

**PER Auskunft**

# **Data model view**

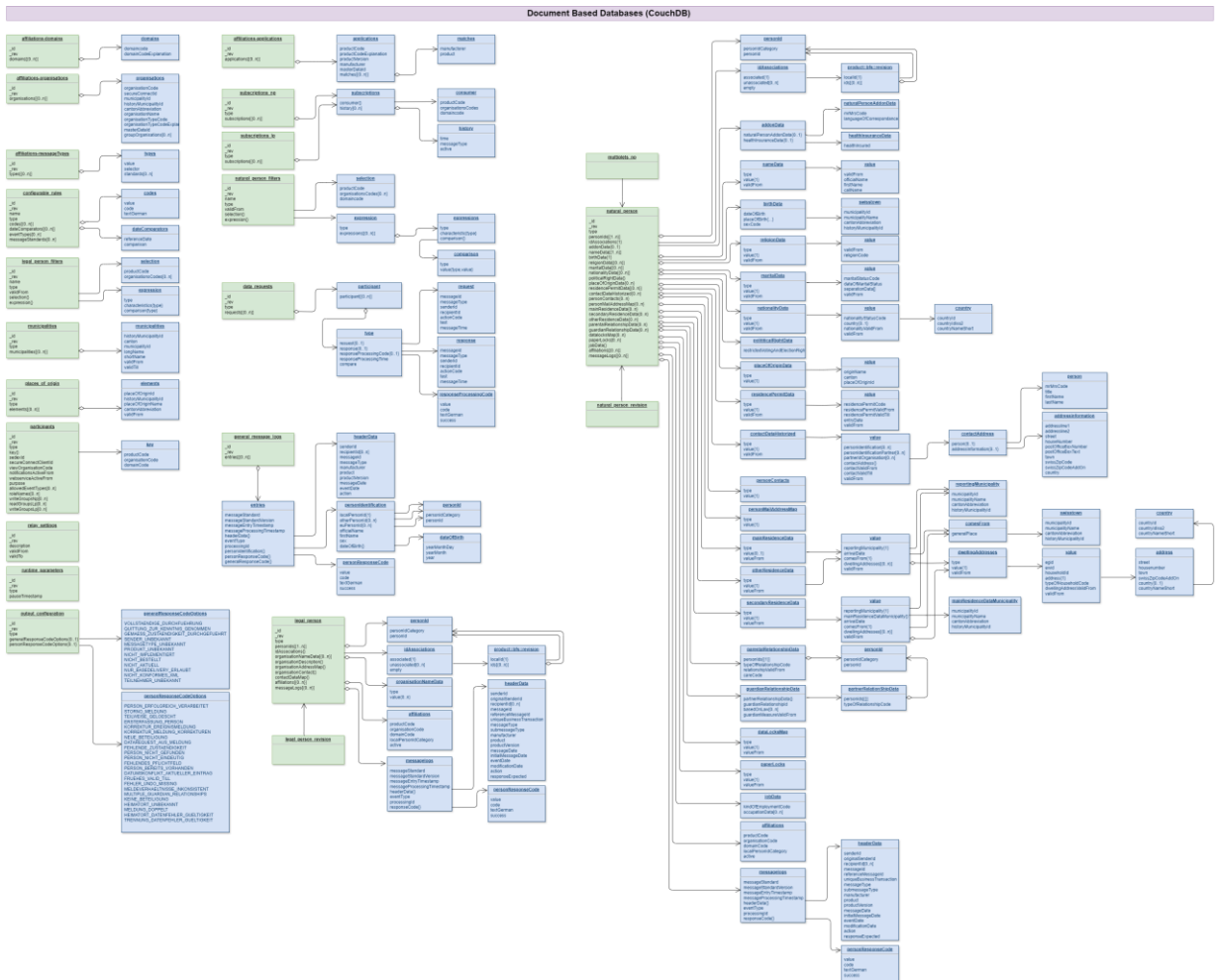
|                |                    |
|----------------|--------------------|
| Classification | public             |
| Version        | 1.0                |
| Date           | October, 31st 2024 |

# Contents

- 1. PER Personenregister Data model ..... 3
  - 1.1 Document Based Database (CouchDB)..... 3
    - 1.1.1 Description of the tables ..... 3
  - 1.2 SQL Based Databases (PostgreSQL) ..... 5
    - 1.2.1 Description of the tables ..... 5

## 1. PER Personenregister Data model

## 1.1 Document Based Database (CouchDB)



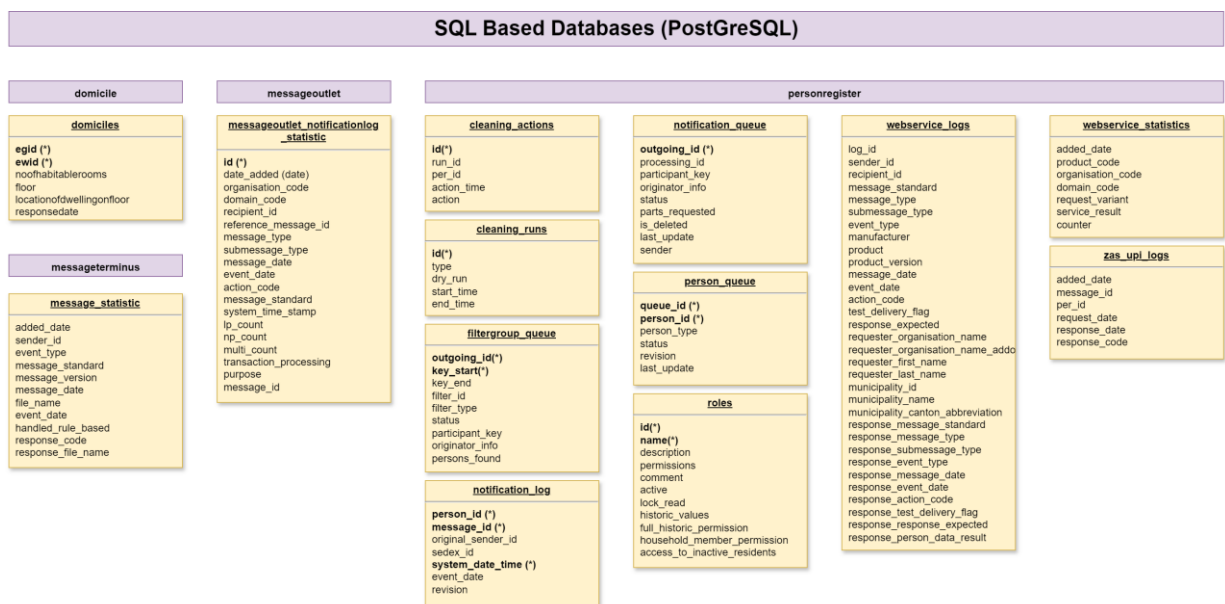
### 1.1.1 Description of the tables

| name of the table          | description   |
|----------------------------|---|
| affiliations-domains       | Contains all domains from an instance. Domains are associated with participants and describe the type of data that the participant is interested in (i.e. tax data, residence data, etc.) |
| affiliations-organisations | Contains all organizations from one instance that have access to PER. An organization is an abstraction of the geographical political entity.   |

| <b>name of the table</b>  | <b>description</b>  |
|---------------------------|---|
| affiliations-messageTypes | Contains the compatible message standards from eCH that PER Personenregister can process.   |
| configurable_rules        | Contains rules that are used to automatically classify responses according to their response codes. Mainly used for triage of messages that resulted in errors.   |
| legal_person_filters      | Contains filters for legal persons. Configurable in AdminGUI. Filters are used to determine which participants need to be automatically notified if person data has changed.  |
| municipalities            | Contains every active and historicized municipality from Switzerland.   |
| places_of_origin          | Contains every municipality in Switzerland.   |
| participants              | Every action in the GUI «PER Auskunft» and the web service, as well as processed messages must be associated to a participant, so it can be checked for permissions and logged accordingly. The known participants are stored here. |
| relay_settings            | Settings for the relay service (relaying original messages to external consumers).  |
| runtime_parameters        | Runtime configuration parameters for Message Terminus (i.e. pause mode).  |
| output_configuration      | Contains configuration if an action is needed after output.   |
| affiliations-applications | Contains all applications from an instance. Applications are associated with participants and describe the software (and to a certain degree, also the protocol) that the participant is using.                                     |
| subscriptions_np          | Contains all active and inactive subscriptions for natural persons. A subscription is an association of a participant with a person. Participants with subscriptions will get notifications, even if no filter selects them.        |
| subscriptions_lp          | Contains all active and inactive subscriptions for legal persons. A subscription is an association of a participant with a person. Participants with subscriptions will get notifications, even if no filter selects them.          |
| natural_person_filters    | Contains all filters for natural persons. Configurable in AdminGUI. Filters are used to determine which participants need to be automatically notified if person data has changed.  |
| data_requests             | Personenregister can request data about a person from a participant. These requests are stored so they can be monitored for data received.  |

| name of the table       | description   |
|-------------------------|---|
| general_message_logs    | Each person in PER has its own message log. Sometimes messages cannot be associated to a (single) person. In this case the log is written into the general message log. |
| legal_person            | Contains legal person data. The data is versioned and historized.   |
| legal_person_revision   | The revisions of the legal_person table. Whenever a person is modified, the previous revision is kept in this database.   |
| multiplets_np           | Contains multiple checks for natural persons. Persons are checked for duplicates (and multiplicatives) and the results are stored here.                                 |
| natural_person          | Contains natural person data. The data is versioned and historized.   |
| natural_person_revision | The revisions of the natural_person table. Whenever a person is modified, the previous revision is kept in this database.   |

## 1.2 SQL-based Databases (PostgreSQL)



### 1.2.1 Description of the tables

| name of the table | description  |
|-------------------|--|
| domiciles         | Information about domiciles (flats per building, orientation of the flat, entrances into the |

| name of the table                       | description   |
|---|---|
|   | buildings). Used to show information to the police and fire fighters.   |
| message_statistic                       | Statistical data about message processing (messages per participant, per organization, etc.). Only viewable in SupportGUI.  |
| messageoutlet_notificationlog_statistic | Notification log of all messageoutlet messages.   |
| cleaning_actions                        | PER can automatically "clean" persons and the results are kept in this database.  |
| cleaning_runs                           | Processing and configuration data for the cleaning runs.  |
| filtergroup_queue                       | Processing data for notifications manually triggered with a filter (go through all persons and notify every participant that passes the filter).                                  |
| notification_queueperson_queue          | Processing data for notifications through regular message processing.   |
| person_queue                            | Processing data for notifications through regular message processing.   |
| roles                                   | Roles are a groups of permissions that are assigned to participants and users. The permissions are used to enforce authorized access only.  |
| webservice_logs                         | Keeps a log of all actions through the web service interface.   |
| webservice_statistics                   | Statistical data about web service access (requests per participant, etc.)  |
| zas_upi_logs                            | Contains logs of the ZAS UPI Service. When a message is processed, sometimes an external service is queried to resolve the VN number of a person. These accesses are logged here. |