

Ledningskollen API V4 Implementation guide

1. Distribution list

The document will be distributed as delivery to the PTS. To be available for PTS and project members.

Suggested audience: Developers implementing Ledningskollen API on account of cable owner.

2. Revision Information

Rev.	Date	By	Comment
PA1	2015-07-30	Daniel Garcia	Creation of document based on API V2
PA2	2015-08-28	Daniel Garcia	Adding info regarding FilterCase using POST
PA3	2015-11-15	Daniel Garcia	Updating info about Closing cases
A	2015-11-25	Daniel Garcia	Deploying LK 4.5
PB1	2016-06-15	Daniel Garcia	Including changes done for V4
PB2	2016-06-29	Daniel Garcia	Changes property ProjectInquiry.IsBroadbandBuilding for AppliesUbl, remove CollaborationInquiry properties
B	2016-09-19	Per Fröjd	Spellchecking and correcting examples for WorkCategories
C	2017-03-30	Per Fröjd	Added notes regarding CreatedWithinBankIdSession property.
D	2026-02-11	Daniel Ström	Added new api-auth endpoint Added area staging endpoints Removed support for xml responses Added new manual-review and notification endpoints

3. Contents

1. Distribution list
2. Revision Information
3. Contents
4. Getting started
 5. Create a user account in Ledningskollen.se to access the API
 - 5.1. Login Ledningskollen API
 - 5.2. New Login Ledningskollen API with managed key
 - 5.3. Logout of Ledningskollen API
 - 5.4. Getting your organization id
6. Getting cases as a cable owner
 - 6.1. Getting a specific cable owner case
 - 6.1.1. Notes
 - 6.1.2. Changes from V3
 - 6.2. Getting a list of cable owner cases
 - 6.3. Getting a list of cable owner cases by filters
 - 6.4. Getting a list of cable owner cases by filters, using a post method
 - 6.5. Paging cases
 - 6.6. Getting case pdf
 - 6.6.1. Cable owner case
 - 6.6.2. Inquirer case
 - 6.6.2.1. Change from V2
 - 6.7. Getting case attachments
7. Handling own organization recipients
 - 7.1. Getting an organization recipient
 - 7.2. Getting a list of organization recipients
8. Replying to cases (as a cable owner)
 - 8.1. Replying to a case
 - 8.2. Reassigning case to another recipient
 - 8.3. Getting correct replies per case
 - 8.4. Mark case as fetched
 - 8.5. Mark case as un fetched
9. Handling Areas
 - 9.1. Getting a specific area
 - 9.2. Getting areas
 - 9.3. Getting areas by type
 - 9.4. Deleting area

- 9.5. Adding area
- 9.6. Patching area
- 10. Posting inquiries
 - 10.1. Getting involved recipients
 - 10.2. Posting attachments
 - 10.3. Getting Work Categories
 - 10.4. Posting inquiries
 - 10.4.1. Change from V3
- 11. Getting inquirer cases
 - 11.1. Getting an inquirer case
 - 11.2. Getting inquirer cases
 - 11.3. Getting inquirer case replies
- 12. Handling inquirer cases
 - 12.1. Post inquirer case confirm
 - 12.1.1. Note
 - 12.2. Post inquirer close case
 - 12.2.1. Note
 - 12.3. Post inquirer cancel case
 - 12.3.1. Note
 - 12.4. Getting ubl information
- 13. Exporting geometries
 - 13.1. Exporting cable owner case geometries
 - 13.2. Exporting area geometries
 - 13.3. Url parameters
- 14. Areas with staging – only affected areas
 - 14.1. Api key auth token
 - 14.2. Create staging area
 - 14.3. Update staging area
 - 14.4. Request deleting area
 - 14.5. Swap staging area with main area
 - 14.6. Check if swap request is approved
 - 14.6. Check if delete request is approved
- 15 Notification messages
 - 15.1. Use case
 - 15.2. Send notification
- 16. Constants used in the API

16.1. Case Types

16.2. Case Statuses

16.3. Recipient Statuses

16.4. Area Types

16.5. Work methods

16.6. Contact way

17. API V3 changes

17.1. Initial changes from V3

17.1.1. Work Category

17.1.2. AppliesUbl (old IsBroadbandBuilding)

17.1.3. PublishableUnderUbl

17.1.4. AnswerIsconfirmed

17.1.5. Get ubl information

17.1.6. Removed properties for CollaborationInquiry

17.2. Initial changes from V4

Changes for BankID

18. Constants used in the API

18.1. Case Types

18.2. Case Statuses

18.3. Recipient Statuses

18.4. Area Types

18.5. Work methods

18.6. Contact way

19. API V3 changes

19.1. Initial changes from V3

19.1.1. Work Category

19.1.2. AppliesUbl (old IsBroadbandBuilding)

19.1.3. PublishableUnderUbl

19.1.4. AnswerIsconfirmed

19.1.5. Get ubl information

19.1.6. Removed properties for CollaborationInquiry

Initial changes from V4

Changes for BankID

4. Getting started

For the test environment use

<https://integration-v4-test-api.ledningskollen.se>

For production environment use

<https://integration-v4-api.ledningskollen.se>

5. Create a user account in Ledningskollen.se to access the API

If you do not have an account for <https://integration-v4-test-api.ledningskollen.se> you can create it here <https://idp-test.ledningskollen.se/account/create>
Contact support at support@ledningskollen.se if you want to be converted to a cableowner.

Note that in the system you can be either a “Cable owner” or an “Inquirer”.

A “cable owner” owns cables in the ground and is the one responsible for answering inquiries from the Inquirers. Their role allows them to get cases (inquiries) which may affect their cables, analyze, and give an answer to the inquirers.

The “inquirers” can be private persons or companies which for different reasons need to do some work in a specific area and use Ledningskollen.se to check if they might cause any problems or damage to any property which might be below/above the area they will work on. The inquirers can create cases as well as manage them (close them when all answers are given, cancel them if their info is incorrect).

5.1. Login Ledningskollen API

Login is done with Basic HTTP Authentication. After successful login, you will obtain a token that will be used in subsequent calls.

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/auth`
- The request must include an Authorization header containing username and password separated by a semicolon (“username:password”) and encoded using base64 encoding.

Example request :

```
POST /api/{version}/auth HTTP/1.1
Auth Header:
Authorization: Basic dGVzdHVzZXI6dGVzdHBhc3N3b3Jk
```

Example response:

```
fb0169f77d27440494af548319b7f569Fri
```

5.2. New Login Ledningskollen API with managed key

Login is done with providing credentials and your organization api key created under api key management. The api key provided must be enabled and valid. After successful login, you will obtain a token that will be used in subsequent calls.

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/v4/api-auth`

Example request:

Example request :

```
GET /api/{version}/user/organization HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response:

```
200 OK
[
  {organization guid} // returns an array of one element
]
```

6. Getting cases as a cable owner

6.1. Getting a specific cable owner case

The get cable owner case method in Ledningskollen API will return a list of CableOwnerCase objects which describe the inquiry as well as any extra information regarding previous replies (answers) from this cable owner.

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/cableownercase/{casenr string}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
POST /api/{version}/organization/{organization guid}/cableownercase/{casenr string} HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response:

```
{
  "Name": "",
  "Recipients": [ .. ],
  "Id": "",
  "Type": "",
  "Created": "",
  "Status": "",
  "Inquirer": { .. },
  "Center": { .. },
  "Regions": [ .. ],
  "CableInquiry": { .. },
  "ProjectInquiry": { .. },
  "CollaborationInquiry": { .. },
  "PlanningInquiry": { .. },
  "EmergencyInquiry": { .. },
}
```

6.1.1. Notes

CableInquiry, ProjectInquiry, CollaborationInquiry, PlanningInquiry and EmergencyInquiry are all Inquiry cases, but for each case, only one of them will have information, the rest will be null. Check the property Type to help you figure out which of them is the one you should use.

If the cable owner is affected more than once, it will be shown as the Recipients array will contain more than 1 recipient.

6.1.2. Changes from V3

The version has introduced the use of some new properties.

- WorkCategories which displays a list of the work the inquirer plans to do. This property applies for CableInquiry and Project inquiry.
- AppliesUbl (IsBroadbandBuilding) is only for Project cases. It indicates that the inquirer will be doing a broadband building work and so UBL law applies for this case.
- PublishableUnderUbl is only for Collaboration cases. It indicates that the inquirer will allow any organization tagged as Network owner (Nätinnehavare) to look at this case. Only involved organization will be able to answer, but the contact information will be available for anyone to see.
 - If this flag is set, then an additional property called UblInfo will have information. The inquirer will indicate an url or an email which can be used by the cable owner in case more information is needed or the cable owner is not part of the case.
 - The properties for this UblInfo are:
 - Url
 - Email
 - ContactName
 - Phone
 - Only Url or email are mandatory values.
- AnswerIsConfirmed, for every CableOwnerCaseRecipient reply a new property is added to indicate if the answer has already been confirmed by the inquirer. If it is false, the cable owner is still allowed to change the answer/reply.

6.2. Getting a list of cable owner cases

The get cable owner cases method in Ledningskollen API.

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/cableownercases/{status}`
 - The status field can be one of these:
 - toconfirm: cases which require your current attention, they are unanswered cases which have not been fetched (taken).
 - confirmed: cases which you have given a final answer.
 - canceled: cases which have been canceled by the inquirer.
 - historical: all cases, use mostly for archiving cases.

- taken: all cases which has been marked as fetched by a cable owner system but no action answer has been given.
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request to get a list of historical cable owner cases :

```
GET /api/{version}/organization/{organization guid}/cableownercases/historical/{optional number of cases to return} HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example request to get a list of unconfirmed cable owner cases :

```
GET /api/{version}/organization/{organization guid}/cableownercases/toconfirm/{optional number of cases to return} HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example request to get a list of confirmed cable owner cases :

```
GET /api/{version}/organization/{organization guid}/cableownercases/confirmed/{optional number of cases to return} HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example request to get a list of canceled cable owner cases :

```
GET /api/{version}/organization/{organization guid}/cableownercases/canceled/{optional number of cases to return} HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example request to get a list of taken cable owner cases :

```
GET /api/{version}/organization/{organization guid}/cableownercases/taken/{optional number of cases to return} HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example request to get a list of unconfirmed cable owner cases for only its affected areas:

```
GET /api/{version}/organization/{organization guid}/cableownercases/toconfirm/{optional number of cases to return}?aType=affected HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response:

```
[
  {
    "Name": "Case example1",
    "Recipients": [ .. ],
    "Id": "",
    "Created": "",
    "Status": "confirmed",
    "Inquirer": { .. },
```

```

"Center": { .. },
"Regions": [ .. ],
"CableInquiry": { .. },
"ProjectInquiry": { .. },
"CollaborationInquiry": { .. },
"PlanningInquiry": { .. },
"EmergencyInquiry": { .. }
},
{
  "Name": "Case example2",
  "Recipients": [ .. ],
  "Id": "",
  "Created": "",
  "Status": "confirmed",
  "Inquirer": { .. },
  "Center": { .. },
  "Regions": [ .. ],
  "CableInquiry": { .. },
  "ProjectInquiry": { .. },
  "CollaborationInquiry": { .. },
  "PlanningInquiry": { .. },
  "EmergencyInquiry": { .. }
}
]

```

6.3. Getting a list of cable owner cases by filters

The get cable owner cases method in Ledningskollen API.

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/cableownercasefilter`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request to get a list of cable owner cases :

```
GET /api/{version}/organization/{organization guid}/cableownercasefilter HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Querystring parameters (optional)

- caseNumber - get the cases after this number
- limit - maximum 100 responses
- from - date (+ time)
- to - date (+ time)
- types - case types (cable indication, emergency, project, collaboration, planning)
- status - case recipient status (toconfirm, confirmed, canceled, historical, taken)
- recipientList - list of Recipient IDs (guid)
- aType - how the case affects the organization (affected, control, other)

Example with parameters

```
GET /api/{version}/organization/{guid 36-characters}
```

```
/cableownercasefilter?types=emergency&types=project&from=2011-12-31&limit=5
```

Example response:

```
[
  {
    "Name": "Case example1",
    "Recipients":
    "Id": "",
    "Created": "",
    "Status": "confirmed",
    "Inquirer": { .. },
    "Center": { .. },
    "Regions": [ .. ],
    "CableInquiry": { .. },
    "ProjectInquiry": { .. },
    "CollaborationInquiry": { .. },
    "PlanningInquiry": { .. },
    "EmergencyInquiry": { .. }
  },
  {
    "Name": "Case example2",

    "Id": "",
    "Created": "",
    "Status": "confirmed",
    "Inquirer": { .. },
    "Center": { .. },
    "Regions": [ .. ],
    "CableInquiry": { .. },
    "ProjectInquiry": { .. },
    "CollaborationInquiry": { .. },
    "PlanningInquiry": { .. },
    "EmergencyInquiry": { .. }
  }
]
```

Example response:

Same as the get method.

6.4. Getting a list of cable owner cases by filters, using a post method

The get cable owner cases method in Ledningskollen API. It is an extension of the get method, as it includes a geographical limitation for the search.

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/cableownercasefilter`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request to get a list of cable owner cases :

```
POST /api/{version}/organization/{organization guid}/cableownercasefilter HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8
```

Data:

```
{ "limit": 100, "caseTypes": ["cable indication","project"], "from": "2015-08-01", "to": "2015-08-30", "areaType": "affected" }
```

CaseFilter class (filter options)

- caseNumber - get the cases after this number
- limit - maximum 100 responses
- from - date (+ time)
- to - date (+ time)
- caseTypes - case types (cable indication, emergency, project, collaboration, planning)
- caseStatus - case recipient status (toconfirm, confirmed, canceled, historical, taken)
- recipientList - list of Recipient IDs (guid)
- areaType - how the case affects the organization(affected, control, other)
- geoDate - a feature collection which includes just one geometry, as a geojson string.

6.5. Paging cases

It is the same call as [Getting list of cable owner cases](#) and you will get the same type of answer. Although, you can have more control by paging the calls, in that way, you can get the next 10 cases or the previous 100 cases from a specific case.

Example request to get a list of cable owner cases before a specific case:

```
GET /api/{version}/organization/{organization guid}/cableownercases/{status}/before/{caseNr string}/{optional number of cases to return} HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example request to get a list of cable owner cases after a specific case:

```
GET /api/{version}/organization/{organization guid}/cableownercases/{status}/after/{caseNr string}/{optional number of cases to return} HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

{status} is any of the statuses listed in 5.2

6.6. Getting case pdf

6.6.1. Cable owner case

Gets a cable owner case as a pdf document. The call is defined by organization guid and case number.

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/cableownercase/{caseNr string}/pdf`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
GET /api/{version}/organization/{organization guid}/cableownercase/{caseNr string}/pdf HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Response:

A stream which contains a inquirer case as a pdf document.

6.6.2. Inquirer case

Gets a inquirer case as a pdf document. The call is defined by case number.

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on /api/{version}/inquirercase/{caseNr}/pdf
- The request must include custom HTTP header "X-Auth-Token" with valid token.
- The should only be done by the case creator or his/her organization

Example request :

```
GET /api/{version}/inquirercase/{caseNr string}/pdf HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Response:

A stream which contains a cable owner case as a pdf document.

6.6.2.1. Change from V2

In the previous version (V2) the url use to end in ".pdf" but we have now changed it to "/pdf". This is to add more security to the system.

6.7. Getting case attachments

Gets an attachment the case may have. The call is defined by case number.

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on /api/{version}/inquiry/{caseNr string}/attachment/{fileName}
- The request must include custom HTTP header "X-Auth-Token" with valid token.
- The file name should be url encoded
- *Optional*: if you want a zip file (for large files) you can add ?zip=true
- *Optional*: if you want to download all attachments as a zip file, use the file name "all"

Example request to get one specific attachment:

```
GET /api/{version}/inquiry/{caseNr string}/attachment/{fileName} HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example request to get one specific attachment as a zip file:

```
GET /api/{version}/inquiry/{caseNr string}/attachment/{fileName}?zip=true HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example request to get all existing attachments as a zip file:

```
GET /api/{version}/inquiry/{caseNr string}/attachment/all HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Response:

A stream which contains the requested document or all documents.

7. Handling own organization recipients

7.1. Getting an organization recipient

Gets an specific recipient by organization guid and recipient id.

These requirements are need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/recipient/{recipient guid}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
GET /api/{version}/organization/{organization guid}/recipient/{recipient guid} HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response:

```
{
  "Id": "",
  "Name": "",
  "ContactInfo": {.. }
}
```

7.2. Getting a list of organization recipients

Gets all recipients by organization guid.

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/recipient`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
GET /api/{version}/organization/{organization guid}/recipient HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response:

```
[
  {
    "Id": "",
    "Name": "",
    "ContactInfo": {.. }
  },
  {
    "Id": "",
    "Name": "",
    "ContactInfo": {.. }
  }
]
```

8. Replying to cases (as a cable owner)

8.1. Replying to a case

Posts a reply any of the case types (Planning, Collaboration, CableInquiry, Project and Emergency).

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/organization/{organization guid }/cableownercase/{caseNr string }/{areaType string}/caserecipient/{recipient guid}/reply`
- The request must include custom HTTP header "X-Auth-Token" with valid token.
- The request must include header "Content-Type" with value "application/json; charset=utf-8"
- The request body needs to contain valid reply in json format

Reply object:

```
{
  "Comment": string (max 2000 characters),
  "Action": list of strings, // there is a pre-defined list of possible actions depending on the case type
}
```

The reply is valid if it fulfills the following requirements:

- Max length of comment is 2000 characters
- Action field can only contain valid actions for type of case on which is replying

Example request :

```
POST /api/{version}/organization/{organization guid}/cableownercase/{caseNr string}/{areaType string}/caserecipient/{recipient - guid}/reply HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8

Data:
{"Comment": "Comment", "Action": ["initialized by us","provided answer"], "Final": false }
```

Example response:

```
{
  "Time": "",
  "Comment": "",
  "Measure": [ .. ]
}
```

8.2. Reassigning case to another recipient

Assigns a case to a different recipient and closes the case for the current recipient.

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/cableownercase/{caseNr string}/{areaType string}/reassign`
- The request must include custom HTTP header "X-Auth-Token" with valid token.
- The body must include a the current and the new recipient ids in json format

Example request :

```
POST /api/{version}/organization/{organizationguid}/cableownercase/{caseNr string}/{areaType string}/reassign HTTP/1.1
```

Headers:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

```
Content-Type: application/json; charset=utf-8
```

Data:

```
{
  "CurrentRecipientId": "recipientId guid",
  "NewRecipientId": "recipientId guid"
}
```

Example response:

```
200 OK
```

```
Standard response for successful HTTP requests.
```

8.3. Getting correct replies per case

This is one of the most complex parts for the system. Knowing what actions are available per case. In order to a facilitate this, we have create a new method so the user can get them.

The available answers depend on the case type as well as the way the organization has been affected (area type).

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/organization/{organization guid} /{areaType string}/{caseType string}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
POST /api/{version}/organization/{organizationguid}/{areaType}/{caseType}
HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8
```

Example response:

```
[
  {
    "Measure": "", // the action name which should be used when replying the case
    "CableOwnerDefinition": "", // text used in Ledningskollen describing this action
    "CableOwnerHelpText": "", // help text used in Ledningskollen which might include helpful
    information to be added in the comment section when replying
    "ShortDescription": "", //short human readable description of the action
    "IsCommentMandatory": true //indicates if for this answer the comment is mandatory or not
  }
]
```

8.4. Mark case as fetched

Changes the recipient status from “requires review” to “under review”. This means that the case has been fetched and indicates that it does not have to be fetched again.

Use this when you have multiple systems using LK Api and you want to set a case aside from the others (other systems) while you take care of it. After taking a case, the user only has three options: either answers the case (reply), assigned it to another recipient or unmark it as taken (manual-review).

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/cableownercase/{string case number}/{areaType string}/caserecipient/{recipient guid}/take`
- The request must include custom HTTP header "X-Auth-Token" with valid token.
- The request must include header "Content-Type" with value “application/json; charset=utf-8”

Example request :

```
POST /api/{version}/organization/{organization guid}/cableownercase/{caseNr string}/{areaType
string}/caserecipient/{recipient guid}/take HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8
```

Example response:

```
200 OK
Standard response for successful HTTP requests.
```

8.5. Mark case as un fetched

When a case has been marked as fetched the case can then be unmarked by calling the manual review endpoint.

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/cableownercase/{string case number}/{areaType string}/caserecipient/{recipient guid}/manual-review`
- The request must include custom HTTP header "X-Auth-Token" with valid token.
- The request must include header "Content-Type" with value "application/json; charset=utf-8"

Example request :

```
POST /api/{version}/organization/{organization guid}/cableownercase/{caseNr string}/{areaType string}/caserecipient/{recipient guid}/manual-review HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8
```

Example response:

```
200 OK
Standard response for successful HTTP requests.
```

9. Handling Areas

From Ledningskollen v4.0, the system supports 2 types of areas, "affected" areas, which are the areas of interest, and the new one is the "control" area, which will not be squarified (they will be stored with the same shape and form as they are sent), read the [4.0 release notes](#) to know more about this new area type.

From V4 of the API, the Geometry property accepts only a geojson representation of a FeatureCollection. See exempld in this page "<http://geojson.org/geojson-spec.html>".

9.1. Getting a specific area

Gets a specific area by organization guid and area guid.

These requirement need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/organization/{organization guid 36-characters}/area/{area guid 36-characters}`

- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
GET /api/{version}/organization/{organization guid}/area/{area guid} HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response:

```
[
  {
    "Id": "area guid",
    "Name": "Name of area",
    "Geometry": "{ \"features\": [{ \"type\": \"Feature\", \"geometry\": { \"type\": \"Polygon\", \"coordinates\": ..... } }, { \"type\": \"FeatureCollection\" } ], // the geometries are specified in one Feature Collection in geojson
    \"Recipients\": {
      \"cable indication\": \"recipient guid\",
      \"emergency\": \"recipient guid\",
      \"collaboration\": \"recipient guid\",
      \"planning\": \"recipient guid\",
      \"project\": \"recipient guid\"
    }
    \"AreaType\": \"affected\" //the alternative is: control
  }
]
```

9.2. Getting areas

Getting all areas (both areas of interest and control areas) by organization guid.

These requirement need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on /api/{version}/organization/{organization guid}/area
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
GET /api/{version}/organization/{organization guid}/area HTTP/1.1
```

Header:

```
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response:

```
[
  {
    "Id": "area guid",
    "Name": "Name of area",
    "Geometry": "{ \"features\": [{ \"type\": \"Feature\", \"geometry\": { \"type\": \"Polygon\", \"coordinates\": ..... } }, { \"type\": \"FeatureCollection\" } ],
    \"Recipients\": { ... }
    \"AreaType\": \"area type\"
  }
]
```

```

},
...{more entries}
}

```

9.3. Getting areas by type

Getting areas by organization guid and type.

The requirements needs to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/area/{area type string}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

The {areatype} can be one of: affected (area of interest) or control (control area)

Example request :

```

GET /api/{version}/organization/{organization guid}/area/{area type} HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3

```

Example response:

```

[
  {
    "Id": "area guid",
    "Name": "",
    "Geometry": "{\n\"features\": [\n{\n\"type\": \"Feature\", \"geometry\":\n: {\n\"type\": \"Polygon\", \"coordinates\": .... }],\n\"type\": \"FeatureCollection\"},,\n    \"Recipients\": { ... }\n    \"AreaType\": \"area type\"\n  },
  ...{more entries}
}

```

9.4. Deleting area

Deleting an area by organization guid and area guid.

These requirements need to be fulfilled:

- The request must be a HTTP DELETE request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/area/{area guid}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```

DELETE /api/{version}/organization/{organization guid}/area/{area guid} HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3

```

Example response:

200 OK
Standard response for successful HTTP requests.

9.5. Adding area

Adding an area by organization guid.

These requirement need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/area`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

When creating an area of interest (affected) or control area (control) the following case recipient rules apply:

	cable indication	project	planning	collaboration	emergency
affected	mandatory	mandatory	mandatory	not allowed	mandatory
control	optional	optional	optional	mandatory	not allowed

Example request :

```
POST /api/{version}/organization/{organization guid}/area HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3

Data:
  "Geometry": "{\"features\": [{\"type\": \"Feature\", \"geometry\": {\"type\": \"Polygon\", \"coordinates\":
  ....}], \"type\": \"FeatureCollection\"},
  "Name": "My New Area",
  "Recipients": {
    "cable indication": "3f41b4a-4a20-4d96-bc58-9012d48062af",
    "project": "3f41b4a-4a20-4d96-bc58-9012d48062af",
    "planning": "16aed5e9-5c89-45ee-a714-ad92b78d3542",
    "emergency": "16aed5e9-5c89-45ee-a714-ad92b78d3542"
  }
  "AreaType": "affected" // possible options list
}
```

Example response:

```
{
  "Id": "area guid",
  "Name": "My New Area",
  "Geometry": "{\"features\": [{\"type\": \"Feature\", \"geometry\": {\"type\": \"Polygon\", \"coordinates\":
  ....}], \"type\": \"FeatureCollection\"},
  "Recipients": { ... }
},
```

9.6. Patching area

Patching/updating an area by organization guid.

These requirements need to be fulfilled:

- The request must be a HTTP PUT request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/area`
- The request must include custom HTTP header "X-Auth-Token" with valid token.
- Name parameter is optional (only if you want to change the name)
- Recipients parameter is optional (only if you want to change any of them). If you want to delete a recipient (only possible for the optional recipients for control areas see 9.5) leave the value blank (i.e "cable indication" : "")

The AreaType is not passed, since it is not allowed to change a control area into an affected area or viceversa.

Example request :

```
PUT /api/{version}/organization/{organization guid}/area HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3

Data:
{
  "Id": "16aed5e9-5c89-45ee-a714-ad92b78d3542", // area guid
  "Geometry": {"features":[{"type":"Feature","geometry":{"type":"Polygon","coordinates":
  ....}}],"type":"FeatureCollection"},
  "Name": "Put name if you want to change it, else leave blank",
  "Recipients": { // set only the recipients you need to change
    "cable indication": "16aed5e9-5c89-45ee-a714-ad92b78d3542",
  }
}
```

Example response:

```
{
  "Id": "area guid",
  "Name": "Name of your area",
  "Geometry": {"features":[{"type":"Feature","geometry":{"type":"Polygon","coordinates":
  ....}}],"type":"FeatureCollection"},
  "Recipients": { ... }
},
```

10. Posting inquiries

The following method should only be used by Inquirers in order to create cases or get them.

10.1. Getting involved recipients

Gets the involved recipients for specific area by GeoJson information.

This method should only be used before creating a case in order to use that information to deselect/exclude organizations which the user already has information.

These requirement need to be fulfilled:

- The request must be a HTTP POST request and have the GeoJson information.
- Request need to be issued on `/api/{version}/inquiry/{inquiry type}/involvedrecipients`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
POST /api/{version}/inquiry/{inquiry type}/involvedrecipients HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8

Data:
{
  "Inquiry": { (properties can be common or separate for each inquiry type)
    "Comment": "Comment about why, or what work needs to be done, max 2000 characters",
    "EndDate": "2014-05-26T00:00:00.0000000+02:00",
    "Geometry": "{\n\"features\": [\n  {\n    \"type\": \"Feature\",\n    \"geometry\":\n    :{\n      \"type\": \"Polygon\",\n      \"coordinates\": ....\n    },\n    \"type\": \"FeatureCollection\" }\n  ],\n  //The geometry should be a FeatureCollection as a geojson string
    \"Name\": \"Name of the case so it's unique to you\",
    \"StartDate\": \"2014-04-26T00:00:00.0000000+02:00\",
    \"NotificationViaSms\": \"0123456789\",
    \"NotifyOnReply\": true,
    \"ExcavationDepth\": 1,
    \"MeetUpAddress\": {\n      \"CityName\": \"city\",
      \"PostCode\": \"area code\",
      \"StreetNameAndNumber\": \"street and number\"
    },
    \"PreferredContactValue\": \"0123456789\",
    \"PreferredContactWay\": \"fax\", // possible options list
    \"PropertyDesignation\": \"FSAS 123:01\",
    \"SiteContactName\": \"Test Testperson\",
    \"SiteContactPhone\": \"0123456789\",
    \"WorkMethods\": [\n      \"machine earthwork\" // possible options list
    ]
  }
}
```

The {inquirytype} can be one of: cableinquiry, collaborationinquiry, emergencyinquiry, planninginquiry or projectinquiry.

Example response:

```
[
  {
    "Organization": {
      "Id": "organization guid",
      "Name": "",
      "OrganizationNumber": ""
    },
    "Confirmed": false,
    "Recipient": {
      "Id": "recipient guid",
      "Name": "",
      "ContactInfo": {
        "Email": "",
        "PhoneNumber": "",
        "FaxNumber": ""
      }
    }
  }
]
```

```
    "Address": ""
  }
},
"Status": "",
"Events": null,
"ResponseTime": 1,
"PhysicalResponseTime": 1
},
...
]
```

10.2. Posting attachments

Posts the attachments before creating the inquiry so you don't have to redo everything if there was an error.

These requirements need to be fulfilled:

- The request must be a HTTP POST request and have a dictionary type of filenames and bytearrays.
- Request needs to be issued on `/api/{version}/inquiry/attachments`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
POST /api/{version}/inquiry/attachments HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3

Data:
{
  "Filename.txt": [239,187,191,65,116,116,97,99,104,109,101,110,116],
  ...
}
```

Executable files (.exe and .dll) will be excluded from the upload.

Example response (you will have to use these ids in order to attach files in inquiries):

```
[
  2027,
  ...
]
```

10.3. Getting Work Categories

Gets a list of work categories which can describe the type of work which will be done for a certain type of case.

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request needs to be issued on `/api/{version}/inquiry/workCategories/{caseType}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
Get /api/{version}/inquiry/workCategories/{caseType} HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response (you will have to use these ids in order to attach files in inquiries):

```
[
  {
    "MainCategory": int,
    "SubCategory": "string",
    "Name": "string",
    "UniqueName": "string", // this value should be sent back when posting an inquiry
    "Order": int
  }
]
```

10.4. Posting inquiries

Posts the inquiries and validates the result. You may want to read the exception details in order to figure out what's required.

These requirement need to be fulfilled:

- The request must be a HTTP POST request and have a create inquiry.
- Request need to be issued on /api/{version}/inquiry/{inquiry type}
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
POST /api/{version}/inquiry/cableinquiry HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Data:
{
  "Attachments": (null or the list of ids from 10.2 - common for all posting of inquiries),
  "ExemptedOrganzations": (null or the list of Organization.Id from 10.1 - common for all posting of inquiries),
  "Inquiry": { (properties can be common or separate for each inquiry type)
    "Comment": "Comment about why, or what work needs to be done, max 2000 characters",
    "EndDate": "2014-05-26T00:00:00.0000000+02:00",
    "Geometry": "{ \"features\": [{ \"type\": \"Feature\", \"geometry\": { \"type\": \"Polygon\", \"coordinates\": [...] } }, { \"type\": \"FeatureCollection\" } ], //The geometry should be a FeatureCollection as a geojson string
    "Name": "Name of the case so it's unique to you",
    "StartDate": "2014-04-26T00:00:00.0000000+02:00",
    "NotificationViaSms": "0123456789",
    "NotifyOnReply": true,
    "ExcavationDepth": 1,
    "MeetUpAddress": {
      "CityName": "city",
      "PostCode": "area code",
      "StreetNameAndNumber": "street and number"
    },
    "PreferredContactValue": "0123456789",
    "PreferredContactWay": "fax", // possible options list
  }
}
```

```

"PropertyDesignation": "FSAS 123:01",
"SiteContactName": "Test Testperson",
"SiteContactPhone": "0123456789",
"WorkMethods": [
  "machine earthwork" // possible options list
],
"WorkCategory": [
  "1_1_EL", "1_3_DIKNING" // use Getting work categories to see options
]
}

```

The `{inquirytype}` can be one of: `cableinquiry`, `collaborationinquiry`, `emergencyinquiry`, `planninginquiry` or `projectinquiry`.

Example response for `/api/{version}/inquiry/cableinquiry`:

```

{
  "PropertyDesignation": "FSAS 123:01",
  "SiteContactName": "Test Testperson",
  "SiteContactPhone": "0123456789",
  "MeetUpAddress": {
    "StreetNameAndNumber": "street and number, area code city",
    "PostCode": null,
    "CityName": null
  },
  "WorkMethods": [
    "Maskinschaktning"
  ],
  "ExcavationDepth": 1,
  "PreferedContactWay": "fax",
  "PreferedContactValue": "0123456789",
  "Attachments": [
    "http://{server}/jwpservices/DownloadFile.ashx?att=true&case=20140425-1035&file=TextFile.txt"
  ],
  "NotifyOnReply": true,
  "NotificationViaSms": "0123456789",
  "CreatedUsing": "webservice",
  "Purpose": "cable indication",
  "Name": "20140425-1035, Name of the case so it's unique to you",
  "Comment": "Comment about why, or what work needs to be done, max 2000 characters",
  "Geometry": "{\"features\": [{\"type\": \"Feature\", \"geometry\": {\"type\": \"Polygon\", \"coordinates\": ...}], \"type\": \"FeatureCollection\"}", // The geometry should be a FeatureCollection as a geojson strings.
  "StartDate": "2014-04-26T00:00:00.0000000+02:00",
  "EndDate": "2014-05-26T00:00:00.0000000+02:00"
}

```

10.4.1. Change from V3

There is some extra information which should be added for certain type cases
For Cable Indication, Project and Collaboration inquiries, the property `WorkCategory` is mandatory.

For Collaboration cases there is a new flag called PublishableUnderUbl which will indicate if the case should be treated within the UBL law. Note, this only applies for organizations which are Network owners (Nätinnehavare).

11. Getting inquirer cases

The following method should only be used by Inquirers in order to create cases or get them.

11.1. Getting an inquirer case

Gets an inquirer case by case number.

The full case information will be returned if the user is the case creator or belongs to the organization who has created the case.

These requirement need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/inquirercase/{string case number}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
GET /api/{version}/inquirercase/{string case number} HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Cache-Control: no-cache
```

Example response:

```
{
  "InvolvedRecipients": [...], // Recipients which have to reply to your inquiry
  "InterestedRecipients": [...], // Recipients which are interested in the area where you have created
  the case. They don't have to reply
  "OtherKnownOrganizations": [...],
  "Id": "",
  "Created": "",
  "Status": "",
  "Inquirer": {...},
  "Center": {...},
  "CableInquiry": {...},
  "CollaborationInquiry": {...},
  "EmergencyInquiry": {...},
  "PlanningInquiry": {...},
  "ProjectInquiry": {...},
  "Regions": [...],
  "Type": ""
}
```

11.2. Getting inquirer cases

Gets inquirer cases that are open or confirmed, the call can has a limit on the amount of cases to be returned but if not set, the system will set a default limit.

These requirement need to be fulfilled:

- The request must be a HTTP GET request.

- Request need to be issued on /api/{version}/user/inquirercase/{limit int}
- The request must include custom HTTP header "X-Auth-Token" with valid token.
- If you add before or after and a case number to the url you can apply paging
- If you add status to the url you can filter specific case statuses

Example request :

```
GET /api/{version}/user/inquirercase/{limit int} HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Cache-Control: no-cache
```

Example request with before and case number :

```
GET /api/{version}/user/inquirercase/before/{casenr string}/{limit int} HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Cache-Control: no-cache
```

Example request with after and case number :

```
GET /api/{version}/user/inquirercase/after/{casenr string}/{limit int} HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Cache-Control: no-cache
```

Example request with status: (confirmed, open, canceled, closed or preliminary)

```
GET /api/{version}/user/inquirercase/{status}/limit int} HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Cache-Control: no-cache
```

Example request with status, before and case number: (confirmed, open, canceled, closed or preliminary)

```
GET /api/{version}/user/inquirercase/{status}/before/{casenr string}/{limit int} HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Cache-Control: no-cache
```

Example request with status, after and case number: (confirmed, open, canceled or closed)

```
GET /api/{version}/user/inquirercase/{status}/after/{casenr string}/{limit int} HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Cache-Control: no-cache
```

Example response:

```
[
{
  "InvolvedRecipients": [...],
  "InterestedRecipients": [...],
  "OtherKnownOrganizations": [...],
```

```
"Id": "",
"Created": "",
"Status": "",
"Inquirer": {...},
"Center": {...},
"CableInquiry": {...},
"CollaborationInquiry": {...},
"EmergencyInquiry": {...},
"PlanningInquiry": {...},
"ProjectInquiry": {...},
"Regions": [...],
"Type": ""
}, ...
]
```

11.3. Getting inquirer case replies

Gets a list of inquiry case recipients for which the inquirer can do confirmations.

Use this method so you only get the replies from the cable owners for a specific case.

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/inquirercase/{string case number}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
GET /api/{version}/inquirercase/{casenr string}caserecipient/toconfirm HTTP/1.1
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Cache-Control: no-cache
```

Example response:

```
[
  {
    "Organization": {...},
    "Confirmed": false,
    "Recipient": {...},
    "Status": "",
    "Events": {...},
    "ResponseTime": 0,
    "PhysicalResponseTime": 0
  },
  {
    "Organization": {...},
    "Confirmed": false,
    "Recipient": {...},
    "Status": "",
    "Events": {...},
    "ResponseTime": 0,
    "PhysicalResponseTime": 0
  }
]
```

12. Handling inquirer cases

The following method should only be used by Inquirers in order to create cases or get them.

12.1. Post inquirer case confirm

Post method for the inquirer to confirm/accept the answer given by a representative of an organization. This method should be called after the cable owner organization has given their final answer.

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/inquirercase/{string case number}/caserecipient/{recipient guid 36-characters}/confirm`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
POST /api/{version}/inquirercase/{caseNr string}/caserecipient/{recipient guid}/confirm HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response:

```
200 OK
Standard response for successful HTTP requests.
```

12.1.1. Note

Only answers given by the InvolvedRecipients can be confirmed. If you confirm a reply from an InterestedRecipient, no errors will be given, but it will take no effect.

12.2. Post inquirer close case

Post method for the inquirer to close the case. This should be called once the inquirer has accepted all the answers from the involved organizations, or the case is no longer valid (after the end date has passed).

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/inquirercase/{caseNr string}/close`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
POST /api/{version}/inquirercase/{caseNr string}/close HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response:

```
200 OK
Standard response for successful HTTP requests.
```

12.2.1. Note

Cases can now be closed even if not all organizations have given an answer. For “Ledningsanvisning” cases, the case can be closed 30 days after the end date. For all other case types, they can be closed after the end date.

12.3. Post inquirer cancel case

Post method for the inquirer to cancel the case. This should be called if the case is no longer valid because the plans have changed or the information given is no longer valid

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/{version}/inquirercase/{caseNr string}/cancel`
- The request must include custom HTTP header "X-Auth-Token" with valid token.
- The request must include header "Content-Type" with value “application/json; charset=utf-8”
- The request body needs to contain valid CancelCase object in json format

CancelCase object:

```
{
  "Reason": string, // exist predefined reasons
  "Comment":string (max 2000 characters)
}
```

The CancelCase object is valid if it fulfills the following requirements:

- Reason of cancellation can be applied on case.
- Max length of comment is 2000 characters.

Example request :

```
POST /api/{version}/inquirercase/{caseNr string}/close HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8

Data:
{ "Reason": "incorrect_information", "Comment": "text" }
```

Example response:

```
200 OK
Standard response for successful HTTP requests.
```

12.3.1. Note

The cancellation of cases is only available for a short period of time. The time depends on the case type and if all the involved organizations have given an answer (then it can't be cancel any more).

12.4. Getting ubl information

This applies for Project cases which are under the UBL law. If the organization has given a valid answer and is willing to collaborate, the inquirer can request the contact information in order to apply for more information.

These requirements are need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on
`/api/{version}/inquirercase/{caseNr}/caserecipient/{recipientId}/ublinformation/recipient/{recipient guid}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
GET /api/{version}/inquirercase/{caseNr}/caserecipient/{recipientId}/ublinformation HTTP/1.1
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
```

Example response:

```
{
  "InformationNotSet": true,
  "ApplicationUrl": "string",
  "ApplicationEmail": "string",
  "ApplicationComment": "string",
  "OrganizationName": "string"
}
```

If there is no information set by the organization then the flag `InformationNotSet` will be set to true. The organization can either set a url where the application can be done or uploaded or it can give an email (which will include also a comment).

13. Exporting geometries

13.1. Exporting cable owner case geometries

Get method for the cable owner case to export the geometries.

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/cableownercase/{caseNr string}/geometries/{coordinateSystem int}/{dataFormat string}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
GET /api/{version}/organization/{organization guid}/cableownercase/{caseNr
string}/geometries/{coordinateSystem int}/{dataFormat string}
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8
```

Example response:

```
200 OK
Standard response for successful HTTP requests.
ByteArray of a zip-file
```

13.2. Exporting area geometries

Get method to export the geometries of an area.

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/{version}/organization/{organization guid}/area/{area guid}/geometries/{coordinateSystem int}/{dataFormat string}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request :

```
GET /api/{version}/organization/{organization guid}/area/geometries/{coordinateSystem int}/{dataFormat string}
Headers:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8
```

Example response:

```
200 OK
Standard response for successful HTTP requests.
ByteArray of a zip-file
```

13.3. Url parameters

For both export cases, export of areas or geometries for cable owner, there is the same format: `/api/{coordinateSystem int}/{dataFormat string}`.

where:

- coordinateSystem can be one of:
 - 2400, 3006, 3007, 3008, 3009, 3010, 3011, 3012, 3013, 3014, 3015, 3016, 3017, 3018, 4326
- dataFormat can be one of:
 - GML, SHAPE, WKT, DXF, MAPINFO, MAPINFO_TAB

14. Areas with staging - only affected areas

14.1. Api key auth token

It is important to notice that all endpoints for the staging areas requires a token from the auth endpoint `/api/v4/api-auth` provided to the X-Auth-Token header. Example login in section 5.2.1

14.2. Create staging area

When creating or updating areas with staging it can require the user to approve the swapping of areas before the swap can be completed. This is done inside of the cable owner application by cable owner admins.

POST method for creating an affected area staging

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/v4/area-staging`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request:

```
POST /api/v4/area-staging
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8

Data:
{
  "name": "Name of area",
  "makeSquare": true,
  "geometries": [
    {
      "type": "Polygon",
      "coordinates": [
        [
          [357523.987, 6392411.381],
          [357525.973, 6392161.596],
          [357830.021, 6392169.618],
          [357841.991, 6392399.401],
          [357523.987, 6392411.381]
        ]
      ]
    },
    {
      "type": "Polygon",
      "coordinates": [
        [
          [359379.972, 6392488.514],
          [359378.026, 6392341.708],
          [359586.022, 6392353.874],
          [359574.008, 6392520.93],
          [359379.972, 6392488.514]
        ]
      ]
    }
  ]
}
```

Example response:

```
200 OK
{
  "ArealId": "00000000-0000-0000-0000-000000000000",
  "Status": "AutomaticallySwapped"
}
```

14.3. Update staging area

PUT method for updating an affected area staging

These requirements need to be fulfilled:

- The request must be a HTTP PUT request.
- Request need to be issued on `/api/v4/area-staging`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request:

```
PUT /api/v4/area-staging
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8

Data:
{
  "Areald": "00000000-0000-0000-0000-000000000000",
  "MakeSquare": true,
  "Geometries": [
    {
      "type": "Polygon",
      "coordinates": [
        [
          [324523.987, 6389411.381],
          [324525.973, 6389161.596],
          [324830.021, 6389169.618],
          [324841.991, 6389399.401],
          [324523.987, 6389411.381]
        ]
      ]
    }
  ]
}
```

Example response:

```
200 OK
{
  "Status": "AutomaticallySwapped"
}
```

14.4. Request deleting area

DELETE method for requesting to remove an area and the staging slot

These requirements need to be fulfilled:

- The request must be a HTTP DELETE request.
- Request need to be issued on `/api/v4/area-staging/{Areald}`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request:

```
DELETE /api/v4/area-staging/{Areald guid}
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8
```

Example response:

```
204 No Content
```

14.5. Swap staging area with main area

PUT method for swapping staging and area slots

These requirements need to be fulfilled:

- The request must be a HTTP DELETE request.
- Request need to be issued on `/api/v4/area-staging/{AreaGuid}/swap`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request:

```
DELETE /api/v4/area-staging/{Areald guid}/swap  
Header:  
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3  
Content-Type: application/json; charset=utf-8
```

Example response:

```
204 No Content
```

14.6. Check if swap request is approved

GET method for getting the swap approval status

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/v4/area-staging/{AreaGuid}/is-swap-approved`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request:

```
GET /api/v4/area-staging/{Areald guid}/is-swap-approved  
Header:  
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3  
Content-Type: application/json; charset=utf-8
```

Example response:

```
200 OK  
True
```

14.6. Check if delete request is approved

GET method for getting the delete approval status

These requirements need to be fulfilled:

- The request must be a HTTP GET request.
- Request need to be issued on `/api/v4/area-staging/{AreaGuid}/is-delete-approved`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request:

```
GET /api/v4/area-staging/{Areald guid}/is-delete-approved
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8
```

Example response:

```
200 OK
True
```

15 Notification messages

15.1. Use case

Send information emails about your integration's status to the assigned email addresses of you API-key settings.

15.2. Send notification

POST method for sending information messages

These requirements need to be fulfilled:

- The request must be a HTTP POST request.
- Request need to be issued on `/api/v4/organization/notification`
- The request must include custom HTTP header "X-Auth-Token" with valid token.

Example request:

```
POST /api/v4/organization/notification
Header:
X-Auth-Token: 9ba0515a9c484ce4b21bf6297e8832c3
Content-Type: application/json; charset=utf-8

Data:
{
  "Header": "Text displayed in the email header",
  "Body": "Text displayed in the email body",
  "NotificationType": "Informational",
  "NotificationLevel": "Info"
}
```

Example response:

```
204 No Content
```

16. Constants used in the API

16.1. Case Types

LK name	API constant name
Ledningsanvisning	cable indication
Projektering	project
Samordning	collaboration
Planärende / Samhällsplanering	planning
Akut	emergency

16.2. Case Statuses

Here is the list of possible case statuses. Notice that case status and recipient status is not the same thing. The case status will change when you as an inquirer confirm, cancel or close a case.

LK name	API constant name
Öppet	open
Bekräftat	confirmed
Makulerat	canceled
Stängt	closed

16.3. Recipient Statuses

Here is the list of recipient statuses. When you as a cable owner get cases you will encounter these statuses. You can also fetch cases depending on the recipient status.

LK name	API constant name
Måste granskas manuellt	toconfirm
Under granskning	taken
Bekräftat	confirmed
Makulerat	canceled

16.4. Area Types

LK name	API constant name
Intresseområde	affected
Bevakningsområde	control

16.5. Work methods

Here is the list of the accepted types of work for each of the case types. Note that project and emergency cases do not have/handle this type of information.

LK name	API constant name
cable indication	<ul style="list-style-type: none"> ● machine earthwork ● milling plowing ● drilling ● blasting ● pilling sheetpiling ● logging ● ground work ● lake routing
project	-
collaboration	<ul style="list-style-type: none"> ● road construction ● walk bike road construction ● fiber channel ● asphalt area ● pipes for water ● pipes for electricity ● pipes for telecommunication ● pipes for heating ● living area ● commercial area ● industrial area ● other
planning	<ul style="list-style-type: none"> ● vehicle road ● pedestrian and bicycle road ● park ● nature ● square ● cables ● living area b ● city center c ● technical facilities e ● commercial area h ● industrial area j

	<ul style="list-style-type: none"> ● office area k ● other
emergency	☒ Same as cable indication

16.6. Contact way

When creating cases, for some type of cases you will need to define a preferred contact way.

Case type	allowed contact way
cable indication	<ul style="list-style-type: none"> ● email
project	<ul style="list-style-type: none"> ● email
collaboration	-
planning	<ul style="list-style-type: none"> ● email
emergency	☒ email

17. API V3 changes

17.1. Initial changes from V3

There are not too many changes from V3, mainly more information has been added to the cases.

17.1.1. Work Category

CableInquiry, ProjectInquiry, CollaborationInquiry and PlanningInquiry have this new property, although it does not apply for CollaborationInquiry nor PlanningInquiry (it is there because of inheritance).

This property give a better indication of what type of work will be done for this case. In order to get a list of valid WorkCategories per case, you can use the request described [here](#).

17.1.2. AppliesUbl (old IsBroadbandBuilding)

This is a flag set for the cable owner. It will indicate if for this case, the UBL law can be applied. This flag may be true for Collaboration or Project cases for which the involved organization is tagged as a broadband build or network owner, respectively.

If the case has this flag set, the network owners (“nätinnehavare”, defined in “*lag (2016:534) om åtgärder för utbyggnad av bredbandsnät*”) have the responsibility to answer with more information than in a normal project inquiry. Example of the extra information that shall be provided in the response is for example buildings, entrances to buildings, masts, wells etc. More or less anything that can be useful for building broadband networks.

17.1.3. PublishableUnderUbl

This flag is only for Collaboration cases. It indicates that any organization that is a Broadband builder (“bredbandsutbyggare”, defined in “*lag (2016:534) om åtgärder för utbyggnad av bredbandsnät*”) may have legal rights to demand collaboration with the project. Only involved broadband builder organizations will be able to answer, but the information in the case will be available for anyone in broadband building organizations to see.

When this flag is set, the property UblInfo is also set which will either have a Url or an email which can be used by the broadband builder to contact the inquirer and apply for collaboration.

17.1.4. AnswerIsconfirmed

For every CableOwnerCaseRecipient reply a new property is added to indicate if the answer has already been confirmed by the inquirer. If it is false, the cable owner is still allowed to change the answer/reply.

17.1.5. Get ubl information

There is also a new method which can be used by the inquirer for Project cases where the cable owner has accepted collaboration for an UBL related case.

The inquirer can request contact information by using this [method](#).

17.1.6. Removed properties for CollaborationInquiry

For the class CollaborationInquiry, we have removed two properties called:

- PreferredContactWay
- PreferredContactValue

These are removed as since API V3, they will return empty or null values as current cases no longer request this information from the inquirer.

17.2. Initial changes from V4

These are the changes to the API for the 5.4 version update to Ledningskollen.

Changes for BankID

On each inquirer case fetched from the API, we now include a property called “CreatedWithinBankIdSession” to indicate if the inquirer was logged in through BankID when creating the inquirer case. If it is false, the inquirer was logged in through normal means (credentials).

This will not indicate whether the inquirer has connected their user to BankID overall, but only if they were logged in through means of BankID.

18. Constants used in the API

18.1. Case Types

LK name	API constant name
Ledningsanvisning	cable indication
Projektering	project
Samordning	collaboration
Planärende / Samhällsplanering	planning
Akut	emergency

18.2. Case Statuses

Here is the list of possible case statuses. Notice that case status and recipient status is not the same thing. The case status will change when you as an inquirer confirm, cancel or close a case.

LK name	API constant name
Öppet	open
Bekräftat	confirmed
Makulerat	canceled
Stängt	closed

18.3. Recipient Statuses

Here is the list of recipient statuses. When you as a cable owner get cases you will encounter these statuses. You can also fetch cases depending on the recipient status.

LK name	API constant name
Måste granskas manuellt	toconfirm
Under granskning	taken
Bekräftat	confirmed
Makulerat	canceled

18.4. Area Types

LK name	API constant name
Intresseområde	affected
Bevakningsområde	control

18.5. Work methods

Here is the list of the accepted types of work for each of the case types. Note that project and emergency cases do not have/handle this type of information.

LK name	API constant name
cable indication	<ul style="list-style-type: none"> ● machine earthwork ● milling plowing ● drilling ● blasting ● pilling sheetpiling ● logging ● ground work ● lake routing
project	-
collaboration	<ul style="list-style-type: none"> ● road construction ● walk bike road construction ● fiber channel ● asphalt area ● pipes for water ● pipes for electricity ● pipes for telecommunication ● pipes for heating ● living area ● commercial area ● industrial area ● other
planning	<ul style="list-style-type: none"> ● vehicle road ● pedestrian and bicycle road ● park ● nature ● square ● cables ● living area b ● city center c ● technical facilities e ● commercial area h ● industrial area j

	<ul style="list-style-type: none"> • office area k • other
emergency	(Same as cable indication)

18.6. Contact way

When creating cases, for some type of cases you will need to define a preferred contact way.

Case type	allowed contact way
cable indication	<ul style="list-style-type: none"> • email
project	<ul style="list-style-type: none"> • email
collaboration	-
planning	<ul style="list-style-type: none"> • email
emergency	<ul style="list-style-type: none"> ∉ email

19. API V3 changes

19.1. Initial changes from V3

There are not too many changes from V3, mainly more information has been added to the cases.

19.1.1. Work Category

CableInquiry, ProjectInquiry, CollaborationInquiry and PlanningInquiry have this new property, although it does not apply for CollaborationInquiry nor PlanningInquiry (it is there because of inheritance).

This property give a better indication of what type of work will be done for this case. In order to get a list of valid WorkCategories per case, you can use the request described [here](#).

19.1.2. AppliesUbl (old IsBroadbandBuilding)

This is a flag set for the cable owner. It will indicate if for this case, the UBL law can be applied. This flag may be true for Collaboration or Project cases for which the involved organization is tagged as a broadband build or network owner, respectively.

If the case has this flag set, the network owners (“nätinnehavare”, defined in “*lag (2016:534) om åtgärder för utbyggnad av bredbandsnät*”) have the responsibility to answer with more information than in a normal project inquiry. Example of the extra information that shall be provided in the response is for example buildings, entrances to buildings, masts, wells etc. More or less anything that can be useful for building broadband networks.

19.1.3. PublishableUnderUbl

This flag is only for Collaboration cases. It indicates that any organization that is a Broadband builder (“bredbandsutbyggare”, defined in “*lag (2016:534) om åtgärder för utbyggnad av bredbandsnät*”) may have legal rights to demand collaboration with the project. Only involved broadband builder organizations will be able to answer, but the information in the case will be available for anyone in broadband building organizations to see.

When this flag is set, the property UblInfo is also set which will either have a Url or an email which can be used by the broadband builder to contact the inquirer and apply for collaboration.

19.1.4. AnswerIsconfirmed

For every CableOwnerCaseRecipient reply a new property is added to indicate if the answer has already been confirmed by the inquirer. If it is false, the cable owner is still allowed to change the answer/reply.

19.1.5. Get ubl information

There is also a new method which can be used by the inquirer for Project cases where the cable owner has accepted collaboration for an UBL related case.

The inquirer can request contact information by using this [method](#).

19.1.6. Removed properties for CollaborationInquiry

For the class CollaborationInquiry, we have removed two properties called:

- PreferredContactWay
- PreferredContactValue

These are removed as since API V3, they will return empty or null values as current cases no longer request this information from the inquirer.

Initial changes from V4

These are the changes to the API for the 5.4 version update to Ledningskollen.

Changes for BankID

On each inquirer case fetched from the API, we now include a property called “CreatedWithinBankIdSession” to indicate if the inquirer was logged in through BankID when creating the inquirer case. If it is false, the inquirer was logged in through normal means (credentials).

This will not indicate whether the inquirer has connected their user to BankID overall, but only if they were logged in through means of BankID.