ADS interfacing manual

Version 2024

Customer: Land-Board

Author: AS Datel

Euroopa Liit
Euroopa
Regionaalarengu Fond

Eesti
tuleviku heaks

Versions

Version	Date	Author	Explanation
V 2022	01.09.2022	AS Datel, Land Board	A new version of the interfacing manual created as a result of the RAAS-3 development project
	11.11.2022	Land Board	Clarification of terms
	30.12.2022	Land Board	Clarification of object classification
V 2023	25.07.2023	Land Board	Clarification of the automatic address renewal process
	11.09.2023	Land Board	Clarification on the In-ADS container solution
	15.09.2023	Land Board	Clarification on parallel addresses
	11.10.2023	Land Board	Sample solution for address selection using In-ADS, Clarification of terms
	24.10.2023	Land Board	Clarifications on the update process for interfacer’s ADS data tables
V 2024	23.04.2024	AS Datel, Land Board	Clarifications regarding new services

Table of contents

1 Introduction

This document is intended for reading by business and IT analysts participating in the design of information systems and other interested parties who want information about the interfacing of the address data system with the information system. You can read more about the address data system in the Estonian Land Board Address Data (ADS) Handbook .

The ADS interfacing guide also includes the document "**ADS X-Tee Services**", which contains descriptions of the X-Tee services provided by ADS. This document always refers to the latest version of the service, i.e. earlier versions do not have all data fields and log events. Additional information about ADS X-tee services can also be found in the address system section of the Geoportal of the Land Board.

In everyday communication, the address seems to be a simple attribute that can be expressed by a single word such as "address", "residence", "location", "postal address" and others. The simple approach is also reflected in the legislative texts. In simpler or poorly thought-out information systems, the address is represented as a single text attribute that can be changed freely, which, depending on the need, is the term -person-, -institution-, -event-, -letter-, etc. on the side. This approach is only suitable under very limited conditions.

In fact, an address is one of the most complex objects in information systems because it has a complex internal structure and dynamics of change. When planning any new information system or rebuilding an existing one, it is imperative to think through the data model for storing address data and the dynamics of address data changes, an integral part of which is the representation of the address on screen forms, printed materials, and the user interfaces for searching and entering the address.

Designers of information systems have arrived at different solutions in the handling of addresses, each of which has its own development story, and these are probably the best solutions in the context of the development stories of specific systems. Nowadays, however, there are few completely independent information systems that do not communicate with other systems. Information systems being part of a larger network has become a basic requirement and an unavoidable necessity. Consequently, it is unavoidable to standardize the structure of the address and the dynamics of change.

The solution, which started in 2007 and is operational today and stipulated in the legislation, is the Address Data System information system managed by the Land Board (abbreviated as ADS information system - hereinafter the abbreviation ADS is used in this document). ADS has standardized the structure of Estonian place-addresses and the dynamics of address data changes through the definition of the internal structure of the address, the implemented X-tee services and the underlying theoretical models. According to the Spatial Data Act, the use of standardized address data issued by ADS is mandatory from March 1, 2012. Based on the Spatial Data Act, address data conforming to the same standards must be used in all data sets belonging to the state information system. It is also strongly recommended and a reasonable choice to use in the design of other information systems or in the reconstruction of existing ones.

Interfacing with ADS is nuanced and different for each dataset. Therefore, not all questions that may arise may be covered in the document

For additional questions, contact ads.abi@maaamet.ee. There is also Frequently asked questions by interfacers (in Estonian) page at Land Board Geoportal.

2 Terms

- **Aadress** (ka koha aadress) - An **address** (also a location address) is a record indicating the location of an object or an identifier for finding an object in a geographical space. The document deals only with site addresses (primarily addresses assigned to cadastral parcels, buildings and parts of buildings). The term "address" is also used to mean a network address (also IP address), an e-mail address, or a memory address in IT, etc., but these addresses are not discussed in this guide.
- **Aadressiandmete süsteem** (ADS ruumiandmete seaduse mõistes) - The **address data system** (ADS in the sense of the Spatial Data Act) is a framework of organizational, technical and legal means that ensures a uniform identification of address objects both in their location and in different data sets, and a uniform organization of place-address assignment and address data processing. The address data system is the basis of the spatial data infrastructure and ensures the maintenance of data collections.
- **Aadressiobjekt** - An **address object** is an object related to land, which has been assigned an address or which has the obligation or possibility to assign an address as a result of legislation. The location of an address object relative to the ground does not change over time, and thus such an object has a fixed location in geographic space. Temporal immutability can of course be viewed as relative, taking into account the duration of human life, because over longer periods of time (centuries and millennia) changes occur sooner or later with all objects.
- **ADOB_ID** – unique identifier of the **address object version** (unique across all versions of all objects).
- **ADR_ID** – unique identifier of the **address version**.
- **ADS_OID** – ADS **object identifier**, identifies an object through versions.
- **EHAK** – **Classification of the administrative and population distribution of Estonia**, which is intended to indicate the territorial location. Counties, municipalities (cities, municipalities) and settlement units (villages, towns, towns within municipalities etc.) are entered into EHAK.
- **EHR (riiklik ehitisregister)** – **National register of construction works** is a database belonging to the state information system, in which the data of buildings (buildings and facilities), including addresses and spatial shapes, are entered. The register of construction works contains documents of procedures related to construction activities and restrictions and limitations placed on buildings.
- **ETAK (Eesti topograafia andmekogu)** – **Estonian topography database** is a database belonging to the national information system, in which spatial data of topographical phenomena of general importance and data describing the content, relationships and context of these phenomena are entered. The output of ETAK are topographic maps, which are used, for example, in map applications of the Land Board (hybrid map, basic map, etc.).
- **In-ADS** - In-ADS is an integratable address lookup interface for the Estonian address data system, which is easy to place in various web-based information systems and allows data to be requested from ADS.

  In addition to standard search, it is possible to save matches found in the search in the consuming information system using the component. In-ADS also includes a map component, and address and object searches can be both text-based and map-based. In-ADS also has a simple map editor functionality.
- **Katastriüksus** (KÜ) - **A cadastral parcel** (CU) is a plot of land (land unit) registered as an independent unit in the cadastre. Unique cadastral characteristics with a definite structure are used to distinguish cadastral parcels. For example, one cadastral code: 78401:101:4712.
- **Kaudne liidestuja** - An **indirect interfacer** is a consumer of ADS data through another system. For example, address data can be obtained from the Population Register or the Business Register.
- **Koha-aadress** - The place-address is a textual-numeric record of the address object based on the hierarchy of the territory's administrative division and official place names e.g. Ida-Viru county, Jõhvi municipality, Jõhvi city, Marja tn 20 (Ida-Viru maakond, Jõhvi vald, Jõhvi linn, Marja tn 20).
- **Koodaadress** – Code address is a uniquely identifying feature of the address formed from the codes of the normalized address components, i.e. a 33-digit number. We recommend using ADR_ID to identify the address.
- **Liidestuja** – An **interfacer** is the organization that manages the database interfaced with the ADS information system. For the purposes of this document, an interfacer is only a consumer of ADS data.

- **Liikluspind** - In the sense of § 44 of the Spatial Data Act, a traffic area = **thoroughfare** is a road (e.g. street, alley, intersection, etc.), the name of which is used in the address of other address objects. The name of the address location can also be assigned to a section of the road.
- **Lähiaadress** - **Proximate address** is a more precise part of the administrative and settlement unit of the place-address. If there is a address area, the proximate address consists of its name with a type word, the name of the address object or the name of the thoroughfare, the type word of the thoroughfare and the address number with the necessary additions.
- **Maaüksus (maatükk)** - A land **unit** (**plot**) is a delimited part of land or water area. The land unit may not be registered in the cadastre. If a land unit is registered in the cadastre, it is called a cadastral parcel.
- **Mitteametlikud aadressiandmed** - **Informal address data** is address data that is not used in location addresses, but is used in human communication when locating an object. Informal address details may be associated
  - with a single object - these are **points of interest** or POIs, for example City Hall (*Linnahall*);
  - with an address - these are **informal areas** (for example, a district municipality, a small island, a part of a city, a subdistrict, a quarter), for example the Rotermann quarter (*Rotrmanni kvartal*).
- **Normaliseeritud koha-aadress** - A **normalized location address** is a text address consisting of structural elements (e. components) of a location address and conforming to business rules. Data sets often contain free-text addresses that can be understood by humans, but may cause problems in machine processing. For example, a small typo (Tatru instead of Tartu) or presentation of structural elements in the wrong order (Mustivere küla, Muuga tee 1, Viljandimaa) is generally not a problem for a person, but in data sets the data must meet specific rules so that they can be processed automatically. To do this, they are brought to a normalized form (in a certain order, with a standard spelling, unnecessary spaces are eliminated and the use of punctuation marks and abbreviations is standardized). For example, the address Aia 16, Tallinn is in normalized form: Harju maakond, Tallinn, Kesklinna linnaosa, Aia tn 16 (Harju county, Tallinn, Kesklinna district, Aia tn 16).
- **Paralleelaadress(id)** - Parallel **address(es)** are multiple location addresses assigned to one address object, which are parallel addresses equal to each other. Often, cadastral parcels located on street corners in cities are assigned parallel addresses according to both streets, e.g. Pargi tn 10 // Pikk tn 3. About 0.4% of all objects have parallel addresses.
- **Primaarobjekt** - A primary object is the most primary of the objects associated with the same address. Determined by automated business rules in ADS.
- **Riiklik kohanimeregister (KNR)** - **National Place Names Register** is a national register, the purpose of which is to collect and register information about Estonian place names, process and store it and make it available to users. The responsible processor is the Land Board; the authorized processor is the Information Technology Center of the Ministry of the Environment.
- **Ruumiaadress** - **Spatial address** is the calculated (analytical) address of the address object, which is obtained by analyzing the spatial shapes of the address object and the spatial shapes of administrative and settlement units. The spatial address consists of the names of administrative and settlement units with species words and the corresponding percentages of the spatial share with an accuracy of 0.01%.
- **Ruumiandmed** - **Spatial data** is data, including data managed in databases, which directly or indirectly points to a specific location or geographical area, and which describes the location (including location address), characteristics and shape of spatial objects in geographical space.
- **Ruumikuju** - **Spatial shape** is the geometric shape of a spatial object. For example, the spatial data set does not contain the data of the house itself, but its spatial shape is stored in the data set.
- **Tuumlinnad** - **Core cities** are all cities as a whole, except in the case of Tartu, Pärnu, Narva-Jõesuu, Haapsalu and Paide, those areas that are essentially not cities. Therefore, in the case of the aforementioned cities, the core cities include the population units of the same name located in them.
- **Täisaadress** - The **full address** is the address text, which consists of the EHAK part and the proximate address, presented in a more general to more specific direction. A distinction must be made between the full address of an object, which includes all parallel addresses, and the full address of a specific address record.
- **Unikaalaadress** - A **unique address** is a place-address whose textual-numerical form is unique among address objects of the same type that require a unique address. Several address objects requiring the

same type of unique address cannot have the same place-address (for example, two residences cannot have the same address).

- **Unikaalaadressi nõudega ala (UN-ala)** - **Unique address requirement area** (**UN area**) is an existing or planned clearly demarcated area with dense or compact population. The borders of the UN areas are managed by the Land Board in the address data system, and the cadastral parcels located in these areas must have a unique location address. In general, the UN area must have defined thoroughfares or address areas, based on which cadastral parcels and buildings are addressed.
- **Väikekoht** - A **address area** is a land area smaller than a settlement (e.g. place, canton, district, horticulture cooperative, cottage cooperative, garage cooperative, etc.), the name of which is used in the address of other address objects. A road located in a address area can be used as an independent address in the sense of a thoroughfare.
- **X-tee** (*X-road*) is a data exchange layer (DXL) developed and launched by the Estonian Information System Agency (RIA), which enables the exchange of data between information systems. A large number of different X-tee services have been created for both data consumers and data providers regarding the data of the address data system.

3 ADS overview

In the following review, the most important aspects for the interfacers consuming ADS data are discussed.

Note: In the following, numerical values from valid data as of **August 2022** are given in italics to the concepts. They are here only to characterize the magnitude of the number of records behind the concept. The number of entries in ADS is constantly changing.

3.1 Address object

Address object *(~2.4 million records*) is the primary concept for interfaces that consume address data.
Legal definition: A geographic address object (hereinafter address object) is an object related to land, which has been assigned an address, or which has the obligation or possibility to assign an address due to the legal act.

Additional explanation: An address object is an object in nature permanently connected to the earth, which has a spatial shape* and on which or inside which you can live, work, store something, move along, etc., and as a result, there is a need to refer to this place or object (region of space) in the usual human language in a way, i.e. with an address.

An address object always has at least one address, but there are also address objects that have multiple addresses, i.e. parallel addresses are assigned.

> * *spatial shape* means a point, a broken line or a surface in the XY coordinate system of the national coordinate system, which defines the shape of the address object projected onto the 0-elevation plane. Z, or vertical dimension, is not used in ADS. Apartments and other building parts that are spatially one above the other are depicted as points.

3.1.1 Address object classification

The table shows the classification and possible levels of ADS objects. For more information on the hierarchy of address components, see Chap 3.2.4.

Code	Name	Init level	Original (origin) register	Note
MK	county	1	Land cadastre	
OV	municipality	2	Land cadastre	
AY	settlement unit	3	Land cadastre	
LO	city district	3	Land cadastre	
VK	address area	4	Place Names Register	
LP	thoroughfare	5	Place Names Register	
CU	cadastral parcel	6, 7	Land cadastre	
EE	residential building	6, 7	Register of construction works	The actual purpose of use of the building and the name of the type may not match, read more below.
ME	non-residential building	6, 7	Register of construction works	
ER	dwelling	8	Register of construction works	
MR	other part of the building	8	Register of construction works	

Attributes

- KOOD – 2-letter type code. The code is assigned to the new type by the system administrator in the ADS system.
- NIMETUS – the name of the object type.
- INIT_TASE – levels of components that this object type can initialize (create new components).
- ORIG_REGISTER – identifier of the original register (e register of origin) providing the data. Each object type can have only 1 origin register where the object data is managed.

The above table shows the typical levels for cadastral parcels and buildings, i.e. generally, the proximate address assigned to the building and cadastral parcel must be at least level 6 or 7 for a correct proximate address. In exceptional cases, the proximate address may also be missing, for example, an unbuilt cadastral parcel outside the area with a requirement for a unique address may have an unassigned proximate address.

ER and MR are considered to be of the same type - the term building part is used as the common denominator. EE and ME objects are also considered to be of the same type - the term building is used as the common denominator.

ADS type EE or ME is assigned to the building during the initial ADS submission (from ETAK or EHR) and cannot be changed later. In the first submission, EE refers to a residential and/or public building and ME to a non-residential building (the corresponding definition is made either by the ETAK mapper or automatically based on the purpose of use of the EHR). When registering a building in ADS, the requirement for a unique address(UN) is calculated for it, i.e. the UN attribute value indicating the residential and/or social nature of the building (read more in Ch.3.3.1).

Address objects classification in details:

- MK – **counties** (*15 records*).
- OV – **municipalities** (*79 records, including 15 cities and 64 rural municipalities*). Municipalities are territories of local municipalities starting from Tallinn, Tartu and ending with Ruhnu and Kihnu. The areas of counties are always fully divided into municipalities. There is no land area unit in Estonia that lies outside municipalities. Most of the internal water bodies and some of the water bodies at the border of the country are also divided between municipalities. Läänemeri, Võrtsjärv, Peipsi, Lämmijärv and Lake Pihkva and Narva Reservoir and Narva River are divided between municipalities only notionally and those bodies of water are not directly part of municipalities. Notional division has been made for situations where an island or holm emerges into a body of water provided above. In such an event it shall automatically be part of the relevant municipality.
- AY – **settlement units** (*4 692 records*), villages, boroughs, small boroughes, towns and cities within the municipality (i.e. town without municipal status). The areas of municipalities are always fully divided into settlement units.
- LO – **city districts** (*13 records*). Only Tallinn and Kohtla-Järve cities have districts in the official classification of administrative and settlement distribution (EHAK) and therefore in ADS. The area of these cities is completely divided between districts. There are no official districts in the remaining cities.
- VK – **address areas** (*~600 records*) are, for example, a gardening cooperative, a summer cooperative or a garage cooperative. These are land areas smaller than a settlement unit, where several addressable objects are relatively close together. Also thoroughfare can be located in some larger address areas. The number of address areas has been in a downward trend since 2007.
- LP – **thoroughfares** (*~17 800 records*) are, for example, a square, a square, a road, a street, a cross, a boulevard and a highway (the list is not exhaustive, other types of words are added if some municipalities designate them as place names). These are the roads by which buildings and cadastral parcels are addressed. In terms of addressing, thoroughfares can participate in the address only within the boundaries of the municipality or settlement unit, i.e. it is generally not possible to provide an unambiguous address only through the thoroughfare name.
- CU – **cadastral parcels** (*~765 000 records*). The division of Estonia's surface between cadastral parcels forms a separate layer compared to the aforementioned divisions. The boundaries of cadastral parcels

may not cross the boundaries of municipalities and, consequently, the county. Cadastral parcels are used to indicate a location if there is no suitable building or part of a building for this purpose.

- EE – **residential buildings** (*~304 000 records*). The type assigned to the building upon initial ADS submission. Buildings for living and daily work. This includes apartment buildings, terraced houses, semi-detached houses, detached houses, farmhouses, cottages and similar buildings where you can live temporarily or permanently, as well as shops, hospitals, retirement homes, prisons, office buildings, hotels, etc. It should be taken into account that the type of building is determined upon its initial registration and it does not change if, for example, reconstruction takes place and the purpose of the building changes. More up-to-date information about the habitability of the building is maintained in ADS by adding and removing the UN attribute (unique address requirement) value. There are ~ 3,400 of the EE buildings without a UN designation.
- ME – **non-residential buildings** (*~684 000 records*). The type assigned to the building upon initial ADS submission. Buildings intended for other purposes, where a person's daily stay is not expected. This includes industrial buildings, warehouses, substations, garages, etc. buildings that are of interest in terms of addressing. The ME building can also be with or without a UN identification. There are ~34,200 ME buildings having UN designation.
- ER – **dwellings** (*~555 000 records*). Parts of the building intended for living. This includes, for example, apartments and row house parts, often the building is also of mixed type, so there are both apartments and commercial premises.
- MR – **other parts of the building** (*~67 000 records*). Addressable parts of buildings not directly intended for living. This includes office premises, less often also parts of production, industrial and warehouse buildings, which need to be specified separately in terms of address. Buildings can have a large number of such building parts that are not addressed. Both ERs and MRs always have a unique address requirement.

3.1.2 Main attributes of an address object

The main attributes of an address object are spatial shape and addresses. All address objects with a shape (except EHAK objects themselves and objects without a spatial shape) are also addressed with a spatial address in ADS. For example, an object can be spatially located in several residential units, i.e. it has several spatial addresses, but the address is always assigned to the object according to only one residential unit. Therefore, databases interfaced with ADS generally do not deal with separate spatial addresses.
The ADS system requests only valid or pending address object data from data submitters as input. Cancelled or outdated data are not additionally registered by ADS, they are created in the case of changes within the ADS system. In interfaced systems where address object data is also stored, if it is necessary to keep this data up-to-date, the address object data must be updated through the corresponding change log service or by some other method.

- ADOB_ID – address object version identifier, unique key field.
- ADOB_LIIK – address object type code.
- ADS_OID – ADS object identifier, identifies the object through versions. The code consists of 10 symbols: the first 2 are an abbreviation consisting of letters expressing the type of object and 8 numbers that are generated by the system.
- ORIG_TUNNUS – object identifier in the original register. Identifies an object within a type. Unique among current versions with a type identifier. Buildings, parts of buildings and cadastral parcels can also be in ADS without a connection to the register of origin. In the case of buildings, this generally means that it is an ETAK-mapped building with an unknown or missing EHR counterpart. In the case of building parts, these are the so-called manually added building parts of ADS, which need to be distinguished on the basis of the address, but which have not been entered into the EHR either for construction technical or other reasons. In the case of cadastral parcels, there are land units that have not yet been entered into the cadastre without a connection to the register of origin (before registration in the cadastre, an address must be assigned to the new unit in ADS). After the implementation of the MinuKataster solution, no more such cadastral parcels will be created from April 2024, the previous ones will be gradually cancelled.

- TEKKIMISE_ALUS – the legal basis for the creation of the version of the object. For example, the data of a document or legislation from the original register, but the legal basis can also be an automatic change or the ADS procedure, and the field can also be entered/changed manually.
- KEHTIV_ALATES – the date of the legal basis of the object's version. Generally coincides with the effective date of object version (field KEHTIV), but may be different (e.g., the date of the EHR document is earlier than the time of the change in ADS). For example, the date of a document or legislation from the original register, but can also be the date of an automatic change or the ADS procedure, and the field can also be entered/changed manually. In general, those interfacing with ADS should use the KEHTIV field instead of the KEHTIV_ALATES field.
- SULGEMISE_ALUS – the legal basis for closing the version. For example, the data of a document or legislation from the original register, but the legal basis can also be an automatic change or the ADS procedure, and the field can also be entered/changed manually.
- KEHTIV_KUNI – the date of the legal basis of closing the object version (if the next version exists, the legal date of its creation). Generally coincides with the closing date of object version (field KEHTETU), but may be different. For example, the date of a document or legislation from the original register, but can also be the date of an automatic change or the ADS procedure, and the field can also be entered/changed manually. In general, those interfacing with ADS should use the KEHTETU field instead of the KEHTIV_KUNI field.
- OLEK – status / state.
    - **K** – (kehtiv) valid or actual - an object that also has a **valid** status in the register of origin, if it has a corresponding connection.
    - **O** – (ootel) object to be designed, e.g. does not yet exist in nature in its finished form, but which has already been assigned an address has **pending** status. The state of the object "pending" generally means that it is an object not fully registered in the register of origin. The EHR code can be generated for the object earlier, if the object is not yet valid in the register of origin (EHR) (for example, the address has already been assigned to the building or building part before the building permit application was issued, but the object has not yet been updated in the EHR). For example, if a cadastral parcel starts to be divided, two or more new pending cadastral parcels are registered on top of it in ADS. When the division is registered in the cadastre, the original cadastral parcel is cancelled and the new pending parcels become valid. This process can take from a few months to several years, from initiation to registration in the cadastre. For example, one building can have both pending and valid building parts. Such a situation can arise, for example, when a building is rebuilt and the distribution of building parts changes. In this case, ADS contains both the existing state of building parts (in valid state) and the new building parts determined by the building permit (in pending state). If the new status is updated in the EHR, the valid building parts are cancelled and the pending building parts become valid.
      A **pending** object is a **current address object** in the ADS system**.**
      A **valid or pending** address object is called the *current address object* in today's context. Thus, current is the common denominator **for both the current and pending states**.
      NB! In older X-tee services K and O objects are generally treated/issued as equally valid objects. In the latest versions of X-tee services (available from April 2024) and extracts, the state of O and K is generally distinguished.
    - **T** – (tühistatud) object has been **cancelled** in the register either because it has ceased to exist in nature, or because it was an object incorrectly entered in the register and has never existed in nature. The object may also be cancelled for technical reasons. For example, when EHR and ETAK records for the same building are merged, the ETAK relationship ADS_OID is cancelled and its ETAK_ID is transferred to the ADS_OID of the EHR building. Or ETAK buildings mapped as two separate buildings are merged into one spatial shape and thus one ADS_OID is cancelled, etc. In addition, the applicant can re-enter the building parts data on the document in the EHR, during which the old building parts are cancelled and new ADS_OIDs are provided. See also chapter 3.3.4.
    - States **K, O and T** are present only on the latest version of the object. When object is cancelled, a new version (ADOB_ID) of the object is not created, but the status of the last current version is changed to T.
    - **V** – (vananenud) **outdated** denotes the historical or old version of the object, there is always a newer version of the same object.

Meanwhile, objects in the ADS system also had a state of **X - pending cancellation** (tühistamise ootel), but this was abandoned during the 2021 developments.

- TAISAADRESS – full address of the object in an optimized text form. Contains all parallel addresses, repeated elements are expressed once.
- LAHIAADRESS – proximate address of the object (levels 4 - 8) in optimized text form. Contains all parallel addresses, repeated elements are expressed once.
- GEOMETRY – spatial shape of the object, see more about the spatial data also on chapter 3.3.3.
- VIITEPUNKT_X / VIITEPUNKT_Y – coordinates of the reference point of the object
- KUJU_MOODUSTUSVIIS – method of forming object's shape:
  M -measured;
  K -mapped;
  D - digitized;
  A – calculated.
- KEHTIV – date and time of receipt of the object's data in the ADS main system, which is also the date when the object's version becomes **valid** for ADS.
- KEHTETU – the date and time the version became invalid in ADS. The time of cancelling the object when OLEK is T.

  If there is a newer version of the object, then generally the KEHTIV of the new version and the KEHTETU of the previous version match, unless the previous version was in a cancelled state in the meantime. Then the time of cancelling the previous version, i.e. KEHTETU, and the time of restoration, i.e. KEHTIV of the new version, are different. When restoring an object, a new version of the object is created, the state of the previous version goes T → V.
- UNIK – whether the address object has a unique address requirement or not. Only relevant for buildings and cadastral parcels. All other address objects must always have a unique address, UNIK is empty or undefined on these objects.
- ETAK_ID – reference to the building in the ETAK system. Can have a value only for buildings.
- HOONE_OID – reference to the building in which the building part is located. Can have a value only for building parts.
- SISSEPAASU_KORRUS – entrance floor of the building part. Can only have value for building parts. It is filled in when provided by the EHR, but the up-to-date information is in the EHR additional data table.

In addition, the address object also has technical attributes that the ADS system uses either to optimize requests or to store additional internal information. It is possible to request more data about an object from ADS extracts and services than is described here (see descriptions of ADS X-tee services). Additional information can be requested from the Land Board at ads.abi@maaamet.ee. The following describes some of the main additional object data that interfaces can use.

Additional data of the register of construction works

Additional EHR data is also kept informative in the ADS system. They are related to the object **ORIG_TUNNUS**, i.e. through the code of the register of construction works .

nimetus	**Name** assigned to the building or building part in the EHR system.
pind	Under-construction **area** in the case of a building, total area of space in the case of a building part.
kasutusotstarbed	List of **intended uses** for the building (list of codes and names, separated by semicolons)
korrus	In the case of a building part, the **floor** of the entrance.
staatus	**Status** of the object in the EHR register.
andmedSeisuga	**Date** as of which the EHR data in the ADS system has been **updated**.

They can also be requested via ADS X-tee services and their changes can be monitored via **ADSobjmuudatusedV7** service **E-event**, but to obtain accurate and up-to-date information, these data should be requested directly from the EHR.

Interfacer connections

Three so-called key-interfacers provide ADS with information about address relationships in the interfaced register. These registers are the population register (RR), the land register (KR) and the business register (AR). ADS also provides information about the address object, i.e., the apartment ownership relationships of the **ADS_OID** real estate book and the relationships of legal entities in the business register, through services and extracts (one object can have several interface relationships), i.e. using **ADSobjmuudatusedV7** can monitor their changes through the **L-event** of the service.

register	KR – Land register (kinnistusraamat). AR/ÄR – Business register (äriregister).
idRegistris	Identification of the object in the interfacers system: either the apartment ownership number or the register code of the business register, respectively.
lisainfo	Special property number in the case of apartment ownership or the name of a legal entity.

Building height data

The address object search and object change services also provide the height data of the building if it is a building related to ETAK:

- **hooneKorgusR** - the height of the eaves of the building in meters with such accuracy as is in the data of the Estonian Topography Database (ETAK). In ETAK, the spatial shape of the building is digitized in a 3D environment with the height of the eaves. The value of the **hooneKorgusR** attribute is found as the averaged difference between the height values of the building's spatial shape (the relative height is calculated for each corner point of the building) and the ground height model compiled on the basis of aerial laser scanning (ALS) or LiDAR data, which is rounded to whole meters. Quality is not checked; negative values are not shown. Negative values are not eliminated directly. Corner points with a relative height of -1 are excluded from the averaging on residential and ancillary buildings.
- **hooneKorgusM** - the maximum (ridge) height of the building in meters with the accuracy as in the ETAK data. Automatically found in aerial laser scanning (ALS) data in areas with 15 or more elevation points per square meter. Attempts are made to exclude chimneys and antennas. The data is updated with each proper ALS result. As a rule, it takes place once a year in larger cities and their surroundings, and according to the receipt of ALS data, every few years also in smaller cities. The stereo mapper can check the data and correct it if necessary.

Points of interest

Point of interest (POI) data is managed outside the ADS system in the POI system of the Land Board. POIs are collected from various registries and public data sources. In the ADS system, POIs are treated as an informal name of an address object, which is not reflected in the official address, but which is informally used to refer to the object, for example *Solaris keskus* (Solaris centre), *Linnahall* (City hall), *Raekoda* (Town hall) etc.

There are more points of interest in the POI system than there are in ADS, because only those that can be associated with a specific address object **ADS_OID**, are entered in ADS. For example, there are no kilometre posts or ATMs in ADS.

The name or location of the point of interest from the source register may have been changed by the ADS administrator. One address object can also have several associated POIs. POIs are not associated with addresses. Through a separate **ADSpoimuudatused** service, the log events that have occurred with points of interest are transmitted to the public with the current data at the moment of the request of the point of interest.

poiId	An identifier that uniquely identifies a POI object.
poiNimi	The name assigned to the point of interest in the ADS system.
poiAlias	The alias assigned to the point of interest in the ADS system.
poiYlemgrupp	Classifier code. The classifier contains the following values: 10 - public administration (riigihaldus) 11 – education (haridus) 12 – health (tervis) 13 - free time (vaba aeg) 14 – services (teenused) 15 – transport (transport) 16 – environment (keskkond) The data contains only the code of the classifier, and the translation of this code is done on the ADS side.
poiGrupp	Name characterizing the group of POI type.
poiAlamgrupp	A name characterizing a subgroup of a POI type.
poiTyypnimi	Characteristic name of the POI type.
poiTyyp	POI type code as a number.
poiX	POI reference point X assigned in the ADS system.
poiY	POI reference point Y assigned in the ADS system.
poiAndmeallikas	The name of the POI source registry where the data originally came from.
poiVID	POI identifier in the source register.
poiAndmeseisKp	Date of last update of POI data from source register.

In the X-tee services related to the address object, the names of the points of interest reviewed by the ADS administrator and marked for distribution are returned uniquely, i.e. repetitions are excluded. This means that if there are e.g. 2 different points of interest associated with the address object, which have the same public distributed main name (alias is not taken into account), then this name is returned once at the object.

3.1.3 Related objects

Cadastral parcels and buildings are considered related objects in the ADS system. Relationships are not calculated for pending cadastral parcels. Relationships between objects are many-to-many. Building parts and cadastral parcels are connected through the building connection.

If the building has a spatial shape, the connection with the spatial analysis is detected in the ADS system. If the building does not have a spatial shape, then it comes from the EHR system, and in this case the relationship of the cadastral parcel added to the building in the EHR is taken into account. These are direct relationships between objects, but indirect relationships can also exist.

Depending on the nature of finding the connection, priority is added to the connection and they can be the following:

- priority 1 (direct connection) detected in the ADS system on the basis of spatial shape;

- priority 2 (direct connection) in the EHR application, the cadastral parcel connection added to the building by the user of the local authority, if there is no spatial shape of the building;
- priority 3 (indirect connection) connection found on the basis of address similarity;
- priority 4 (indirect link) the CU link added to the building by the user of the local government in the EHR application, which does not coincide with the direct link identified on the basis of the spatial shape of the building.

The connection is created between the ADS_OID values and only the current data is taken into account. Cancelling an object deletes its associations.

Object family

In ADS, the concept of object family is still used. The object family always consists of exactly 1 cadastral parcel, buildings directly related to it and parts of these buildings. If the building is directly connected to several cadastral parcels, it belongs to several families.

Business rules for calculating the relationship of objects in the ADS system

Priority 1 connections are identified with a **spatial analysis** either by the cadastral parcel or by the building. In the case of CU, spatially related buildings are found, in the case of a building, spatially related CUs are found.

Algorithm for finding the relationship between the CU and the building, if both are surface objects. The ratio of the area of the common part of the spatial shapes of the building and CU to the area of the building is found.

1) If the ratio is less than 10%, then these objects are not related.
2) If the ratio is 10-40%, then the absolute value of the area of the common part is considered.
    a. If the area of the common area is less than 16m2, the objects are considered to be related to priority 2.
    b. If the area of the common part is larger than 16 m2, then the objects are related to priority 1.
3) If the ratio is greater than 40%, then the objects are associated with priority 1.

A CU and a building are associated with priority 1 if one of them is a point and the other is a surface, and the point is on a surface.

If both shapes are a point, then the objects are associated with priority 1 if the points match the coordinates to 2 decimal places.

If the analysis was done based on the building and the building had both priority 1 and 2 connections, priority 2 connections are deleted (not taken into account). If priority 1 connections did not occur, but priority 2 connections did, the priority of the connection with the largest share is changed to 1, the rest are deleted (not taken into account).

If the analysis was done based on the CU, priority 2 connections identified with the spatial analysis are not taken into account.

As a result of the spatial analysis, only priority 1 connections are saved to the object.

Priority 2 and 4 connections are created during the registration of additional EHR data, when information is received from the EHR register about the addition or removal of a connection. The new association is registered with priority 2 if the building has no shape, and with priority 4 if the building has a shape. A link removed in the EHR registry is also deleted from ADS if it is either priority 2 or 4 in ADS.

When a new version is created in the ADS system, if a shape has been added to the building, all priority 2 connections will be changed to priority 4, if the spatial analysis has not changed them to priority 1 before.

Priority 3 relationships are created for the object only during the follow-up activity of the relationship calculation. The system finds buildings with the same address based on the cadastral parcel or CU with the same address based on the building. An address with the same ADR_ID value is considered the same address.

If the corresponding connection between the objects is not yet found, the system adds it with priority 3. If the corresponding connection is found (with any priority), the system does not change the priority of the connection. If the analysed building or CU has a registered connection with priority 3, which the request for the same address no longer provided, the system removes this connection.

3.2 Address

Another important concept is **address** (*~1.36 million records).* An address is a readable and understandable text in human language that points to the address object. Several address objects can correspond to one address, because several objects in nature (for example, several buildings) can have the same address. Thus, by expressing the location only with the precision of the address, we may not uniquely define an object (e.g., a building) in nature (for example, a house in the countryside with outbuildings all have the same name, a house in the core city with sheds has the same street name and a number as a proximate address). In order to uniquely define an object, an address object must be added to the address automatically or by the user in information systems. In addition, it must be taken into account that one object can have several addresses, i.e. **parallel addresses.** There are few objects with parallel addresses, but when interfacing with ADS, the complexity associated with them must be taken into account. For example, in the context of a specific use case, it must be decided whether all parallel addresses of an object or a specific one are to be saved and displayed. For example, in the case of the object of apartment ownership in the Land register, it is correct to save and display all parallel addresses of this object. But, for example, only one specific address must be saved and displayed as the address of a legal person in the Business register. Also, in most use cases related to persons, the accuracy of one specific address is required, because, for example, in the case of a dormitory house, each household has a different address in use, even though one dwelling as a whole has two addresses (parallel addresses).

Addresses change according to changes in address objects and decisions of local governments (e.g. assigning a new name to a cadastral parcel or changing the name of a thoroughfare). Most changes to address objects (including assigning new addresses, changing and invalidating existing ones) are also within the competence of local governments. Most of the changes in the administrative and settlement distribution are established either by legislation of the Government of the Republic, the Minister or the Director General of the Land Board. The decision to merge a municipality is made by the councils of the merging municipalities. In all the aforementioned changes, local residents and other affected persons are also involved in the administrative procedures.

3.2.1 Location address and address states

An address consists of up to 8 components. Each level can contain only 1 component in the address composition. Some components (levels) may be missing, some may be mandatory. For each object type it is known to which level its address can be assigned.

An interfacers address can be either normalized or not. A normalized address consists entirely of components only. In a non-normalized address, some part may be expressed through components, but some or all of the address is expressed textually. There is an X-tee service for the normalization of addresses, and the geocoding service also offers the possibility of normalization, see more precisely in the chapter 4 on the interfacing process.

The actual location address must not exist in a non-normalized form. Only historical data and foreign addresses may not be normalized.

An address can only be created from valid components.

A new address is created in ADS immediately in a valid state.

When components are changed (versioned), the addresses containing the changed component are also automatically versioned in the ADS system. Addresses containing an outdated component version are in an outdated state. Address associations with an object are not automatically versioned, therefore new address versions may initially be left without an object association. In order to change the relationship of objects, ADS is waiting for confirmation from the register of origin or changing the relationship through the procedural system.

The structural composition of the address cannot be changed. It is not possible to replace the components in the address, in this case it is a matter of creating a new address, not changing the existing one

If a component is cancelled, the addresses containing the component will also be cancelled. The cancelled address may still remain on some objects at the beginning, the ADS system does not remove it by itself, but waits for a change by the registry of origin.

If the cancelled address no longer appears on any valid object, an additional **attribute** is added to the address - **connections lost**. This event also generates an additional changelog S - loss of connections for the address.

3.2.2 Main attributes of an address

- ADR_ID – unique identifier of the address (version ID).
- KOODAADRESS – address code formed from component codes + version number, which makes the value unique.
- TAISAADRESS - a textual address consisting of the names of the components of all levels in a more general to more specific direction.
- LAHIAADRESS – a textual address consisting of the names of the 4th - 8th level components in a more general to more specific direction.
- VIITEPUNKT_X – 6-digit east-west axis coordinate of the representative point of the address, corresponds to the Y coordinate of the L-Est system.
- VIITEPUNKT_Y – 7-digit north-south axis coordinate of the representative point of the address, corresponds to the X coordinate of the L-Est system.
- OLEK – address status:
  K - is an address consisting of only **valid** components;
  T - at least 1 of the components of this address has been **cancelled**;
  V - **outdated** e.g. at least 1 of the components at this address has received a new version. There is a newer version from the same address that may be valid or revoked.
- KEHTIV – the date and time of the **start of validity** of the address (version) in the ADS system.
- KEHTETU – the date and time of the **end of validity** of the address (version) in the ADS system. Blank on valid address.
- SIHTNUMBER – postal code of the address.
- ASUMI_NIMI – names of unofficial regions related to the address, separated by commas.
- ASUMI_ALIAS – aliases of unofficial regions related to the address, separated by commas.
- PRIMAAR_OID – ADS_OID of the primary object of the address.
- TEHNILINE – 1-is the technical address; empty - normal address.

In addition to the listed ones, the id, code and name of all its components in long and short form are also stored with the address.

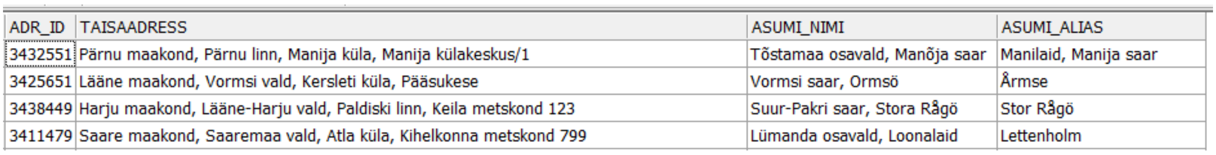
Unofficial areas

In the ADS system, unofficial districts and subdistricts, as well as quarters, small inhabited islands and district municipality's, which may also have an official place name, but which do not officially participate in addressing, are kept as unofficial regions. For example, *Kadriorg, Kassisaba, Rääma, Supilinn, Rotermann* quarter.

The management of these objects is carried out in the national register of place names. The KNR system automatically submits all object-related changes to ADS. With spatial analysis, the names of all regions in the area of which the representative point of the address (ADR_ID) is located are added to one ASUMI_NIMI field as an address attribute, separated by a comma. The system ranks the names of the informal areas associated with the address in order of importance:

- 1 – sub-municipality (*osavald*),
- 2 – small island (*väikesaar*),
- 3 – city district (*linnaosa*),
- 4 – sub-district (*asum*),
- 5 – quarter (*kvartal*).

If one region has several names, other forms of names go separately in the ASUMI_ALIAS field, e.g:



ADR_ID	TAISAADRESS	ASUMI_NIMI	ASUMI_ALIAS
3432551	Pärnu maakond, Pärnu linn, Manija küla, Manija külakeskus/1	Tõstamaa osavald, Manõja saar	Manilaid, Manija saar
3425651	Lääne maakond, Vormsi vald, Kersleti küla, Pääsukese	Vormsi saar, Ormsö	Årmse
3438449	Harju maakond, Lääne-Harju vald, Paldiski linn, Keila metskond 123	Suur-Pakri saar, Stora Rågö	Stor Rågö
3411479	Saare maakond, Saaremaa vald, Atla küla, Kihelkonna metskond 799	Lümanda osavald, Loonalaid	Lettenholm

Event A is generated for the address in **ADSaadrmuudatusedV7** service log, if the value of the asumi_nimi or asumi_alias attribute actually changed at the address.

Unofficial regions as area objects themselves are not issued through ADS X-tee services, but spatial shapes of regions are available through the ADS public application. Changes to unofficial areas can be followed through the X-tee service ***nimeobjektiLogi*** provide by the Place Name Registry by tracking changes to the following types:

- unofficial city district (KNR type 10311),
- unofficial sub-district (KNR type 10312),
- quarter (KNR type 10315),
- inhabited small island (KNR type 30406),
- sub-municipality (KNR type 10103).

Technical address

From the spring of 2022, the addresses will have a separate identifier "technical". If this characteristic has a value (is marked), then it is a technical address. If this identifier is empty, then it is not a technical address, it is a so-called normal location address.

Technical addresses are generally created for such cadastral parcels that are formed for technical objects. Mostly, for example, when registering cadastral parcels under roads ("Vahi tee T3", "Kadaka tee L1", "T-11371 Keila-Ohtu", etc.). The place-address must be assigned to these cadastral parcels, but the need to find them is generally limited to the activities of specific institutions (maintenance of roads and railways, land management, etc.). Such addresses are generally not used to locate people, events, or other objects.

Consumers of addresses can optionally exclude technical addresses from their lists based on this feature to reduce excessive noise and possible erroneous address selections. We recommend doing this if dealing with such addresses is not appropriate based on the specifics of the database.

The "technical" attribute is always automatically assigned to an address by the system if all the following conditions are met and is automatically removed if these conditions are no longer met:

- the address ends with level 6;

- 6th level component has no subordinates, i.e. no other addresses can be found that are more precise than it;
- the name of the level 6 component contains either a number or a technical species word;
- no current building is associated with the address.

The technical species word must appear as a separate word in the name of the component, i.e. it must be written separately from the rest of the name. A space is considered a word separator, and this space must appear before the technical species word. If the name is only a species word, then the name is not technical. For example: "Boulevard" is not technical, "Mere boulevard" is. It is not case sensitive.

Technical species words are managed by the Land Board in the ADS procedural application. Although there are more cadastral parcels serving so-called technical objects, not all generic words used in the names of the respective units have been treated as technical in this context. This is because it is difficult to draw the line. For example, "power station" or "waste station" can sometimes be conventional built units, can also be workplaces, etc., although they refer to a technical facility. So, as of August 2022, the solution is narrower and the technical species words are as follows (they may change):

Tehniline liigisõna	**Meaning of the technical species word:**
allee	alley
juurdelõige	street cut
juurdelõike	(street) extension
kergliiklustee	light traffic road
km	km
käik	passage
liiklussõlm	traffic node
liiklussõlme	traffic junction
lõik	(road) section
maantee	highway
mnt	highway species word abbreviation
pst	boulevard species word abbreviation
puiestee	boulevard
põik	cross (cross street)
raudtee	railway
ringtee	roundabout
ristmik	intersection
sissesõit	entry
t-	
tee	the road
tupik	dead end
tänav	street
väikekoht	address area
väljak	square
õueala	outdoor area
ühismaa	common ground
üldmaa	general land
ülesõit	crossing
ümbersõit	detour

When assigning/removing the attribute "technical" to an existing address, the system registers a new log event T - technical to the service **ADSaadrmuudatusedV7**. If a new address is created and it immediately gets the "technical" attribute in the initial state, no T-event will occur in addition to the I-event. If the address is revoked, the "technical" attribute retains its status and does not change for revoked addresses. When the address is restored, an R event is generated, and after that the system recalculates the characteristic "technical".
The technical identifier is not calculated for previously cancelled addresses.

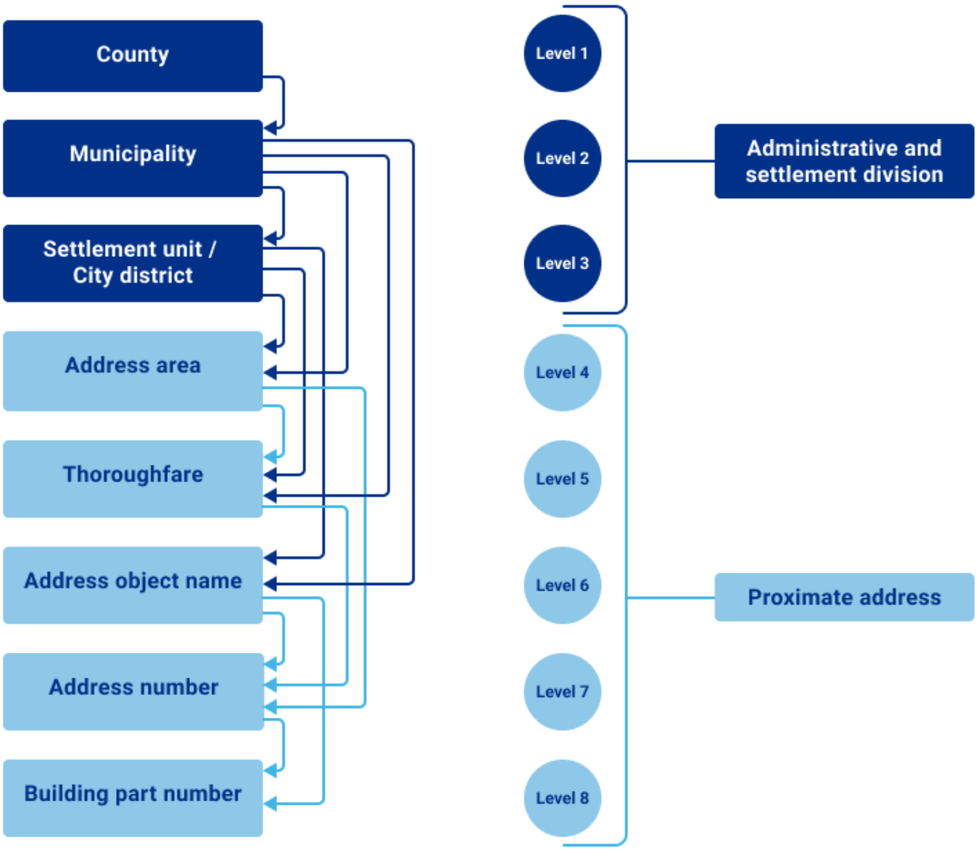
3.2.3 Address components (levels)

The text of the address is structured and its components are called **address components** *(~1.36 million records*) (*they are called structural elements in the spatial data law*). There are eight address components and they are hierarchically interdependent. All components except level 1 components have a relationship with some other component to which it belongs. In this way, the components form a hierarchical, tree-shaped structure.

Level	Name	Possible upper levels
1	county (maakond)	
2	municipality (omavalitsus)	1
3	settlement unit / city district (asustusüksus / linnaosa)	2
4	address area (väikekoht)	2, 3
5	thoroughfare (liikluspind)	2, 3, 4
6	address object name (nimi)	2, 3
7	address number (aadressinumber)	4, 5, 6
8	building part number (korterinumber)	6, 7

The main attribute of the address component is the name in the NAME data field.

Hierarchical dependency of address components



Explanations of levels:

Levels 1 and 2 must always be completed (except for the county's own address, where only level 1 is completed)
Level 3 may be missing.
Level 4 can only follow Level 2 or 3;
Level 5 can only follow Level 2, 3 or 4;
Level 6 can only follow Level 2 or 3 (i.e. 4+6 and 5+6 are not allowed);
The 7th level can only follow the 4th, 5th or 6th level (i.e. it cannot follow e.g. the village name directly - 3+7 is not allowed);
8th level can only follow 6th or 7th level

Additional information:

The settlement unit level is not always fulfilled in the address. Level 3 is missing in the address if the city is both a municipality and a settlement unit within the same boundaries. Then at level 3 the name of the city is not duplicated. The 3rd level can be filled in the address of a self-governing city if there are city districts or if the city has additional population units, including the city of the same name (core city). In the latter case, the city name is duplicated in both the 2nd and 3rd level in city addresses within the city. For clarification:

- There are 47 cities as settlement units in Estonia, of which 10 are also municipalities within the same borders and 5 so-called inner-city cities.
- The borders of 10 cities as municipalities exactly overlap with the borders of the city of the same name as a settlement unit - in the addresses, the name of the city appears only once at the 2nd level, e.g. "*Ida-Viru maakond, Narva linn, Anne tn 10*"; The 2nd self-governing city, i.e. Tallinn and the city of Kohtla-Järve, have official districts that are also used for addressing (the district is on the 3rd level).
- In the remaining 5 self-governing cities, in addition to the inner-city city of the same name (as a settlement unit), there are also towns, townships and/or villages, i.e. the borders of the self-governing city and the city as a settlement unit do not overlap. In these 5 self-governing cities, which are the city of Paide, the city of Haapsalu, the city of Tartu, the city of Pärnu and the city of Narva-Jõesuu, the name of the city must be written twice (on the 2nd and 3rd level) in the city addresses within the city, for example "Pärnu Maakond, Pärnu linn, Pärnu linn, Aasa tänav 10". The address of other residential units in the city of Pärnu is, for example, in the form "*Pärnu maakond, Pärnu linn, Audru alevik, Aia tn 12*".
- 32 cities are regular cities within the municipality - the city name is at level 3, for example "*Järva maakond, Türi vald, Türi linn, A. Haava tn 2*".
- In this document, we refer to Tallinn, Kohtla-Järve as core cities and those cities that are municipalities and settlement units within the same borders and those that are cities within the municipality. In other words, core cities are all cities as a whole, except for the cities of Tartu, Pärnu, Narva-Jõesuu, Haapsalu and Paide, those areas that are essentially not cities. Thus, the core cities are the population units of the same name in the case of the above-mentioned cities.

Postcodes

Postal code / zip code (e.g. 78313) is an address attribute, not an address object attribute. If there are several objects at the same address, they have the same postal code. If one object has several addresses, this object can also have several postal codes.

Each address can have exactly one postal code. Although there are also addresses that practically have several different postal codes in use, based on the address, there can always be only 1. Others are special-purpose postal codes that arise from the needs of the postal service of the institution located at that address and not from the address itself.

The ADS system does not have to provide an overview of all postal codes in use, but only of the postal codes associated with the address. That is, the postal codes assigned to a specific legal entity cannot be found in the ADS data - e.g. the postal code of the address Suur-Amerika tn 1 is 10122, the postal codes of the Ministry of Finance (postal code 15006) and the Ministry of Social Affairs (postal code 15027) located at the same address are not found in the ADS data.

Not every address needs to have a postal code. Addresses associated with buildings or parts of buildings with a unique address requirement (UN) and where the address contains level 6, 7 or 8 must have a postcode. If, for example, there is a building with a UN, which only has an address specified up to the village level, then this address does not need to have a postal code. Also, the postal code does not have to appear in the addresses of thoroughfares and address areas.

A postal code can also be assigned to those addresses that do not have to have it.

The rules for assigning postcodes are established by the universal postal service provider (Omniva), and its task is also to assign postcodes to new addresses. The task of the ADS system is to store this information and check its correctness as far as possible.

There are 2 ways to set postcodes in the ADS system:

- **when the representative point of the address is located in the postcode area; these areas are prepared outside the ADS system (by Omniva);**
- **through the exceptions table.**

Postcodes were added to ADS in 2019. Prior to this, postal codes were not assigned to cancelled and obsolete addresses.

An N-type event is generated for the address in the **ADSaadrmuudatusedV7** service - postal code change - if the postal code in the address record changes, including when the postal code is added or removed. The system immediately calculates the postal code for the new address. An I-event occurs and no additional N-event occurs. When updating an address, the new version will initially have the same postal code as the previous version. If the address is restored, the postal code will be recalculated. If the address is cancelled, no follow-up action will be taken with the postal code.

Main attributes of the address component

- KOMP_ID – version ID of the component
- TASE – Mandatory reference to the component's hierarchical level.
- KOOD – Mandatory component code in 32-system. Unique within the tier among valid versions. Does not change when versioning a component. Levels 1 and 2 are EHAK codes without leading zeros.
- NIMETUS – Mandatory component name with abbreviation of species word if species word exists. Also called component short name.
- NIMETUS_LIIGIGA – Mandatory name of the component with the species word in accordance with the form prescribed by law. In the absence of a species name or its abbreviation, equal to the name. Also called the long name form of the component. In reality, they only differ in level 4 and level 5 components. For all other levels, they are equal.
- YLEMKOMP_TASE – reference to the level of the parent component.
- YLEMKOMP_KOOD – reference to the parent component code.
- KEHTIV – valid version start time in ADS system.
- KEHTETU – time when the version became invalid. A version becomes invalid when a new version occurs or when the last binding of the initializing address object associated with the component becomes invalid.

Within the parent component, both the name and the name-with-type must be unique in valid versions to ensure the address uniqueness requirement.

The latest current version of the component has a KEHTIV (*valid*) date filled and an KEHTETU (*invalid*) date left blank. The outdated version has both dates filled. The last current version of the deprecated component has an invalid date filled in.

Address objects initialize address components of a certain level. Initiating levels are defined in the address object's types classifier.

A level 1 - 5 component is always initialized by only 1 address object. The name of this component is actually the name of this object. If this address object name changes, the component is versioned. As a chain reaction, all addresses that contain the changed component change. Also, a component becomes invalid if the object that initializes it becomes invalid. As a chain reaction, all addresses containing this component become invalid.

Level 6 - 8 components can have several initiating objects in parallel, such as a cadastral parcel and a building. Components are not versioned if one of the objects that initialize them changes, but when the value (name or number) changes, a new component is always created and the object starts to initialize it. If there are no more initializing objects left for the old component, it becomes invalid. If it remained, the old one will also remain valid, and all addresses containing this component will also remain valid.

Synonyms

An address component may also have informal name variants, called synonyms, registered. Synonyms are associated with the component as a whole and not with the version of the component, i.e. a change in the official form of the name does not necessarily lead to changes in the synonyms. Synonyms are not versioned. They can be removed from the component and added to the ADS system based on business logic.

3.2.4 Address text generation and address code

For each address, the text form of the address is also formed from the component names. A distinction is made between full address text (includes all levels) and close address text (from level 4). In the case of a full address, it is necessary to distinguish between the full address of an object (including all its parallel addresses) or the text of the full address of a single address record. The full address is presented in a more general to more specific direction, for example *Harju maakond, Tallinn, Kristiine linnaosa, Mustamäe tee 51*.

The address text is formed by defined rules. Therefore, if the interface offers the user the opportunity to select an address by component and wants to form the text of the full address from the selection, the following principles must be used:

- levels 1 to 5 are separated by a comma followed by a space (even if levels 4 and 5 are filled at the same time);
- if both level 6 and 7 are fulfilled, there is a slash between them;
- level 8 is always preceded by a hyphen (without a space);
- there is a gap between levels 4 and 7 and 5 and 7;
- there may be slashes inside the level component.

For all levels, the long (adjective) name form of the component is generally used. For the 4th and 5th levels, a short form of the name is used for the sake of brevity (Pärnu maantee vs Pärnu mnt). The short and long form of the name can theoretically differ at other levels, but in practice they are mostly the same.
Parallel addresses are separated by a double slash (… // …), before and after which spaces are left (for example, *Pärnu maakond, Pärnu linn, Pärnu linn, Kapteni tn 4 // Pärnu maakond, Pärnu linn, Pärnu linn, Sadama tn 5*).
More about parallel addresses see ADS handbook (EST).
Several identical address texts **with different ADR_IDs** may be valid at the same time.
This is not an error, but mainly results from the fact that the addresses of the thoroughfare and the cadastral parcel serving the thoroughfare have the same address text. In the first case, the address ends with a level 5 component and in the second case with a level 6 component, which are identical.

For example, (the first is the address of the thoroughfare, the second is the address of the cadastral parcel):

Full address	Address as components	ADR_ID

Rapla maakond, Rapla vald, Valtu küla, **Pargi põik**	1->71-Rapla maakond;
	2->668-Rapla vald;
	3->8971-Valtu küla;
	5->0M4T-Pargi põik
		3185526

Rapla maakond, Rapla vald, Valtu küla, **Pargi põik**	1->71-Rapla maakond;
	2->668-Rapla vald;
	3->8971-Valtu küla;
	6->8J41-Pargi põik
		3217488

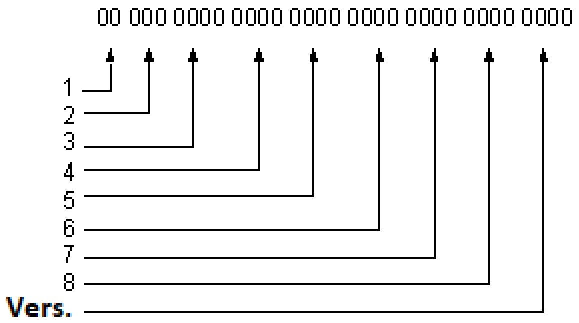
For example, some combinations of proximate addresses:

• Proximate address text from joined components:	• Address component: level > code > text	• Code address
• 4+7 proximate address: „Tuuletare AÜ 26“	• **1 -> 45 - Ida-Viru maakond** • **2 -> 803 - Toila vald** • **3 -> 8914 - Valaste küla** • **4 -> 25AZ - Tuuletare aiandusühistu** • **7 -> 0ACU - 26**	• **45803891425AZ000000000ACU00000000**
• 5+7+8 proximate address: „Väljaku tn 12-3“	• **1 -> 37 - Harju maakond** • **2 -> 653 - Rae vald** • **3 -> 2377 - Jüri alevik** • **5 -> 0FE0 - Väljaku tänav** • **7 -> 3QFE - 12** • **8 -> 59B0 - 3**	• **37653237700000FE000003QFE59B00000**
• 6+7+8 proximate address: „Mereranna/1-4“	• **1 -> 37 - Harju maakond** • **2 -> 353 - Kuusalu vald** • **3 -> 1007 - Kolga-Aabla küla** • **6 -> 0C38 - Mereranna** • **7 -> 9NLB - 1** • **8 -> EBKG - 4**	• **3735310070000000000C389NLBEBKG0000**
• 6+8 proximate address: „Sepikoja-1“	• **1 -> 84 - Viljandi maakond** • **2 -> 899 - Viljandi vald** • **3 -> 1794 - Heimtali küla** • **6 -> 3ZL5 - Sepikoja** • **8 -> B1L8 - 1**	• **848991794000000003ZL50000B1L80000**

The address code or code address is compiled from the codes of the address components. If there is no level in the address, the code of the missing level is replaced by a zero code, i.e. a code consisting of zeros (for example, “0000”). To ensure the uniqueness of the versions, the version number of the address is also added.

The address code is a 33-digit code, where each level corresponds to the next number of characters:

Level 1: 2 characters
Level 2: 3 characters
Level 3: 4 characters
Level 4: 4 characters
Level 5: 4 characters
Level 6: 4 characters
Level 7: 4 characters
Level 8: 4 characters
Version: 4 characters



The version ID also conveys the state of the address. Version ID 0000 is at a valid address. Historical addresses get version ID 10000 - version serial no. Thus, the first historical version has the characteristic 9999, the second 9998, etc. This provides an opportunity for convenient ordering of versions in queries.

To clarify the code address: if the address consists of the same components, but the component changes, the code address levels 1-8 remain the same, the ADR_ID changes. For example, if the name of the thoroughfare changes, the code of the level 5 component remains the same (because the object that initiates the component is the same), but its version changes. Accordingly, all addresses containing this thoroughfare name become obsolete and a new version (ADR_ID) is created from the address.

The function of the **normalization service** is to read non-normalized text from ordered and controlled input data and try to arrange it into components. If such a component actually exists in ADS, it associates the address with the component.

3.3 The relationship between an address and an address object and its dynamics

The need for two individual terms ‘address object’ and ‘address’ comes from life itself, as in daily use the objects and their addresses are not singularly connected.

EHAK objects always have 1 address, but the other **objects can have several addresses at the same time, i.e. parallel addresses**. About 0.4% of all objects have parallel addresses.
The presence of multiple addresses (parallel addresses) per object is related to road intersections and buildings that have several doors facing different streets and thus have received multiple addresses (parallel addresses). Also, parallel addresses are assigned to cadastral parcels that have many buildings with different addresses, and the addresses of these buildings are all used as parallel addresses for this cadastral parcel.

More complex relations between the address and the object occur in complex buildings, where there are apparently several different buildings, but in fact the buildings are connected to an underground part (e.g. a garage). There are not many such buildings, but more are being built every year. In the case of complex buildings, it is a building registered as a single entry in the register of construction works, on which several spatial shapes have been formed. These spatial shapes are in ADS as separate objects, i.e. each shape has a separate ADS_OID, the ORIG_TUNNUS of one corpus is conventional, the ORIG_TUNNUS of other corpuses contains a hyphen and an additional number (e.g. 123456789-1) in addition to the main EHR code. Thus, from the EHR point of view, one object has several addresses, but in ADS it is technically different objects and each corpus generally has one address. For example, terraced apartment blocks can be registered as a complex building in ADS.

Assigning multiple addresses to building parts is generally not allowed. An exception is some living spaces in dormitories, where, in the technical sense of construction, the rooms of one apartment (living space) are numbered and used by different households. However, it is necessary to assign parallel addresses to such residences.

Several objects often correspond to one address. An address ending with the EHAK level can also have several objects. That is, although the addresses of EHAK objects must be unique, for example, there can be a cadastral parcel with the same EHAK address. For example, the address *Lääne maakond, Lääne-Nigula vald, Liivi küla* appears both as a settlement unit *Liivi küla* (village) and on the unbuilt cadastral parcel CU00178328 located there.
Starting with level 6, it is not practical to create a unique address for each potential object, because it would unnecessarily overload the concept of address, but at the same time, it is justified that all objects that could potentially have an address should be registered in the ADS information system, because these objects may also need pointing. Thus, there are usually several objects with the same address, but several objects of the same type requiring a unique address cannot have the same address. Following of this requirement is not 100% guaranteed, however, because assigning a unique address to objects with a unique address requirement generally requires an administrative procedure. There may also be some exceptions where unique addresses are not actually necessary for objects with a UN attribute valued. Read more in chapter 3.3.1.

There are about 2.4 million current objects, 1.36 million valid addresses. About one claim has more to do with an address than an object.
In other words, there is a higher probability of finding more objects at one address than if one object is assigned to different addresses. As a rule, addresses containing the components of a proximate address with many objects are complexes with many buildings.

For interfacing, it is generally necessary to use both an address and an object association. In order to make it easier to select an object link from among several objects with the same address, ADS provides primary object information. See more in chapter 3.3.2.

3.3.1 Unique address requirement (UN-attribute)

Al Since 2014, address objects have been equipped with a unique address requirement attribute.

The requirement for a unique address originates from the Spatial Data Act, according to which a unique address is a place-address whose textual-numerical form is unique among address objects of the same type that require a unique address. Several address objects requiring the same type of unique address cannot have the same location address. This does not mean that all objects with a unique address requirement have a unique address, but that another object with the same UN attribute value cannot have the same address. In addition to objects of the same type, types of buildings, i.e. EE and ME, and types of building parts, i.e. ER and MR, are considered to be of the same type.

For example, a cadastral parcel has one residential building with a UN requirement and 2 auxiliary buildings that do not have a UN requirement - in this case, the address of the residential building does not have to be different from the address of the auxiliary buildings. If there are several residential buildings, the addresses of these residential buildings must be unique among themselves. At the same time, depending on the situation, there may be several objects of the same type with UN identifiers, but using a unique address is not necessary or practical. For example, valid and planned or pending CUs (not yet fully registered in the cadastre) can have the same address, because often, for example, as a result of merging parcels, the newly created parcel is assigned the address of an earlier cadastral parcel, etc. There may also be a valid building with a UN attribute that is to be demolished and a construction permit for a new building with a UN attribute issued in its place. In nature, they cannot exist simultaneously (the old building must be demolished before building the new one), although in ADS, the records of the building to be demolished and the building to be built exist in parallel for some time. In this case, assigning a unique address to the buildings is not required. In addition, there are a number of objects with a uniqueness conflict for which the local authority has not yet assigned unique addresses.

If it is necessary to assign a unique address to buildings, a special addition (so-called additional number) is added to supplement the address number or name level address. Technically, the special addition is always added to level 7, i.e. the address number and the additional number with a slash are included in the level 7 component (e.g. Metsa tn 5/1). If a special suffix is added to the name, a slash as a separator between the 6th and 7th levels is automatically added to the address by the system, i.e. the 7th level only has an additional number (e.g. Maasika/2). A letter suffix can also be added to the address of the building, but this results in changing the address of the cadastral parcel (adding a parallel address). Different solutions are further described in the Address data handbook.

Address objects that require a unique address are:

1) residential or public building;
2) buildings in which there are living spaces, i.e. apartments and other building parts that need to be distinguished based on the address;
3) buildings where legal entities are located;
4) buildings in which universal postal service access points and parts of the aforementioned buildings are located;
5) land units located in an existing or planned clearly demarcated densely populated or compactly populated area (hereinafter an area with a unique address requirement);
6) land units whose place-address contains the name of the place of address;
7) apartments and other building parts that need to be distinguished based on the address.

In general, the reasons for awarding the UN designation are as follows (all reasons are equal, not in order of priority):

- **the object type is ER or MR**
- **the object type is CU and the object have a 4+7 level address**
- **the object type is CU and the object have a 5+7 level address**
- **the object type is CU and its spatial shape is located at least 50% in the UN area**
- **the object type is CU and at least 1 UN-building is associated with it**
- **the object type is CU, EE or ME and at least 1 RR or ÄR object of the interfaced register is associated with it**
- **the object type is EE or ME and is associated with at least 1 EHR purpose of use causing a UN designation (the purpose of use classifier is internally valued once in ADS)**
- **ETAK building type is "residential or public"**
- **current building parts (ER and MR objects) are located in the building (EE or ME objects)**
- **no interface objects from the RR or ÄR registers are related to the building, the building has no EHR usage purposes at all, the building has no ETAK connection, there are no building parts in the building, but the address object type is EE**

Reason for removing UN-requirement:

- **the object does not have any reasons for assigning a UN designation**

In addition to the above, it is possible for the ADS administrator to assign or remove the UN value manually (for example, in the case of an erroneous ÄR object connection, the UN requirement can be removed from the shed, etc.).

The UN attribute value is automatically assigned to the building upon initial registration in ADS (when submitted from EHR or ETAK) based on the ADOB type, i.e. EE receives a UN requirement attribute value and ME does not receive a UN requirement attribute value.

- The ADOB type of the EHR building is automatically determined based on the purpose of use, i.e. if there is at least one purpose of use that causes a UN requirement, the ADOB type becomes EE, in other cases ME.
- The ADOB type of an ETAK building is determined based on the mapped building type: a residential or public building is EE and a service or production building is ME.

The ADOB type cannot be changed after the initial registration of the building in ADS. With nightly post-processing, the UN attribute value for buildings is calculated based on additional data, and then the ADOB type no longer plays a role. In other words, if a building has at least one UN requirement reason, it gets a UN requirement (retains the UN attribute value), if there is no UN requirement reason, the UN attribute value is removed (or not added).

As long as the building with EHR connection does not have an ETAK connection and EHR additional data, i.e. the purpose of use information is not yet in ADS, then the UN requirement remains according to the ADOB type. This is a situation where, for example, an application for a building permit has been submitted and enforced by ADS, but the document has not yet been fully registered in the EHR. That is, from the current data of the EHR, a valid usage purpose has not yet been transmitted to ADS over the X-tee. Also, for example, in the case of the housings of a complex building, there is no additional data (the purpose of use is mapped to the so-called basic code). In other words, the EE has a UN requirement and the ME is without a UN designation, unless the ME building also has building parts or an interface connection that cause the ME building to have a UN requirement. If additional EHR data is added to the building, i.e. purpose of use or ETAK connection, they will be taken into account for the UN attribute value, i.e. then the ADOB type no longer plays a role.

Thus, the ME building may have a UN attribute value and the EE building may not have a UN attribute value, which is why it is more important to consider the UN attribute value than the object type when interfacing with ADS.

This attribute is only relevant for buildings and cadastral parcels. All building parts have always UN attribute value set. The rest of the object types have UN requirements designation not valued, i.e. empty, because essentially they always have the obligation to assign a unique address.

When the UN attribute changes, the object is versioned (ADOB_ID changes), so it can be monitored through the object's change log events (there is no separate log event referring to the UN attribute) in the service: **ADSobjmuudatusedV7.**

UN-areas

Since March 16, 2018, areas with a unique address requirement, or UN areas, have been added to the address data system. In the UN area, the requirement to assign a unique address to the cadastral parcels applies, as well as the cadastral parcels located in the UN area that are built or subject to building must be addressed according to the thoroughfare name or address area name.

The requirement to consider UN areas comes from the spatial data act (*RAS* § 43 (2) p 5), i.e. instead of the earlier densely populated area and the area with the obligation of detailed planning, addressing now depends on the location in the UN area. For example, UN areas are taken into account in the ADS business rules when adding a UN requirement to a cadastral parcel, as well as for other requirements where the location of an object in a city, town or settlement area (in a densely populated area) was previously taken into account.

The consideration of UN areas instead of the former densely populated area and the area with the obligation of detailed planning is related to the fact that often the entire area of a city, town or small town is not compactly populated, and therefore the requirement of addressing, for example, thoroughfare name does not have to be applied to the entire city, town or small town. At the same time, however, in several developing regions, the population has actually become very dense, for example, the population density characteristic of a city has essentially expanded beyond the borders to the territory of a village, or an area with a compact population has emerged in the centre of some villages, which is why it is necessary to use addressing based on the thoroughfare name or a address area name to ensure findability. Often, the municipality has already provided addresses according to the thoroughfare name in the areas described when making plans.

The rules for the formation of UN areas are not specified in the legislation and can be based on § 43 (2) point 5 and § 48 (1) of the RAS. When forming UN areas, the Land Board has primarily based its needs on fact-based addressing, which is why the UN area is based not only on the densely populated areas designated by the municipality, but also on those areas where, in fact, compact buildings have developed. Thus, several conditions have been taken into account when forming UN areas, including:

- densely populated areas specified by the local government in the general plan;
- areas with compact population, which have been found due to the combined effect of the number of buildings, density, purpose of cadastral parcels, population register connections, presence of

thoroughfare names, etc. In doing so, the density and appearance of cities specific to Estonia, the purpose of use of buildings, etc. are taken into account;

- additional conditions and exceptions, for example
    - some dense/compact areas of coastal villages with historical place names, etc., have been excluded from the UN area;
    - most of the core cities are UN areas as a whole, also address areas are always UN areas (because they have an established address-by-address address, and it is also historically used in cities and necessary to ensure findability);
    - in core cities, towns and small towns, the conditions of compactness (distance, number of buildings, etc.) are different than in villages.

For clarification: in cities/towns/small towns, the feel (usual) is a little different (in most cases, the thoroughfare names *are also determined throughout, etc.) and therefore in cities/towns/small towns, one must proceed from a more established situation, i.e. the distances between buildings may be slightly larger, but nevertheless it is a UN area. At the same time, there should not automatically be a UN area due to the* thoroughfare *names designated between the farms/fields on the outskirts of, for example, a smaller city/town/small town. It is often reasonable to correct the coverage area of* thoroughfare *names in such areas instead. We also consider that a new UN area does not have to be created in every old kolkhoz centre with barns, workshops, etc., because it is not in accordance with the practice.*

UN areas can be viewed in the ADS public service map application:
https://xgis.maaamet.ee/xgis2/page/link/NDQYwEz - UN sites are displayed at a scale of 1:1 to 1:600,000.

3.3.2 Primary object

As described above, there are often multiple address objects associated with the same address. In order to assign one representative point to an address, the ADS system first identifies the most important or primary object among all objects with the same address. The address point of the primary object becomes the representative point of the address. In most cases, the primary object provided by the ADS system is also suitable for the interfacers to create an automatic object connection. For more details, read the recommendations in chapter 4.5.

In the case of a change of the primary object (event O) and a change of the address point (event P), a different log event is generated for the address, through which their change can be monitored via the **ADSaadrmuudatusedV7** X-tee service. The primary object was calculated in ADS in the spring of 2022, so there is no reference to the primary object for previously invalidated addresses.

In order to decide whether the primary object found by ADS matches the need arising from the usage history of the interfacers, the rules for calculating the primary object of ADS are described below.

Rules for selecting the primary object and the representative point of the address

For addresses with only levels 1-5, the reference point is always the reference point of the object that initializes the most precise component, i.e. the address itself. This object is also the primary object of the address. If such an address is also owned by another object (for example, a building), these objects do not affect the location of the representative point of the address.

The rules apply to addresses with levels 6-8. The primary object is selected, and its address point is simultaneously also the representative point of the address. If the primary object has no shape, the address proxy is removed if it exists.

The representative point of the address is changed only if the coordinates of the address point of the primary object found as a result of the following analysis differ from the coordinates of the existing representative point by 2 decimal places.

Selecting the primary object if there are spatial objects associated with the address

In this case, only objects with spatial shape at this address participate in the analysis, and objects without spatial shape are discarded.

For each subsequent step, only the most prioritized objects based on the previous step remain in the further analysis.

1. First of all, the priority of address objects is looked at based on type. Types order by importance
ER,
MR,
EE, ME (these types are considered to be equal)
CU.
If there were several objects with the highest priority type, only those are further analysed.

Building parts with a spatial shape

1R1 If the highest priority type was ER or MR, then the EHR relationship is looked at as the next condition. An object with an EHR connection is preferred.

1R2 in the case of objects with multiple EHR connections, objects with a valid status are preferred over objects with a pending status.

1R3 in the case of several objects with the same highest priority status, the one that also has additional EHR data is preferred.

1R4 If several have additional data, the object with the higher ADOB_ID of the first version is preferred.

Buildings with a spatial shape

1H1 in the case of types EE and ME, the second rule applies to checking the UN requirement attribute. If there is only one object with a UN attribute set, it is the primary object. If there is more than one building with a UN designation or no building with a UN designation is found, the analysis continues in the corresponding branch.

A) There is more than one building with UN attribute designation associated with the address:

1HA1. Status checking is applied first – actual (valid) objects are preferred over objects with a pending state.

1HA2. If the previous step did not reach one object, then the reasons for the UN designation are looked at next, and the building with the most reasons for the UN designation is preferred. Considerable reasons for the UN designation are:
1) The purpose of use is the cause of the UN designation . (*Note: Purposes of use are taken into account only in the case of building state K, and the purposes of use are found on the housing of a complex building based on the first half of the orig_tunnus i.e. orig. identifier*.)
2) ETAK type is a residential or public building.
3) Presence of building parts.
4) Manually designated UN attribute.
5) Interfacer connections (the number of connections is not important, but their presence or absence).

1HA3. If this too should not guarantee a single result, then the connection check is applied to the buildings with the largest number of reasons for the UN designation, i.e. the buildings are placed in the order of priority based on the following:
1) The building has both EHR and ETAK connection.
2) The building only has an EHR connection.
3) The building only has an ETAK connection.

1HA4. If the previous step did not reach one object, but comparable buildings have equally 1 reasons for assigning the UN designation, then the object whose reason for the UN designation has a higher priority according to the list in point 1HA2 will be preferred.

1HA5. If the previous steps did not reach one object (i.e. if the objects equally have more than 1 reason for assigning a UN requirement or the same reason for the same UN attribute), but the buildings remaining in the further analysis during the previous steps are all related to the EHR, then the additional EHR data will be taken into account. Additional data is considered only for state K, and additional data is found on the complex building body based on the first half of the orig_tunnus (orig. identifier). The EHR status is read from the additional data and the highest priority is given priority:
1) exists (*olemas*);
2) under construction (*püstitamisel*);
3) planned (*kavandatav*);
4) the rest are equally of the lowest priority:
demolished, under demolition, not implemented, deleted, unassigned, no additional EHR data
(*lammutatud, lammutamisel, realiseerimata, kustutatud, määramata, EHR lisaandmed puuduvad*).

1HA6. If all the buildings remaining in the analysis do not have an EHR connection or the previously verified EHR status of the buildings is equivalent, then the area control is applied as the next rule. A building with a larger surface area is preferred.

1HA7 If even in this way one building is not reached, the building with the higher ADOB_ID of the first version is selected.

B) Only non-UN buildings are associated with the address:

1HB1. Status checking is applied first - valid objects are preferred over objects with a pending state.

1HB2. If this too should not guarantee a single result, the buildings are checked for connections, i.e. the buildings are prioritized based on the following:
1) The building has both EHR and ETAK connection.
2) The building only has an EHR connection.
3) The building only has an ETAK connection..

1HB3. If the previous steps did not reach one object, but the buildings left for further analysis during the previous steps are all EHR-connected, then the additional EHR data will be taken into account. Additional data is considered only for state K, and additional data is found on the complex building body based on the first half of the orig_tunnus (orig. identifier). The EHR status is read from the additional data and the highest priority is given priority:
1) exists (*olemas*);
2) under construction (*püstitamisel*);
3) planned (*kavandatav*);
4) the rest are equally of the lowest priority:
demolished, under demolition, not implemented, deleted, unassigned, no additional EHR data
(*lammutatud, lammutamisel, realiseerimata, kustutatud, määramata, EHR lisaandmed puuduvad)*.

1HB4. If all the buildings remaining in the analysis do not have an EHR connection or the previously verified EHR status of the buildings is equivalent, then the area control is applied as the next rule. A building with a larger surface area is preferred.

1HB5 If one of the buildings is not reached, then the building with the higher ADOB_ID of the first version is selected.

If one building has been reached, an additional analysis is performed to obtain a representative point of the address. Cadastre units with the same address are found, the spatial shape of which is a surface (as you know, point-shaped units created in ADS can also be present) and which have a common part with the building under consideration. A temporary joint object is formed from these common parts (if there should be several of them).

Next, it is checked whether the address point of the selected highest priority building is located inside this compound object. If located, the address point of the building is suitable as a representative point of the address. If it is not located, the address point is placed inside a temporary compound object (the reference point position is calculated with respect to the compound object with the same algorithm as usual in ADS).

Cadastral parcels

1K1 in the case of type CU, the status check is applied as the first rule - valid objects have higher priority than pending objects.

1K2 as the next rule, the building control of the cadastral parcel applies. Building connections are in the following order of priority:
1) the cadastral parcel is built-up with connection UN building with 1st priority;
2) the cadastral parcel is built-up with connection UN building with 2nd priority;
3) the cadastral parcel is built-up with connection non-UN building with 1st priority;
4) the cadastral parcel is built-up with connection non-UN building with 2nd priority.

1K3 If the previous step did not reach one object, preference is given to the cadastral parcel that has more buildings with the highest priority.

1K4 If the cadastral parcels have the most priority relationship equally (e.g. two cadastral parcels have equally one priority 1 relation UN building), then the cadastral parcel with a higher number of priority 1 and 2 relation buildings is preferred.

1K5 If one cadastral parcel was not reached beforehand, or the analysed cadastral parcels do not have the previously described buildings at all, then the UN code check is applied. If only one cadastral parcel with a UN requirement is found among the previously screened units, then it is the primary object.

If there is more than one UN cadastral parcel, these UN cadastral parcels remain in the further analysis.

If there are no UN cadastral parcels, then all previously screened cadastral parcels remain in the further analysis.

1K6 The next rule is the area control. The unit with the largest area is preferred.

1K7 If this too should not guarantee a single result, the cadastral parcel with the largest ADOB_ID of the first version is selected.

Selecting the primary object if no spatial object is associated with the address

For each subsequent step, only the most prioritized objects based on the previous step remain in the further analysis.

2. First, the priority of the address objects is looked at. Valid non-spatial objects with the observed address are ordered based on the importance of the type as follows (decreasing order of importance):
ER,
MR,
EE, ME (these types are considered to be equal)
Objects with the highest priority type are taken under further observation (there may be several of them).

Building parts

2R1 If the highest priority type was ER or MR, then the EHR relationship is looked at as the next condition. An object with an EHR connection is preferred.

2R2 in the case of objects with multiple EHR links, objects with a valid status are preferred over objects with a pending status.

2R3 in the case of several objects with the same highest priority status, the one that also has additional EHR data is preferred.

2R4 If several have additional data, the object with the higher ADOB_ID of the first version is preferred.

Buildings

2H1 in the case of types EE and ME, the second rule applies the UN requirement check. If there is only one UN object, it is the primary object. If there is more than one UN building or no UN building is found, the analysis continues in the relevant branch.

C) More than one UN building is associated with the address:

2HC1. Status checking is applied first - valid objects are preferred over objects with a pending state.

2HC2. If a step did not reach an object, then the UN reasons are looked at next and the building with the most UN reasons is preferred. Considerable UN reasons are:
1) Purpose of use is the basis of UN. (*Note: Purposes of use are taken into account only in the case of building state K, and the purposes of use are found on the housing of a complex building based on the first half of the orig_tunnus (orig. identifier)*)
2) ETAK type is a residential or public building.
3) Presence of building parts.
4) Manually designated UN attribute.
5) Interfacer connections (the number of connections is not important, but their presence or absence).

2HC3. If the previous step did not reach one object, but the comparable buildings have equally 1 UN reasons, then the object whose UN reason has a higher priority according to the list in point C2 is preferred.

2HC4. If the previous steps did not reach one object (i.e. if the objects equally have more than 1 UN reason or the same only UN reason), but the buildings remaining in the further analysis during the previous steps are all EHR connected, then the EHR status check is performed on the buildings, i.e. the highest priority is given priority:
1) exists (*olemas*);
2) under construction (*püstitamisel*);
3) planned (*kavandatav*);
4) the rest are equally of the lowest priority:
demolished, under demolition, not implemented, deleted, unassigned, no additional EHR data (*lammutatud, lammutamisel, realiseerimata, kustutatud, määramata, EHR lisaandmed puuduvad)*.

2HC5. If one of the buildings is not reached, then the building with the higher ADOB_ID of the first version is selected.

D) Only non-UN buildings are associated with the address

2HD1 The status check is applied first - valid objects are preferred over objects with pending status.

2HD2. If the previous steps did not reach one object, but the buildings left for further analysis are all EHR-connected, then the buildings are checked for EHR status, i.e. the highest priority is given priority:
1) exists (*olemas*);
2) under construction (*püstitamisel*);
3) planned (*kavandatav*);
4) the rest are equally of the lowest priority:
demolished, under demolition, not implemented, deleted, unassigned, no additional EHR data (*lammutatud, lammutamisel, realiseerimata, kustutatud, määramata, EHR lisaandmed puuduvad)*.

2HD3. If even in this way one building is not reached, the building with the higher ADOB_ID of the first version is selected.

3.3.3 Spatial data of addresses and address objects

The purpose of the address is to deliver (to ensure the findability of objects). Thus, spatial shapes play an important role in ADS.

- The shape of EHAK objects is a surface.

- The shape of address areas is a surface.
- The shape of the thoroughfares is a fig.
- The shape of the buildings is a surface (in rare cases it can also be a point or a line, i.e. it needs to be arranged).
- The shape of a cadastral parcel is a surface, the geometry of a land unit not entered in the cadastre can be a point or a surface.
- The shape of the building part is a point. It is mostly located in the centroid of the building, but it can be shifted on the surface of the building to the door next to the corresponding thoroughfare, according to which the part of the building is addressed (in the case of a building with a parallel address).

All EHAK objects and address locations (thoroughfares, address areas) always have a spatial shape. The spatial shape of these objects can also consist of separate pieces.

At the same time, it must be noted that many address objects in ADS do not yet have a spatial shape. This results from cases where the entry of a building without a spatial form in the register of construction works is not yet linked to the spatial form mapped in ETAK. In particular, such buildings without an associated spatial form are production or ancillary buildings without a UN designation (~223,000). With the UN designation (especially residential or social buildings) there are mostly room shapes, and also the previously described connections between the mapped ETAK buildings and the buildings of the register of construction works have been made.

99.99% of addresses have a reference point. This is ensured because, in most cases, there are several objects with the same address, and generally at least one of them also has a spatial shape (the representative point of the address is chosen from among objects with a shape related to the address, i.e. the reference point of the address is the characteristic point of the most primary object).
There are a very small number (less than 100 addresses) of cases where a non-spatial object is the only object with an observable address. Then there is no representative office at the address either. Most likely, such addresses without a reference point are not someone's place of residence or place of business, which is why interfacers usually have no contact with these addresses.

All previously mentioned data errors are indicated in ADS and the municipalities deal with the organization of these data. Therefore, the number of buildings without a spatial image and addresses without a reference point is decreasing.

Since three types of reference point coordinates are issued in ADS statements and services, it is important to distinguish which point is more precisely.

Types of reference points:

1. **The object's reference point, usually located in the centre of the object's spatial shape.**
    a. **Can be saved, for example, from the X-tee service ADSobjotsingV8, ADSobjjarglasedV4.**
    b. **There is no separate log event in the X-tee service when it changes, you have to monitor changes in shape ADSobjmuudatusedV7 in the service.**
    c. **Object reference point data is in ADS Public service extract No. 5 (*Jõusolevad aadressiobjektid* - Effective Address Objects) or in FTP extract AADRESS_OBJEKTID, KEHTETUD_JA_KEHTIVAD_AADRESSIOBJEKTID ja KEHTETUD_JA_KEHTIVAD_AADRESSIOBJEKTID_KUJUGA.**
2. **The address point of the object, i.e. the connection point between the object and the address - can be located on the building, e.g. at the entrance or elsewhere, i.e. it can be manually edited. For example, some buildings with parallel addresses have each address shifted to the side of a specific street, but this has not been done outright. Generally, however, the object is automatically located in the centre, but when the shape of the object changes, it is not automatically recalculated, as it may have been manually changed. Thus, an object may be slightly offset from the centre during its life cycle, even when it is not manually changed.**
    a. **Can be saved, for example, from the X-tee service ADSobjotsingV8, ADSobjjarglasedV4, ADSobjajalooline.**

   b. **The change of this point is logged in the service ADSobjaadrmuudatusedV5 -K- event.. That is, this point can change without changing the object or address. But with a change of address or object change, it can be updated from the X-tee services ADSobjmuudatusedV7 and ADSobjaadrmuudatusedV5.**
   c. **Object address point coordinates are in ADS Public service extract No. 1 (*Jõusolevad aadressiobjektid koos aadressidega* - Address Objects in Force with Addresses) and in FTP extract ADOB_AADRESSID.**

3. **The representative point of the address or the reference point of the address - is chosen from among the address points of the objects on the basis of priority rules, so that the address generally refers to the most important object related to this address. That is, the representative point of the address is the address point of the primary object.**
   a. **Can be saved from the X-tee service ADSaadrotsingV5, ADSnormal, ADSaadrjarglasedV4**
   b. **When changing, a separate log event is generated in the -P- X-tee service ADSaadrmuudatusedV7.**
   c. **The coordinates of the representative point of the address are in ADS Public service extract No. 9 (*Aadressid koos komponentide ja ajalooga* - Addresses with components and history) and in FTP extract AADRESSID_KOMPONENTIDEGA.**

The reference point of the object characterizes the address object, the address point of the object characterizes the relationship between the object and the address, the representative point of the address characterizes the address.

Interfacing with ADS the interfacer must decide for himself whether and which of the above three reference points is relevant in a specific data set/place of use (and then also use the services, extracts issued by the corresponding point). Generally, the description of the services or the statement will tell you which specific point isissued.
In the In-ADS service and the X-tee gazetteer service, generally the address point of the object is issued for objects that have spatial data.
Additionally, please note that services and statements may have X and Y reversed.
That is, for example, this is the case in the X-tee service:

aadressipunktX	Double (min 6300000 max 6700000)	The x-coordinate of the address point of the object, i.e. the connection point between the object and the address.
aadressipunktY	Double (min 300000 max 800000)	The y-coordinate of the address point of the object, i.e. the connection point between the object and the address.

In the extract, the same coordinates are given as follows:
- viitepunkt_x (decimal) – 6-digit east-west axis coordinate of the object address point, corresponds to the Y coordinate of the L-Est system
- viitepunkt_y (decimal) – the 7-digit north-south axis coordinate of the object's address point, corresponds to the X coordinate of the L-Est system

3.3.4 Actions with address object

The life of address objects in the ADS information system proceeds through the following events:

1. **Addition of address objects** – for example, the municipality creates new address objects in the register of construction works in connection with the creation of new buildings, or creates a new cadastral parcel through the MinuKataster and assigns it an address, or a building mapped by ETAK is added to the ADS.

   Substantial change or versioning of address objects – when the data of the address object changes, the existing version of the address object is marked as invalid and a new record of the address object version is added. A new ADOB_ID is created. A common characteristic of a chain of versions is the ADS_OID, which is an invariant size across different versions of the same ADS object. For example, if the municipality changes the address of the address object of the cadastral parcel through the procedural application of the address data system or the address of the address object of the building through the register of construction works, a new record of the version of the address object is created, but the object identifier ADS_OID does not change. However, ADS_OID is not uniquely related to an object in nature and may change for technical reasons. See more below. Versioning is caused by changes to the following attributes:

- ORIG_TUNNUS (if the missing value is assigned, it cannot be changed in the future)
- OLEK
- UNIK
- HOONE_OID

Versioning is also caused by changing the shape and address of an object.

2. **Version correction of the address object** – a version correction is a small change of the address object that does not provide a basis for creating a new version of the address object. The name and address of the object must not change with the version correction. The shape can change with a version correction only in the topological scope. With a version correction, the legal basis of the object and its date may change. Several additional features of the object also change with the version correction, from which changes in ETAK_ID, SISSEPAASU_KORRUS, KUJU_MOODUSTUSVIIS are logged. A change to the rest of the additional features does not cause a version correction log.
3. **Cancellation of the address object** – the address object is cancelled if the object disappears from nature or is mistakenly added. An object may also be merged with another object, so it is simply no longer considered a separate object. The address object remains in the database, but its status has been cancelled and the expiration date has been met. Upon cancellation, no new version is created, i.e. the last current version becomes cancelled.
4. **Restoring the address object –** action to compensate for erroneous cancellation. Used to restore previously revoked address objects.

More complex operations such as joining or splitting address objects take place through several operations. For example, the joining of address objects can occur as a sequence of events:

1. Changing or versioning the address object that remains permanent.
2. Cancellation of address objects that disappear after merging.

or

1. Cancelling all address objects to be merged.
2. Addition of a new joint address object.

For example, the so-called flickering of the spatial shape of the object has caused questions for interfacers. In general, the EHR building connected to ETAK has an ETAK-mapped space shape, but the local government has the option to change it (over-digitize, usually via EHR). Both EHR and ETAK provide data to ADS. Therefore, in terms of the shape of the space, the final say remains.

For example, in EHR, a building permit is initially obtained with one shape, then ETAK maps the completed building and says the actual shape according to the eaves, then EHR can, in turn, submit the room shape of the building after the foundation with the use permit document, etc. Sometimes, the ADS administrator arranges the shape manually - for example, the applicant has digitized the approximate shape of the building through the EHR, then the data is still read in the ADS based on the shape received from ETAK. That is why the shape and the way it is formed can so-called fluctuate and this must be taken into account.

In addition, it must be taken into account that, in rare cases, the location of an object may change due to the combined effect of data organization works and different data sources, so that it refers to different objects in nature. For example, a building is given a spatial shape by the local authority based on existing materials, but later it turns out that it is wrong and it is "lifted" to the right place. ADS_OID remains the same, a new version of the object is created, but the location of the associated building in nature is completely different from the previous version

Typical cases of cancelling objects

An address object is cancelled (invalidated) usually when the object is lost (e.g. demolition of a building), but it can also happen during data organization, for example when the existing building is split into parts or when parts are merged together.

The most important cases of invalidation of ADS_OIDs of cadastral parcels.

- The cadastral parcel is divided into new cadastral parcels - all new units get new ADS_OIDs and the old one becomes invalid.
- Cadastral parcels are merged - old ADS_OIDs become invalid and a new one is created.
- The cadastral parcel is closed - this can happen, for example, by a court order or in the event that the local government itself has decided to cancel its order when carrying out land reform - the ADS_OID will disappear and nothing will be created in its place.

Invalidation of building ADS_OIDs.
In general, buildings appear and disappear either through EHR operations or through updating ETAK data. For example, if the building is demolished, dilapidated or the building permit is not implemented. Or if data is corrected, e.g. a building mapped by ETAK or entered in the EHR is actually a facility or small building that does not require addressing and thus is cancelled in ADS. There is also a very small number of exceptional cancellations during data management, e.g. cancellation of EHR duplicate, cancellation of a wrong complex building body (e.g. when expanding a building with a permit, a new part is mistakenly made as a separate room by the applicant) etc. ETAK and EHR buildings can be linked together in ADS.

- When several ETAK buildings are merged, the ADS_OID and ETAK_ID of one (the largest) building to be merged, or the ADS_OID of the object that has an EHR connection attached to it, generally remains valid. The ADS_OIDs of the other parts to be merged will be nullified (become invalid).
- When the building is divided into several buildings, the previous ADS_OID with ETAK_ID will remain for the larger emerging building or the one with an EHR connection, a smaller or an emerging building without an EHR connection will receive a new ADS_OID and ETAK_ID.
- If an EHR building without or with a shape and an ETAK building with a shape are associated with each other, the ADS_OID of the ETAK building will be cancelled and then the ETAK_ID with the shape will be added to the EHR building. When EHR and ETAK records are merged, the ADS_OID of the EHR building remains valid (that is, the object without the same ORIG_ID remains with the same ADS_OID "for the rest of its life") and the ADS_OID of the ETAK building is cancelled (its ETAK_ID is transferred to the ADS_OID of the EHR building). *When adding a new building via EHR, it is also possible to add an EHR code to the existing ETAK building (i.e. then the ADS_OID of the ETAK building will not be cancelled), but this option is planned to be abandoned with EHR development, i.e. in the future, associations will still be made by cancelling the ETAK building.*

As of August 2022, there are still a large number (~223,000) EHR buildings without a spatial shape. In ADS, these buildings appear twice (i.e. for the same building, there is an ETAK building record with a spatial shape and an EHR building record without a spatial shape) - sooner or later they are so-called pairs, therefore building records are exchanged daily with ADS_OID- cancelled in ADS. In 8 years, ~375,000 EHR buildings have received such spatial relations.

Invalidation of ADS_OIDs of building parts.

- ADS_OIDs for building parts will be cancelled if the building is cancelled (for the above reasons).
- Building parts do not exist (anymore) in reality.
- Building parts are cancelled during the data arrangement, as there is no need to differentiate based on the address. For example, there have been wrongly numbered parts of the building in the adjacent building: sauna no. 1, garage no. 2, etc. Or it is a single part of a building:
    - e.g. in a terraced house designed as a complex building, each booth has one part of the building that does not need to be distinguished based on the address - the unique address of the booth/block is used at the building level.
- Building parts are rebuilt (e.g. two building parts are built together). It is also possible that during the renovation the entire distribution of building parts in the house changes or the original construction project is changed. That is, even though some or all building parts numbers are the same, physically they are not the same building parts as on the original plan. Then, in general, the data of the existing

building parts are not changed, but the existing ones are cancelled and new ones are entered in the EHR. (Sometimes the old and new state of building parts of the building can temporarily exist in parallel, i.e. there is a distinction between valid and pending building parts).

- Technical causes. For example, due to EHR technical reasons, the applicant cannot take the building parts submitted with the construction notification to the use notification document. Then the building parts of the construction notice are cancelled and new ones come with the use notice.

3.3.5 Address change

The change of address is completely subject to the change of address objects following:

1. **Upon addition of an address object**, if a suitable address is not found, an address will be created automatically. Any address object is always added with valid addresses that are selected from existing addresses or a new address is created. The creation of an address object, connection or connections with an address or addresses and, if necessary, the addition of addresses always takes place simultaneously, i.e. in one transaction.
2. **When the address object changes** i.e. when a new version of the object is created, the addresses are analysed analogously to the addition and, if necessary, the addresses are changed automatically. More specifically, when changing the address of an object, the previous address link is removed and a new address link is added. That is, the object's previous address (ADR_ID) becomes invalid (if there are no more current objects with this address) and a new valid address (ADR_ID) is added, if this address has not previously been valid for any object.
3. **The error correction of the address object** has no effect on the address.
4. **When an address object is cancelled** the related addresses that were not linked to any valid address object are automatically cancelled (in one transaction). For example, if an address object representing an administrative unit is cancelled (when two municipalities join, the address objects of these two municipalities are cancelled and a new municipality is created), then all the addresses of the joined administrative units are cancelled in cascade. However, cancelling address objects representing, for example, buildings or parts of a building causes the related addresses to be cancelled only if the address object being cancelled is the last of the valid ones for that address (i.e. there are no other objects with that address).
5. **Restoring an address object** is the reverse of cancellation, i.e. previously invalidated addresses are restored. When restoring the object, the existing valid address may also be used, i.e. the address may not be restored.

3.3.6 Sequence of cancelled components, addresses and objects

As a follow-up, the ADS system tries to find replacement components, addresses and objects for cancelled components, addresses and objects.
It is anticipated that there may be a significant time lag between the cancellation of a component and the emergence of a replacement component. This time difference is the smallest for EHAK objects, because the land cadastre processes EHAK changes as a package, EHAK changes of one area are processed together. However, the ADS reporting logic causes changes to be reported independently: cancellations are reported first, then additions and changes. From the point of view of ADS, it does not matter if the time difference between submissions is 1 minute or 1 day.
For component succession to occur, the replacement component must be at the same level as the cancelled component. For address succession, it is not important that the replacement address consists of the same levels as the cancelled address. For object succession, the replacing object must initiate the same levels as the object to be replaced, and the type of object also depends: a cadastral parcel can be succeeded by a cadastral parcel, a building can be succeeded by a building, while residential can replace non-residential and vice versa.

Adding or removing a ancestors/descendants does not result in a changelog. Descendants can change over a long period of time and may not be 100% accurate - based on this knowledge, it is decided whether and how

ADS succession information is used in the consuming dataset. Descendants are given as of the time when the consumer asks for them (generally when reading the change event of the corresponding address or object) and as of the corresponding time, operations with the descendants are generally already performed in the consuming dataset, so later changes of ancestors/descendants generally no longer play a role for this consumer. If you want to update the status of descendants later, you can do so using extracts or you can request a new status of descendants for an object cancelled during a certain period (X-tee services **ADSobjjarglasedV4**, **ADSaadrjarglasedV4**).

In the latest versions of the services, ancestors and descendants are generally issued in a logical order:

- **Ancestors are ordered according to the time of invalidation in descending order, i.e. the most recently valid ancestor is the first.**
- **Descendants are ordered according to the time of validation of the first version in ascending order, i.e. the descendant that has existed the longest is the first.**

The sequence of cadastral parcels

The sequence of cadastral parcels is always identified by spatial analysis, because all cadastral parcels have a shape and it is a surface. Point-shaped cadastral parcels, which do not yet have an identifier, do not participate in the analysis of descendants.

In order to calculate the sequence of objects, the events of adding, deleting and changing the shape of an object are analysed. Depending on the event, cancelled objects overlapping the current shape are found, or vice versa.

Current is a descendant of a cancelled object if one of the following two conditions is met:

- **The absolute value of the overlapping surface must be greater than 300m2 (from September 2022, the parameter can be changed).**
- **The share of the overlapping surface in the smaller object must be greater than 20% (August 2022, the parameter can be changed).**

If the controllable parameter is changed in the system, it will be applied to the calculations of succession relationships that will take place in the future. Previously found sequences are not recalculated because they have already been issued to consumers through the Services.

Sequence of buildings

In the case of buildings, the difference is that not all buildings have a shape. If the cancelled building has a shape, the successors are found among the current ones in the same way as in the case of cadastral parcels using the method of surface overlap with spatial analysis.

A descendant of a cancelled object is found among valid ones if one of the following two conditions is met:

- **The absolute value of the overlapping surface of the space shape of the valid and cancelled object must be greater than 16m2 (August 2022, the parameter can be changed).**
- **The share of overlapping surface in the smaller object must be greater than 50% (August 2022, the parameter can be changed).**

A current building with the same EHR code is also considered the successor of the cancelled building. Such a succession can theoretically occur when a building of one type is cancelled and a building of another type is created.

Sequence of building parts

The successor of the cancelled part of the building is the part of the building with the same designation, either in the same building, if the building is current, or in the successor of the building, if the building has also been cancelled.

If there are several parts with the same designation in the building, the part with the same nearest address becomes the successor. If the proximate address does not match any, then all parts with the same designation will be the successor.

The name of the level 8 component of its address is considered to be the designation of the building part.

It is not excluded that the part of the building has several addresses with different signs. In this case, all building parts with such designations become successors, i.e. there may be several successors.
The same logic applies to the current part of the building when searching for ancestors. Ancestors are also searched both in the same building and in the building's ancestors. The building may be current, the cancelled part of the building with the same designation becomes the predecessor.

Description of the process of calculating address sequences

An ADS system needs address sequences more than component sequences. Data analysis showed that the sequence of addresses is more informative than that of components, and that the life cycle of components and addresses can still be different, even though at first glance they seem quite similar.

Component change events are only handled for levels 1-5. For levels 1 - 3, the sequence of components and addresses is calculated completely independently. 4. -5. when identifying the sequence of level components, the sequence of corresponding addresses is also added. Each component has exactly 1 direct address associated with it. In addition, address change events are also processed.

Consistency of level 6-8 components is derived from address sequence. Only address change events are analysed, in addition to related object change events if they changed addresses.

Address sequences are added depending on the levels specified in the address, which also correspond to the related object types:

- EHAK addresses - only levels 1 - 3 are specified;
- Address area and thoroughfare addresses - 4th - 5th level is specified and 6th - 8th level is not present;
- Addresses of buildings and cadastral parcels - levels 6-7 and no level 8 are specified;
- Addresses of Building Parts - Level 8 is set.

Address sequences are identified in the ADS system during post-processing of logs. Computation is triggered by processing add, change, cancel, and restore events for components, addresses, and objects.

There is extensive business logic for identifying address succession, which includes a number of different methods for identifying a possible successor. Since several objects are generally associated with an address, there are usually more possible sequence relationships for addresses. Addresses become invalid every day, mostly because the addresses of the address objects are changed, but the ADR_ID also changes when the address text remains the same, but the address has changed, e.g. the EHAK level (EHAK code). The address object is more permanent and the objects have fewer succession relationships, so it is suitable to use the address object as the main relationship to keep the address data up-to-date. A more detailed description of the business logic of automatic data update is in the chapter 4.6.

3.3.7 Life cycle of addresses in the bigger picture

According to the Spatial Data Act, the setters (enforcers, changers and cancelers) of place-addresses are city and municipality governments (local governments). Therefore, the largest data providers are local governments, which determine place-addresses in connection with land management activities, planning activities and construction activities.
In addition, major changes to administrative and settlement units are made in the land cadastre on the basis of international agreements, the *Riigikogu*, the Government of the Republic, ministers or the legislation of the Director General of the Land Agency and are transmitted to the ADS information system. Local governments also determine and change the place names of addresses, i.e. address areas and thoroughfares, by legislation. Their changes are received by ADS through the Place Name Register (KNR).
In case of less important changes, legal acts are not adopted and the changers are individual officials of the local government, ETAK mappers and officials of the Land Board.

The Land Board (*Maa-amet*) plays the biggest role in harmonizing data between data from different sources. For example, if the administrative boundary changes, the Land Board makes the corresponding address changes for cadastral parcels, buildings and parts of buildings in the ADS information system.
The Land Board also deals with smaller-scale spatial changes. Small adjustments are called topological rearrangements. For example, if a river is mapped to a different place, the location of the administrative border

in the cadastre may also be changed, but certainly not to such an extent that buildings and cadastral parcels get a new address as a result. Addresses are not changed during topological arrangement. Major changes, including changing boundaries to the extent that also changes addresses, are made only on the basis of the relevant legislation of the above-mentioned important decision-makers.

3.3.8 Sources of Addresses

Sources through which address object data in the ADS information system can change are:

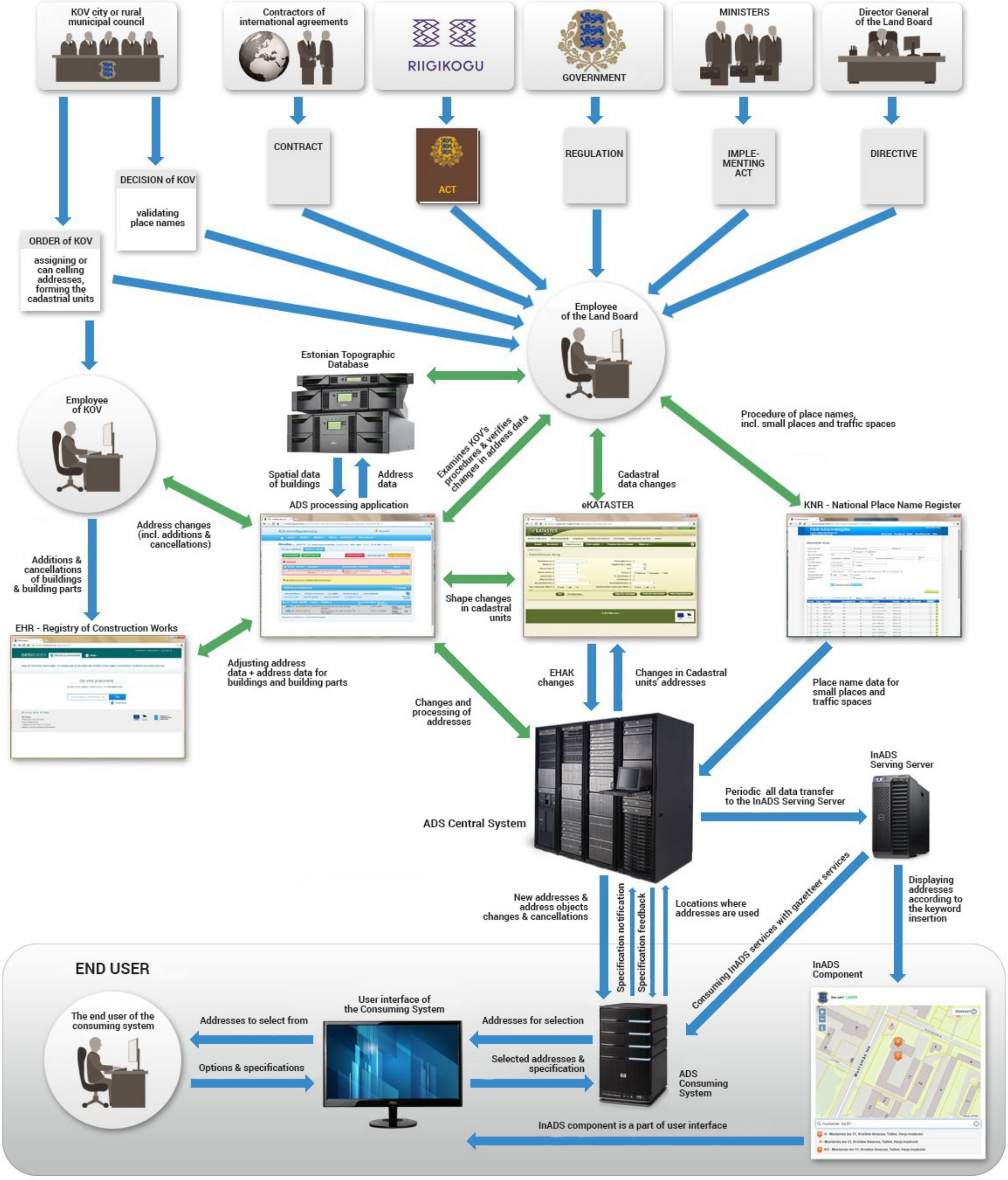
1. **Address data system procedural application**, or ADS information system procedural application. Through the application, local government officials can change existing address objects, addresses and other address data. The Land Board also performs various address operations in the procedural application (e.g. in case of changes to EHAK).
2. **Estonian topographic database** (ETAK) – mapping data. When registering a building in ETAK, the building's identification data is transferred to the ADS information system, where the local government assigns its address through the procedural application. By default, the address of the cadastral parcel is provided by ADS. ETAK also provides information on changes in the spatial shape of the building, including destruction, as well as information on the type.
3. Register of construction works (EHR) - addresses for most buildings and building parts, for some buildings also room layout. The spatial shape of the EHR building and the address of both the building and the part of the building can be changed in ADS (changes are transmitted to EHR and ETAK), while the same changes can also be made in EHR (in this case, changes are transmitted to ADS). ETAK transmits changes to the spatial shapes of a large number of buildings to ADS (ADS transmits them to EHR). A building can have a connection to both EHR and ETAK at the same time, or only to both. The address must be assigned to the building and part of the building before its final updating in the EHR, so there may be a temporary situation where there is a building or part of a building with an actual EHR connection in the ADS, which cannot be found in the current data of the EHR. On the ADS side, such objects are mostly distinguished by a **pending** (*ootel*) status.
4. **e-Cadastre** (*e-Kataster)* (land cadastre) – changes and deletions (closures) of cadastral parcel shapes, changes to the legal basis. The creation of a cadastral parcel is not initiated by *e-Kataster*. A cadastral parcel as an ADS object must appear in the ADS information system before it can be registered in the e-Cadastre. Such cadastral parcels are pending in ADS and without a cadastral identifier. In addition, *e-Kataster* deals with the management of the entire life cycle of EHAK objects (level 1-3) (creation, all changes and closure). ADS only uses EHAK data, but does not make changes to them. With the introduction of the MinuKataster solution from April 2024, changes to cadastral parcels will no longer be received through this process.
5. **Place name register** (KNR) - thoroughfare names and address area names and all their life cycle changes.
6. MinuKataster - the creation of cadastral parcels with an identifier (also in the case of pending parcels), the change of the cadastral unit's purpose and connection to the Land register.

In addition, there are specifying sources whose actions may cause changes in addresses. The most important of them are:

- **Population register** (RR - *Rahvastikuregister*), from which information on ADS connections of residence and additional addresses is transmitted to ADS. RR relationship information may cause the need to correct address data in ADS (e.g., adding a UN identifier to a non-residential building).
- **Land register** (KR - *Kinnistusraamat*), through which information on the address object connections of the real parts of the apartment property is received.
- **Business register** (AR - *Äriregister*) provides information on the legal entity relationships of addresses and objects.

In addition to the above, all systems that interface with the ADS information system have the right to make proposals for specifying or changing addresses.

A general picture of address data regulations and simplified data flows between systems is as follows:


KOV city or rural municipal council
Contractors of international agreements
RIIGIKOGU
GOVERNMENT
MINISTERS
Director General of the Land Board
CONTRACT
ACT
REGULATION
IMPLE-MENTING ACT
DIRECTIVE
DECISION of KOV
validating place names
ORDER of KOV
assigning or can celling addresses, forming the cadastrial units
Employee of the Land Board
Employee of KOV
Estonian Topographic Database
Spatial data of buildings
Address data
Examines KOV's procedures & verifies changes in address data
Procedure of place names, incl. small places and traffic spaces
Cadastral data changes
ADS processing application
eKATASTER
KNR - National Place Name Register
Address changes (incl. additions & cancellations)
Additions & cancellations of buildings & building parts
EHR - Registry of Construction Works
Shape changes in cadastral units
Adjusting address data + address data for buildings and building parts
Changes and processing of addresses
EHAK changes
Changes in Cadastral units' addresses
Place name data for small places and traffic spaces
InADS Serving Server
Periodic all data transfer to the InADS Serving Server
ADS Central System
New addresses & address objects changes & cancellations
Specification notification
Specification feedback
Locations where addresses are used
Consuming InADS services with gazetteer services
Displaying addresses according to the keyword insertion
END USER
The end user of the consuming system
Addresses to select from
Options & specifications
User interface of the Consuming System
Addresses for selection
Selected addresses & specification
ADS Consuming System
InADS Component
InADS component is a part of user interface

4 Interfacing with ADS

4.1 Legal framework

Interfacing with ADS is governed by three main pieces of legislation:

Spatial Data Act §59 (1) and (2)

> *The use of ADS* ***is mandatory for information holders when maintaining databases and processing address data*** *related to* location addresses*.*

Public Information Act §43[9](3)

> *The use of ADS as a* ***system securing the state's information system is mandatory*** *for maintaining all state and local government databases.*

ADS regulation §4

> *(1) The keeper of the database processing the address data and the place-address determiner are obliged to* ***use the address data of the ADS information system to*** *determine and* ***process the address data****.*
>
> *(2) In order to use the address data system within the meaning of §59 of the Spatial Data Act, the* ***data collection is directly or indirectly interfaced with the ADS*** *information system.*

The conditions for interfacing with ADS are specified in more detail in the ADS regulation (EST). In other words, it is not enough that the address comes directly or indirectly from ADS, but according to § 4 (4) of the ADS regulation, the information holder must do the following in his database in order to interface with the ADS information system:

1) align the place-addresses in use in your database with the place-address structural elements of the ADS information system and create other necessary connections to ensure the interoperability of the database;
2) associate the place-addresses of your database with the place-address identifier of the ADS information system or with the identifier of the address object;
3) ensure the connection of the data used in its database with the up-to-date address data of the ADS information system (including the place-address identifier and, if necessary, the identifier of the address object). It is not necessary to update the archived data links.

In a simplified way, addresses must be converted to the normalized form of ADS, equipped with ADS identifiers and kept up-to-date. The following is a more detailed description of how to do this.

It is recommended that data collections belonging to the state information system also familiarize themselves with the terms of approval of the Land Board [EST] in RIHA. RIHA is the administrative system of the state information system. https://geoportaal.maaamet.ee/docs/aadress/ADS_X-tee_services.pdf

4.2 Means of interfacing with ADS

Technically, various tools can be used to interface with ADS. The X-tee services offered by ADS are mainly used for interfacing, see more from ADS X-tee services document.

The In-ADS service is suitable for address search and selection. This is an additional opportunity to use the web-based user interface in addition to X-tee services. This service is suitable for systems that do not need to update the stored address, as well as for those who want to create an initial connection of the address through a fast and convenient web application and later update the data through X-tee. This means that the use of the In-ADS component can be combined with the ADS X-path services, because both use exactly the same unique features.

Geocoding service can be used to organize existing data and connect it to ADS.

Extracts of address data that are available can be used to bootstrap and in certain cases update the data:

a) From the public application of ADS https://xgis.maaamet.ee/adsavalik/extracts, see also additional information under the HELP link;
b) From Land boards FTP server ftp://ftp.maaamet.ee/ADS_valjavotted/ADS_valjavotted/, see "readme.pdf" file; You can use, for example, Filezilla or WinSCP to access ftp.maaamet.ee.

Extracts in the public application of ADS and on the FTP server of the Land Board have a different structure and at a different creation time (in the public application, they are updated with an interval of 30 days, extracts for the FTP are produced on the 1st of every month).

4.3 Working plan for interfacing with ADS

In the working process of interfacing with ADS, the following most important stages can be identified:

1. Determining the initial state of the interfacing system and selecting the level of interfacing.
2. Achieving the interfacing readiness of the interfacing system ((re)construction of data structures, address processing logic and user interfaces).
3. Loading address data from the ADS information system and automatic normalization of existing addresses, i.e. matching with ADS addresses.
4. The use of an interfacing system as a transition period, where the addresses left over from the automatic encoder are coded by hand.
5. The interfacing system has completely transitioned to using ADS.

4.3.1 Determining the initial situation and choosing the way and level of interfacing

Determining the initial situation must include at least the following activities:

1. Find places in the data model that have contact with the address. Mapping where addresses are used and stored as classifiers in the system's data models.
2. Same operation for user interface and machine interface layers. Places for entering, searching and issuing addresses (including reporting, publications, statistics) must be found in the user interface. Machine interfaces mean interfaces for receiving or transmitting address data to other systems.
3. If there are background processes related to the processing of addresses, they should also be named and the contact with the address should be described.
4. Name and articulate the most important business processes related to addresses, i.e. you need to understand how addresses are created, changed and lost, and whether they should also be updated.

The choice of the interface level largely depends on the system's existing features and requirements, but implementation and maintenance costs should also be taken into account.

Sometimes, as a result of the analysis, it may turn out that the system actually does not need the stored address at all in its later operation. In this case, you could consider giving up the address in this system altogether. For example, cases where the address stored in the database is never used in practice, contact is made with the subject in rare cases and then instead by e-mail or phone.

Another place of savings can be found if it is necessary to save the addresses, but it is not necessary to keep them up-to-date later. **In practice, such use cases are very rare.**

It is often believed that if a subject's address changes, he must notify the database himself, i.e. there is no need for an automatic data update. A distinction must be made between a change of address in connection with moving a person, place of business, etc., and a change of address of the same building/apartment, e.g., due to a change in the EHAK or street name, numbering, etc. address arrangement. In the first case, the subject must notify the change of location himself, while in the latter case, addresses must be automatically updated in the database belonging to the state information system on the basis of ADS data.

In addition, it is necessary to analyse whether the system necessarily needs to specify the ADS address (if the ADS does not contain the address to be searched for). In 2022, the completeness of address data is quite good, so it is very rare that an address is missing from ADS. **Therefore, in general, it is sufficient to use the addresses registered in ADS, and allowing clarification is not justified.** Waiver of the requirement to specify the address allows saving a large amount of development resources when creating the system. In most cases, the reason for the lack of an address is instead the fact that an incorrect address is used, which differs from the address assigned to the object by the local authority. In very exceptional cases, however, it may be necessary to add clarifications to the address, for example, in the address data system, the address does not include the numbers or signs of buildings, floors, terminals, warehouses, etc., which may be needed in the database. Such information is stored in a separate data field if necessary. When selecting an address from In-ADS, it is also possible to add clarification to the address, if the exact address cannot be found at the level of the building part.

The **choice of identifiers** to be saved depends on whether the data needs to be updated, whether the data is forwarded to the following databases, whether it is necessary to reconcile information with a third database based on address data or to make inquiries somewhere, etc. Storing the master key is also mandatory for time-locked addresses, i.e. the stored address/object must have a reference to ADS. You also have to think about what data will be needed later, for example, to make statistics, etc. For example, does the EHAK part have to be stored by level, zip code, unofficial area information, etc.

We recommend consulting with the Land Board during the analysis phase of interfacing with ADS. The Land Board is often asked how exactly to interface, including which service to use, which identifiers to save, etc. There is no one-size-fits-all answer to this question, because the **exact business logic of the interfacing depends on the needs of the specific data collection and the use case of each specific data field**. In most cases, the data sets have several address fields with different content and purpose, and the business logic of their ADS use is also different, for example, the venue of the event does not need to be updated, but the address related to the activity license must be up-to-date. Some addresses are taken from ADS through X-tee services, some from In-ADS, some are inherited directly from the Population Register, Business Register, etc.

In other words, the exact technical solution depends on the specific place of use.

<u>Some more general tips and ideas are as follows:</u>

- In-ADS is well suited for the primary selection of address and address object, but for automatic updating of addresses, you generally need to use ADS X-tee services. In other words, the user is offered, for example, the In-ADS user interface with a card to select an address, from which the user selects a specific address object, on the basis of which the customer's address data is later kept up-to-date.
- In order to automatically update addresses as much as possible, it is necessary to save both the address and the object link. If an address object is changed in ADS (the address associated with it or the object itself is changed, e.g. the object is cancelled), this change is also made in the base of the consuming system (from the X-path log services of ADS) and from this so-called ADS copy of the consuming system to the internal through the services, a change is also made to the address object related to the customer, event, contract or the like, if a specific address related to the customer or the like must be kept up-to-date.
- For some consumers (mainly the private sector) it has been sufficient to use only In-ADS, i.e. with a certain regularity (e.g. once a year) the customer is asked to update his address data himself. Or, in the case of smaller volumes, it is also possible (e.g. once a month) to update addresses on the basis of public csv extracts from ADS.
- In case of smaller volumes, it is also possible to try to update the address on behalf of the customer based on the ADS address object identifier (ADS_OID). For example, if there are few addresses and operations with them, in every situation where the address is used (e.g. an event or

a request), you can check whether the address of the given object is still the same and whether the object is still valid with an automatic query of In-ADS. If the object is not valid, the customer must be asked to update the address or implement further automation, e.g. use the object's descendants request (X-tee service) and/or find a new object link via the address (if the address itself is still valid) etc.

- If the exact location of a specific object is not so important and possible inaccuracies in updating the address are tolerable, then it may be sufficient to save only the address link (ADR_ID).
- It all depends on the purpose and how up-to-date the address data needs to be kept. It must be taken into account that the machine can never fully cover the entire innovation, i.e. there are a small number of situations where a person has to intervene.
- Sometimes it is not necessary to use ADS directly, but it is sufficient to use data from registers directly interfaced with ADS. For example, the addresses of natural and legal persons are inherited from the population register and the business register (addresses are also updated from these registers through the connection of persons).
- Regular synchronicity checks must also be performed when interfacing with ADS. For this purpose, you can use the public extracts from ADS (see chapter 4.2) or request a special data set from the address data department of the Land Board (*Maa-amet)* ads.abi@maaamet.ee.

4.3.2 Achieving interface readiness

Achieving the readiness of an interfacing system usually requires development work in this area, which is the most expensive in terms of work volume. It is advisable to involve workforce who have already done the corresponding design work in the reconstruction design, because understanding the concepts of address and address object and their dynamics may take more time and resources than preparing the corresponding reconstruction projects.

Those who interface with the ADS system need to analyse their data in use against ADS data in order to do the preliminary work necessary for interfacing. Booting is also required to create a copy of ADS on the interfaced system. It is not recommended to do this through X-tee services, but using public statements generated from the ADS system. Extracts provide the ability to download address data available on X-tee, but making extracts via X-tee services is troublesome, suboptimal and very time-consuming. It is recommended to request only changes through X-tee services.

ADS extracts can be used to bootstrap ADS data:

a) From the public application of ADS https://xgis.maaamet.ee/adsavalik/extracts, see also additional information under the HELP link;
b) From Land boards FTP server ftp://ftp.maaamet.ee/ADS_valjavotted/ADS_valjavotted/, see "readme.pdf" file; You can use, for example, Filezilla or WinSCP to access ftp.maaamet.ee.

Extracts in the public application of ADS and on the FTP server of the Land Board have a different structure and at a different creation time (in the public application, they are updated with an interval of 30 days, extracts for the FTP are produced on the 1st of every month).

4.3.3 Normalization of addresses

The address normalization process is designed to encode manually entered addresses or parts thereof, i.e. to match valid ADS addresses.

If the addresses do not need to be updated, the data can be left intact in all existing address usage locations and only the query and user interface locations can be rebuilt to work with old and new data in terms of the query.
If the addresses must be up-to-date and it is necessary to link the active address data with ADS, then it is inevitably necessary to normalize the addresses, i.e. align the existing data with the official addresses of ADS. If

you are lucky, it will be possible to automatically code all addresses and there is no need to do manual work. In practice, such luck does not occur and the situation where some addresses remain operational in the old solution must be taken into account in the system development plan. For example, when matching the addresses of the Population Register, nearly 70% correspondence was achieved in 2011, 91% in 2012, 96% in 2014, and 99.99% in 2022. The remaining addresses will continue to be corrected and linked manually.
The address coding (normalization v. association) process can also be done proactively (as one-off jobs outside the system), i.e. during the transition, only the most recently added or changed addresses are coded. Coding addresses can be a very time-consuming process (lasts several weeks for larger systems) and it is not possible to keep the system out of action all this time.

Normalization process

Depending on the volume of addresses that need to be normalized and the technical possibilities, either ADS X-tee services or a geocoder or In-ADS gazetteer service can be used for this activity (read more in the In-ADS document In-ADS developer guide).

If normalization is performed via the X-path, the address is first tried to be normalized with the **ADSnormal** service. In the input of the service, the codes of the components and the non-normalized proximate address are given. If the service finds exactly one address and the ADR_ID is valued in it, it is stored in the address record, the address match is considered found, and the status of the address is set to the value of the status received from the service (as a rule, it is -K-, i.e. valid (*kehtiv*)). If the service finds more than one address, this result is ignored and address normalization remains manual.

If no address was found with the **ADSnormal** service, the **ADSaadrotsingV5** service is tried. The service input is shown as text, which is made up of the names of all used components and the user-entered text NAME_VASTE_PUUDUB. If the service finds exactly one address and adrId is valued in it, it is stored in the ADR_ID_ADS column of the address record, the address match is considered found and the address status is set to the status value received from the services. If the service finds more than one address or ADR_ID is unvalued, this result is ignored and address normalization remains manual.

4.3.4 Transition period

During the transition period, addresses excluded from automatic coding will be manually coded. This requires the creation of a separate job or jobs. Through such workplaces, the uncoded (without ADS connection) addresses must be screened and the corresponding ADS addresses must be searched for, using aids, including the ADS public application provided by the Land Board (*Maa-amet*), which also contains extracts, a geocoder or using the capabilities created by the In-ADS component.

The list of uncoded addresses should be continuously shortened as the work progresses, because new uncoded addresses cannot be produced in the system. Inevitably, some of the addresses remain unlinked manually, despite human efforts.

There are three recommendations for such addresses:

1) they should be left uncoded and during the entire life of the system it is taken into account that there may be addresses in the form of text in places where addresses are used.
2) cut off the inaccurate part of the address and encode the address at a more general level. This is also a reasonable choice, because the inaccurate part is not usable anyway, as it does not contain information competent to indicate the location.
3) The third option is to contact the data providers to specify the address.

4.3.5 Ready interface

For each system interfaced with the ADS information system, a situation should eventually come where the problems of normalization of addresses have been solved in one way or another, and the old way of using addresses can be completely abandoned. Depending on the system, it can take a very long time (even years) to get there, but if the necessary resources are available, any system can be converted to use ADS data.

In connection with the continuous development of the ADS system, the consuming databases also need to modernize the ADS interface: adopt the latest versions of X-tee services, improve the business logic of updating data, supplement address data with new data fields. For example, post code information was added to the ADS system in 2019. Until now, consumers used different solutions to process postal codes. From 2019, postal codes are conveniently available with addresses via ADS.

4.4 Interfacing levels

The interfacing of the ADS information system can be classified into different levels depending on the needs of the systems.
In the first approximation, interfacing is divided between three basic levels: indirect interfacing and two levels considered as direct interfacing: light interfacing and full basis interfacing.

The levels of ADS consumption are described below, but before making a decision and realizing it, please pay attention to the chapter 4.3 Working plan for interfacing with ADS

L0 Indirect interfacing - the information system does not directly interface with the ADS information system. Address data from another system is used, for example data from the Population Register (RR) or Business Register (ÄR). No new addresses are entered in the information system, and existing ones are replaced at the interfaced system objects according to their system processes. The system does not create any new connection between objects (subjects) obtained from the external register and location addresses. *For example, it is not allowed to change the relationships between persons and place-addresses inherited from the Population Register*. Even in the case of indirect interfacing, it is necessary to periodically check the data synchronicity with the ADS data. In general, even with an indirect interface, it is necessary to store the ADS identifiers in your system.

L1 Light interfacing - the information system does not save the ADS address or address object data as a whole. User interfaces are served with In-ADS and only at the level of address components if necessary. Address data (including address object data) is taken from the ADS information system in real time as needed.

L2 Full interfacing - the information system takes a copy of all ADS addresses, address objects and their relationships, and this data is regularly updated as a background process. A full-base interface is useful when it is necessary to achieve maximum speed when servicing user interfaces and it is not tolerable a situation where entering addresses in user interfaces gets stuck due to the possible unavailability of the X-tee services of the ADS information system. Such time criticality is a topic, for example, in operational management systems. Soon it will also be possible to use In-ADS as a container solution with every information system, which ensures the availability and continuity of ADS data. For more information, please contact inads.abi@maaamet.ee.

L1 and L2 are divided into sublevels:

A - Time-fixed address data - address data is used only in the context of the time of its use, and subsequent updating of address data is not necessary.
B - Address data kept up-to-date - the address data recorded for the subject or object of the interfaced register must be automatically updated according to the change of addresses, while some inaccuracy accompanying the update is tolerable.

When designing the ADS interface of the information system, the appropriate level must be defined. Within the system, different levels can be applied to different parts to avoid excessive costs for development and system maintenance. In addition to the main and sub-levels given, there may be other combinations according to the specifics of the interfaced register.

In terms of development costs, interfacing using In-ADS is the cheapest. Implicit interfacing is also likely to be cheaper than lightweight interfacing. The most expensive is the full base interface in terms of development and operation.

Deciding on the choice of interface levels boils down to two main questions:

1. Do we need maximum operating **speed**?
2. Do we need **up-to-date** addresses?

Does maximum WORK SPEED matter?	Need to UPDATE addresses?	Suggested itinerary::
NO > is suitable **Light interfacing (L1)**	NO	L1A – time-fixed *
	YES	L1B – up-to-date
YES > is suitable **Full interfacing (L2)**	NO	L2A – time-fixed *
	YES	L2B – up-to-date

**time-fixed: not changing in time, i.e. the state at the moment of saving to the interfaced system is in use, the change of which is not monitored and the history is not kept.*

Achieving maximum speed when servicing user interfaces requires keeping copies of the ADS address and/or address object data in the consuming system, including using the In-ADS container solution (available soon) may also be sufficient. It also ensures better reliability because there is no real-time dependency on an external system. Keeping a copy of the address data in the consuming system requires a larger database data space compared to light interface.

The need to update addresses means creating mechanisms for updating stored address data, including the need to solve the problem of multiple successors.

The need to specify the address leads to the requirement to inform the Land Board about the need to specify the address. The user interface and the database structure serving it also become more complex.

Abandoning the history of the address object simplifies to some extent the follow-up action caused by the change of the address object in the consuming system. The frequency of replacing the address object's primary keys and related foreign keys is significantly reduced.

The levels of interfacing are described in more detail below.

4.4.1 Indirect interfacing (L0)

In indirect interfacing, the address data is obtained from another system together with the data of this system. There is no need to enter the address in the consuming system. This guide does not dwell on this way of interfacing. In the case of indirect interfacing, the following must be requested from the data-giving system:

- ADS attributes: ADR_ID, ADOB_ID, ADS_OID or koodaadress (code address) must be stored in the address data. We recommend using the ADR_ID or ADS_OID value.
- If up-to-date data is necessary, then their change data must also be continuously received from the corresponding register of origin. For example, if the registry in turn distributes RR or ÄR addresses, it must ensure that this data is up-to-date in its registry.
- Refined addresses must be distinguished from ADS addresses, because addresses that do not fully comply with the ADS standard may be less reliable.

4.4.2 Light interfacing (L1)

A light interfacing is probably the predominant choice in information systems where a possible delay of a few seconds when entering an address at the user interface level (X-tee request time) can be tolerated and it is possible to survive at a reasonable cost if the X-tee service consumed to service the user interface should become unavailable (e.g. network connection interruption). Instead of the X-tee service, the In-ADS service can be used for the search, in which case the risk of interruption is reduced (in the case of a container solution, this risk disappears). See also chapter 4.5.

In the case of a light interface, only the address components are taken into the consuming system, either as a whole or only at the first three levels (EHAK levels). In the user interface, the corresponding levels may be offered as options and/or the search for a specific address is performed based on the text (string) entered by the user from the **ADSaadrotsingV5** service of the ADS information system X-tee. The service starts, for example, by pressing the "Search" button. As a result of the service, a list of addresses matching the text entered by the user is offered as a response. The user selects a suitable address and it is saved directly to the concept of the place of use or in a central address table. Addresses are added to the consuming system according to the intensity of input of different addresses. It is not necessary to pull all addresses and/or address objects into the consuming system.

Light interfacing is further divided into levels A and B.

Time-fixed address data (L1A)

Using address data that is time-fixed means that there is no need to update already stored address data. These are addresses that do not need to be processed later or used for analysis (there is no need to make statistics or release up-to-date data to other consumers). This is a natural feature for capturing, for example, events or a location that only has meaning in a given time. For example, addresses on an archived document or an already issued invoice will not be changed later.

In doing so, it should be noted whether there is a need to perform location-based analyses later, e.g. to highlight the distribution of invoices issued over a 10-year period based on addresses. If such a need exists, the corresponding business objects (invoice, archived document) will remain in time (their address data will not be updated), but in view of the needs of later analysis, the logic of address data lineages must be built on top of them (e.g. external attributes of the document or invoice container).

But let's get back to the handling of address data, which is only time-fixed.

Due to time-consuming address handling in light interface, it is sufficient to use only **address components** and **addresses** from ADS. The address selected from the address search is saved either as text only or, if necessary, (also) by component to the corresponding subject/object (for example, the location of the document signing). The identifier of the address version, or ADR_ID, is generally stored as the primary key.

Address data kept up-to-date (L1B)

The use of up-to-date address data means the need to update the stored address data. This need is also immediately present when there is a requirement to perform address-based analysis or statistics. Also, if there is a need to provide an address in the system for concepts whose data needs to be kept up-to-date. Or if the client of the system interfaced with ADS, with whom the data is exchanged, needs to keep the address data up-to-date.

Typical examples here are the address of a person's residence, the address of a company, the address of a service point (e.g. a doctor's office), the location address of an object (e.g. a museum) or the address of a building. This means a situation where already saved addresses need to be periodically updated by background processes based on changes coming from ADS.

NB! Here we are talking about the change of address of the same location (e.g. building, apartment). For example, if the local government changes the location address (including street name or numbering or place name) or if there is a change in the administrative and population distribution. If the location associated with the subject changes, for example the person moves, such address changes cannot be received through ADS. Such information must be provided by the data provider himself, or it is inherited from the original register (e.g. the Population Register) that maintains the relevant subject and address relationship information.

When dealing with address data that goes along with time, we cannot avoid the term "**address object**". The address and the master key of the address object are stored at the point of consumption, because finding the necessary descendants through the address object is more effective. Storing the address only as text is not a good choice to meet the up-to-date requirement, because the processing of text data on a massive scale is still too IT resource-intensive.

As you know, the address object has a many-to-many relationship with the address, and here you have to consider how to select one address object when the user selects the address field. Looking at the address, the existence of several address objects is rather widespread than an exception (for example, the same place-address cadastral parcel, the dwelling on it and three sheds have the same place-address).

The most complete solution consists of three parts:

1. Conventional address selection (provision of suitable addresses corresponding to the string typed in the box, making selections from the selection list from top to bottom, or a hybrid of the two options).
2. Automatic narrowing of the number of address objects according to the selected address. If only one address object corresponds to the selected address, the user is no longer bothered with an additional choice. The master key of the selected address and the corresponding address object is saved.
3. If two or more address objects correspond to the selected address, the most perfect solution should offer to select these objects on the background of the map, because there is a lack of raw data suitable for the informed selection of the address object. The address object is best characterized by its location and shape on the map. The primary object association provided by ADS can be used to create an automatic association if it matches the needs of the interface. See also chapter 4.5.

The implementation of the third point is definitely the most labour-intensive for the consuming system, especially if the consuming system does not otherwise display data in the background of the map. However, the complexity is alleviated by the web-based component In-ADS, which can be easily integrated into the systems (see related descriptions above). If it is not recommended to use the display/selection option on the map, one alternative is to use the primary object provided by ADS or to enhance the automatic field selection of your address object to the point where there is always one object in the selection that corresponds to the place of use of the interface address.

In ADS, there is an identifier of the primary object of an address, which marks one object as more important than others in the context of a given address. For the primary object, see above chapter 3.3.2.

A high-quality and context-appropriate choice of object connection is very important so that further automatic changes take place correctly, i.e. e.g. addresses related to persons are not changed incorrectly. Read more about the renewal process in chapter 4.6.

4.4.3 Full interfacing (L2)

A full interfacing is practical if there is a need to ensure maximum speed and reliability when entering addresses in user interfaces. The table of addresses, objects and their relationship (+ additional tables if necessary) is kept on the side of the interfaced register as a copy of the so-called ADS. In this case, it is necessary to store all the data in the options and search answers in the database of the consuming system. Along with this, there is a need to update the data stored in the database with the recommended regularity of 1 time per day, but not less often than 1 time per week, because too long a gap between data updates accumulates too much unprocessed data and there is a risk that the corresponding log request services of the ADS information system will not be able to make the desired changes. to fulfil the request order within one night. In case of major EHAK changes, there may be tens of thousands of changes in one day (hundreds of thousands in exceptional cases). On normal working days, the number of changes in different log entries (components, addresses, objects) is around 2000-3000.

Soon it will be possible to use an In-ADS container solution instead of a copy of ADS. This means that the data consumer can add one or more copies of In-ADS to their infrastructure, which updates itself every night over the X-path. This ensures greater operational reliability and the operation of the In-ADS component (address search) even in those very rare situations where the Land Board's In-ADS does not work.
This solution is intended for large consumers of ADS data, for whom reliability is also very critical (e.g. Population Census, with a large number of simultaneous users or the Emergency Centre, for whom it is critically important that the address search works at all times).
For smaller consumers, the In-ADS infrastructure provided by the Land Board is sufficiently capable and reliable. If you are interested in using the container solution, you should contact the Land Board (Maa-amet) at inads.abi@maaamet.ee to receive further information.

The full interfacing is also divided into levels A and B. In other words, in the case of a full-base interface, there is a need for updating in any case - the data in the so-called ADS copy must be updated. But additionally, it must be decided whether the address related to the subject will be updated through the copy of the ADS or it will remain for some time.

Time-fixed address data (L2A)

Similar to level L1A, subsequent updating of addresses is not important, but maximum operating speed and reliability are important.

At the L2A level, it is sufficient to use only **address components** and **addresses** from ADS. The address is needed to fill out the selection lists of the user interfaces. Can also offer component-wise address selection. After making choices by the user, the address is uniquely known and its text and master key (usually ADR_ID) are saved to the subject/object of the consuming system.

Address data kept up-to-date (L2B)

In the same way as the level of L1B consumption, the handling of the **address** and the **address object** is also necessary for the L2B interface. Unlike the L1 level, in the L2 level it is necessary to maintain the addresses and address objects in the consuming system in a complete or basic set as a copy of the ADS database. Keeping addresses and address objects in the consuming system ensures the best speed of serving user interfaces. Addresses and address objects are loaded in their entirety as an extract from the ADS information system when the system is set up and are later kept up-to-date as relevant background processes.

The rules for selecting one address object for binding to the subject and one field from several descendants for the address object are the same as the rules described in the L1B interface level.

Interface level L2B is predictably a typical choice for those interfacing with operational services, mail handling and other time-critical systems. A typical need for time-critical systems is to update addresses, and it is not acceptable to have an ongoing workflow disrupted due to the unavailability of ADS X-path services.

4.5 Address search and selection

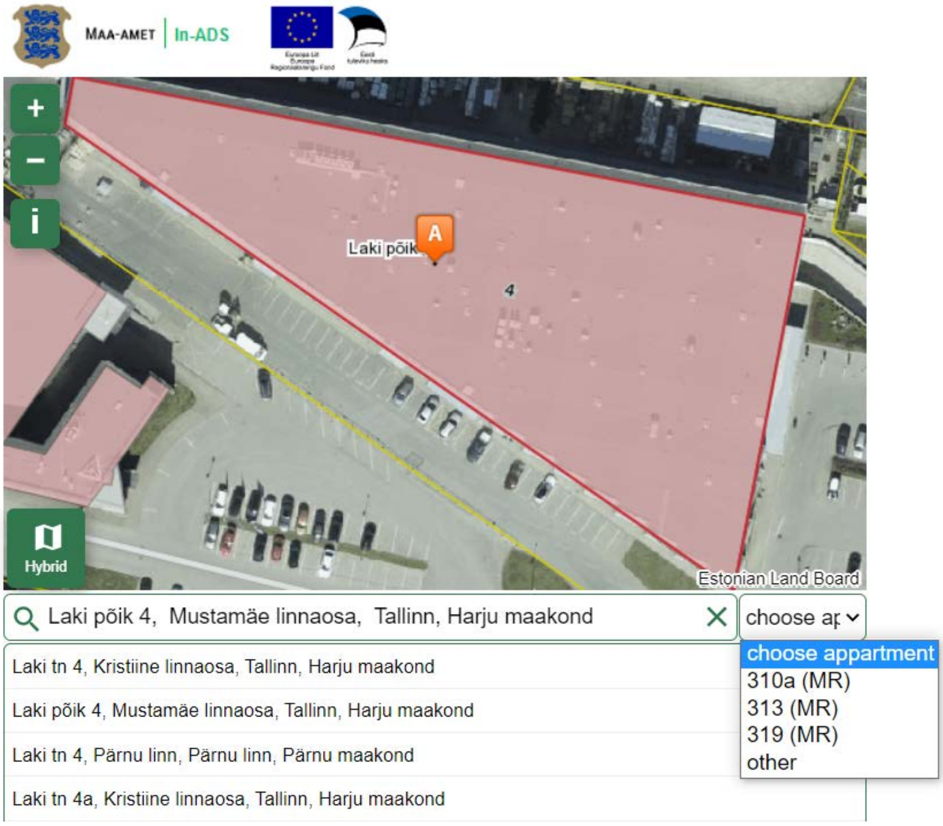
4.5.1 Address search

Three typical user interface solutions are used to select an address field (for example, to a person or an event):

- address selection from a text box with a suggestion list;
- address selection from top to bottom (EHAK) selection lists;
- and their hybrid solution, where a part of the address (for example, the EHAK level) is selected from top to bottom, and then the address is entered into a text box with a list of suggestions.

Address selection from a text box with a recommendation list is a solution where the user starts typing any part of the address into the box and the system offers a list below the box of addresses corresponding to the entered text (parts of words, numbers). Additional filter conditions for address searches.
For example, if only cadastral parcels are dealt with in the place of use, then the object type CU can be added to the filter so that the selection comes only from the addresses of cadastral parcels or that the selection of building and apartment can be used, etc.

In the case of a full-interfacing, the corresponding address search can be made by the consumer, but the Maa-amet offers an integrable **In-ADS-i**, which covers the described need.



Laki põik 4, Mustamäe linnaosa, Tallinn, Harju maakond

MAA-AMET
In-ADS
Laki põik
A
4
Hybrid
Estonian Land Board
choose ap
choose appartment
310a (MR)
313 (MR)
319 (MR)
other
Laki tn 4, Kristiine linnaosa, Tallinn, Harju maakond
Laki põik 4, Mustamäe linnaosa, Tallinn, Harju maakond
Laki tn 4, Pärnu linn, Pärnu linn, Pärnu maakond
Laki tn 4a, Kristiine linnaosa, Tallinn, Harju maakond

The In-ADS web service is an integrable address search using data from the address data system, which can be easily placed in various web-based information systems. In-ADS always has the most accurate and up-to-date addresses, because thanks to the nightly data update, the interface offers the data that took effect in the address data system as of the previous day.

In-ADS is fast and convenient. Through In-ADS, all interested parties can receive correct Estonian addresses free of charge with the possibility to save them with the necessary ADS attributes in their information system. The service is suitable for all institutions that expect users of their web applications to enter an address.

Through In-ADS, it is made convenient for the consumer (addresses are already offered during the entry process) and information about the selected address is transmitted to the application in a systemically processed form. Most of the ADS data described above are available via In-ADS (including e.g. zip code, primary object, information on related objects, etc.).

It is also possible to use object-based search. In In-ADS, addresses can also be searched by unofficial address data, such as points of interest or unofficial areas. Therefore, if the user does not know the exact address, but knows the name of the building/institution, he can search based on it (e.g. *Ülemiste keskus* (Centre), *Märjamaa Gümnaasium* (Gymnasium), *Ungari saatkond* (Hungarian Embassy)) and choose the official address match. Also, if the user does not know his valid address and searches by historical address, he is offered a valid address match. Coordinates, cadastral code, etc. can also be used as input for the search. In addition, there are different priority rules when ordering matches already in use in In-ADS, so that a more accurate match and a more primary object are in front, technical addresses are in the back (can also be excluded altogether), etc.

In-ADS also includes a map component, where address and object searches can be both text-based and map-based. In addition, a static map can also be created without a search, for example to display the location of an institution.

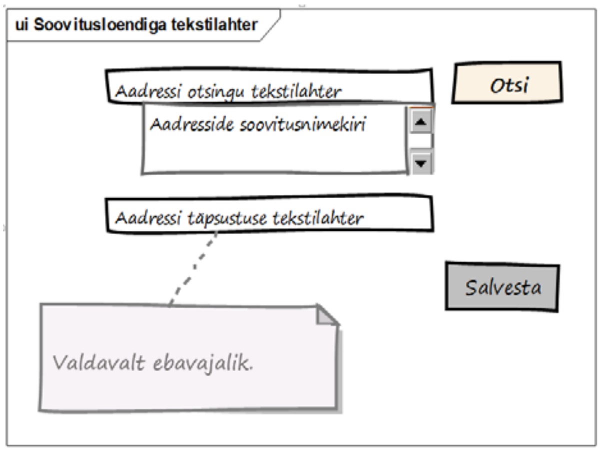
Address data can also be queried without an integrable user interface using the Gazetteer URL query, which returns results in JSON or XML format.

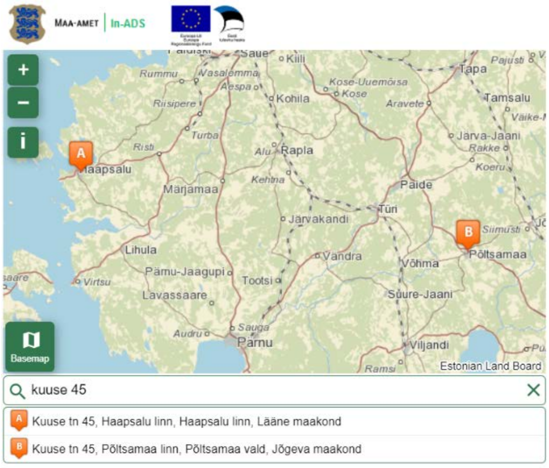
There are many different ways to set up In-ADS. See more on the Geoportal of the Land Board and the In-ADS guides.

All in all, integrating In-ADS is easier and cheaper than building your own address lookup on the interfaced database side. If system reliability and speed are important, In-ADS can also be deployed as a container solution (in this case, In-ADS is handled by the customer's infrastructure and updated nightly). For address search, you can also use the ADS address search or object search X-tee service, or combine it by level with the selection of the interface from the ADS copy, etc.

User interface type solutions

1. The user can move the focus to the recommendation list and select a suitable address there. When using X-tee queries, it makes sense to link offering suggestions to clicking the 'Search' button next to the cell, as challenging the X-tee query is time-consuming. When using In-ADS, the selection recommendations appear quickly without pressing the Search button. You can make it so that when you click on the selection, the selection and immediate saving takes place. A map can also be displayed to help.


ui Soovitusloendiga tekstilahter
Aadressi otsingu tekstilahter
Otsi
Aadresside soovitusnimekiri
Aadressi täpsustuse tekstilahter
Salvesta
Valdavalt ebavajalik.



kuuse 45 ✕

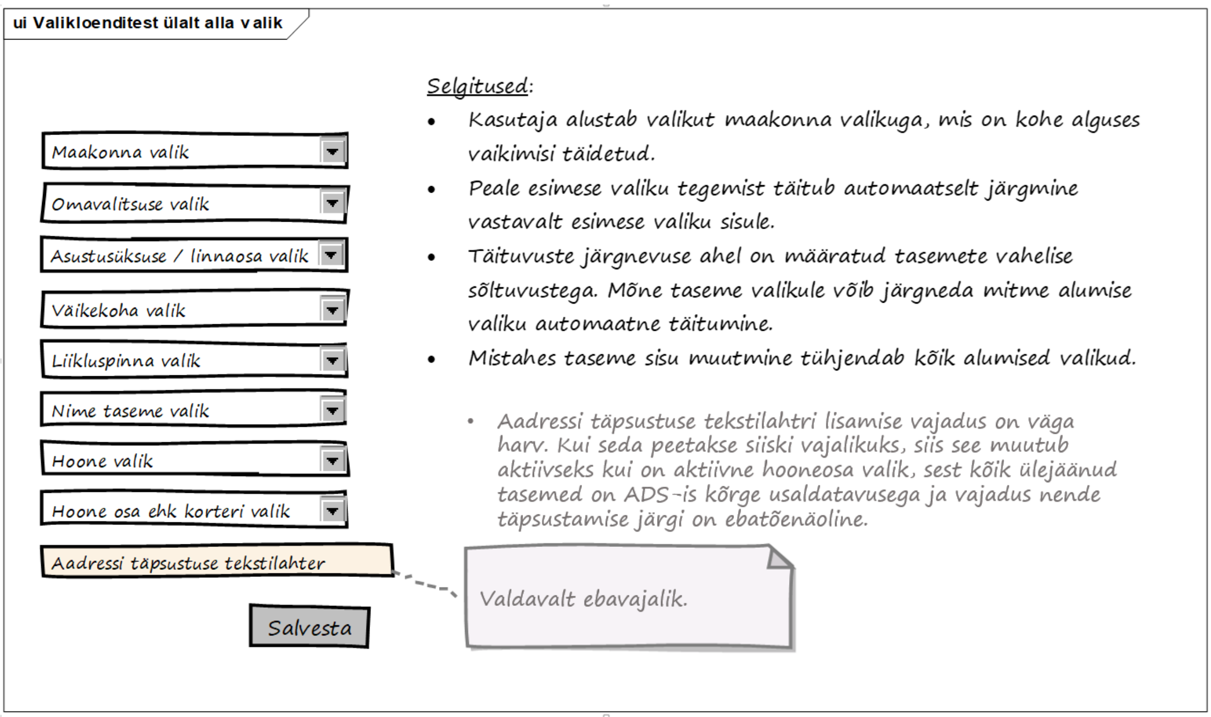
A Kuuse tn 45, Haapsalu linn, Haapsalu linn, Lääne maakond

B Kuuse tn 45, Põltsamaa linn, Põltsamaa vald, Jõgeva maakond

<u>Note: Mostly unnecessary (Valdavalt ebavajalik</u>)
The need to add an address specification text box is very rare. If it is considered necessary, it becomes active when the building part selection is active because all other levels above are of high confidence in ADS and the need to specify them is unlikely (see following examples).

2. Address selection from top-down selection lists is based on address levels. It starts from level 1 and when it is selected, the levels below are filled according to the subordination relationships of the levels.

Top to bottom selection sample:


ui Valikloenditest ülalt alla valik
Maakonna valik
Omavalitsuse valik
Asustusüksuse / linnaosa valik
Väikekoha valik
Liikluspinna valik
Nime taseme valik
Hoone valik
Hoone osa ehk korteri valik
Aadressi täpsustuse tekstilahter
Salvesta

<u>Selgitused</u>:

- Kasutaja alustab valikut maakonna valikuga, mis on kohe alguses vaikimisi täidetud.
- Peale esimese valiku tegemist täitub automaatselt järgmine vastavalt esimese valiku sisule.
- Täituvuste järgnevuse ahel on määratud tasemete vahelise sõltuvustega. Mõne taseme valikule võib järgneda mitme alumise valiku automaatne täitumine.
- Mistahes taseme sisu muutmine tühjendab kõik alumised valikud.

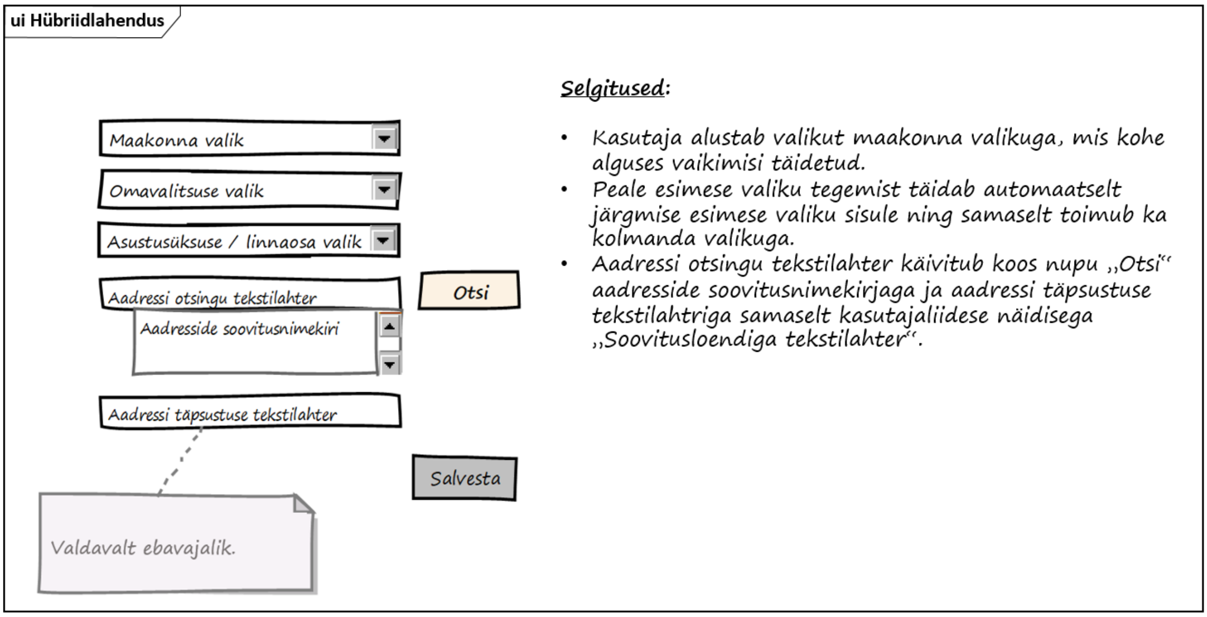
- Aadressi täpsustuse tekstilahtri lisamise vajadus on väga harv. Kui seda peetakse siiski vajalikuks, siis see muutub aktiivseks kui on aktiivne hooneosa valik, sest kõik ülejäänud tasemed on ADS-is kõrge usaldatavusega ja vajadus nende täpsustamise järgi on ebatõenäoline.

Valdavalt ebavajalik.

<u>Explanations</u>:

- The user starts the selection with the county selection, which is filled in by default at the very beginning.
- After making the first selection, the next one is automatically filled in according to the content of the first selection.
- The chain of succession of responsibilities is determined by the dependencies between the levels. Selection of some levels may be followed by automatic completion of several lower selections.
- Changing the contents of any level clears all lower options.

Example of hybrid solution:


ui Hübriidlahendus
Maakonna valik
Omavalitsuse valik
Asustusüksuse / linnaosa valik
Aadressi otsingu tekstilahter
Otsi
Aadresside soovitusnimekiri
Aadressi täpsustuse tekstilahter
Salvesta
Valdavalt ebavajalik.

Selgitused:

- Kasutaja alustab valikut maakonna valikuga, mis kohe alguses vaikimisi täidetud.
- Peale esimese valiku tegemist täidab automaatselt järgmise esimese valiku sisule ning samaselt toimub ka kolmanda valikuga.
- Aadressi otsingu tekstilahter käivitub koos nupu „Otsi" aadresside soovitusnimekirjaga ja aadressi täpsustuse tekstilahtriga samaselt kasutajaliidese näidisega „Soovitusloendiga tekstilahter".

Explanations:

- The user starts the selection with the county selection, which is filled in by default at the very beginning.
- After making the first choice, the next one is automatically filled in according to the content of the first choice, and the same happens with the third choice.
- The text box for address search behaves together with the "Search" button, a list of suggested addresses and the text box for specifying the address in the same way as the "Text box with suggested list" user interface example.

4.5.2 Address and object selection

In order for an address to fulfil its purpose in the database, it must be of the highest possible quality. **Depending on the data set and the context, different conditions can be set on the selection**. The requirement with which address must be selected is set by the consuming system itself. For example, in some data sets/data fields, only the address at the local authority level is sufficient, in others you have to choose an exact proximate address. In addition, it may be checked on the part of the database consuming ADS data that, if there are building parts, the address of the building part can also be selected. Here, however, the specifics of the place of use of the address must be taken into account.

In the case of physical persons and residential premises, the requirement to select a part of the building (apartment) is relevant - for example, the residence of a person in the population register cannot remain at the level of an apartment building, but must still be given with the accuracy of a specific apartment. In the case of legal entities and non-residential premises, such a strict requirement is generally not justified, although it is appropriate to at least warn the user that the building also has addressed building parts.

For example, the following type check can be applied to an address selection:

1) In the case of legal entities and natural persons, it is always mandatory to choose an accurate local address. An accurate local address means that at least the 6th or 7th address level is fulfilled.

2) Once an address up to at least level 6 or 7 has been selected, checks must also be applied in the background:

- Is there an EE or ME object associated with this address (i.e. building);
- As an address, you must not choose an address that corresponds only to an empty (unbuilt) cadastral parcel;

- It is possible that an additional check must be applied to ensure that the selected object has a unique address requirement (so that no one registers at a substation etc. to a non-residential address).

3) In the case of a natural person, an additional check is made that the address can end with level 6 or 7 only if level 8 is missing. If a level 6 or level 7 component is followed by level 8, the natural person must select an address with level 8 accuracy.

Additional information: Levels 6 and 7 can occur together, but it is enough if only one is completed. That is, with the building part, the following combinations can be 6(+8,) 6+7(+8), 5+7(+8). The second option, i.e. the 6+7(+8) combination is very rare, but here it must be taken into account that if the 6th level is already selected and the 7th level also follows, then the user does not have to be forced to go to the 7th level. For example, there are two buildings Maasika (6) and Maasika/1 (6+7) - then only Maasika can be selected. It is enough to additionally check whether the UN building also corresponds to Maasika's address. But if the last level of the selected address, i.e. level 6 or 7, is followed by level 8, then the natural person must go up to level 8.

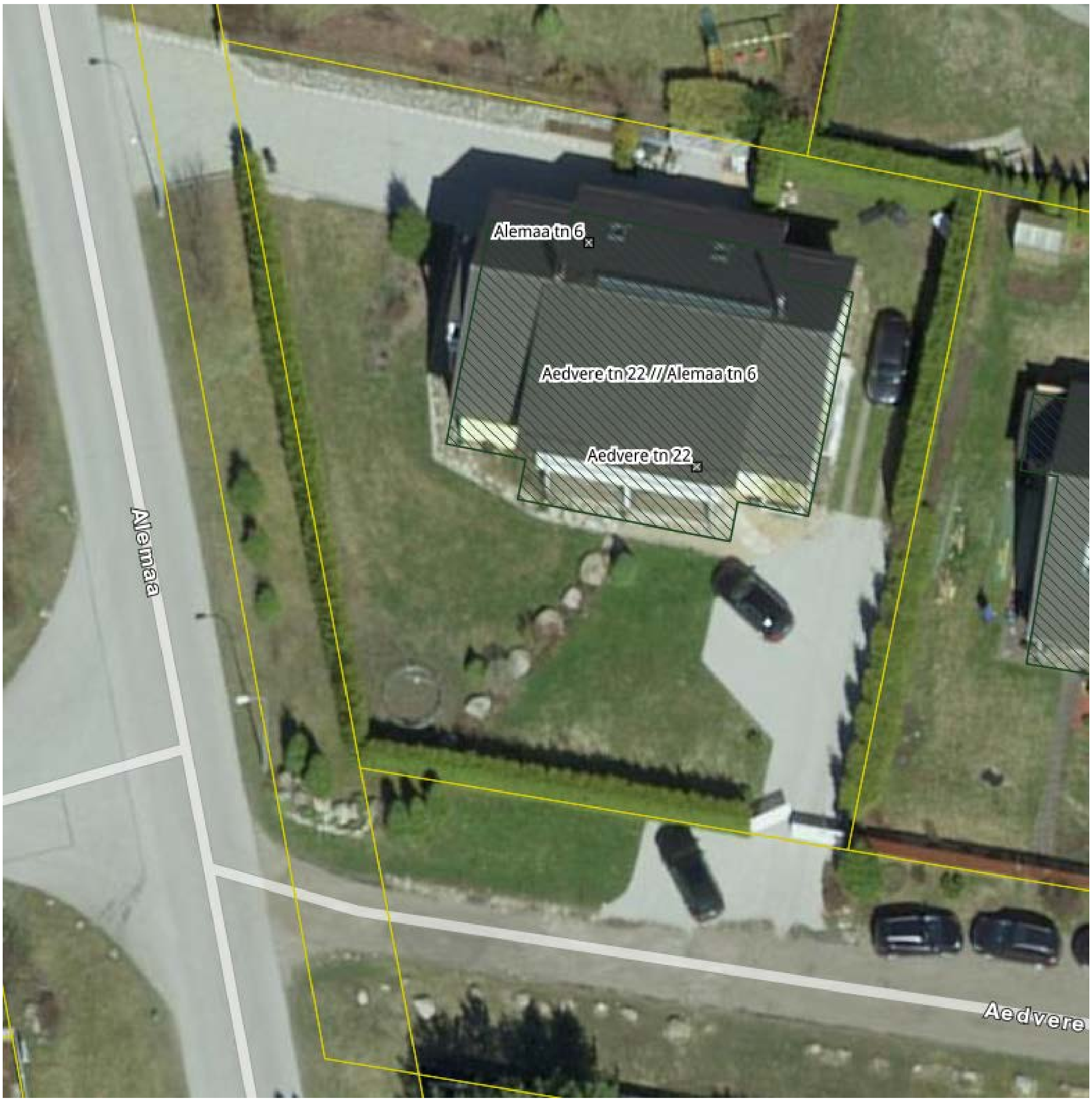
It should be taken into account that, based on the attributes available through the address data system, it is not possible to identify, for example, buildings suitable for living and/or registering a place of residence with 100% accuracy at any given time within the limits of current technological possibilities. Namely, the quality of ADS data depends on other parties. Buildings are added to ADS during mapping (from the Estonian topography database - ETAK) and through the National register of construction works (EHR). The frequency of mapping is 1 year in larger settlements, but there may be an interval of 4 years, the up-to-dateness of the documentation of buildings in the EHR may differ from the real situation according to the documents provided by the owner or the absence of documents, etc. Therefore, ADS also do not have real-time information about when an object appears or disappears in nature.

However, there are certain attributes that can be applied to the address selection as needed. For example, the addresses of all pending address objects can be excluded from the selection - pending objects are generally such objects that have not yet been fully registered in the register of origin (MinuKatasterEHR). At the same time, for example, in the case of building parts, it may be the case that, although the use permit has not yet been formalized for the building parts submitted to ADS with a building permit and these building parts are not yet current in EHR, but these building parts may have already been built and in reality it is necessary for persons to already use the addresses of these pending building parts. Similarly, the valid state does not always mean that the address object actually exists. For example, in the case of buildings, a valid building that has been actualized in EHR and then made valid in ADS, may not yet be built. However, in the case of a building, for example, the presence of an ETAK_ID connection (i.e. the building is mapped by ETAK) can give an additional indication that the building also exists in nature.

In addition, it must be considered that the UN attribute generally refers to a residential and/or public building, but ADS currently does not distinguish between residential buildings and public and commercial buildings. There is also no dataset on which to make this distinction. If, however, you want to make the default selection as narrow as possible, to reduce, for example, the registration of residence in public buildings (which also have a UN attribute), you can also use the information on the purpose of use of the her building, which is also issued by ADS as informative in its services. At the same time, considering the purpose of use of the EHR building also excludes a number of residences not registered in the EHR or those whose EHR data do not correspond to reality.

NB! When interfacing, the complexity resulting from parallel addresses must be taken into account. In the case of parallel addresses, it has to be decided whether the object or the address is primarily important from the point of view of the interfaced register. If dealing primarily with object relations, all parallel addresses of an object are generally displayed. For example, all parallel addresses of the ADS object corresponding to the apartment ownership in the Land register are saved and displayed. But, for example, one specific address is always chosen for people's residences. For example, a dormitory-style apartment has parallel addresses to distinguish between rooms - each household uses one specific address, even though they share the same object.

For example, a two-family house has parallel addresses because each family has a separate entrance from different streets and the households use different addresses:



Technically, a building as an object has several addresses, i.e. ADS_OID version is associated with two addresses (ADR_ID). Table 1:

ADS_OID	ADOB_ID	ADR_ID	Full address text corresponding to ADR_ID
EE03458185	9840241	2122946	Harju maakond, Tallinn, Haabersti linnaosa, Aedvere tn 22
EE03458185	9840241	2122947	Harju maakond, Tallinn, Haabersti linnaosa, Alemaa tn 6

The full address of the building. Table 2:

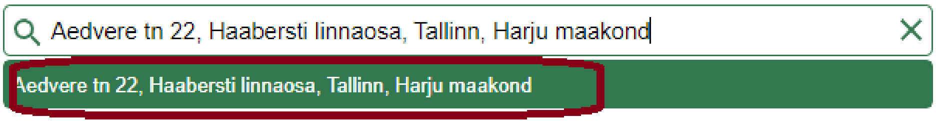
ADS_OID	Full address text of the object
EE03458185	Harju maakond, Tallinn, Haabersti linnaosa, Aedvere tn 22 // Alemaa tn 6

Object's parallel address text does not have its own identifier, only each separate address record has a unique ADR_ID.

For example, if the consumer manages a database where all residential buildings in Tallinn are collected, then it is appropriate to save and display all parallel addresses of the object. Depending on the more precise use case, it remains to be decided whether the addresses are stored as separate records with an identifier (as in table 1) or whether the object identifier is sufficient and the full address of the object is stored only as one text (as in table 2).

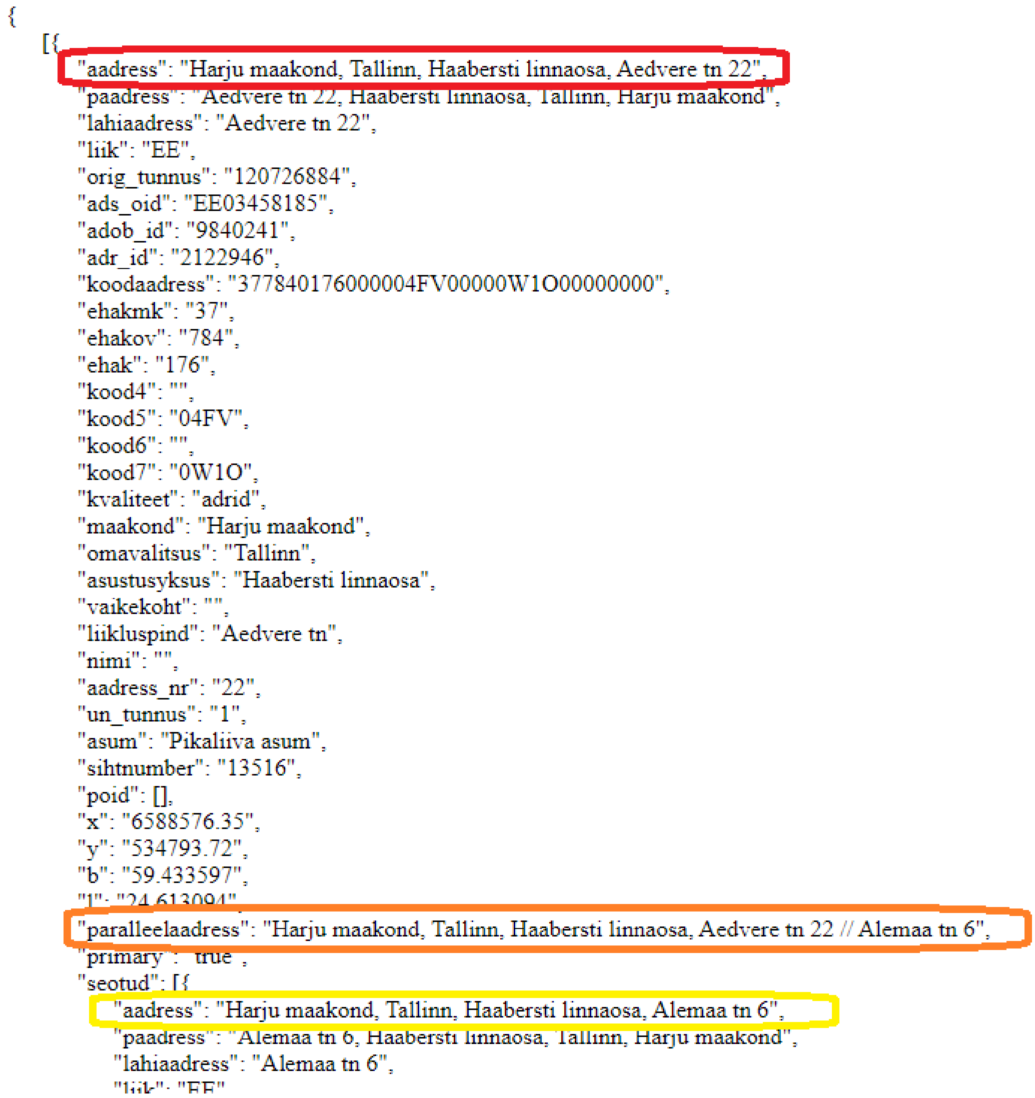
However, if it is, for example, address data collected for the provision of some service (e.g. garbage collection), one specific address where the person lives and where the service is provided must be selected and saved. The given example then has two separate contracts/persons with the same ADS_OID (object), but different ADR_IDs (addresses).

In general, in most cases dealing with personal addresses, it is necessary to select and save one specific address, not all parallel addresses of an object. Therefore, it is necessary to pay attention to the fields in the various ADS services and extracts, which specific address is provided and which of them must be recorded at the point of use. For example, the In-ADS json response contains the address selected by the user in the user interface plus the full address of the object related to this address (with all its parallel addresses) and also other addresses related to the object (i.e. object addresses that were not selected by the user) in separate fields .
For example:



Aedvere tn 22, Haabersti linnaosa, Tallinn, Harju maakond

Aedvere tn 22, Haabersti linnaosa, Tallinn, Harju maakond



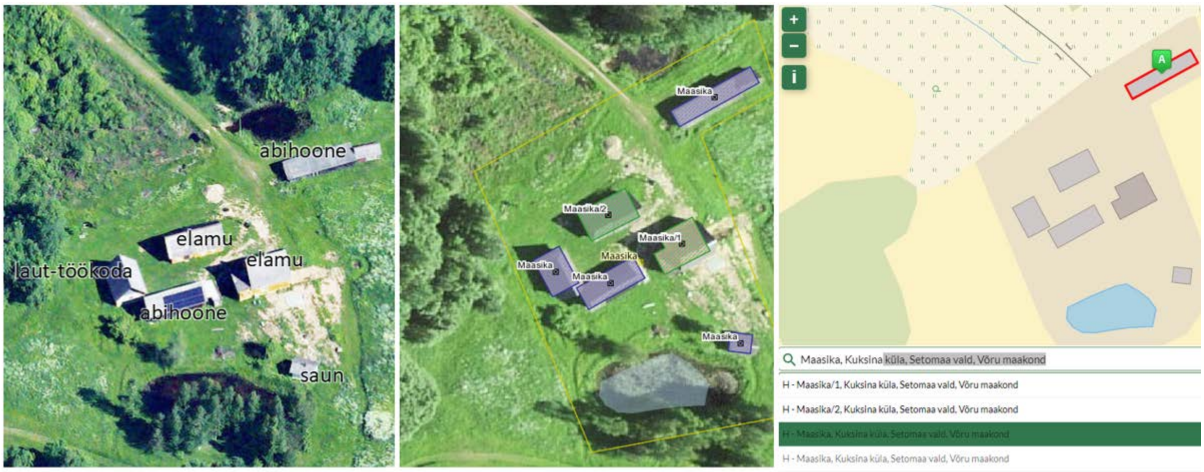
```
{
    [{
      "aadress": "Harju maakond, Tallinn, Haabersti linnaosa, Aedvere tn 22",
      "paadress": "Aedvere tn 22, Haabersti linnaosa, Tallinn, Harju maakond",
      "lahiaadress": "Aedvere tn 22",
      "liik": "EE",
      "orig_tunnus": "120726884",
      "ads_oid": "EE03458185",
      "adob_id": "9840241",
      "adr_id": "2122946",
      "koodaadress": "3778401760000004FV00000W1O000000000",
      "ehakmk": "37",
      "ehakov": "784",
      "ehak": "176",
      "kood4": "",
      "kood5": "04FV",
      "kood6": "",
      "kood7": "0W1O",
      "kvaliteet": "adrid",
      "maakond": "Harju maakond",
      "omavalitsus": "Tallinn",
      "asustusyksus": "Haabersti linnaosa",
      "vaikekoht": "",
      "liikluspind": "Aedvere tn",
      "nimi": "",
      "aadress_nr": "22",
      "un_tunnus": "1",
      "asum": "Pikaliiva asum",
      "sihtnumber": "13516",
      "poid": [],
      "x": "6588576.35",
      "y": "534793.72",
      "b": "59.433597",
      "l": "24.613094",
      "paralleelaadress": "Harju maakond, Tallinn, Haabersti linnaosa, Aedvere tn 22 // Alemaa tn 6",
      "primary": "true",
      "seotud": [{
          "aadress": "Harju maakond, Tallinn, Haabersti linnaosa, Alemaa tn 6",
          "paadress": "Alemaa tn 6, Haabersti linnaosa, Tallinn, Harju maakond",
          "lahiaadress": "Alemaa tn 6",
          "liik": "EE"
```

In order to keep the address data up-to-date, in addition to the address, it is also necessary to select the highest possible quality object connection. As you know, the address object has a many-to-many relationship with the address, and here you have to consider how to correctly select the address object when the user selects the address field. Looking at the address, the existence of several address objects is rather the rule than an exception (for example, the cadastral parcel, the residential building on it and three sheds have the same place-address). If there are two or more address objects in the selection, the most perfect solution should offer the possibility to select the object on the map, so that the user can point to the correct building on the map or visually confirm the selection made. For the informed selection of the address object, the user does not have enough information to make the correct choice by looking only at textual records. The address object is best characterized by its location and shape on the map. The primary object association provided by ADS can be used to create an automatic association if it matches the needs of the interface.
The primary object provided by ADS is the one of all the objects with the same address that could be suitable for creating an object relationship in most cases.
To identify the primary object, ADS uses a number of business rules, which can be viewed above if desired. Whether the primary object offered by ADS also matches the consumer's need depends on the use case.

More background on the primary object

If a land unit has several objects and they have different addresses, ADS does not say which of these objects is the most important. The primary object is the attribute of the address (ADR_ID), i.e. the most important object in terms of a specific address.

Example 1: a cadastral parcel has two residential buildings and three auxiliary buildings. The residential buildings have unique addresses Maasika/1 and Maasika/2, the cadastral parcel and the auxiliary buildings have Maasika.
The user searches for the address Maasika and selects it - the most primary object with this address is the biggest auxiliary building. In fact, the user should have chosen either Maasika/1 or Maasika/2 (each with a different ADR_ID and primary object).



Maasika, Kuksina küla, Setomaa vald, Võru maakond

H - Maasika/1, Kuksina küla, Setomaa vald, Võru maakond
H - Maasika/2, Kuksina küla, Setomaa vald, Võru maakond
H - Maasika, Kuksina küla, Setomaa vald, Võru maakond
H - Maasika, Kuksina küla, Setomaa vald, Võru maakond

For example, In-ADS offers the addresses of UN objects Maasika/1 and Maasika/2, which are residential buildings (UN buildings), as the first two responses to the input "Maasika, Kuksina". However, the user can still choose the Maasika (highlighted by green marker A) object, i.e. the auxiliary building. Generally in this situation it is necessary to direct the user to select a building with a UN identifier (Maasika/1 or Maasika/2).

Example 2: The cadastral parcel has parallel addresses: Metsa tn 5 // Kase (2 different ADR_IDs). The building located on the cadastral parcel has the address Metsa tn 5, there is no parallel address. i.e. the historical farm name Kase is only on the cadastral parcel, the building only has an address according to the thoroughfare name. The user searches for his farm name Kase and selects it. At this address, the only and therefore the most primary object is the cadastral parcel. Generally in this situation it is necessary to direct the user to select the address of the building: Metsa tn 5.

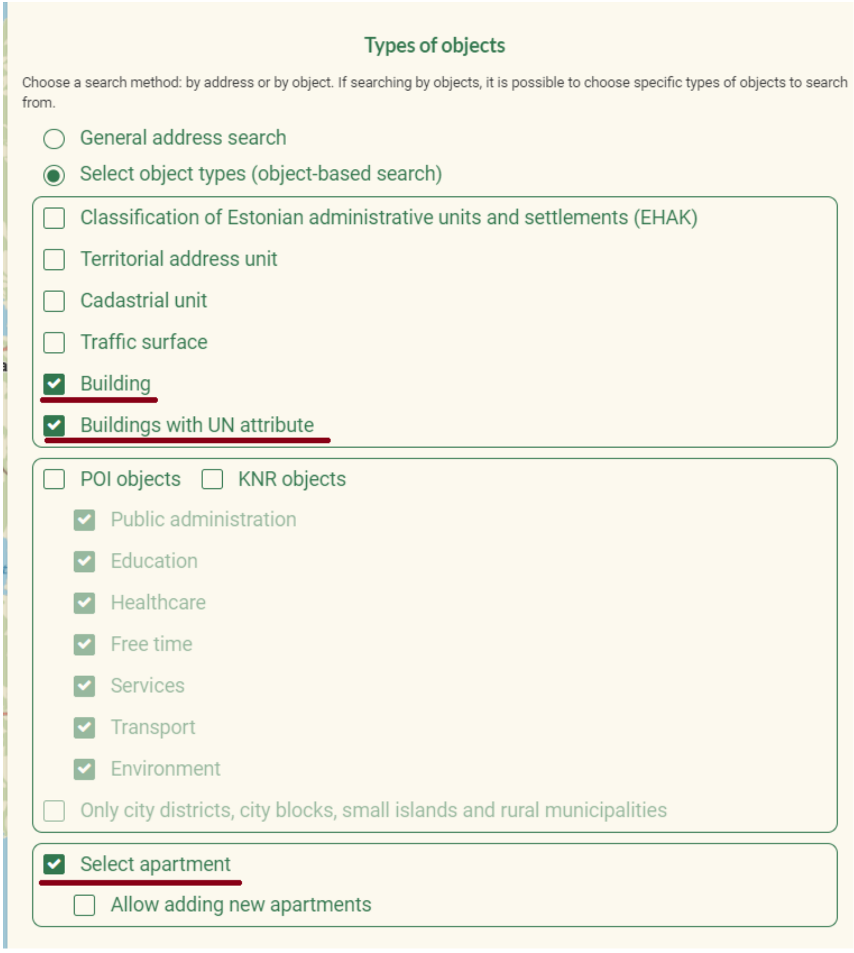
In conclusion:

If the exact object in nature is important for the consuming system, then in order to guide the selection of the correct address and object, it is good to display the selection on the map and, if necessary, based on the specifics of the database, add filter conditions already when performing the address search and/or perform additional automatic checks on the user's selection.

4.5.3 Sample address selection solution using In-ADS

The sample database collects persons' residential addresses. Therefore, the user should select the address primarily from the addresses of residential buildings and apartments.
It must be taken into account that ADS today does not distinguish residential buildings with one specific characteristic, so it is not possible to make the default selection from residential addresses only. However, it is reasonable to narrow down the selection to at least buildings with a UN attribute, which are generally residential and/or public buildings (i.e., commercial buildings are also included) and to exclude cadastral parcels from the selection. Also, in In-ADS, you can't narrow down the selection to only dwellings (all parts of the building are always included in the selection).

The user is offered In-ADS address search with the map application. On the 2nd page of the In-ADS code generator ("search type"), select "Building", "Buildings with UN attribute" and "Select apartment" as the objects from which the search will be performed:



Types of objects

Choose a search method: by address or by object. If searching by objects, it is possible to choose specific types of objects to search from.

○ General address search
● Select object types (object-based search)

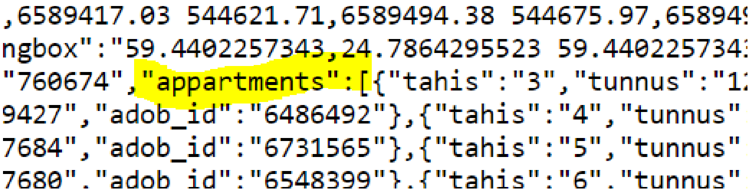
□ Classification of Estonian administrative units and settlements (EHAK)
□ Territorial address unit
□ Cadastrial unit
□ Traffic surface
☑ Building
☑ Buildings with UN attribute

□ POI objects □ KNR objects
☑ Public administration
☑ Education
☑ Healthcare
☑ Free time
☑ Services
☑ Transport
☑ Environment
□ Only city districts, city blocks, small islands and rural municipalities

☑ Select apartment
□ Allow adding new apartments

If there are addressed building parts in the building, user must select a building part. It must be checked whether the user has chosen the address of a building that also has building parts, i.e. the user should actually choose an address with the accuracy of a building part and not be satisfied with the address of the building. The In-ADS user interface displays the selection of building parts as a "vali korter" drop-down menu, but there is no control built into In-ADS that forces the user to select an apartment. **This control must be on the interfacers side.** Therefore, if the user has selected a building type object from the user interface (values

"EE" or "ME"), it must always be checked whether there are building parts in this building. This can be done by the interfacers own ADS database or by using the In-ADS gazetteer service using the ADS_OID of the object selected, providing it to the gazetteer service in the "address" field. For example:
https://inaadress.maaamet.ee/inaadress/gazetteer?address=EE03343692&appartment=2
Check the presence of the "appartments" element in the query response:

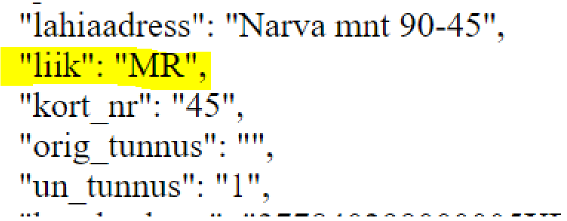


```
,6589417.03 544621.71,6589494.38 544675.97,65894
ngbox":"59.4402257343,24.7864295523 59.440225734
"760674","appartments":[{"tahis":"3","tunnus":"1
9427","adob_id":"6486492"},{"tahis":"4","tunnus"
7684","adob_id":"6731565"},{"tahis":"5","tunnus"
7680","adob_id":"6548399"},{"tahis":"6","tunnus"
```

If this element is missing, building is a suitable object as a residence. If the building parts exist, the user is given, for example, the following message: "The selected building also has addressed building parts. Please enter the address of the apartment!" (in Estonian: „Valitud hoones on ka adresseeritud hooneosi. Palun sisesta korteri aadress!“.)

It should be taken into account that non-residential apartments are also considered as building parts in In-ADS, but the choice of non-residential apartments (MR) is not really preferred in this example. If the user selects an MR object, they must be asked for additional confirmation that the choice is correct, because a non-residential apartment cannot, as a rule, be a residence.

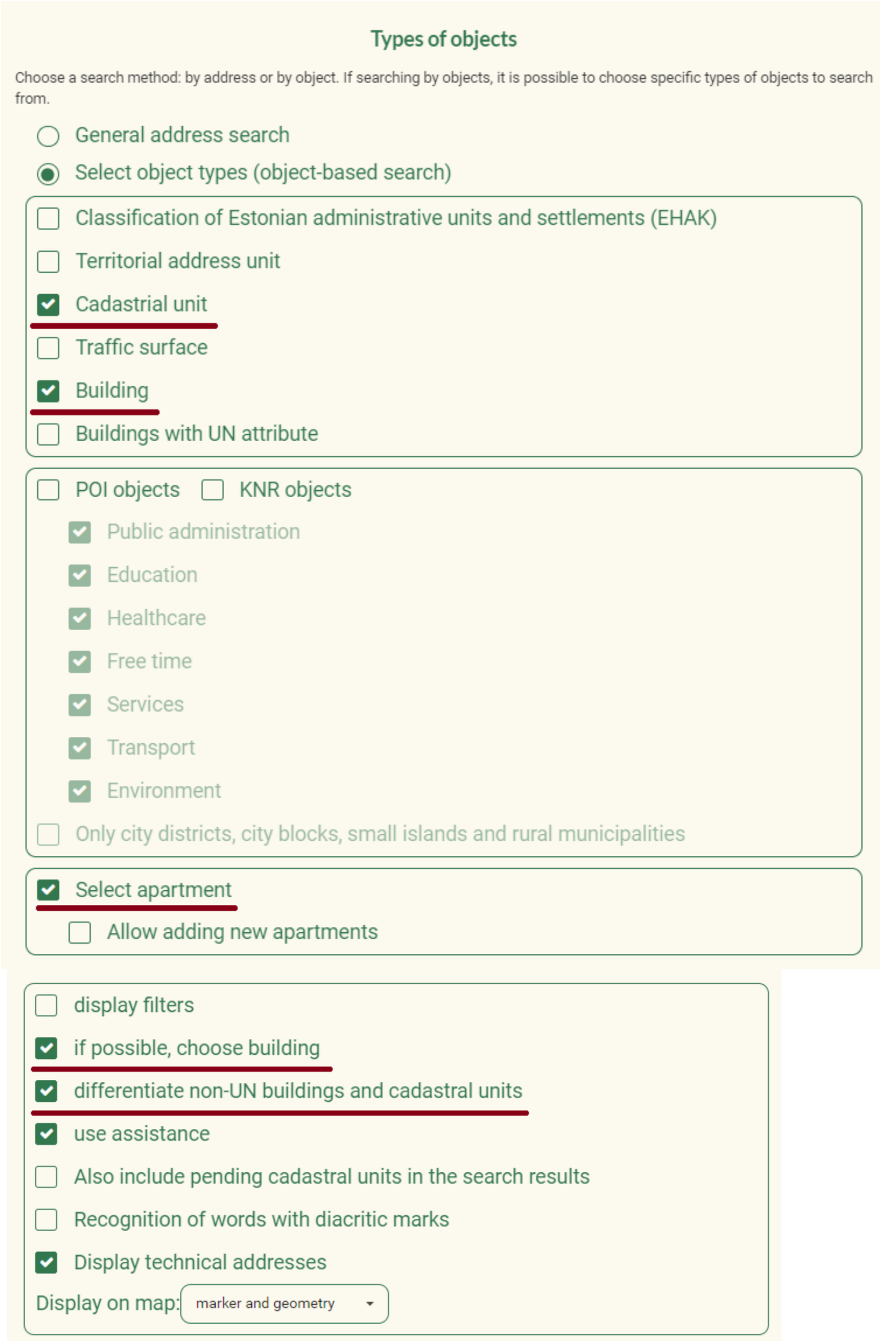
The object type is the "liik" element in the In-ADS UI JSON output:



```
"lahiaadress": "Narva mnt 90-45",
"liik": "MR",
"kort_nr": "45",
"orig_tunnus": "",
"un_tunnus": "1",
```

If the user does not find a suitable address in the default selection, it may be possible to search for an address in a wider selection. For example, the person lives in a trailer, in a container building, etc., which is not mapped and is not registered in EHR (so the building is not in ADS) or the auxiliary building has been converted into a residence, but the relevant documents have not been formalized, so the building does not have a UN attribute. Then it should also be possible to select the address of an empty cadastral parcel or the address of a non-UN building, but in general the user should justify this choice and this wider range of addresses should be separate from the default selection (slightly hidden). The same checks should also be applied in the wider selection, so that if the address of the building is selected, which also includes building parts, the user is directed to select the address of the apartment. Or if the address of the cadastral parcel is selected and it is built up, the user is directed to select the address of the building.

An example of how to implement the extended selection.
When generating the In-ADS code, cadastral parcels and non-UN buildings (auxiliary buildings) must also be included in the results. The option "if possible, choose building" should be marked, this does not allow the user to select the cadastral parcel object if this cadastral unit has buildings, so the user is forced to make a more precise choice. We recommend also leaving the "differentiate non-UN buildings and cadastral units" option, which means that UN buildings are better highlighted to the user:



Types of objects

Choose a search method: by address or by object. If searching by objects, it is possible to choose specific types of objects to search from.

○ General address search

● Select object types (object-based search)

□ Classification of Estonian administrative units and settlements (EHAK)

□ Territorial address unit

☑ Cadastrial unit

□ Traffic surface

☑ Building

□ Buildings with UN attribute

□ POI objects □ KNR objects

☑ Public administration

☑ Education

☑ Healthcare

☑ Free time

☑ Services

☑ Transport

☑ Environment

□ Only city districts, city blocks, small islands and rural municipalities

☑ Select apartment

□ Allow adding new apartments

□ display filters

☑ if possible, choose building

☑ differentiate non-UN buildings and cadastral units

☑ use assistance

□ Also include pending cadastral units in the search results

□ Recognition of words with diacritic marks

☑ Display technical addresses

Display on map: marker and geometry

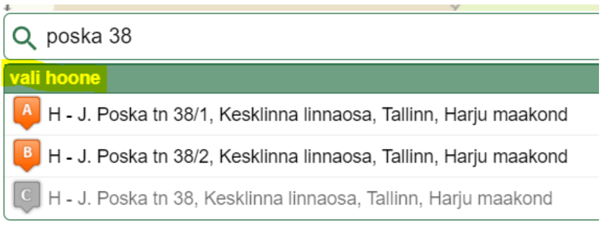
Even with the extended selection, it must be checked that if the building object ("EE" or "ME") is selected, whether there are building parts in this building (see description above).

If the user has selected a cadastral parcel or a non-UN building, additional checks must be made:

- **Does this cadastral parcel have buildings?** If the cadastral parcel has buildings, the user must be directed to select a building.
- **Are there UN buildings in the same cadastral unit** as the selected non-UN building? If the user has selected a non-UN building and this cadastral unit also has UN buildings, the user must be directed to select a UN building. If the user has already moved on to a wider selection, it may be reason why a building with a UN attribute is not selected. Therefore, it is perhaps not reasonable to set the choice of a building with a UN attribute as a strict requirement in the wider selection, but a check in the background should take place and brought to the attention of the user.

- **Does the selected building have building parts?** If the selected building has building parts, the user must be directed to select a building part.

When making a query through the In-ADS user interface, the user is not allowed to select the address of a cadastral parcel if this cadastral parcel has buildings (if the corresponding condition was turned on during code generation). The system opens a selection of buildings to the user and asks to select a building:



poska 38

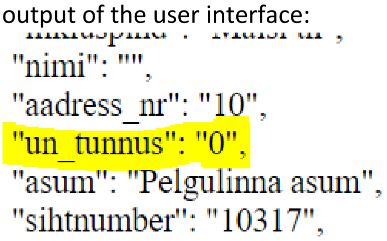
vali hoone

A H - J. Poska tn 38/1, Kesklinna linnaosa, Tallinn, Harju maakond

B H - J. Poska tn 38/2, Kesklinna linnaosa, Tallinn, Harju maakond

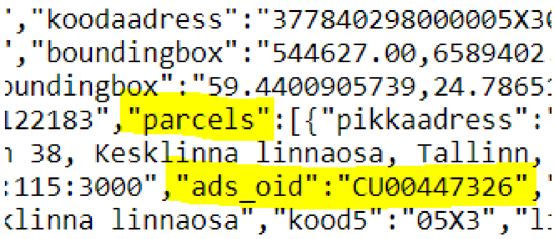
C H - J. Poska tn 38, Kesklinna linnaosa, Tallinn, Harju maakond

Whether the selected building is a UN building or not can be checked based on the "un_tunnus" element in the JSON output of the user interface:



```
"nimi": "",
"aadress_nr": "10",
"un_tunnus": "0",
"asum": "Pelgulinna asum",
"sihtnumber": "10317",
```

A value of 1 means that it is a UN building. A value of 0 means that it is a non-UN building.
If the user has selected a building without a UN attribute, it as to checked in the background whether the cadastral parcel also has a building with a UN identifier, using the In-ADS gazetteer service. For this, the cadastral parcel related to the selected non-UN building is first found (based on the ADS-OID of the non-UN building):
https://inaadress.maaamet.ee/inaadress/gazetteer?address=ME00642457&seos=1
Check the element „parcels":



```
',"koodaadress":"377840298000005X3(
',"boundingbox":"544627.00,6589402
oundingbox":"59.4400905739,24.7865:
122183","parcels":[{"pikkaadress":'
ı 38, Kesklinna linnaosa, Tallinn,
:115:3000","ads_oid":"CU00447326",'
clinna linnaosa","kood5":"05X3","l:
```

Then it is checked that the given cadastral parcel does not have UN buildings with a query:
https://inaadress.maaamet.ee/inaadress/gazetteer?address=CU00447326&seos=1

Check objects in the "buildings" section of the output. In this specific example, both UN buildings and building parts are returned, so the user should select the address of the building part. Or if the result would be UN buildings in the "buildings" section, the user should be notified, for example: "You have selected a non-residential building address. The cadastral unit also has residential and/or public buildings - please check and confirm the correctness of the address or correct the address." (In Estonian: „Valisite mitteelukondliku hoone aadressi. Katastriüksusel on ka elu- ja/või ühiskondlikke hooneid – palun kontrollige ja kinnitage aadressi õigsust või parandage aadressi.").

If the user still selects an **MR object, a non-UN building or an empty cadastral parcel** (they cannot select a built-up cadastral parcel, because then the user is directed to choose from among the buildings), then the user can also be asked the reason why he chose such an address object (because as a rule it cannot be a place of residence) - this information can be forwarded to the Land Board, who can, if necessary, improve the address data in cooperation with the local government.

Some of the possible standard reasons that can be provided to the user as a default value (in Estonian):

- *The place of residence is in a trailer, container building (etc.) on the selected cadastral parcel. (In Estonian: Elukoht on katastriüksusel soojakus, konteinerhoones vmt.)*
- *The residence is temporarily in an auxiliary building. (In Estonian: Elukoht on ajutiselt kõrvalhoones.)*
- *The new building is not registered. (In Estonian: Uus hoone puudub valikust.)*
- *The building data in the register do not correspond to the real situation. (In Estonian: Hoone andmed registris ei vasta reaalsele olukorrale.)*
- *The apartment data in the register do not correspond to the real situation. (In Estonian: Eluruumi andmed registris ei vasta reaalsele olukorrale.)*
- *Other reason, please specify:… (In Estonian: Muu põhjus, palun täpsusta: …)*

4.6 Data update processes

Changes to addresses and address objects occur every day. When analysing the business logic of interfacing with ADS, it is decided whether the addresses must be up-to-date and whether any kind of time delay, manual work, etc. can be tolerated. It depends on how the process of updating address data is structured.

In order to keep addresses up-to-date, both an address and an object association are generally required. The quality of the initial object relation also determines the quality of further changes. For more information on choosing an address and object, see chapter 4.5. The business logic of updating the addresses of each database depends on the specifics of the specific database/data field and the level of interface, but the example given in chapter 4.6.1 can be used as an example.

As long as an object is valid, updating data relies on changes to that object. If the object itself becomes invalid, there are generally 2 possibilities:

1. The address object has one successor - there is no significant complexity, the old object must be replaced with a new one in the corresponding places of use.
2. The address object has several successors or there is no successor - it is necessary to make relevant logic in the consuming system or, in the case of user intervention, decision points where a new address object must be found manually or the loss of validity must be taken into account when this matter ends (for example, no more letters are sent to this address).

Automatic finding of offspring may not give the best result in all cases. The error can never be drastic, because the successor address object is never in a completely different location. A typical multiple succession occurs when address objects are divided into smaller parts (in the case of compound buildings, so-called splitting into several separate buildings) or in other spatial redistribution (in the case of cadastral parcels, for example, land management operations).

The best accuracy of finding a successor is only needed in systems where the accuracy of the location pointing to the address is of utmost importance. Manually finding successors to address objects means creating a corresponding user interface on the side of the consuming system, which contains at least a list form of unlinked address objects (objects for which the system could not find a 1:1 successor in ADS) and a detail form where it is possible to select one for a specific address object as a successor by ADS provided address object.

In order to select an address or an object, it is inevitably necessary to show the existing and successor address objects on the background of the map. In-ADS is made with a map window where you can make choices. The selection of a suitable address object should take place in the context of the operational logic of the information system. For example, if the address object is related to a person and this address object lost its validity and there are several successors, it may be necessary to specifically interrogate the person or to compare other data in the consuming system to accurately select one successor.

ADS X-tee services can be used to update address data. See more in the document ADS X-tee services.

Synchronization comparisons with ADS data must be performed from time to time. For this purpose, public extracts from ADS can be used (see chapter 4.2). For some databases, extracts are also suitable for updating data.

In-ADS is suitable for searching and selecting addresses/objects, but not for keeping data up-to-date. If the volumes are small and you want to update the data for some reason using In-ADS, the following options are examples:

1) Inherit the currently valid address from In-ADS based on the subject-related object.
2) If such an object is no longer valid, inherit a new connection object based on the address of the object related to the previous subject - but it may not be the actual successor of the cancelled object or the right object to follow the life map of;
3) inherit a new object from the location on the basis of other coordinates stored in connection with the previous object or subject, if necessary apply additional rules (e.g. choose the one from the found

objects and the addresses of the selected object that has the same address as before, etc.) - if the same type of object is not there (object is no longer in the wild) or it is part of a building, then it does not help;

4) ask the subject to update their address data themselves.

4.6.1 An example of the process for automatically updating addresses

This process could be used in the case where the connection of a subject (e.g. a person) to one specific address object and address is recorded in the data set (in the case of parallel addresses of the object, a specific one is selected from them). In this case ADS_OID, ADOB_ID, ADR_ID, code address, address text, local address, address levels and postal code are saved.

1. Changes to objects are inherited with the **ADSobjmuudatusedV7** service:
   - logId – the largest logId value of the object in the interface database;
   - andmevektor – 001 (data vector: address records);

   The data inherited in point 1 is stored in the interface database.

2. Address changes are inherited with the **ADSaadrmuudatusedV7** service:
   - logId – the largest address logId value in the interface base;
   - nSyndmused – true.

   The data inherited in point 3 is stored in the interface database (to another table).

3. Changes to objects in the interface base are processed (**syndmus='U'**):

   a) The object is associated with only 1 address - the data is updated (adob_id, ads_oid, adr_id, koodaadress, täisaadress, sihtnumber – all data will be updated if there are differences).

   b) An object is associated with multiple addresses:

   - If one address has the same adr_id as the subject is currently using, the data is updated (data of the object, because the address does not change);
   - If there is no same adr_id, but only exactly one address has the same code address without version number (levels 1-8) as the address associated with the subject (data is updated).
   - If the object does not have any address with the same ADR_ID or code address associated with the person, it can be checked whether one of the proximate addresses of the object's new addresses matches the proximate address part of the person's address. If it matches, select it as the person's new address and update the data. This step can help automatically process cases where there is a change in addresses at the EHAK level (levels 1-3).
   - If it is a part of a building that has parallel addresses, the one that has the same 8th level name as the address previously associated with the person is selected from the new parallel addresses for the person (so that, for example, if the address of the part of the building has the same symbols and the address at the building level has changed, then automatically to choose a new address).

   c) If the address of the subject was not updated in point 3, the main user must update the address manually or direct it to the person to specify the address. It is the interfacer's choice how the person's address data is handled until new address is given, e.g. whether the person's address data is made empty or the connection with outdated data is maintained.

4. Object cancellations in the interface base are processed (**syndmus='D'**):

   a) For each record, the information of the descendants is inherited with the **ADSobjjargalsed** service. Descendants of a canceled object are inherited until information about the descendants is received, but at most up to 14 days after adding the 'D' event to the interface base, because later additions of sequences are rare and subjects cannot be left with a canceled object indefinitely.

- Adobid – adobes of the cancelled object being processed;
- adsOid – does not transmit;
- andmeVektor – 001 // data vector: address records;

The data of the inherited descendants is stored in the interface database.

b) Descendants are processed:

- There were several descendants - the record is marked as processed and the subject's data is not updated.
- There is one descendant:
  (1) Only 1 address is associated with the descendant: the subject's data is updated.
  (2) Several addresses are associated with the descendant; they are selected in order of priority according to the following logic:

    (a) An address with the same addr_id as the subject's current address is selected;
    (b) An address is selected that has the same code address without the version number (levels 1-8) that is currently valid for the subject (only 1 address can be with this code address to automatically identify).

c) If the data was not updated in point 4 (there was no descendant or one descendant could not be identified), the person must intervene (the subject is asked to update its address data).

5. If the objects' data have been updated, i.e. the persons' address data are up-to-date, the postal code data for all addresses is updated based on the N-event logs received from the **ADSaadrmuudatusedV7** service.

4.6.2 The logic of updating the main data tables of the ADS copy

The business logic for updating basic data tables is described below. The exact structure of the tables and additional tables depend on the place of use, for example, the interface can store related objects, information of descendants, additional EHR data, etc.

WSDLs for X-tee services are available: https://x-tee.ee/catalogue/EE/GOV/70003098/ads

The first 100 responses are issued when all ADS X-path services are executed, to receive the following responses, the request must be repeated by supplementing the input, e.g. with the last received log number, i.e. the log number acts as a bookmark. In different services, the role of a bookmark may have a different information field.

To update the data, it is recommended to contact the X-tee services in the early hours of the day, but not immediately after the date change, so as not to end up inheriting the data before the ADS system synchronization processes are finished. Since there are many data processing processes and they are dependent on each other in time sequence, it is difficult to recommend a specific time of day. Generally, however, by three o'clock in the morning, the processes could be finished and the service servers are using the most recent dataset.

It is not recommended to make mass requests via X-tee. ADS extracts are created to boot larger amounts of data (see chapter 4.3.2).

Updating address components

1. The process starts working at the time specified in the start schedule.
2. The process calls the ADS information system's X-tee service **ADSkompklassif**.
   Service input:
   - LogId = last processed log entry id.

- The service returns up to the next 100 log entries in log_id order. If no log entries are returned, then all ADS changes have been processed.
  End of activity.

3. Process the returned log entries one at a time.
4. If in the log entry syndmus = -**I**-, -**U**- or -**R**-, then the component is added (Insert), changed (Update) or restored (Restore). To overcome errors and inconsistencies that may occur during synchronization, for all these events, the component should be added to the **ADS_KOMPONENT** table, if it is not there, it should be updated if it is present. For example, if there is a -U- event in the log for a component that does not exist, then there is an unhandled -I- event from an earlier phase, or the initialization was incorrect. This fact should not become an obstacle in the processing of changes. Check the existence of the entry based on the level+code value. Make the following assignments:
   - TASE <= Tase
   - KOOD <= Kood
   - NIMETUS <= Nimetus
   - NIMETUS_LIIGIGA <= NimetusLiigiga
   - YLEMKOMP_TASE <= YlemkomponendiTase
   - YLEMKOMP_KOOD <= YlemkomponendiKood
   - KEHTIV <= LogStamp, if syndmus = -I- and -U- the former value remains
   - KEHTETU <= remain blank or make blank
5. If in the log entry syndmus = -**D**- the corresponding component is cancelled (Delete). If such a component does not exist in the ADS_KOMPONENT table, no error should occur. If it exists, find the component based on the level+code value. To attribute:
   - KEHTETU <= LogStamp.
6. If the log entry has syndmus = -**S**- it is a case of the complete disappearance of the component, i.e. this component has no connection with a valid address object. If such a component does not exist in the ADS_KOMPONENT table, no error should occur. If it exists, find the component based on the level+code value. To attribute:
   - SURNUD <= LogStamp.
7. When all the log entries have been read, remember the ID of the last processed log entry and repeat p. 2 described activities.

Updating addresses and address objects

Updating addresses and address objects is a topic at interface sublevel B. In the case of **light interfacing, i.e. basic level L1**, only the addresses and/or address objects stored are updated. In the case of **full interfacing, i.e. basic level L2**, for all addresses and address objects stored as ADS clones in the base of the place of consumption. The main difference is only in the number of records that need to be updated. The complexity of the interface design and the number of records requested from the ADS information system X-tee queries are the same for both L basic levels.

In general, **three main tables** need to be updated: **the object table, the address table, and the object-address association table**. A separate X-tee service can be used to update each table, and depending on the specific X-tee service, the event can have a different meaning. For example:

- **Events of address changes can be monitored from the service ADSaadrmuudatusedV7**. The events are based on the code address, e.g. I mean adding the address to the ADS system, etc.
- **Changes to the object, including object address data, can be monitored through the ADSobjmuudatusedV7** service. Events are based on ADS_OID: I for ADS_OID addition, D is ADS_OID cancellation, etc.
- **Changes in object and address relationships** can also be monitored with the separate X-tee service **ADSobjaadrmuudatusedV5**. That is, the events are based on the relationship between the object and the address: The I event means the addition of a connection between the address and the object, and the D event means that the object and the address are no longer connected. In the case of the K event, a change in the address point of the object has occurred.

There are additional events in the address change service **ADSaadrmuudatusedV7**:

- Postal code change **(N) - Sihtnumbri muudatus** - an event that occurs when a postal code is added, changed or removed from an address. Not related to address change event. The zip code changes at the address independently of other events.
- Informal Area Name Change **(A) - Mitteametliku piirkonna nime muudatus** - Event generated when an area name is added, changed, or removed from an address. Not related to address change event. The region name changes at the address independently of other events.
- Top priority object change **(O) - Prioriteetseima objekti muudatus -**refer to the address for the event of adding, changing or removing the highest priority object.
- Technical identifier change **(T) - Tehnilise tunnuse muudatus** – an event that occurs when the "technical" identifier is assigned/removed to an address.
- Change of point coordinates **(P) - Punkti koordinaatide muutumine** - the event of a change in the coordinates of the representative point of the address. Coordinates can change when an address is assigned to or removed from a new object, or when the shape or importance of an object associated with an address changes.
- Loss of connections **(S) - Seoste kadumine** - this event is only possible for invalid addresses. The invalid address may be attached to another valid object. This event occurs when the last link between an address and a valid object is removed, in other words, when that address no longer remains the address of any valid object.

Object changes service **ADSobjmuudatusedV7** are additional events:

- Change of related objects **(H) - Seotud objektide muudatus** - building change event.
- Change of associated interfaced objects **(L) - Seotud liidesobjektide muudatus** – the interfaced object associated with the address object has either been added, removed, or the data of the interface object has changed.
- EHR additional data change **(E) - EHR lisaandmete muudatus** – additional data from the EHR register of the building or part of the building was created or changed.

If these address or object attributes need to be handled on the interfaced register, then these events must also be read from the service and additional data fields (for example related objects) must be requested in the input. From the return of the service, add the corresponding data fields to the table of addresses or objects, accordingly. If the interface also needs to know the relationships between objects, the relationships between cadastral parcels and buildings are kept in a separate table (multi-to-multiple hoone_ads_oid and ky_ads_oid relationships). Cadastral parcels and building parts are connected to each other through building connections (in the objects table, the building_oid field is filled for the building part).

It must be taken into account that in certain cases it is not enough to only monitor the so-called additional event, i.e. for example T event is not given in addition to address I event, etc., i.e. only reading T event is not enough to update the technical feature. That is, in general, in addition to the so-called normal I, U, R, D, S events, an additional event must also be read if the corresponding attribute is used in the interfaced system.

A separate service **ADSpoimuudatused.v1** has been created for changes related to points of interest. Points of interest are stored in a separate data table in the interfaced system and linked to the address object table via ADS_OID.

See details in the description of X-tee services.

***Updating of addresses**, i.e. table ADS_AADRESS*

1. The process starts working at the time specified in the start schedule.
2. The process calls the ADS information system's X-tee service **ADSaadrmuudatusedV7**.
   Service input:
   - LogId = last processed log entry id.

- AadressKomp = *true.*
- Additionally, indicate in the input which log events are of interest (see X-tee service description for more details).
- The service returns up to the next 100 log entries in log_id order. If no log entries are returned, then all ADS changes have been processed.
  End of activity.

3. Process the returned log entries one at a time.
4. If in the log entry syndmus = -**I**- or syndmus = -**U**- or -**R**-, then it is the addition, change or restoration of a valid address. For event -I-, a new record must be created, and for event -R-, an existing record must be found and its data changed. In the case of a -U- event, you must first check whether the vanaAdrId (old ADR_ID) is also filled in the log entry and compare whether the vanaAdrId and the adrId are the same. If they are the same, look for the existing ADR_ID record and update its data (quite a rare case, but not excluded). If they are different, create a new ADR_ID record (see also point 5). Regardless of whether a new record is being created or an existing one is being modified, the correspondences to the **ADS_AADRESS** table are as follows:
    - ADR_ID <= adrId
    - KOODAADRESS <= Koodaadress
    - TAISAADRESS <= Taisaadress
    - LAHIAADRESS <= Lahiaadress
    - OLEK <= -**K**-
    - KEHTIV <= kehtiv
    - KEHTETU <= null // indefinite validity
    - TASE1_KOOD <= ADSTase1.Kood
    - TASE1_NIMI <= ADSTase1.Nimetus
    - TASE1_NIMI_PIKK <= ADSTase1.Nimetus_liigiga
    - TASE2_KOOD <= ADSTase2.Kood
    - TASE2_NIMI <= ADSTase2.Nimetus
    - TASE2_NIMI_PIKK <= ADSTase2.Nimetus_liigiga
    - TASE3_KOOD <= ADSTase3.Kood
    - TASE3_NIMI <= ADSTase3.Nimetus
    - TASE3_NIMI_PIKK <= ADSTase3.Nimetus_liigiga
    - TASE4_KOOD <= ADSTase4.Kood
    - TASE4_NIMI <= ADSTase4.Nimetus
    - TASE4_NIMI_PIKK <= ADSTase4.Nimetus_liigiga
    - TASE5_KOOD <= ADSTase5.Kood
    - TASE5_NIMI <= ADSTase5.Nimetus
    - TASE5_NIMI_PIKK <= ADSTase5.Nimetus_liigiga
    - TASE6_KOOD <= ADSTase6.Kood
    - TASE6_NIMI <= ADSTase6.Nimetus
    - TASE6_NIMI_PIKK <= ADSTase6.Nimetus_liigiga
    - TASE7_KOOD <= ADSTase7.Kood
    - TASE7_NIMI <= ADSTase7.Nimetus
    - TASE7_NIMI_PIKK <= ADSTase7.Nimetus_liigiga
    - TASE8_KOOD <= ADSTase8.Kood
    - TASE8_NIMI <= ADSTase8.Nimetus
    - TASE8_NIMI_PIKK <= ADSTase8.Nimetus_liigiga
    - SIHTNUMBER <= SIHTNUMBER
    - ASUMI_NIMI <= maPiirkond
    - ASUMI_ALIAS <= maPiirkondAlias
    - PRIMAAR_OID <= primaarseimObjekt
    - TEHNILINE <= tehniline

- STAMP_UPD <= LogStamp

5. In addition to the above, if in the log entry syndmus = -**U-** and **vanaAdrId** (old ADR_ID) is filled in, first compare whether the vanaAdrId and the adrId are the same in the log record. If they are the same, it is a version correction (the data was updated in point 4). If they are different, search for a record based on this vanaAdrId and assign (there is no error if there is no record):
   - OLEK <= -**V**- // outdated (vana)
   - KEHTETU <= LogStamp
   - In addition, **replace vanAdrId with adrId values in places of consumption.**
     NB! If there are many places of possible consumption in the system, it is advisable to solve the exchange of keys as a separate process. In this case, a file must also be added to the **ADS_AADRESS** table to store the vana_adr_id.
6. If in the log entry syndmus = -**D-**, then the address is cancelled. Then look for a record based on adrId and assign (there is no error if there is no record):
   - OLEK <= -**T**- // cancelled (tühistatud)
   - KEHTETU <= kehtetu
7. If in the log entry syndmus = **-N-** it is a change of postal code. The corresponding field in the table must be changed, other data remain the same.
8. If in the log entry syndmus = **-A-** it is a change in the MA area. The corresponding field in the table must be changed, other data remain the same.
9. If in the log entry syndmus = **-O-** it is a change of the primary object. The corresponding field in the table must be changed, other data remain the same.
10. If in the log entry syndmus = **-P-**, it is a change of the representative office of the address. The coordinates of the representative point in the table should be changed, other data remain the same.
11. If in the log entry syndmus = **-T-**, it is a change in the technical characteristic of the address. The corresponding field in the table must be changed, other data remain the same.
12. If in the log entry syndmus = -**S**-, then the address has completely disappeared from circulation, i.e. this address has no connection with a valid address object. For example, an invalid address may initially remain attached to a valid object. When the last link between an address and a valid object is removed, the date and time of the link loss can be assigned to the address based on the LogStamp of the -S- event log.
13. When all the log entries have been read, remember the ID of the last processed log entry and repeat p. 2 described activities.

***Updating of objects**, i.e. table ADS_OBJEKT*

1. The process starts working at the time specified in the start schedule.
2. The process calls the ADS information system's X-tee service **ADSobjmuudatusedV7**.
   Service input:
   - LogId = last processed log entry id.
   - Andmevektor = 111 // data vector: alphanumeric data, spatial data and address records.
   - AadressKomp = *true.*
   - Additionally, indicate in the input which log events are of interest (see X-tee service description for more details).
   - The service returns up to the next 100 log entries in log_id order. If no log entries are returned, then all ADS changes have been processed.
     End of activity.
3. Process the returned log entries one at a time.
4. If in the log entry syndmus = -**I**-, -**U**- or -**R**-, it is the addition, change or restoration of a valid object. In case of event -I- and -R-, a new record must be created. In the case of a -U- event, you must first check whether the log entry also contains vanaAdobId and compare whether vanaAdobId and adobId are the same. If they are the same, find the existing ADOB_ID record and update its data. If they are different, create a new ADOB_ID record (see also point 5). Regardless

of whether a new record is created or an existing one is changed, the correspondences with the **ADS_OBJEKT** table are as follows:

- ADOB_ID <= adobId
- VERS <= find the number of entries with the same ads_oid in the standalone ADS and add 1
- ADOB_LIIK <= ObjektiLiik
- ADS_OID <= adsOid
- TAISAADRESS <= Taisaadress
- LAHIAADRESS <= Lahiaadress
- VIITEPUNKT_X <= TsentroidX
- VIITEPUNKT_Y <= TsentroidY
- OLEK <= olek
- KEHTIV <= kehtiv
- KEHTETU <= null
- HOONE_OID <= HooneOID
- ORIG_TUNNUS <= origTunnus
- UNIK <= unikaalne
- STAMP_UPD <= LogStamp

a. Add to the **ADS_OBJ_AADR** table relations with addresses, which may be several, as follows

- ADOB_ID <= just added adobId
- ADR_ID <= adrId in the address element
- VIITEPUNKT_X <= aadressipunktX
- VIITEPUNKT_Y <= aadressipunktY

It should be taken into account that the separate change of the address point of the object cannot be learned from the object change service ADSobjmuudatusedV7 (these points can change even without the change of the object). So to update the address point of the object, you must also read the K-events from the ADSobjaaadmuudatusedV5 service and update these address points of the object accordingly. Also, the connection table of objects and addresses (ADS_OBJ_AADR) can entirely be updated from the ADSobjaadrmuudatusedV5 service.

5. In addition to the above, if in the log entry syndmus = -**U-** and **canaAdobId** is filled in and it is different from adobId or syndmus = **-R-**, then search for a record based on vanaAdobId and assign (there is no error if the record is missing):
   - OLEK <= -V-
   - KEHTETU <= LogStamp
6. If in the log entry syndmus = -**D**-, it is a cancellation of the object. Then look for the record based on adobId and assign (if there is no record, there is no error):
   - OLEK <= -T-
   - KEHTIV <= does not change, remains as it was
   - KEHTETU <= kehtetu
7. When all the log entries have been read, remember the ID of the last processed log entry and repeat p. 2 described activities.