Property Service Re-indexing

Promotion Steps
- Promote **property-services-db:v1.1.2-ba1d2417-170**
- Add https://github.com/egovernments/configs/blob/master/egov-indexer/property-services.yml to indexer

For Property tax module following two new indexes are created. Please refer index configuration in https://github.com/egovernments/configs/blob/master/egov-indexer/property-services.yml

- Index for properties (index name "property-services")
- Index for property assessments (index name "property-assessments")

- **Index for properties ("property-services")**
    - It will Index property records
    - The records will be picked from following kafka topics
        - save-property-registry :- The records on this topic are for newly created properties
        - update-property-registry :- The records on this topic are for any update on existing property
        - cancel-property-registry :- Records for cancelled properties
        - property-registry-legacyIndex :- This topic is used for reindexing of old records
    - Every time any new property is created new record is pushed on elastic search index, if existing property is updated then the existing record on index is updated. So this index stores latest information about any propertyId
    - As key for indexed records we are using combination of propertyId,tenantId fields of the property
    - Property last modified time is used as timestamp

- **Index for properties assessments ("property-assessments")**
    - It will Index property assessment records
    - The records will be picked from following kafka topics
        - save-pt-assessment :- The records on this topic are for newly created property assessments
        - update-pt-assessment :- The records on this topic are for any update on existing assessments

- cancel-property-assessment :- Records for cancelled assessments
- pt-assessment-legacyIndex :- This topic used for reindexing of old records
  - As key for indexed records we are using combination of assessment number and tenantId
  - On update or cancel assessment existing record on index is modified, so we are keeping only latest information about any assessment.
  - Assessment created time is used as timestamp

Updates from existing indexing structures

- All the PIIs are removed from Index
- On creation of new property, new record would be pushed on property index and on creation of new assessment, new record would be pushed on assessment index. And on cancellation, updates etc existing record would be updated on both indexes.
- In earlier case property details were kept inside "propertydetail" in property object, but now this field is removed and information kept inside it is part of property object itself
- Earlier if the user updates property information the new information was saved in "propertydetail" array along with old property information entries, but now we are not storing any property history and both records on property and assessment index will keep latest information about property and assessment respectively.
- Earlier property object itself used to contain assessment information, but now assessment information is separated and kept on separate assessment index
- Records on both property and assessment index contains workflow information
- Few fields are modified and added in the property object, for more information please refer "Property" object in property service API contract https://github.com/egovernments/municipal-services/blob/master/docs/property-services/property-services.yml

Since the record getting indexed on property index contains same structure (although few fields removed) the information about any field can be obtained from there.

- The structure of ward and tenantData is as before

Impact on Dashboard

- Total count of properties in the system would be total number of records on property index

- For counting assessments please use assessment index, the records includes assessment number and financial year fields which are being used in current Dashboard
- Get "usageCategoryMajor" from property object instead of propertydetail, only latest value of this field would be available for any property
- For all fields which were present in "propertydetail" except assessment information get them from property object
- Assessment index records contain propertyId field which can be used to get count of total assessment for a property from assessment index

Property Services: Tenant and date range wise re-indexing Utility

- Need to have criteria based re-indexing because in any production environments we would not be doing or needing entire data reindexing
- It would be for either for one tenant or for certain criteria like all residential properties
- Added a plainsearch API endpoints for multiple tenantIds and date range wise reindexing
- fromDate() and toDate() are to be converted to respective unix timestamps.
- Query parameters are tenantId(can be single or multiple), fromDate and toDate which are to be converted to respective unix timestamps.
- Basically Unix timestamp is a method to track time as running total of seconds. The count starts from Jan 1st 1970, which is considered as Unix Epoch. Unix timestamp is the number of seconds elapsed between a given date and unix epoch.

Reindexing commands

- **Property Index**

```
curl -X POST \
  http://egov-indexer:8080/egov-indexer/index-operations/_legacyindex \
H 'Cache-Control: no-cache' \
H 'Content-Type: application/json' \
H 'Postman-Token: d380bebb-383f-1b7c-76d1-10c1dc07dd06' \
d '{
  "RequestInfo": {
    "apiId": "string",
    "ver": "string",
    "ts": null,
    "action": "string",
    "did": "string",
    "key": "string",
    "msgId": "string",
    "authToken": "b843ef27-1ac6-49b8-ab71-cd0c22f4e50e",
    "correlationId": "e721639b-c095-40b3-86e2-acecb2cb6efb",
    "userInfo": {
```

```
      "id": 23299,
      "uuid":"e721639b-c095-40b3-86e2-acecb2cb6efb",
      "userName": "9337682030",
      "name": "Abhilash Seth",
      "type": "EMPLOYEE",
      "mobileNumber": "9337682030",
      "emailId": "abhilash.seth@gmail.com",
      "roles": [
        {
          "id": 281,
          "name": "Employee"
        }
      ]
    }
  },
  "apiDetails":{
      "uri":"http://property-services:8080/property-services/property/_plainsearch",
      "paginationDetails":{
        "offsetKey":"offset",
        "sizeKey":"limit",
        "maxPageSize":50,
        "startingOffset": 0
      },
      "responseJsonPath":"$.Properties"
  },
  "legacyIndexTopic":"property-registry-legacyIndex",
  "tenantId":"pb.amritsar"
}'
```

- **Assessment Index**

```
curl -X POST \
  http://egov-indexer:8080/egov-indexer/index-operations/_legacyindex \
H 'Cache-Control: no-cache' \
H 'Content-Type: application/json' \
H 'Postman-Token: d380bebb-383f-1b7c-76d1-10c1dc07dd06' \
d '{
  "RequestInfo": {
    "apiId": "string",
    "ver": "string",
    "ts": null,
    "action": "string",
    "did": "string",
    "key": "string",
```

```
    "msgId": "string",
    "authToken": "b843ef27-1ac6-49b8-ab71-cd0c22f4e50e",
    "correlationId": "e721639b-c095-40b3-86e2-acecb2cb6efb",
    "userInfo": {
      "id": 23299,
      "uuid":"e721639b-c095-40b3-86e2-acecb2cb6efb",
      "userName": "9337682030",
      "name": "Abhilash Seth",
      "type": "EMPLOYEE",
      "mobileNumber": "9337682030",
      "emailId": "abhilash.seth@gmail.com",
      "roles": [
        {
          "id": 281,
          "name": "Employee"
        }
      ]
    }
  },
  "apiDetails":{
    "uri":"http://property-services:8080/property-services/assessment/_plainsearch",
    "paginationDetails":{
    "offsetKey":"offset",
    "sizeKey":"limit",
    "maxPageSize":50,
      "startingOffset": 0
    },
    "responseJsonPath":"$.Assessments"
  },
  "legacyIndexTopic":"pt-assessment-legacyIndex",
  "tenantId":"pb.amritsar"
}'
```

- **Tenant and Date wise Indexing**

```
curl -X POST \
 http://localhost:8280/property-
services/property/_plainsearch?fromDate=25112020&toDate=26112020&tenantId=pb.amritsar \
H 'Cache-Control: no-cache' \
H 'Content-Type: application/json' \
d '{
"RequestInfo": {
"apiId": "Rainmaker",
"ver": ".01",
"ts": null,
"action": "search",
```

```
"did": "1",
"key": "",
"msgId": "20170310130900|en_IN",
"authToken": null,
"correlationId": "654938aa-ed8a-4d8f-8bd9-a2b73db9cbfe",
"userInfo": {
"id": 24710,
"userName": "EMP1",
"name": "EMP ONE",
"type": "EMPLOYEE",
"mobileNumber": "8888888888",
"emailId": "emp1@egovernments.org",
"roles": [
{
"id": null,
"name": "WS Counter Employee",
"code": "WS_CEMP",
"tenantId": "pb.amritsar"
},
{
"id": null,
"name": "WS Document Verifier",
"code": "WS_DOC_VERIFIER",
"tenantId": "pb.amritsar"
},
{
"id": null,
"name": "Employee",
"code": "EMPLOYEE",
"tenantId": "pb.amritsar"
}
],
"tenantId": "pb.amritsar",
"uuid": "b81e378d-0bcf-4ea9-a7eb-d24163c74b8e"
}
}
}
}'
```