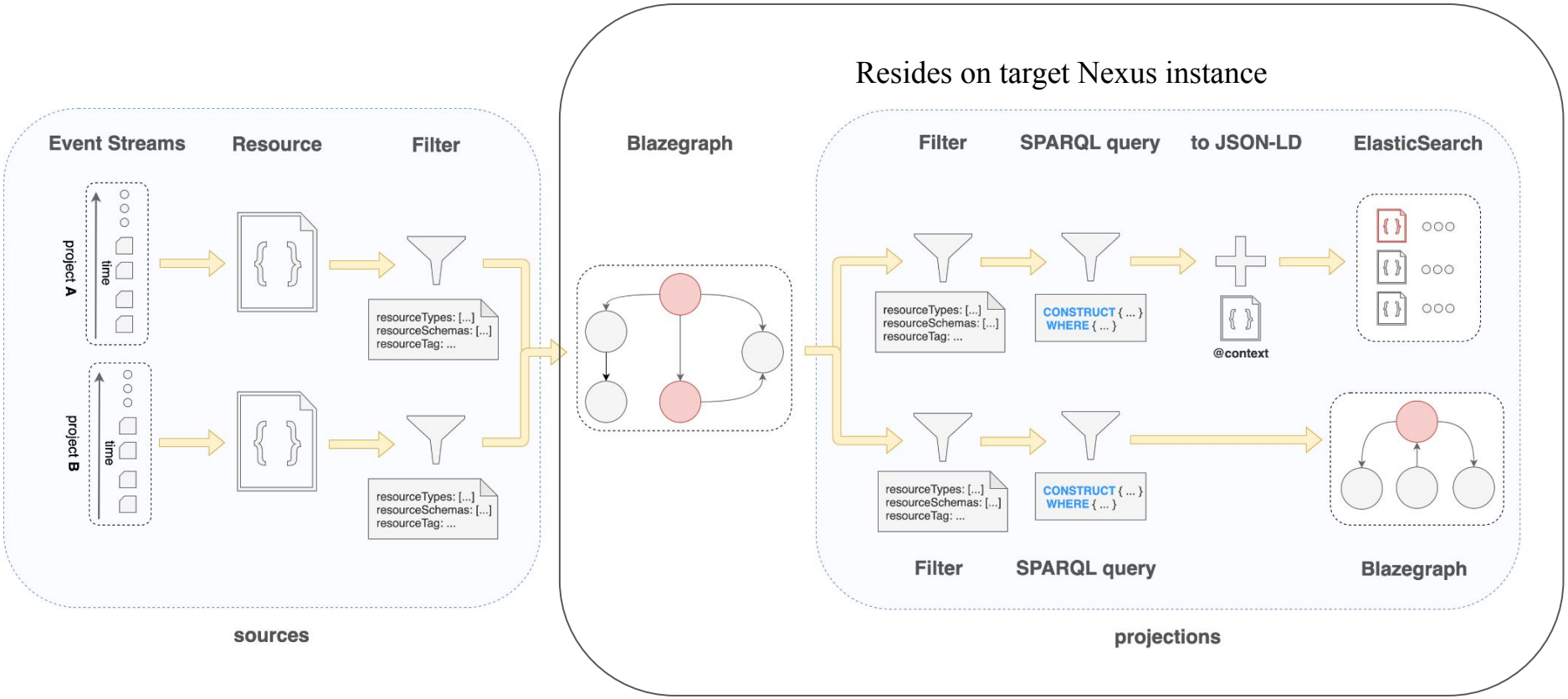
ETL - Pipeline Automation



1. Pull data from source systems using Apache NiFi and load as-is to staging
2. Process raw data from staging using Apache NiFi and Python and load to Blue Brain Nexus (Knowledge graph)
3. Create additional BlazeGraph and Elasticsearch indexes as required
4. Create projections from Elasticsearch to Postgres as required for data warehouse support
5. Make data available to be accessed through Analytics tools and services
6. Write back analytics results into the Knowledge Graph for continuous learning in the Graph.

Data Sharing

BBN Composite Views - Data Flow



BBN Composite View - Sources

Source Name	Description
ProjectEventStream	Data from the same project is used as a source for the view.
CrossProjectEventStream	Data from another project in the same organization is used as a source for the view.
RemoteProjectEventStream	Data from another project in a remote Nexus instance is used as a source for the view.

Authorization for Composite Views

For RemoteProjectEventStream, the view requires a token to be passed as a parameter.

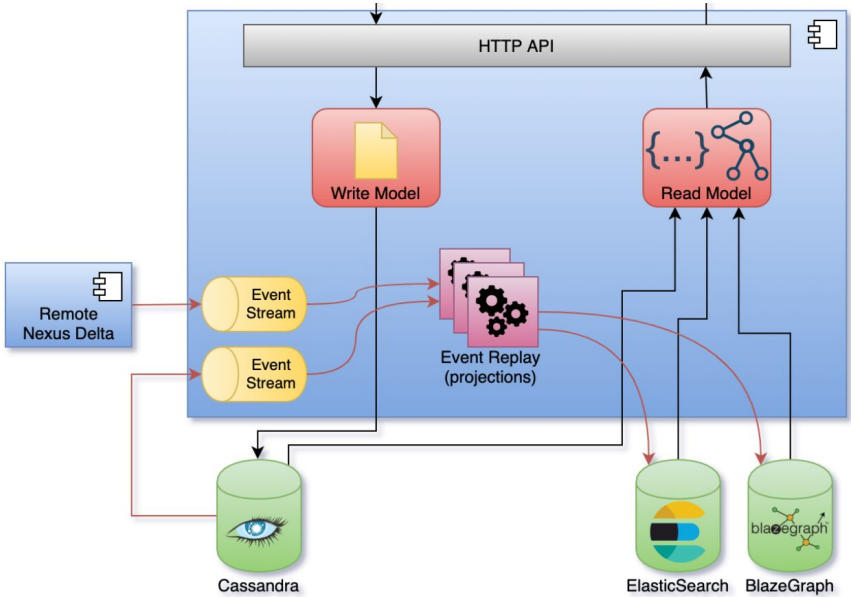
```
{
   "sources": [
      {
         "@id": "{sourceId}",
         "@type":
"RemoteProjectEventStream",
         "project": "{project}",
         "endpoint": "{endpoint}",
         "token": "{token}",
         "resourceSchemas": [
"{resourceSchema}", ...],
         "resourceTypes": [
"{resourceType}", ...],
         "resourceTag": "{tag}"
      }
   ],
   ...
}
```

where…

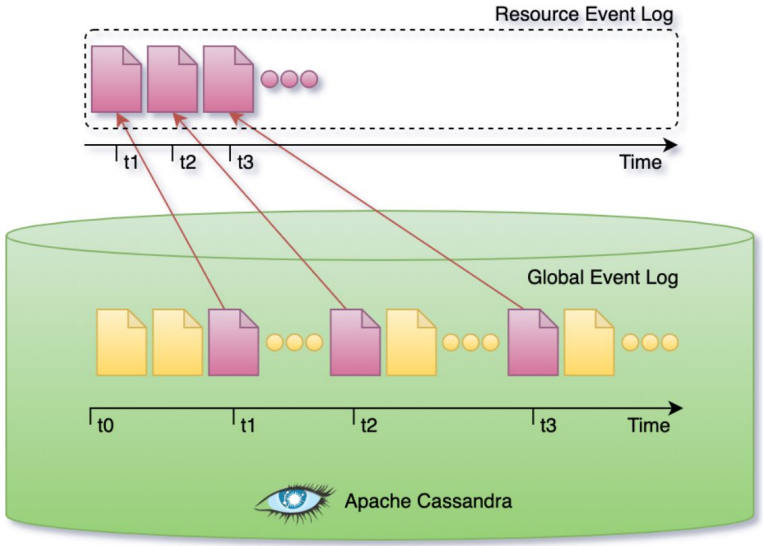
- {sourceId}: Iri - The identifier of the source. This field is optional. When missing, a randomly generated Iri will be assigned.
- {project}: String - the remote project (in the format ‘myorg/myproject’).
- {endpoint}: Iri - the Nexus deployment endpoint.
- **{token}: String - the Nexus deployment token. This field is optional. When missing, the Nexus endpoint will be accessed without authentication.**
- {resourceSchema}: Iri - Selects only resources that are validated against the provided schema Iri. This field is optional.
- {resourceType}: Iri - Select only resources of the provided type Iri. This field is optional.
- {tag}: String - Selects only resources with the provided tag. This field is optional.

BBN Event Stream

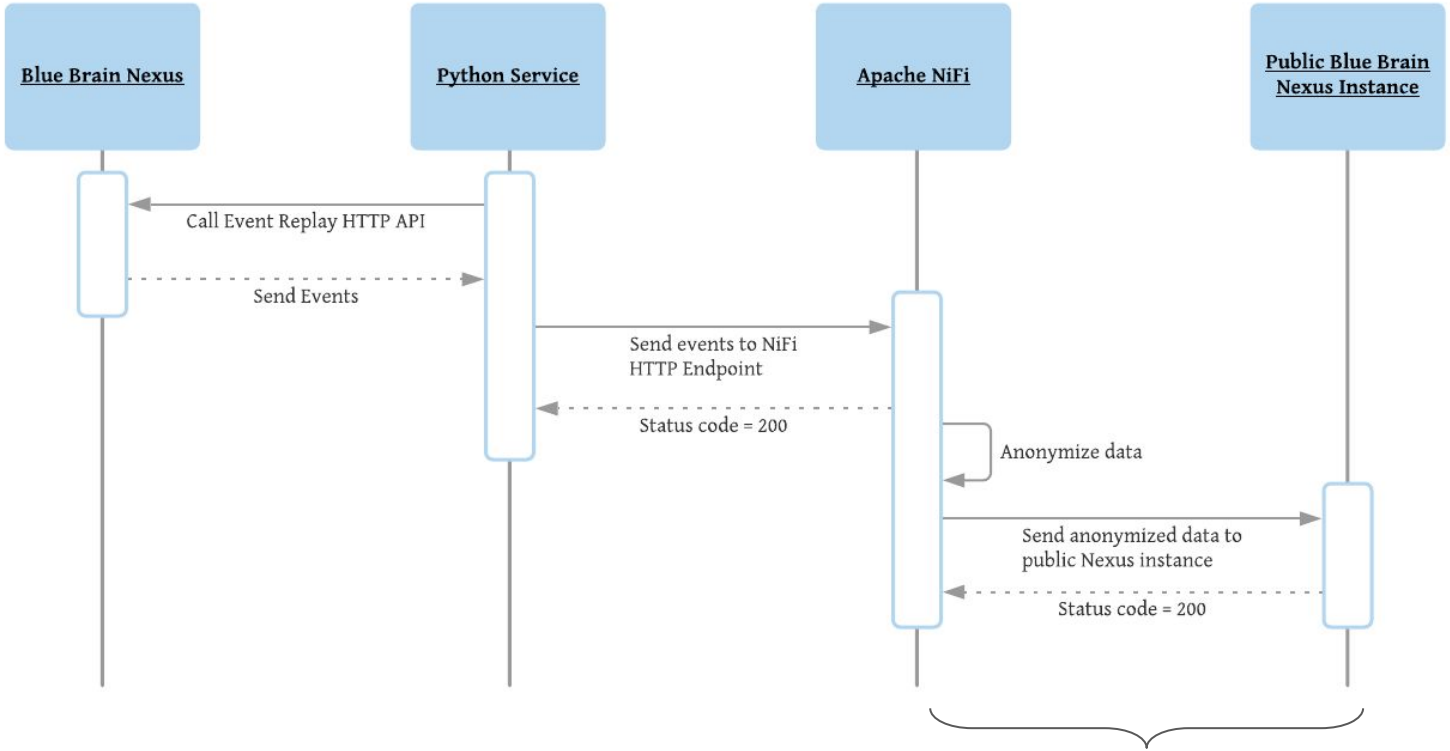
Nexus records all operations performed on resources in Apache Cassandra as events. This image shows how the events are used to construct the views in ElasticSearch and BlazeGraph. The events can also be accessed by HTTP API to push the events to another data store.



This image describes how all the resource events are stored in a single Global Event Log in Apache Cassandra. Therefore, when you access the HTTP API, you will receive events for all resources on the HTTP path.



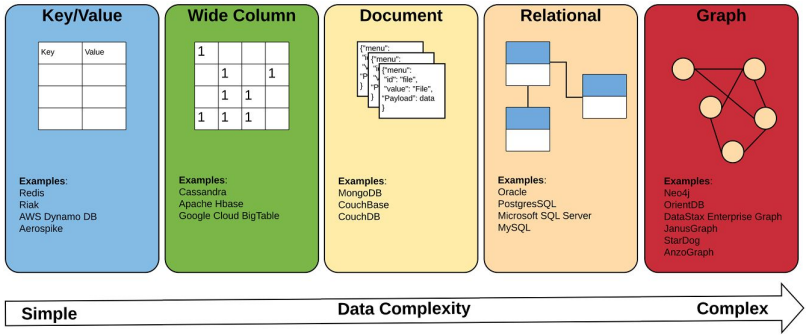
BBN Event Stream - Demo



Not shown in demo but described for use-case completeness.

Supporting Analysis - Querying Data

Querying the data - supporting different use cases



https://livebook.manning.com/book/graph-databases-in-action/chapter-1/v-3/

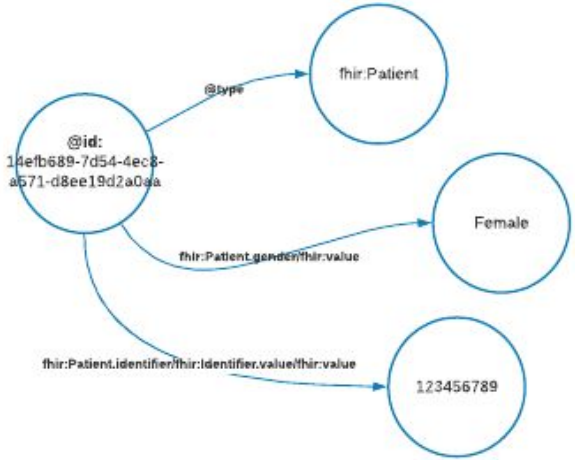
Data Store	Usage
elasticsearch	**Population-level search** Use for search-engine like queries - text based searches that returns hits based on relevance.
blazegraph™	**Patient-level data exploration** Use when you want to retrieve different types of data for a particular node and you want to analyze the relationships between the node and other data points.
PostgreSQL	**Population-level analysis** Primary use case is to support our dashboards and reports. Returns large amounts of data in a consistent format.

Querying the data - SPARQL Query

SPARQL Query

In a SPARQL query, imagine performing a graph traversal. You're starting at a particular node and tell your query which edges to traverse.

The syntax for a graph query follows a 'subject → predicate → object' pattern or 'starting node → edge → ending node'.



Test | adeel | nxv:defaultSparqlIndex

```
1 prefix fhir: <http://hl7.org/fhir/>
2
3 select ?p_identifier where {
4     ?p a fhir:Patient .
5     ?p fhir:Patient.identifier/fhir:Identifier.value/fhir:value ?p_identifier .
6     ?p fhir:Patient.gender/fhir:value 'Female' .
7 }
```

Execute SPARQL query

p_identifier
123456789

Querying the data - Elasticsearch query

```
{
    "from" : 0, "size" : 100,
    "query": {
        "match_phrase": {
            "note_content": {
                "query": "ADD concerta", "slop": 10
            }
        }
    }
}
```

```
    "total": 0
},
"hits": {
    "hits": [
        {
            "_score": 0.739864,
            "_id": "https://www.camh.ca/kcnietl/clinical_note_3",
            "_index": "delta_7d9c0a67-71fe-4ee9-abaf-53195a555484_d65c0a99-9134-4978-bc7a-9801f3ab150b_3",
            "_source": {
                "@id": "clinical_note_3",
                "@type": "clinical_note",
                "note_content": "Contact Information for the Referral Source [~~~] [~~~] 455 [~~~] [~~~] [~~~], ON [~~~]
                    Phone: [~~~] [~~~] Dr. [~~~], Thank you for referring Mr. [~~~] [~~~] to us. Today I saw [~~~] at
                    the Mood and Anxiety outpatient clinic at [~~~] and reviewed [~~~] case with Dr. [~~~], staff
                    psychiatrist. As you know [~~~] is a [~~~] year [~~~] [~~~] who lives in [~~~] by [~~~]. [~~~] works
                    as a mechanical engineer. [~~~] is currently not in any relationships and does not have any [~~~] or
                    pets. Reason for Referral/Presenting Problem/Chief Complaint [~~~] was referred to us by yourself
                    for management of symptoms of depression. History of Present Illness Today [~~~] stated that [~~~]
                    symptoms of depression started about 4 years ago. At that time [~~~] was experiencing low mood,
                    anhedonia, low energy, low motivation and poor concentration. [~~~] went toToronto [~~~] General
                    hospital at that time and was followed-up by a psychiatrist and a counsellor for couple of months
                    and got started on sertraline and bupropion which were helpful. [~~~] stopped [~~~] medications
                    after 2 years because [~~~] was feeling 'numb' and although [~~~] did not feel sad anymore, [~~~]
                    was not able to enjoy anything. [~~~] started feeling depressed again in [~~~] and Dr. [~~~] started
                    [~~~] on citalopram which was gradually increased to 30mg po daily. [~~~] has recently been started
                    on concerta for possible diagnosis of ADD. [~~~] has not found citalopram helpful. [~~~] stated that
                    [~~~] mood is sad most of the time. [~~~] used to enjoy music and reading which [~~~] is not
                    interested in anymore. [~~~] has low motivation and [~~~] concentration is poor. [~~~] has problems
                    falling asleep and has fatigue and low energy during the day. [~~~] sometimes thinks that would have
                    preferred to be dead but doesn't have any suicidal plan. [~~~] stated when [~~~] got started on
                    sertraline after about a month [~~~] experienced an episode where [~~~] only needed 1-2 hours of
                    sleep. During that time [~~~] mood was elevated and [~~~] energy level was high. [~~~] thoughts were
                    racing and [~~~] freinds were telling [~~~] that [~~~] was talking faster than usual. [~~~] was
```

concerta Aa Abl .* 1 of 5