Modelling of a privacy language and efficient policy-based de-identification

Armin Gerl

N°d'ordre NNT : 2019LYSEI105

THESE de DOCTORAT DE L'UNIVERSITE DE LYON
opérée au sein de
INSA Lyon
En cotutelle internationale avec
Universität Passau

Ecole Doctorale ED 512
Informatique et Mathématiques

Spécialité / discipline de doctorat :
Informatique

Soutenue publiquement le 5/12/2019, par :
Armin GERL

Modelling of a Privacy Language and Efficient Policy-based De-identification

---

Devant le jury composé de :

Bertino, Elisa	Directeur de Recherche	Purdue University	Rapporteure
Benzekri, Abdelmalek	Professeur des Universités	Université Paul Sabatier-Toulouse	Rapporteur
Granitzer, Michael	Professeur des Universités	Universität Passau	Examinateur
Lenz, Richard	Professeur des Universités	Friedrich-Alexander Universität	Examinateur
Cuppens, Frédéric	Professeur des Universités	Télécom Bretagne	Examinateur
Brunie, Lionel	Professeur des Universités	INSA Lyon	Co-directeur de thèse
Kosch, Harald	Professeur des Universités	Universität Passau	Co-directeur de thèse
Bennani, Nadia	Maître de Conférences	INSA Lyon	Co-directeurice de thèse

Département FEDORA – INSA Lyon - Ecoles Doctorales – Quinquennal 2016-2020

SIGLE	ECOLE DOCTORALE	NOM ET COORDONNEES DU RESPONSABLE
CHIMIE	**CHIMIE DE LYON** **http://www.edchimie-lyon.fr** Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage **secretariat@edchimie-lyon.fr** INSA : R. GOURDON	M. Stéphane DANIELE Institut de recherches sur la catalyse et l'environnement de Lyon IRCELYON-UMR 5256 Équipe CDFA 2 Avenue Albert EINSTEIN 69 626 Villeurbanne CEDEX **directeur@edchimie-lyon.fr**
E.E.A.	**ÉLECTRONIQUE, ÉLECTROTECHNIQUE, AUTOMATIQUE** **http://edeea.ec-lyon.fr** Sec. : M.C. HAVGOUDOUKIAN **ecole-doctorale.eea@ec-lyon.fr**	M. Gérard SCORLETTI École Centrale de Lyon 36 Avenue Guy DE COLLONGUE 69 134 Écully Tél : 04.72.18.60.97 Fax 04.78.43.37.17 **gerard.scorletti@ec-lyon.fr**
E2M2	**ÉVOLUTION, ÉCOSYSTÈME, MICROBIOLOGIE, MODÉLISATION** **http://e2m2.universite-lyon.fr** Sec. : Sylvie ROBERJOT Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 INSA : H. CHARLES **secretariat.e2m2@univ-lyon1.fr**	M. Philippe NORMAND UMR 5557 Lab. d'Ecologie Microbienne Université Claude Bernard Lyon 1 Bâtiment Mendel 43, boulevard du 11 Novembre 1918 69 622 Villeurbanne CEDEX **philippe.normand@univ-lyon1.fr**
EDISS	**INTERDISCIPLINAIRE SCIENCES-SANTÉ** **http://www.ediss-lyon.fr** Sec. : Sylvie ROBERJOT Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 INSA : M. LAGARDE **secretariat.ediss@univ-lyon1.fr**	Mme Emmanuelle CANET-SOULAS INSERM U1060, CarMeN lab, Univ. Lyon 1 Bâtiment IMBL 11 Avenue Jean CAPELLE INSA de Lyon 69 621 Villeurbanne Tél : 04.72.68.49.09 Fax : 04.72.68.49.16 **emmanuelle.canet@univ-lyon1.fr**
INFOMATHS	**INFORMATIQUE ET MATHÉMATIQUES** **http://edinfomaths.universite-lyon.fr** Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage Tél : 04.72.43.80.46 **infomaths@univ-lyon1.fr**	M. Luca ZAMBONI Bât. Braconnier 43 Boulevard du 11 novembre 1918 69 622 Villeurbanne CEDEX Tél : 04.26.23.45.52 **zamboni@maths.univ-lyon1.fr**
Matériaux	**MATÉRIAUX DE LYON** **http://ed34.universite-lyon.fr** Sec. : Stéphanie CAUVIN Tél : 04.72.43.71.70 Bât. Direction **ed.materiaux@insa-lyon.fr**	M. Jean-Yves BUFFIÈRE INSA de Lyon MATEIS - Bât. Saint-Exupéry 7 Avenue Jean CAPELLE 69 621 Villeurbanne CEDEX Tél : 04.72.43.71.70 Fax : 04.72.43.85.28 **jean-yves.buffiere@insa-lyon.fr**
MEGA	**MÉCANIQUE, ÉNERGÉTIQUE, GÉNIE CIVIL, ACOUSTIQUE** **http://edmega.universite-lyon.fr** Sec. : Stéphanie CAUVIN Tél : 04.72.43.71.70 Bât. Direction **mega@insa-lyon.fr**	M. Jocelyn BONJOUR INSA de Lyon Laboratoire CETHIL Bâtiment Sadi-Carnot 9, rue de la Physique 69 621 Villeurbanne CEDEX **jocelyn.bonjour@insa-lyon.fr**
ScSo	**ScSo*** **http://ed483.univ-lyon2.fr** Sec. : Véronique GUICHARD INSA : J.Y. TOUSSAINT Tél : 04.78.69.72.76 **veronique.cervantes@univ-lyon2.fr**	M. Christian MONTES Université Lyon 2 86 Rue Pasteur 69 365 Lyon CEDEX 07 **christian.montes@univ-lyon2.fr**

*ScSo : Histoire, Géographie, Aménagement, Urbanisme, Archéologie, Science politique, Sociologie, Anthropologie

The Cotutelle-PhD has been conducted within the framework of the *International Research & Innovation Center on Intelligent Digital Systems (IRIXYS)*.

The Cotutelle-PhD has been financially supported by the *Deutsch-Französische Hochschule (DFH) / Universitè franco-allemande (UFA)*.

Armin Gerl: *Modelling of a Privacy Language and Efficient Policy-based De-identification*, September 2019

ABSTRACT - ENGLISH

The processing of personal information is omnipresent in our data-driven society enabling personalized services, which are regulated by privacy policies. Although privacy policies are strictly defined by the General Data Protection Regulation (GDPR), no systematic mechanism is in place to enforce them. Especially if data is merged from several sources into a data-set with different privacy policies associated, the management and compliance to all privacy requirements is challenging during the processing of the data-set. Privacy policies can vary hereby due to different policies for each source or personalization of privacy policies by individual users. Thus, the risk for negligent or malicious processing of personal data due to defiance of privacy policies exists.

To tackle this challenge, a privacy-preserving framework is proposed. Within this framework privacy policies are expressed in the proposed *Layered Privacy Language (LPL)* which allows to specify legal privacy policies and privacy-preserving de-identification methods. The policies are enforced by a *Policy-based De-identification (PD)* process. The PD process enables efficient compliance to various privacy policies simultaneously while applying pseudonymization, personal privacy anonymization and privacy models for de-identification of the data-set. Thus, the privacy requirements of each individual privacy policy are enforced filling the gap between legal privacy policies and their technical enforcement.

ABSTRACT - FRENCH

De nos jours, les informations personnelles des utilisateurs intéressent énormément les annonceurs et les industriels qui les utilisent pour mieux cibler leurs clients et pour amééliorer leurs offres. Ces informations, souvent trés sensibles, nécessitent d'être protégées pour réguler leur utilisation. Le Règlement Général sur la Protection des Données (RGPD) est la législation européenne, récemment entrée en vigueur en Mai 2018 et qui vise à renforcer les droits de l'utilisateur quant au traitement de ses données personnelles. Parmi les concepts phares du RGPD, la définition des règles régissant la protection de la vie privée par défaut (privacy by default) et dès la conception (privacy by design). La possibilité pour chaque utilisateur, d'établir un consentement personnalisé sur la manière de consommer ses données personnelles constitue un de ces concepts. Ces règles, malgré qu'elles soient bien explicitées dans les textes juridiques, sont difficiles à mettre en œuvre du fait de l'absence d'outils permettant de les exprimer et de les appliquer de manière systématique – et de manière différente – à chaque fois que les informations personnelles d'un utilisateur sont sollicitées pour une tâche donnée, par une organisation donnée.

L'application de ces règles conduit à adapter l'utilisation des données personnelles aux exigences de chaque utilisateur, en appliquant des méthodes empêchant de révéler plus d'information que souhaité (par exemple : des méthodes d'anonymisation ou de pseudo-anonymisation). Le problème tend cependant à se complexifier quand il s'agit d'accéder aux informations personnelles de plusieurs utilisateurs, en provenance de sources différentes et respectant des normes hétérogènes, où il s'agit de surcroit de respecter individuellement les consentements de chaque utilisateur.

L'objectif de cette thèse est donc de proposer un framework permettant de définir et d'appliquer des règles protégeant la vie privée de l'utilisateur selon le RGPD. La première contribution de ce travail consiste à définir le langage LPL (Layered Privacy Language) permettant d'exprimer, de personnaliser (pour un utilisateur) et de guider l'application de politiques de consommation des données personnelles, respectueuses de la vie privée. LPL présente la particularité d'être compréhensible pour un utilisateur ce qui facilite la négociation puis la mise en place de versions personnalisées des politiques de respect de la vie privée.

La seconde contribution de la thèse est une méthode appelée Policy-based De-identification. Cette méthode permet l'application efficace des règles de protection de la vie privée dans un contexte de données multi-utilisateurs, régies par des normes hétérogènes de respect de

la vie privée et tout en respectant les choix de protection arrêtés par chaque utilisateur. L'évaluation des performances de la méthode proposée montre un extra-temps de calcul négligeable par rapport au temps nécessaire à l'application des méthodes de protection des données.

CONTENTS

1 INTRODUCTION 1
1.1 User Concerns . . . 2
1.2 Company Concerns . . . 4
1.3 Key Issues . . . 7
1.4 Research Questions . . . 8
1.5 Key Contributions . . . 10
1.6 Organization of the Thesis . . . 11
2 PRIVACY LANGUAGE REQUIREMENTS 13
2.1 Privacy Policy Structure . . . 13
2.2 Legal Compliance . . . 15
2.2.1 Information to be Provided . . . 15
2.2.2 Data Subject Rights . . . 18
2.3 Human-readability . . . 19
2.3.1 Textual Representation . . . 20
2.3.2 Privacy Icons . . . 21
2.4 Access Control . . . 22
2.5 De-identification Capabilities . . . 22
2.6 Provenance . . . 23
2.6.1 Data Processing . . . 24
2.6.2 Data Production . . . 24
3 DE-IDENTIFICATION BACKGROUND 27
3.1 Personal Data Categorization . . . 27
3.1.1 Multidimensional Data . . . 29
3.1.2 Transaction Data . . . 30
3.1.3 Longitudinal Data . . . 30
3.1.4 Graph Data . . . 31
3.1.5 Time Series Data . . . 32
3.1.6 Summary . . . 33
3.2 Anonymization . . . 33
3.2.1 Suppression . . . 33
3.2.2 Generalization . . . 35
3.2.3 Deletion . . . 36
3.2.4 Summary . . . 37
3.3 Privacy Models . . . 38
3.3.1 Record Linkage . . . 39
3.3.2 Attribute Linkage . . . 41
3.3.3 Table Linkage . . . 43
3.3.4 Probabilistic Attack . . . 44
3.3.5 Privacy and Utility Trade-off . . . 45
3.3.6 Reasoning on Privacy Models . . . 47
3.4 Personal Privacy . . . 49
3.4.1 Personal Privacy Approaches . . . 49

3.4.2 Reasoning on Personal Privacy 51
3.5 Pseudonymization 52
3.5.1 Pseudonym Generation 53
3.5.2 Implementation Patterns 56
3.5.3 Reasoning on Pseudonymization 58
4 RELATED WORK 59
4.1 Classification of Privacy Languages 59
4.1.1 Access Control Policy 61
4.1.2 SLA Policy 64
4.1.3 Privacy Policy Transparency 65
4.1.4 Privacy Policy Preferences 68
4.1.5 Privacy Policy Enforcement 69
4.1.6 Discussion 73
4.2 Positioning 74
5 LAYERED PRIVACY LANGUAGE (LPL) 79
5.1 Concepts 79
5.1.1 Privacy Policy Structure 79
5.1.2 Legal Compliance 80
5.1.3 Human-readability 80
5.1.4 Access Control 81
5.1.5 De-identification Capabilities 82
5.1.6 Provenance 83
5.1.7 Naming 84
5.2 Formal Definition 85
5.2.1 Super-elements 85
5.2.2 Elements 89
5.3 Life Cycle 102
5.3.1 Creation 104
5.3.2 Negotiation 106
5.3.3 Pre-Processing 107
5.3.4 Storage 108
5.3.5 Transfer 108
5.3.6 Usage 109
5.4 Legal Compliance 110
5.4.1 Privacy Policy 110
5.4.2 Data Subject Rights 112
5.5 Access Control 116
5.5.1 Support Structures 117
5.5.2 Entity-Authentication 120
5.5.3 Purpose-Authorization 121
5.5.4 Entity-Authorization 122
5.5.5 Data-Authorization 124
5.6 Privacy Policy User Interface 125
5.6.1 User Interface Layout 125
5.6.2 Negotiation View 129
5.6.3 Creation View 132

5.6.4 Discussion . . . 133
5.7 Provenance . . . 135
5.7.1 Data Processing . . . 136
5.7.2 Data Production . . . 137
5.8 Discussion . . . 140
6 POLICY-BASED DE-IDENTIFICATION 143
6.1 Overview . . . 144
6.1.1 Data-warehouse Scenario . . . 144
6.1.2 Process Sequence . . . 147
6.1.3 Data Structure – DataWrapper . . . 151
6.2 Pseudonymization . . . 153
6.3 Personal Privacy Anonymization . . . 155
6.3.1 Minimum Anonymization . . . 156
6.3.2 Global Minimum Anonymization . . . 157
6.4 Privacy Model . . . 161
6.4.1 Set Maximum Anonymization . . . 162
6.4.2 Set Privacy Group . . . 163
6.4.3 Privacy Model Substitution . . . 164
6.4.4 Apply Privacy Models . . . 171
6.5 Discussion . . . 172
7 EVALUATION 175
7.1 Policy-based De-identification Benchmark Framework 175
7.1.1 Data Provider . . . 176
7.1.2 Policy Provider . . . 177
7.1.3 Policy-based De-identification . . . 178
7.1.4 Benchmark Configuration . . . 180
7.2 Experiments . . . 181
7.2.1 Policy-based De-identification Overhead . . . 181
7.2.2 Personal Privacy – Minimum Anonymization . 189
7.2.3 Personal Privacy – Global Minimum Anonymization . . . 196
7.3 Conclusion . . . 208
8 CONCLUSION AND FUTURE WORK 211
8.1 Conclusion . . . 211
8.2 Future Work . . . 216
8.2.1 Extended Scenarii . . . 216
8.2.2 Company Compliance . . . 218
8.2.3 User Experience . . . 219

I APPENDIX

A POLICY-BASED DE-IDENTIFICATION – ARX ALGORITHMS 223
A.1 Data-set Transformation . . . 224
A.2 Anonymization Hierarchy Transformation . . . 224

BIBLIOGRAPHY 227

LIST OF FIGURES

Figure 1.1 Privacy policy scenario for the processing and transfer of personal data. 1
Figure 2.1 Core structure and elements of a privacy policy. 14
Figure 2.2 Visualization of the *Layered* approach as proposed by the *Article 29 Working Party* [18] [124]. 21
Figure 3.1 Example of graph data in the context of a social network. A 'friend' relationship is expressed between the different users. 32
Figure 3.2 Anonymization hierarchy for the German postal-code for Passau '94032' based on suppressing first the last character of the postal-code using the replacement character '*'. The anonymization level is steadily increasing starting with level '0' and ending with level '5'. 34
Figure 3.3 Anonymization hierarchy for the domain 'sex'. The scope of original values at the anonymization level '0' is limited. The anonymization of the original value leads to the replacement of it with 'ANY' denoting a generic upper class for both 'M' for male and 'F' for female. 35
Figure 3.4 Anonymization hierarchy for the domain 'age'. The scope of original values at the anonymization level '0' has a broad range. Each subsequently higher anonymization level covers a sub-range of the original values. The generic value 'ANY' is used for the highest anonymization level. 36
Figure 3.5 Anonymization hierarchy for the QI attribute job for the patient table (see Table 3.10). 40
Figure 3.6 Example scenario for *Patient Identifier (PID)* generation based on Lablans et al. [165] using pseudonymization. 54
Figure 4.1 Classification of privacy languages in categories with *Security-Focus* and *Privacy-Focus* [121]. The classification is based on the works of Kumaraguru et al. [164] and Kasem-Madani and Meier [151]. 61

Figure 4.2 Anonymization hierarchy for the German postal-code for Passau '94032' (see Figure 3.2) with both the *Minimum Anonymization Level* '1' and *Maximum Anonymization Level* '3' detailed. Furthermore, the possible range for values during later anonymization is denoted. 76
Figure 5.1 Core privacy policy structure of the Layered Privacy Language. 80
Figure 5.2 Entity-Relationship-Model showing the complete structure of the *Layered Privacy Language (LPL)*. All elements and attributes are shown. The inheritance of *UIElement* and *Entity* is denoted. 88
Figure 5.3 Life-cycle of LPL. 104
Figure 5.4 *Data Subject Right* scenario showing a schematic response generation based on the the individuals' personal data and LPL privacy policy. . . . 113
Figure 5.5 Process chain for the *Policy-based Access Control (PAC)*. 117
Figure 5.6 Possible structure of *Entity-Hierarchy* based on the scenario in Section 5.3). The {}-node is a default node with no access rights assigned. . 119
Figure 5.7 *Purpose-Hierarchy* showing a possible inheritance hierarchy for purposes based upon the online shop scenario of Section 5.3. 120
Figure 5.8 *Negotiation View* for a LPL privacy policy stating the processing of personal data for marketing, research, and billing. The layout elements based on the *Negotiation View* are highlighted as follows: 1: *Policy Header*, 2: *Privacy Icon Overview*, 3: *Purpose Overview*, 4: *Purpose Detail*, and 5: *Policy Information*. 127
Figure 5.9 Place-holder *Privacy Icons* indicating the processing of personal data for marketing, research, and billing purposes. 128
Figure 5.10 Anonymization method settings for the attribute `postal-code` using the*Creation View*. . . 132
Figure 5.11 *Creation View* for a raw LPL privacy policy lpp_{raw} stating the processing of personal data for marketing, research, and billing. 134

Figure 5.12 *Data Processing* use case based upon the *Transfer* phase scenario. Data is transferred from user ds_{U1} to company c_{C1} under the LPL privacy policy $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$. The data is furthermore transferred to c_{C2} under the policy $lpp_{ds_{C1}\text{-}\{dr_{C2}\}}$, which has to be validated against the previous policy. 137
Figure 5.13 *Data Production* use case, demonstrating the production of new data based upon the personal data of several users. The derived data is associated with a new LPL privacy policy incorporating the distinct privacy policies of the users as underlying privacy policies for enforcing *Provenance* and credibility. 139
Figure 6.1 Data-warehouse scenario in which records with personal data from different sources are collected for further processing. Each record is linked to a (personalized) LPL privacy policy. A LPL privacy policy is detailed in excerpts. . 146
Figure 6.2 Anonymization hierarchy for the domain $'age'$. The scope of original values at the anonymization level $'0'$ has a broad range. Each subsequently higher anonymization level covers a sub-range of the original values. The generic value $'ANY'$ is used for the highest anonymization level. 150
Figure 6.3 *Policy-based De-identification (PD)* process step sequence after the *Policy-based Access Control (PAC)* has been completed for a request. 152
Figure 6.4 *Minimum Anonymization* example for the attribute postal-code. 158
Figure 6.5 *Global Minimum Anonymization* example for the attribute postal-code. 159
Figure 6.6 Example for *Set Maximum Anonymization* process for the attribute postal-code. 163
Figure 6.7 Examples for *Privacy Model Substitution* showing *Complete Substitution*, *Attribute Substitution*, *Model Substitution*, and *Model Combination*. . . . 166
Figure 6.8 Example for *Privacy Model Substitution* process. 172
Figure 7.1 Framework architecture for *Policy-based De-identification* benchmark evaluation. 176
Figure 7.2 Run-time comparison of *Policy-based De-identification* to *Apply Privacy Models* for different privacy models. 186

Figure 7.3 Run-time difference for all configurations from Table 7.3 of the *Policy-based De-identification Overhead* experiment. All data-sets are used and visualized in the graph via their size, i.e. number of records (see Table 7.1). 187
Figure 7.4 Percentage run-time difference for all configurations from Table 7.3 of the *Policy-based De-identification Overhead* experiment. All data-sets are used and visualized in the graph via their size, i.e. number of records (see Table 7.1). . . . 188
Figure 7.5 Line graph showing the overall run-time in seconds for different number of records with personal privacy (PP) on all attributes '9' using the *Minimum Anonymization*-algorithm. *5-Anonymity* is applied on different data-set sizes of *IHIS*. 192
Figure 7.6 Line graph showing the overall run-time in seconds for different numbers of attributes with personal privacy using the *Minimum Anonymization*-algorithm. *5-Anonymity* is used on different data-set sizes of *IHIS*. The number of records with personal privacy (PP) '0.4' is constant. 195
Figure 7.7 Line graph showing the overall run-time in seconds for different number of records with personal privacy (PP) on all attributes '9' using the *Global Minimum Anonymization*-algorithm. *5-Anonymity* is is on different data-set sizes of *IHIS*. 197
Figure 7.8 Line graph showing the overall run-time in seconds for different numbers of attributes with personal privacy using the *Global Minimum Anonymization*-algorithm. *5-Anonymity* is used on different data-set sizes of *IHIS*. The number of records with personal privacy (PP) is set to '0.4'. 201
Figure A.1 *Policy-based De-identification (PD)* process step sequence with the ARX specific processes after the *Policy-based Access Control (PAC)* has been completed for a request. 223

LIST OF TABLES

Table 2.1 Legal requirements of information to be provided by privacy policies according to Art. 12 - 14 [110, Art. 12 - 14] [119]. 17
Table 2.2 Overview of *Data Subject Rights* of Art. 15 - 22 *GDPR* [110, Art. 15 - 22]. 19
Table 3.1 Example of a multidimensional data-set detailing various attributes classified as explicit identifier (EI), quasi-identifier (QI), sensitive data (SD) and non-sensitive data (NSD). Each row represents personal data of a user for a lottery. The NSD 'Lucky#' is randomly assigned to the person. 29
Table 3.2 Example of data-set of transactional data for an online shop selling french food specialities. Each transaction is represented by a '1', while no transaction is denoted by a '0'. For example Alice bought 'Baguette' and 'Cheese'. 30
Table 3.3 Example of data-set of longitudinal data in the context of continuous blood glucose measurements. Each row represents a measurement for a patient representing date and time of the measurement, as well as the measured blood glucose level. For example, Bob measured his blood glucose level six times over the course of two days. 31
Table 3.4 Example of data-set of time series data regularly measuring the yearly salary of individuals. Each row represents the regular measurements for one person. The dimensionality of the data-set is growing regularly with each measurement, e.g., each year in this example. 32
Table 3.5 Summary of personal data-set categories. . . . 33
Table 3.6 Suppression of the postal-code of Bob to anonymization level '2' based on the example multidimensional data-set in Table 3.1. 35
Table 3.7 Generalization of the sex and age of Bob to anonymization level '1' based on the previously suppressed data-set in Table 3.6. 36
Table 3.8 Deletion of both the ID and name of Bob based on the previously generalized data-set in Table 3.7. 37

Table 3.9	Summary of anonymization methods.	38
Table 3.10	Patient table with EI attributes deleted to prevent identification.	40
Table 3.11	Available external knowledge with EI and QI attributes.	40
Table 3.12	3-anonymous patient table based on Table 3.10. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 3.4, and 3.5.	41
Table 3.13	Distinct 3-diverse patient table based on Table 3.10. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 3.3, and 3.5.	42
Table 3.14	4-anonymous external knowledge table based on Table 3.11. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 3.3, and 3.5.	44
Table 3.15	Non-exhaustive list of privacy models and their classification according to the attack models they mitigate.	48
Table 3.16	Summary of personal privacy approaches.	52
Table 3.17	Mapping store for ID for the data-set in Table 3.1 in which the ID of the record of Bob is stored.	57
Table 3.18	Mapping store for name for the data-set in Table 3.1 in which the name of the record of Bob is stored.	57
Table 3.19	Pseudonymization of the ID and name of Bob replacing the corresponding original values based on the previously generalized data-set in Table 3.7.	57
Table 3.20	Summary of approaches for pseudonymization.	58
Table 4.1	Overview of fulfilled requirements of related privacy languages of the *Access Control Policy* and *SLA Policy* category. An 'X' denotes the fulfilment of the requirement, an '(X)' denotes the partial fulfilment of the requirement, and a '-' denotes that the requirement is not fulfilled.	63
Table 4.2	Overview of fulfilled requirements of related privacy languages of the *Privacy Policy Transparency*, *Privacy Policy Preferences* and *Privacy Policy Enforcement* category. An 'X' denotes the fulfilment of the requirement, an '(X)' denotes the partial fulfilment of the requirement, and a '-' denotes that the requirement is not fulfilled.	67

Table 5.1 Overview over all elements and their formal definition. Bold styled sets are tuples inheriting an order. 87
Table 5.2 Fulfilled legal requirements of Art. 12 - 14 [110, Art. 12 - 14] by LPL. 112
Table 5.3 General structure of *Entity-Lookup Table* shown for the scenario introduced in Section 5.3 as an example. 119
Table 6.1 3-anonymous patient table based on Table 3.10. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 6.2, and 3.5. 149
Table 6.2 Result patient table for the usage of *Personal Privacy Anonymization* on a 3-anonymous patient table (see Table 6.1. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 6.2, and 3.5. 149
Table 6.3 Result patient table for the usage of *Personal Privacy Anonymization*. The value of the attribute age is anonymized to meet the individuals privacy requirements. Anonymization is conducted with the anonymization hierarchy shown in Figure 6.2. 151
Table 6.4 Result *3-anonymous* patient table for the usage of *Personal Privacy Anonymization* before *Privacy Models*. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 6.2, and 3.5. 151
Table 6.5 Requested data-set for the data-warehouse scenario depicted in Figure 6.1 which is represented by DW_{req}. 154
Table 6.6 Pseudonymous data-set for the data-warehouse scenario depicted in Figure 6.1 which is represented by DW_{psm}. 155
Table 6.7 Bijective mapping table for the pseudonymized data-set of the data-warehouse scenario. 155
Table 6.8 Data-warehouse scenario data-set pseudonymized and anonymized using the *Minimum Anonymization*-algorithm. 160
Table 6.9 Data-warehouse scenario data-set pseudonymized and anonymized using the *Global Minimum Anonymization*-algorithm. 160

Table 6.10 Data-warehouse scenario data-set which fulfils the properties of *3-Anonymity*. The data-set is pseudonymized and anonymized using either the *MA*- or *GMA*-algorithm for *Personal Privacy Anonymization*. 161
Table 6.11 Reduced list (see Table 3.15) of privacy models and their classification according to the attack models they mitigate. 166
Table 6.12 Example *Privacy Model Substitution Table* for the privacy models *k-Anonymity*, *(c,l)-Diversity* and *t-Closeness*. 170
Table 7.1 Overview of used data-sets for the evaluation. #*Attr* denotes number of attributes. *Hierarchy Size* defines for each attribute the depth of the hierarchy. #*Records* denotes the number of records. #*DV* denotes the number of distinct attribute values. 177
Table 7.2 Overview of privacy-preserving frameworks providing privacy model implementations. . . 180
Table 7.3 Benchmark results for the *Policy-based De-identification Overhead* evaluation. All measurements are displayed in seconds. Calculations of the Δ_{time}, $\Delta_{\mathrm{percent}}$, $\%_{\mathrm{MA}}$, $\%_{\mathrm{sub}}$, and $\%_{\mathrm{pm}}$ and have been conducted with nanosecond precision measures, therefore deviations of displayed values and calculations can occur. The column *Overall*, representing the full run-time and *PM*, representing the run-time of the privacy model application, are used for the overhead calculation. Labels have been shortened according to following scheme: (*MA*: Minimum Anonymization, *Max*: Set Maximum Anonymization, *Group*: Set Privacy Group, *Sub*: Privacy Model Substitution, *PM*: Apply Privacy Models, *Overall*: Overall Execution Time). . . . 184
Table 7.4 Overview of IHIS attributes, the size of the corresponding anonymization hierarchies, and the exclusive upper *Minimum Anonymization Levels* used in the experiments. 190
Table 7.5 Overall run-time of the PD process using the *Minimum Anonymization*-algorithm for varying number of records with personal privacy settings (PP). The number of attributes used for personal privacy settings (#Attr) is fixed to '9'. 191

Table 7.6 Detailed run-time measurements for each individual process of PD using *5-Anonymity* on the *IHIS* data-set with 100,000 records while varying the number of records with personal privacy settings PP. Labels have been shortened according to following scheme: (*MA*: Minimum Anonymization, *Max*: Set Maximum Anonymization, *Group*: Set Privacy Group, *Sub*: Privacy Model Substitution, *PM*: Apply Privacy Models. 193

Table 7.7 Overall run-time of the PD process using the *Minimum Anonymization*-algorithm for varying number of number of attributes used for personal privacy settings (#Attr). The proportional number of records with personal privacy settings (PP) is fixed to '0.4'. 195

Table 7.8 Overall run-time of the PD process using the *Global Minimum Anonymization*-algorithm for varying number of records with personal privacy settings (PP). The number of attributes used for personal privacy settings is fixed to '9'. 197

Table 7.9 Detailed run-time measurements for each individual process of PD using *5-Anonymity* on the *IHIS* data-set with 100,000 records while varying the number of records with personal privacy settings PP. Labels have been shortened according to following scheme: (*GMA*: Global Minimum Anonymization, *Max*: Set Maximum Anonymization, *Group*: Set Privacy Group, *Sub*: Privacy Model Substitution, *PM*: Apply Privacy Models. 199

Table 7.10 Overall run-time of the PD process using the *Minimum Anonymization*-algorithm for varying number of number of attributes used for personal privacy settings (#Attr). The proportional number of records with personal privacy settings (PP) is fixed to '0.4'. 200

Table 7.11 Run-time difference of the *Policy-based De-identification* (PD) comparing the *Minimum Anonymization (MA)*-algorithm with the *Global Minimum Anonymization (GMA)*-algorithm. The time difference $\Delta_{\texttt{time-difference}}$ and percentage difference $\Delta_{\texttt{percent-difference}}$ is calculated. The *IHIS* data-set with 10,000 records is used. *5-Anonymity* is used as the privacy model. 204

Table 7.12 Measurement of distinct values #DV and number of anonymization hierarchy elements #HE for the processed data-set at the beginning of the PD process (Input Data-set) and before *Apply Privacy Models* (Pre-PM Data-set). $\Delta_{\#DV_{\#Attr}}$ and $\Delta_{\#HE_{\#Attr}}$ are put into relation to $\Delta_{\texttt{percent-difference}}$ is calculated. The *IHIS* data-set with 10,000 records is used. *GMA* is used for *Personal Privacy Anonymization* and*5-Anonymity* is used as the privacy model. 205
Table 7.13 Run-time difference of the *Policy-based De-identification* (PD) comparing the *Minimum Anonymization (MA)*-algorithm with the *Global Minimum Anonymization (GMA)*-algorithm. The time difference $\Delta_{\texttt{time-difference}}$ and percentage difference $\Delta_{\texttt{percent-difference}}$ is calculated. The *IHIS* data-set with 1,000,000 records is used. *5-Anonymity* is used as the privacy model. 206
Table 7.14 Measurement of distinct values #DV and number of anonymization hierarchy elements #HE for the processed data-set at the beginning of the PD process (Input Data-set) and before *Apply Privacy Models* (Pre-PM Data-set). $\Delta_{\#DV_{\#Attr}}$ and $\Delta_{\#HE_{\#Attr}}$ are put into relation to $\Delta_{\texttt{percent-difference}}$ is calculated. The *IHIS* data-set with 1,000,000 records is used. *GMA* is used for *Personal Privacy Anonymization* and*5-Anonymity* is used as the privacy model. 207

LISTINGS

Listing 5.1 Pseudocode describing the authorization of purposes of a lpp utilizing *Purpose-Hierarchy*. The *Entity-Hierarchy* is assumed to be accessible within the method. 122
Listing 5.2 Pseudocode describing the authorization of a requesting entity for an authorized purpose utilizing *Entity-Hierarchy*. The *Entity-Hierarchy* is assumed to be accessible within the method. 123
Listing 5.3 Pseudocode describing the authorization of data from authorized purposes. 124

Listing 5.4 Pseudocode to determine the origin source of a specific data attribute. 138
Listing 6.1 Pseudocode of the determination of pseudonymization methods that have to be applied on the data-set. 154
Listing 6.2 Pseudocode of the *Minimum Anonymization*-algorithm. 156
Listing 6.3 Pseudocode of the *Global Minimum Anonymization*-algorithm. 159
Listing 6.4 Pseudocode for the *Set Maximum Anonymization*. 163
Listing 6.5 Pseudocode for the *Set Privacy Group*. 165
Listing 6.6 Pseudocode for the *Privacy Model Substitution*. 171
Listing A.1 Pseudocode for the *Data-set Transformation*. . . 224
Listing A.2 Pseudocode for the *Anonymization Hiearchy Transformation*. 225

1

INTRODUCTION

In day to day life, personal data is collected and processed continuously to provide personalized services easing one's daily tasks. However, privacy protection of personal data is a fundamental right requiring that personal data can only be processed for defined purposes with the consent of the user or on legitimate interest based on a law [98, Art. 8]. Thus, enforcing personal data protection is essential.

In the context of the processing of personal data by companies, the privacy policy is the core for expressing and regulating the usage of one's data. Assuming that a user registers for a service offered by a company, a privacy policy about his data has to be agreed upon. This agreement can hereby include the consent to individual purposes. Only after the agreement of the user, his personal data may be processed by the company.

Furthermore, data can be transferred from the company to a third party, e.g., for outsourcing processes or data trading as part of the business model of the company (see Figure 1.1).

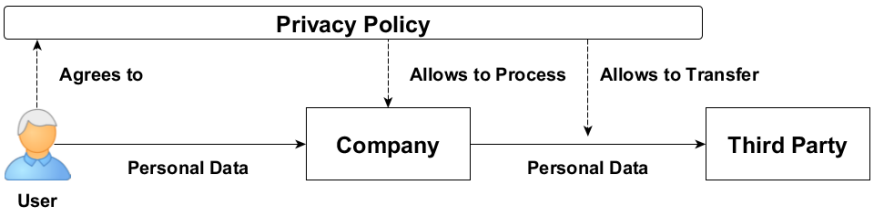


Figure 1.1: Privacy policy scenario for the processing and transfer of personal data.

The key actors within this privacy policy scenario are the user agreeing on the privacy policy and the company providing the privacy policy and processing the personal data. Transfer to a third party, e.g., another company for B2B business, is optional and lies in the responsibility of the company and must be agreed upon by the user through the privacy policy. This simple scenario is applicable for most situations of life in which personal data is processed, for example daily activities like shopping, transportation, socializing and connecting with others, health care, home automation and many more.

Both users and companies have their own concerns regarding the protection of privacy in the context of the processing of personal data. In the following, those concerns are highlighted and several key issues are derived which are correlated to privacy policies for both companies and users. Detailed description of the tackled research questions and used methodology follow.

1.1 USER CONCERNS

From the user's point of view, privacy should be provided at any time minimizing the personal data exposure, while the quality and performance of the service is maintained. In contrast, processing personal data is common and necessary during various aspects of daily life, e.g., online shopping, to provide user-centric services. These contradicting objectives require the regulation via privacy policies, allowing the user to be informed about the processing of his personal data while having control over the processing of his personal data. Thus, the main concerns for the user are the transparency and control of personal data use while the quality and performance of the service is maintained, which are illustrated respectively by *Scenarios 1*, *2* and *3*.

Scenario 1: Online shops use their customers' (users') personal information to provide personalized offers and specifically market new products to target audiences. Furthermore, customers behaviour and interests are tracked to recommend products they might be interested in. These services are highly connected, such that the usage of a search engine looking for a TV series, e.g., *'Game of Thrones'*, is tracked and shared with other services. This information about the interests of the user leads to product recommendations for merchandise while using online shops. The relation of the usage of the search engine of one company, which derives the users interest, and the recommendation of merchandise by another company, utilizing the information about the users' interest, may not be clear from the point of view of the user. It is not transparent for the user how the online shop received his personal data, i.e. interest in *'Game of Thrones'* for the recommendation of products.

Such services allow online shops to gain added value through a potentially increase in their revenue. For users, these services may be positively or negatively perceived, depending on their personal attitude. This can be perceived positively, adding value to the shopping experience of the customer because recommended products match exactly what the user is interested in. Inversely, this may be perceived as a violation of privacy, because personal information, i.e. interest in *'Game of Thrones'* is shared without the known agreement of the user. This perception may be based on a non-transparent agreement stating the sharing of personal data from the search engine, or

the online shop not informing the user about reasoning behind the personalized recommendations. In both cases the user is unable to reconstruct the agreement sharing this personal data for either the online shop or the search engine.

Therefore, transparent presentation of privacy policies is a key issue [110, Art. 13 - 14]. In the above use case the search engine could express in a clear and concise way that the submitted search terms are tracked and can be used by third parties for personalized offers. The online shop could furthermore specify that the user is recommended specific products based on tracked search terms of the search engine. Therefore, the flow of personal data as well as its usage would be more understandable for the user.

Scenario 2: In social networks, privacy issues can be perceived in a similar way. Users stay in contact with dear friends and family, get in touch with new individuals or groups, or forge new business relationships. To enable the full potential of such networks, the user is encouraged to create a personal profile, including various personal information on age, gender, education, interests, and profile picture. Messages, pictures and videos are exchanged to share one's thoughts and experiences with others. This sharing of information is the crux of the matter for users, due to the non-transparent rules and settings of the social network provider. One cannot be sure that personal information is only shared with his intended audience nor is aware of how to change his personal settings. Furthermore, if the information is shared with third parties, the user has no more control over future processing of his personal data.

Thus, the user needs a transparent and fine-grained control over his personal data, such that only specific user groups, e.g., trusted family and friends, can have access to a specified part of his information. This includes situations in which personal data is shared to third parties. Hereby, the user shall have the ability to specify the way how his personal data is handled. Furthermore, it should be considered that the default settings for the privacy control protect the user [110, Art. 25], so that only through explicit interaction personal data is shared to a greater extent [110, Art. 7].

Scenario 3: This concern is even more present when sensitive information of users is processed, e.g., personal data on health, religion or sex. The access to such data should be limited by the usage or purpose. According to the *GDPR*, such data has to be treated with special care [110, Art. 9] Assuming the user had a severe car accident and is treated by the first responders, the emergency physicians require information for fast and effective treatment of the user to save his life. For example, the blood type for transfusions or possible information about allergies or incompatible medication. Such information should only be accessible to the emergency physician in case of such an accident scenario, otherwise a treating physician, or anybody else, has

no reason to access the personal information.

Those concerns are publicly known, discussed and can be mitigated by the user with additional effort. Users can directly influence the privacy of their personal data by the choice of alternative privacy-friendly services, e.g., search engines, private modes in web-browsers preventing tracking [209] [210], or due to detailed engagement with privacy settings of a service. But, this requires users to actively decide and search for privacy-friendly alternatives and settings, thus requiring additional effort. Despite privacy being a concern for many people, only few actually act to preserve their privacy [256]. This phenomenon, in which users express privacy concerns but do not act accordingly, is denoted as *Privacy Paradox* [246] [26] [143] [24]. For example, it has been shown that users, despite being sufficiently familiar with technology, claim to be concerned about privacy are not willing to invest time, effort or money to protect their privacy [25]. This unwillingness of the users has to be considered for the design of privacy solutions, which should be easy to use.

Thus, the user has several concerns. First, the user has to be informed about the processing of his personal data in a transparent way. Second, the user has to have control over the processing of his personal data for different purposes even when data is transferred or traded to third parties. Third, the control over the processing of his personal data has to be designed in an easy and accessible way. The user should be able to personally decide on a trade-off between his privacy and the functionality of the service.

1.2 COMPANY CONCERNS

From a company point of view, the personal data of their users is a valuable asset, which is essential for many business models. However, if a privacy incident, i.e. data breach, happens then the reputation of the company can be affected. A survey showed that before a data breach happens users typically only discuss the perceived quality of the product or service. But after a data breach occurred also other aspects of the companies reputation are payed attention to, i.e. the customer orientation or performance of the corporate [70]. Thus, the companies reputation can be affected by privacy incidents, which changes the users' perception of the company.

Besides, companies are also enforced by law to protect users' privacy. In fact, within the *European Union* (EU), the *General Data Protection Regulation 2016/679* (*GDPR*) entered into force on 25th May 2018 [110, Art. 99 (2)]. This legal framework is designed to standardise privacy laws for all member states of the EU, to protect and empower citizens' privacy, and to revise how companies approach data privacy. The principles *Privacy by Design* and *Privacy by Default*, as well as the

introduction and strengthening of *Data Subject Rights*, the rights a user has, are hereby most noticeable.

Privacy by Design, an already existing concept adopted by the *GDPR*, denotes that all technical systems have to include at design time appropriate technical and organisational measures to preserve privacy [110, Art. 25 (1)].

Furthermore, *Privacy by Default* denotes that technical systems have to provide the users with privacy-friendly settings. Therefore, by default only necessary personal data shall be processed for a purpose. This includes that no pre-emptive data collection for future processing shall be conducted, that personal data processing is limited to the scope of the intended purpose, and the transfer of data is limited by default and can only be extended with prior intervention of the user [110, Art. 25 (2)]. Thus, the users' privacy is protected by default in the best way possible and requires no further interaction or effort by the user.

Both *Privacy by Design* and *Privacy by Default* motivate strict regulations and restrictions on companies processing personal data. Before personal data can be processed, the company has to present the privacy policy to the user, expressing which personal data is processed for which purpose, and the user has to agree to it. Hereby, purposes must have a legal basis or have to be explicitly consented to by the user. Consent has to be given *freely* and *informed* to be valid [110, Art. 7]. Examples for invalid consent violating the *freely* condition would be pre-ticked boxes or assumed consent due to inactivity of the user [110, Recital 32]. The term *informed* indicates that the user should know about the contents of the privacy policy to a certain degree, but no further details are given for this legal requirement in the *GDPR*. In practice, consent is usually given by ticking a check-box. This individual choice of users has to be taken into account for the processing of personal data, while the user shall remain in control over the processing of his personal data such that his decision can be altered at any time, e.g., revocation of consent for a specific purpose. Thus for each user, individual consent decisions have to be managed and differentiated during processing in an efficient way.

If a privacy policy is agreed upon, the users' rights are strengthened by the *GDPR*. These rights grant the user the power to request detailed information from the company about the processing of their personal data [110, Art. 15]. Other rights enable the user to control or restrict the processing of his personal data. This includes the *Right to be Forgotten* (or *Right to Erasure*) [110, Art. 17], and the *Right to Object* the processing of personal data at any time [110, Art. 21]. Furthermore, the *Right to Data Portability*, introduced as a novelty, allows the user under specific prerequisites to receive a copy of all his personal data or let the data be transferred from one company to another. The latter is neither restricted to specific domains nor limited to companies of a specific

size. The strengthening of user rights puts companies in a strong responsibility processing personal data under the legal framework of the *GDPR*. Fines for the violation of the *GDPR* can be set up to 20.000.000 EUR or up to 4% of the total worldwide annual turnover of the preceding financial year, whichever is higher [110, Art. 83].

This poses the obligation to answer *Data Subject Right* requests. This can be quite challenging, e.g., considering a user who requests all his personal data that is processed within a company including their corresponding purposes, or a user who requests the deletion of his personal data. This requires the identification of the individual and the association of his personal data within the company, which becomes more complex with heterogeneous services and increasing size of the company due to diverse allocation of personal data.

Thus, efficient fine-grained control over the enforcement of the policy has to be considered throughout business processes to preserve privacy and minimize the required overhead. This affects not only global companies but each legal entity processing personal data, e.g., (voluntary) clubs, public services, self-employers, start-up companies, putting additional stress on them.

Lastly, when a company intends to process vast amounts of personal data for a specific purpose, it has to consider the privacy policies of each individual. Due to data collection from different sources, e.g., web-services, with varying privacy policies, a differentiation between policies has to be made during the processing of personal data. This issue increases in complexity even further, if previously introduced personalized privacy policies are assumed, such that even for the same service various policies exist. A manual differentiation and determination of a unified privacy level for a specific processing purpose is hereby not feasible if a high volume of personal data with corresponding policies has to be processed. Therefore, scalable and efficient determination of a unified privacy level based upon various privacy policies is desirable.

Once a policy has been determined, the processing of personal data should be conducted in a privacy-preserving way, such such that individuals' privacy requirements are satisfied. To enable privacy-preserving processing, various de-identification methods have been proposed including the application of pseudonymization methods, anonymization methods and privacy models. A privacy model hereby defines properties of a data-set to prevent re-identification of an individual. But various privacy models exist with different properties for different use cases, such that the selection of the appropriate method requires extensive expert knowledge and time-consuming manual interaction. Therefore, an efficient process considering privacy policies on-the-fly during the processing of high volume of personal data is a challenge for a company.

The application of de-identification methods on a data-set alters it and usually reduces the overall quality of the resulting data-set. To allow the company to reliably process personal data to derive meaningful results, the quality reduction must be limited in a way that it remains usable for the intended purpose, e.g., billing. Otherwise, to introduce an example ad absurdum, a user could define that his bank account must not be processed by an online shop after he received the order preventing the payment deduction from his account. Therefore, a dialectic approach has to be implemented to define a balance between the privacy requirements of the user and the processing requirements of the company.

Privacy should not hinder business processes but be integrated to protect individuals while their personal data is processed in a trusted way.

1.3 KEY ISSUES

Privacy has many facets which have to be considered for supporting a holistic management approach. The user, as the source of personal data, expresses his concerns about his personal data processing. But as detailed by the *Privacy Paradox*, users are, generally speaking, not willing to put additional effort into the protection of their privacy. Thus, they have to be protected by default which is realized in the EU by the legal framework *GDPR*.

This protection comprises transparent information on the processing of the personal data and control due to strengthened rights, which are expressed and regulated within the privacy policy. Both transparency and control over the processing of personal data are the main challenges. Transparency is a challenging task as privacy policies are commonly presented as legal text which makes them hard to comprehend and hard to be consulted by the user. To enable transparency, privacy languages have not only to be machine-readable but also human-readable. The second issue is how to efficiently enforce the users' control over personal data. Besides legal policies express the handling of personal data, no technical measures for preserving privacy are directly bound to such policies. Therefore, the user can only trust the company to process the personal data only for the defined purposes. But this is also an issue for companies, which intend to comply with the legal framework for which they are responsible. The processing of personal data according to the privacy policy is hereby a core challenge which has further aspects to be considered. Efficient processing of personal data is essential for companies. On the one hand, this requires the preservation of privacy according to the agreed on privacy policies of individuals. On the other hand, the utility of the data-set has to be preserved such that the data is still useful for the intended purpose. Thus, a trade-off between privacy and utility has

to be considered. Moreover, user requests regarding their *Data Subject Rights* have to be supported by technical means to support the data protection officer in his task.

The gap between legal requirements and their technical realization using privacy-preserving technologies is due to the lack of a machine-readable representation of privacy policies. The goal of this thesis is to formalize privacy policies in machine-readable format integrating legal requirements of the *GDPR* and privacy-preserving technologies and to enable the enforcement of efficient policy-based and user-guided processing of personal data.

1.4 RESEARCH QUESTIONS

To reach the goal of this thesis, the research questions (RQ) are stated in the following. Furthermore, the respective research approaches are detailed in the following.

RQ1 How to represent legal privacy policies in a machine-readable format which complies to the legal requirements of the General Data Protection Regulation in the EU while privacy guarantees are defined?

The first research questions *RQ1* requires a holistic understanding of privacy in the context of the legal framework of the *GDPR* as well as de-identification methods, used to preserve privacy, from the computer science domain. Therefore, a set of requirements is derived from both domains with the goal to express machine-readable privacy policies taking into account the concerns of users as well as the personal data processing companies.

From a legal point of view, the required contents and representation as well as the overall structural composition of a privacy policy are considered. But also privacy policy related concepts have to be considered, e.g., *Privacy by Design*, *Data Subject Rights*, and *Consent*. For example the differentiation between purposes based on consent or legitimate interest has to be considered within the privacy language to enable appropriate consent management. But also the transparent and human-readable presentation of the privacy policy has to be supported while consent negotiation is enabled.

From a privacy-preserving point of view, appropriate methods have to be identified that enable the preservation of privacy and can be used to define privacy guarantees in the context of a privacy policy. The appropriate privacy-preserving methods shall hereby provide the user with a fine-grained control over the access as well as the privacy of his personal data through the personalization of the privacy policy. Furthermore, this shall enable companies processing the personal data of several users to efficiently determine and apply the required privacy-

preserving methods while their own requirements for processing personal data are met.

To answer *RQ1*, a set of requirements for a privacy language representing privacy policies is derived. Related privacy languages are evaluated against those requirements to identify a research gap in the expression of privacy guarantees with de-identification methods in privacy languages. Based on the derived requirements, the reasoning for the *Layered Privacy Language (LPL)* and its formalization is detailed . Furthermore, LPL is qualitatively evaluated according to the given requirements including the discussion of prototype implementations.

With the definition of a privacy language that fulfils the requirements of *RQ1*, a personal privacy policy can be assumed for each user that details the individuals' privacy requirements. Therefore, the conditions under which personal data can be processed can vary for each user. Considering a data-warehouse that sources its data from various sources, the policies for processing the data can also vary. This has to be taken into account before processing a data-set, such that the individuals' privacy is preserved while the data can be processed in a meaningful way by the company. Thus, the second research requestion is:

RQ2 How can machine-readable privacy policies, expressing privacy-preserving methods, be utilized to efficiently preserve the privacy of individuals when a set of users' personal data is requested for processing?

Enabling privacy of individual users while their personal data is processed by companies is the core of the second research question *RQ2*. Assuming a data-warehouse scenario in which personalized privacy policies are stored for individual users alongside their personal data, the de-identification of the requested data has to be conducted for each request due to varying combinations of data records with their corresponding privacy policies. The personal privacy policies may vary hereby regarding the consented purposes or defined de-identification methods, which may be introduced by the individuals' privacy requirements. These individual privacy requirements can be altered at any time due to the control of the user which results in dynamic privacy requirements over time which have to be reconsidered for each processing of personal data.

Furthermore, the interplay of the de-identification methods has to be considered, because they alter the original values and therefore affect the quality of the requested data-set. Thus, the de-identification process chain has to be carefully crafted considering the properties of the de-identification methods. The automatic determination of appropriate de-identification methods from various privacy policies, iff different methods are defined, is additionally subject to research and has to be integrated within the process chain.

The processing of possibly millions or billions of privacy policies introduces additional overhead compared to the sole application of de-identification methods, because privacy requirements have to be derived from the set of personalized privacy policies. The efficiency of this de-identification process chain is quantitatively evaluated. The evaluation includes a detailed run-time performance analysis and how the data-set as well as the privacy policies affect the run-time. Moreover, the impact of personalization of policies by users on the de-identification process is quantified.

To answer *RQ2*, the *Policy-based De-identification (PD)* process is introduced, which combines and integrates pseudonymization, personal privacy anonymization, and privacy models, using LPL as a basis. Furthermore, the process is evaluated in detail with a focus on efficiency both with and without the introduction of personal privacy policies.

1.5 KEY CONTRIBUTIONS

The key contributions of this thesis include the proposal of the *Layered Privacy Language (LPL)* (see Chapter 5) and the *Policy-based De-identification (PD)* process (see Chapter 6).

LPL models privacy policies while it incorporates various requirements detailed in Chapter 2 – *R1 Privacy Policy Structure*, *R2 Legal Compliance*, *R3 Human-readability*, *R4 Access Control*, *R5 De-identification Capabilities*, and *R6 Provenance*. Although some of these requirements have been addressed by related works (see Chapter 4), they have not been brought together in one privacy language. Especially, the fulfilment of the requirement *R5 De-identification Capabilities* is emphasized in LPL, allowing the definition of anonymization and pseudonymization methods as well as privacy models. The inclusion of such methods within privacy languages has been understudied in related works, although they are a valuable asset in the definition of privacy that extends classical approaches for privacy languages based on access control. In addition, a personalized LPL instance is intended for each user, which allows the distinction of personalized privacy settings, i.e. each user can decide for which purpose what data is processed by whom with which de-identification settings. Combined with the fulfilment of the remaining requirements, LPL is able to represent privacy policies compliant to the GDPR that are intended to be presented to the user, enable provenance, and facilitate the privacy-preserving processing of personal data due to access control and de-identification. Thus, LPL is intended as a holistic approach to model, present and process privacy policies.

The second main contribution – the PD process – demonstrates how various distinct (personalized) LPL privacy policies can be used to efficiently determine and apply the de-identification methods applied on the data-set while the individuals' privacy settings are guaranteed (see

Chapter 6). This challenge is detailed according to a data-warehouse scenario in which personal data from various sources, i.e. with various LPL privacy policy settings, are combined and requested for being processed for a specific purpose. Therefore, two main challenges had to be overcome: First, the sequence of methods applied for pseudonymization, personal privacy anonymization, and privacy models has to be set to guarantee privacy while the data quality is preserved as best as possible. Second, various different LPL privacy policies have to be accounted to during the PD process, thus for each of the de-identification methods – pseudonymization, personal privacy anonymization, and privacy models – algorithms are proposed to identify and apply the respective de-identification methods while the requirements of all LPL policies are fulfilled. In addition, two distinct algorithms – *Minimum Anonymization (MA)* and *Global Minimum Anonymization (GMA)* – are proposed to realize personal privacy anonymization. Compared to the isolated usage of de-identification methods for specific use cases in the literature, the PD process highlights how de-identification methods can be combined in an approach to fulfil personal privacy requirements as well as preserve the privacy of a whole data-set based on personalized LPL privacy policies. Furthermore, it is shown that the core-algorithms of the PD process add a relative minor run-time overhead compared to the usage of privacy models, thus are efficient. In addition, the usage of the GMA algorithm in combination with privacy models demonstrates that the baseline run-time of privacy models can be significantly undercut in a scenario using personal privacy settings (see Chapter 7).

1.6 ORGANIZATION OF THE THESIS

The remaining of the thesis is structured as follows: In the following Chapter 2, requirements for a privacy language that expresses privacy policies are derived. Chapter 3 details background information on the de-identification methodology that is considered throughout the thesis. Related work is detailed in Chapter 4 classifying related privacy languages according to the requirements for a privacy language in Chapter 2 showing a research gap. Furthermore, the approach of this thesis is positioned according to related works. Chapter 5 details the reasoning of the *Layered Privacy Language* and formalizes the proposed privacy language. Additionally, the fulfilment of the requirements for a privacy language by LPL is detailed and discussed. The *Policy-based De-identification (PD)* process is detailed in Chapter 6 based on a data-warehouse scenario. The evaluation of the run-time efficiency of the PD process and the impact of personal privacy settings on the run-time of the PD process are detailed and discussed in Chapter 7 for which a suitable test framework is introduced. Lastly, the work is concluded and an outlook for future work is given in Chapter 8.

2

PRIVACY LANGUAGE REQUIREMENTS

This chapter details the requirements for a privacy language that expresses privacy policies. In the following, a set of requirements for a *GDPR*-compliant privacy language is derived, which enables privacy-preserving processing of personal data based Gerl et al. [121]. A privacy language can be denoted as a specialization of a domain specific language (DSL) in the context of privacy. Privacy itself is a complex and vast field, which is not only tackled in computer science but also in social and legal sciences. Several aspects have to be considered for a privacy language that represents and enforces legal privacy policies. On the one hand, the legal view on privacy, i.e. the *GDPR* [110], has to be considered to comply with the current legal privacy framework in Europe. On the other hand, available technologies and methods to realize privacy, i.e. give privacy guarantees, have to be considered. Therefore, a set of requirements – *R1 Privacy Policy Structure*, *R2 Legal Compliance*, *R3 Human-readability*, *R4 Access Control*, *R5 De-identification Capabilities*, and *R6 Provenance* – is defined and detailed in the following.

2.1 PRIVACY POLICY STRUCTURE

The first requirement for a privacy language that represents privacy policies is denoted as:

> *R1* The base structure of a policy language has to match the structure of legal privacy policies.

Hereby, a legal privacy policy defines, at its core, purposes for which personal data is processed. Furthermore, it is specified from which entity the personal data originates and by which entity the personal data is processed. It has to be noted that, legally, privacy policies are voluntary regulations that a company follows, while a privacy notice, i.e., Art. 12 - 14 [110], is required by all companies processing personal data and can be embedded in privacy policy. For simplicity the terms 'privacy policy' and 'privacy notice' are considered to define the same in the remaining of this work.

Thus, the following core elements can be defined for a privacy policy:

- **Privacy Policy**: Denotes all purposes of processing of personal data of an individual.
- **Data Source**: The individual from which the personal data originates.
- **Data Recipient**: The entity which processes the personal data.
- **Purpose**: Denotes the reason and extent of the processing of personal data.
- **Data**: Denotes the personal data that is subject to processing.

Therefore, a *Privacy Policy* denotes the processing of personal *Data* of a *Data Source* for *Purposes* by *Data Recipients* (see Figure 2.1). Thus, a privacy policy regulates what personal information is processed by whom for which reason.

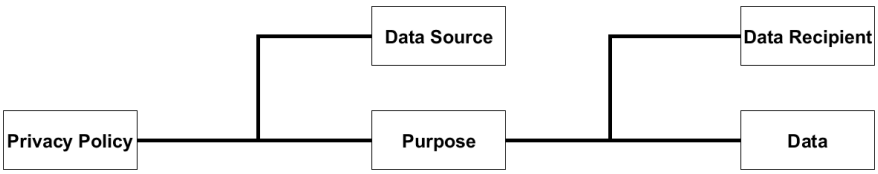


Figure 2.1: Core structure and elements of a privacy policy.

This core structure of a privacy policy is not only specific to the *GDPR*, but a generic description of privacy rules based upon the upon the individuals' perception of privacy. Privacy is perceived by individuals as a time and space in which they can be autonomous and have a limited and protected communication [274]. Hereby, privacy is interpreted as the dynamic process which gives or limits access to (personal) information with the goal to achieve balance between actual and desired privacy [13]. This interpretation of privacy has been extended by Petronio [211] with its *Communication Privacy Management (CPM)* theory. The CPM theory states that privacy is a range of complete openness to complete closeness, which is regulated by people via a dialectic approach. Initially personal information is owned by the individual (data source) itself, but it can be shared and distributed to others (data recipients) such that the ownership is distributed to many [211] [212]. If ones' privacy is violated, then corresponding privacy rules are adopted by the individual, e.g., information is no longer shared with specific individuals [67] [68]. Therefore, core elements of a privacy policy can be matched to the perceived privacy according to the CPM theory.

Legal frameworks follow similar definitions of privacy. The *GDPR* denotes the structure of privacy policies in Art. 12 - 14 [110] and

uses similar legal terms, i.e. *Data Subject* [110, Art. 4(1)], *Purpose of Processing* [110, Art. 5], *Personal Data* [110, Art. 4(1)] and *Recipient* [110, Art. 4(9)]. The purpose-based approach is hereby one of the core principals of the *GDPR* for the processing of personal data [110, Art. 5(2)]. Hereby, the *Controller* [110, Art. 4(7)] is responsible and accountable for the processing of personal data. The generic notation of the previously introduced core structure of a privacy policy slightly deviates from the notation of the *GDPR* to indicate its compatibility with other legal frameworks. Comparable approaches can be found in other legal frameworks, e.g., in the *Health Insurance Portability and Accountability Act (HIPAA)* [71] [219] which denotes national standards for the protection of health information. The *HIPAA* covers health plans and health care providers in the USA. Furthermore, the *California Consumer Privacy Act (CCPA)* [63] follows several principles of the *GDPR* and intends to strengthen the privacy rights of individuals. Unlike the *HIPAA*, the *CCPA* applies to all businesses that collect personal data. Its key aspects are, that individuals are informed about the collection of personal data and personal data trading including the right to refuse the sale of personal data. Furthermore, individuals have the right to access their personal data.

Therefore, the detailed core structure of privacy policies follows both the individuals' perception of privacy as well as the legal requirements of the *GDPR* for privacy policies, but should also be applicable for other legal frameworks.

2.2 LEGAL COMPLIANCE

The next requirement for privacy languages that express privacy policies is:

> *R2* A privacy language has to comply with the intended legal framework.

Considering the scope of the thesis, the creation of a privacy language that complies with the legal framework of the *GDPR* is the goal. The *GDPR* denotes *Data Subject Rights* in its third chapter, which information has to be provided to the *Data Subject*. Furthermore, it details the different rights of the *Data Subject*. In the following, this chapter of the *GDPR* is analysed and several requirements for privacy languages are derived.

2.2.1 *Information to be Provided*

General provisions are defined in Art. 12 *GDPR*, which enable the *Data Subject* to exercise his rights [110, Art. 12]. Furthermore, Art. 13

denotes the information that has to be provided to the *Data Subject* iff his data is collected [110, Art. 13]. Similarly structured is Art. 14 *GDPR*, denoting the information that has to be provided iff personal data is not collected directly from the individual, e.g. personal data is collected by a third party [110, Art. 14]. Although none of the above explicitly denotes privacy policies as the instrument to fulfil the legal requirements, in practice privacy policies are the de-facto standard for it. In the following, relevant legal requirements are denoted (see Table 2.1) which are derived from Art. 12 - 14 of the*GDPR* by an in detail analysis of Gerl and Pohl [119].

First, it is stated that the information provided to the *Data Subject* for both the privacy policy, but also the later detailed *Data Subject Rights* [110, Art. 15 - 22], have to be provided in an easy and plain language; this is especially important for children [110, Art.12 (1) Sentence 1]. Furthermore, the privacy policy can be provided in a written or electronic form, but an oral presentation of the information is also considered [110, Art.12 (1) Sentence 2].

Unless the *Controller* is not able to identify the *Data Subject*, the *Data Subject Rights* have to be implemented [110, Art. 12 (2)]. Further details on a required response time [110, Art. 12 (3)] and the protection of the *Controller* from excessive requests [110, Art. 12 (5)] are given.

Lastly, Art. 12 denotes that the information, which has to be provided according to the Art. 13 and 14 requirements, can be provided in combination with standardised icons. Those *Privacy Icons* shall hereby provide an overview over the intended processing. Iff they are provided electronically, they should be machine-readable [110, Art. 12 (7)].

Table 2.1: Legal requirements of information to be provided by privacy policies according to Art. 12 - 14 [110, Art. 12 - 14] [119].

GDPR	
Article	**Requirement**
Art. 12(1) Sentence 1	Clear and Plain Language
Art. 12(1) Sentence 2	Written or Electronic Information
Art. 12(2)	Data Subject Rights Realization
Art. 12(7)	Standardised Icons
Art. 13(1)(a), Art. 14(1)(a)	Contact Details of Controller
Art. 13(1)(b), Art. 14(1)(b)	Contact Details of Data Protection Officer (DPO)
Art. 13(1)(c), Art. 14(1)(c)	Purpose and Legal Basis
Art. 13(1)(d), Art. 14(2)(b)	Legitimate Interest
Art. 14(1)(d)	Categories of Personal Data
Art. 13(1)(e), Art. 14(1)(e)	Recipients of Personal Data
Art. 13(1)(f), Art. 14(1)(f)	Third Country Transfer
Art. 13(2)(a), Art. 14(2)(a)	Storage Period
Art. 13(2)(b), Art. 14(2)(c)	Information: Data Subject Rights
Art. 13(2)(c). Art. 14(2)(d)	Information: Withdraw Consent
Art. 13(2)(d), Art. 14(2)(e)	Information: Lodge a Complaint
Art. 13(2)(e)	Information: Required Data
Art. 14(2)(f)	Source of Personal Data
Art. 13(2)(f), Art. 14(2)(g)	Automated Decision-Making

Due to the similar structure and content of Art. 13 and Art. 14 requirements can be described combined in the following. The privacy policy shall provide the identity and contact details of the *Controller* (or several if *Joint Controllers* [110, Art. 26]) [110, Art. 13 (1)(a), Art. 14 (1)(a)] and the responsible *Data Protection Officer (DPO)* [110, Art. 13 (1)(b), Art. 14 (1)(b)] have to be provided.

The *Purposes of Processing* and their legal basis have to be given [110, Art. 13 (1)(c), Art. 14 (2)(c)], whereas the legitimate interests have to be defined if pursued by the *Controller* or a third party [110, Art. 13 (1)(d), Art. 14 (2)(b)]. Furthermore, the *Data Subject* has to be informed about the data categories of his collected personal data [110, Art. 14 (1)(d)]. Additionally, the *Recipients* of such data [110, Art. 13 (1)(e), Art. 14 (1)(e)] have to be stated. Iff data is transferred to a *Third Country*, a country which does not fall under the legislation of the *GDPR*, the destination of the transfer as well as implemented *Safeguards* to ensure protection have to be made transparent [110, Art. 13 (1)(f), Art. 14 (1)(f)]. The duration of the storage of personal data has to be provided,

i.e. when personal data is deleted [110, Art. 13(2)(a), Art. 14(2)(a)]. The *Data Subject* has to be further provided with information on his right to conduct *Data Subject Rights* [110, Art. 13(2)(b), Art. 14(2)(c)], withdraw *Consent* [110, Art. 13(2)(c), Art. 14(2)(d)], and lodge a complaint [110, Art. 13(2)(d), Art. 14(2)(e)]. Furthermore, the *Data Subject* has to be informed about whether or not the provision of personal data is required for the purpose [110, Art. 13(2)(e)]. The source of personal data has to be defined whereas it has to be denoted if this source is publicly available [110, Art. 14(2)(f)]. Lastly, the *Data Subject* has to be informed if automated decision-making is performed [110, Art. 13(2)(f), Art. 14(2)(g)].

Thus, a wide variety of information has to be encapsulated within the privacy policy to inform the *Data Subject* about the processing of his personal data. A privacy language, which intends to comply with *GDPR*, has therefore to consider and model several complex requirements (see Table 2.1). Furthermore, the implementation of *Data Subject Rights* has to be considered because their fulfilment can be supported by information that corresponds to the already detailed information required by Art. 12 - 14 *GDPR*.

2.2.2 *Data Subject Rights*

Data Subject Rights are intended to empower the *Data Subject* to be in control of the usage of his personal data. Although *Data Subject Rights* should rather be seen as processes and therefore do not explicitly have impact on a privacy language, they should also be considered due to their dependency to the previously defined requirements.

For example, the *Right of Access by the Data Subject* denotes that the *Data Subject* has the right to receive the information if his personal data is processed by the *Controller*. If this is the case, the *Data Subject* shall be granted access to the personal data as well as additional information, e.g., *Purposes of Processing*, categories of personal data, *Recipients*, or the information to the right to lodge a complaint. The information shall be provided as a copy or by electronic means in a commonly used electronic form [110, Art. 15]. It can be observed, that the required information that has to be provided to the *Data Subject* is also postulated by Art. 13 and 14 of the *GDPR*. A privacy language containing all this information also contains personal data on the *Data Subject*. Therefore, for each *Data Subject*, an instance of the privacy language, i.e. personal privacy policy, is correlated. Furthermore, the privacy language can be queried to fulfil the *Data Subject Right* request. Because, the privacy policies contain personal data, it has to be ensured that the requesting entity has to be identified beforehand to authorize access to his personal data, otherwise private information may be inferred.

Table 2.2: Overview of *Data Subject Rights* of Art. 15 - 22 *GDPR* [110, Art. 15 - 22].

Article	Data Subject Right
Art. 15	Right of Access by the Data Subject
Art. 16	Right to Rectification
Art. 17	Right to Erasure
Art. 18	Right to Restriction of Processing
Art. 19	Notification Obligation
Art. 20	Right to Data Portability
Art. 21	Right to Object
Art. 22	Automated Individual Decision-making

The creation of adequate processes is hereby not trivial, although it might seem so for the *Right of Access by the Data Subject*, as other rights require a deeper integration into business processes of the *Controller*, e.g., the *Right to Restriction of Processing* [110, Art. 18], or even some standardization between *Controllers*, e.g., the *Right to Data Portability* [110, Art. 20]. An overview over all *Data Subject Rights* is given in Table 2.2, which should be considered for the design of a privacy language to enable or at least facilitate their implementation.

2.3 HUMAN-READABILITY

Privacy languages, commonly designed to be machine-readable, that intend to represent privacy policies require to be understandable by common users [120]. This is not only required by the *GDPR* [110, Art. 12], but is a general concern to close the gap between the expression of policy statements and their enforcement. Otherwise, additional processes have to be put into place to synchronize the human-readable privacy policy with the enforcing privacy language, e.g., for updates in the privacy policy, introducing overhead and error-proneness. Thus, it can be concluded that systematic usage of privacy languages for privacy policies has the requirement:

R3 A privacy language has to be human-readable.

Several aspects related to the term 'human-readable' are considered essential for a privacy language, representing both the textual representation of the privacy policy as well as the usage of pictograms for visual stimuli, namely *Privacy Icons*.

2.3.1 *Textual Representation*

The textual representation of the privacy policy shall facilitate the user to understand its contents. Therefore, it is necessary that the privacy language incorporates the possibility to define human-readable text.

But this shall not only be done by a unique occurrence, e.g., adding one text element for representing the whole policy, but thoughtfully integrated within required elements of the privacy language. In other words, several elements of the privacy language should have human-readable texts such that not only the texts themselves but also the structure of the privacy language improve the understandability. Therefore, the privacy language structure shall be enhanced by several human-readable texts. Additionally, internationalization is a concern and has to be tackled, such that several human languages can be supported.

The key intention for providing human-readable text within the privacy language is transparency and understandability, which has in the context of privacy policies several facets. Bertino et al. [36] denote five key dimensions:

- *Record Transparency*: The user has to be informed about all aspects of the collection of personal data, e.g., what data is collected and by whom.
- *Use Transparency*: The user has to be informed about the usage of his data, including the purpose and recipients of data, and applications processing the data.
- *Disclosure and Data Provisioning Transparency*: The user has to be informed about the transfer of his data to other entities, e.g., companies, and the terms of the transfer. Therefore, it has to be communicated if the data is sold and what mechanisms are used for the transfer of data, e.g., an encrypted connection.
- *Algorithm Transparency*: The user has to be informed about the algorithms processing his personal data for, e.g., recommendations or automated decisions.
- *Law and Policy Transparency*: The user has to be informed about the available laws and regulations applicable to his personal data.

This complex topic is also subject of the *GDPR*, which covers various aspects as detailed before [110, Art. 12 - 14].

Furthermore, *Article 29 Working Party* recommends the usage of the *Layered* approach in digital environments to ensure transparency for the *Data Subject* (see Figure 2.2). This approach is intended to provide the user with several linked layers of information, instead of providing all information at once, trying to avoid to overwhelm the *Data Subject* [18]. This shall make privacy information more accessible. The first

layer can hereby provide concise information on the key elements of the privacy policy, e.g., the purposes of the processing. Furthermore, the first layer should focus on information that could surprise the user or has the most impact on the processing of the personal data. Therefore, the most important information for the user is presented in the first layer. The second layer on the other hand, would give more details on specific elements, whcih have lesser importance for the user, to cover all legal requirements of Art. 13 and 14 [124] [99]. This approach is not a novelty within the context of *GDPR*, but has been already proposed alongside other recommendations for the design of privacy policies [65] [190].

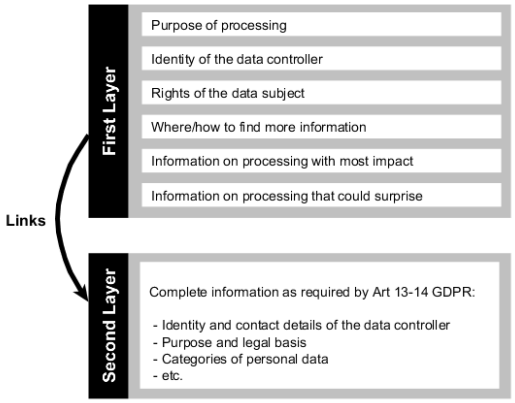


Figure 2.2: Visualization of the *Layered* approach as proposed by the *Article 29 Working Party* [18] [124].

2.3.2 *Privacy Icons*

Next to textual representation of the required information, also standardised icons are considered by the *GDPR* for transparency [110, Art. 12(7)]. They are intended to give the *Data Subject* an overview on the processing of his personal data. Furthermore, privacy icons have to be machine-readable. A privacy language should hereby enable the usage of privacy icons or incorporate them as detailed by Gerl [112].

Currently, no privacy icon set is officially standardised for the *GDPR*, but several approaches have been proposed. The 'official' proposal of privacy icons has been evaluated with a negative result, showing that they are not suitable for broad public use [213]. But other privacy icon sets have been proposed, e.g., *DaPIS* [234], *Privacy Icons Project* from Mozilla [189], or *Iconset for Data-Privacy Declarations* [183]. Furthermore, methodologies have been proposed to develop privacy icons [206] [233]. For the design of privacy icons, or privacy policies in general, the target user group has to be considered. Therefore, dedicated design decisions have to be taken for each of the user groups, e.g., children [84].

Because no standardised set of privacy icons is given yet, showing that the design of such icons is not trivial, no detailed requirement for the incorporation of privacy icons in privacy languages can be given. It can be noted though, that privacy icons should give an overview over the processing of personal data for transparency purposes.

2.4 ACCESS CONTROL

Considering the before mentioned *Communication Privacy Management (CPM)* theory, which states that privacy is perceived as a range from complete openness to complete closeness [211], mechanisms for controlling who has access to personal data have to be incorporated in a privacy language. Access control mechanisms allow exactly this, thus the requirement states that:

> *R4* A privacy language has to enable purpose-based access control.

A privacy language should hereby consider both the source, e.g., a user, and the recipient, e.g., a company, as identifiable entities. The data flow should be hereby controllable between any combinations of such entities, e.g., user and company, user and user, or company and company. Furthermore, access to personal data should be purpose specific, e.g., a company can use the phone-number of Bob for emergency contacts but not for advertisement.

Therefore, fine-grained access control is required for authenticating and authorizing the requesting entity to access personal data. Similar problem statements have been worked on various other domains, e.g., privacy policies for mobile devices [59], access control in cloud [236] and IoT [103] [278] environments. *GDPR* requires a purpose-based processing of personal information [110], thus a differentiation is necessary. This has been addressed for relational databases [62]. Especially in the domain of health care, in which very sensitive and private information is stored and processed, purpose-based access control mechanisms are required. Therefore, hippocratic database systems have been proposed [9] [39], which goal is to enable privacy [122]. Privacy meta-data, which could be expressed using a privacy language, is hereby utilized to strengthen the access constraints to the data [7].

2.5 DE-IDENTIFICATION CAPABILITIES

Next to access control mechanism, privacy can be guaranteed by de-identification methods. De-identification hereby relates to the alteration of data in such a way that the corresponding individual cannot be

identified. Considering the *Communication Privacy Management (CPM)* theory [211] again, the de-identification of data allows the *Data Subject* to define in a fine-grained manner the condition of the information that is shared with others. For example Bob can specify that his friends know his exact address, while his colleagues know only the city he lives in. Therefore, partial reduction of information can be achieved. Thus, the requirement states that:

> *R5* A privacy language has to define de-identification methods.

De-identification can be achieved by two basic approaches, anonymization and pseudonymization.

- Pseudonymization replaces the original value with a *pseudonym*, which can be related or unrelated to the original value, for later authorized re-identification [196] [217] [216] [95] [201].
- Anonymization hides the information of the original value to a certain degree to preserve its semantic, so that on the one hand the privacy of the individual is preserved and on the other hand the information is still useful, e.g., for data mining [226] [181] [57] [283].

GDPR mentions pseudonymization as a viable method for securing the processing of personal data, but it does not restrict the usage of other methods [110, Art. 4 (5), Art. 32, Recital 28]. Furthermore, it is noteworthy that *Pseudonymization* according to *GDPR* specifies that personal data is processed that is no longer associated with the individual [110, Art. 4 (5)], which can be both achieved by anonymization and pseudonymization as defined before. Real anonymization (no risk or re-identification) of personal data is questioned due to the various sources of publicly available data that can be utilized to identify individuals [271]. But, several advanced methods have been based upon anonymization and pseudonymization, which offer various properties to express privacy, limit the risk of re-identification, and are used in practice. A background on relevant de-identification methods and concepts is detailed in Chapter 3.

2.6 PROVENANCE

When data is transferred to other *Controllers* the origin, i.e. the *Data Subject*, of the personal data may be lost resulting in the loss of the rights of the *Data Subject* due to the missing proof of his ownership. Thus, the requirement states that:

R6 A privacy language has to enable provenance.

After personal data is transferred, the origin of the personal data should be identified, such that *Data Subject Rights* of the corresponding user can be preserved and checked. Therefore, it is necessary that the policy remains linked to the personal data. A *Controller* must be accountable for the handling of personal data, especially if it is traded or transferred to third parties, e.g., in cloud context scenarios [154] [170] or for scientific purposes [81] [83] [82]. Hereby, two use cases – *Data Processing* and *Data Production* – have to be considered, which are detailed in the following.

2.6.1 *Data Processing*

When personal data is processed as is, e.g., transferred or sold to a third party, it has to be ensured that the data source, as well as the agreed privacy policy, is also transferred. The data source, e.g., the user expressed by the privacy policy, has to be identifiable to claim his *Data Subject Rights* [110, Art. 12]. The privacy policy has to be transferred (and enforced) to guarantee that the personal data is only processed according to its purposes, e.g., a policy defines that the personal data may only be used for marketing purposes by a specific recipient. Assuming this policy is not transferred to this recipient, the recipient may process the personal data for other unauthorized purposes.

Furthermore, the agreed policy may be further refined in an additional policy, e.g., limiting the processing rights for third parties. The refinement of the privacy policy must always be within the scope of the previously agreed upon policy, therefore refined policies and their verification and validation processes is required [110, Recital 50].

2.6.2 *Data Production*

However, personal data is not only transferred, but also merged and processed producing new data which can be used for advanced processes, e.g., decision making, statistics or machine learning. Hereby, the data may be processed from various data sources each with their own privacy policy.

Thus, the combination of different policies and their validation have to be considered while new data values are produced based on them. The goal is hereby to be able to trace the data flow back to individual sources of data, to allow verifiable results and processes. Furthermore, attacks which address the data to be processed, e.g., poisoning attacks in machine learning [23], can be mitigated or unveiled.

In the following, de-identification methodology is detailed, which is considered for the *R5 De-identification Capabilities* of the *Layered Privacy Language* and the *Policy-based De-identification* process.

3

DE-IDENTIFICATION BACKGROUND

This chapter gives an overview on state-of-the-art de-identification methodology. Various concepts and approaches are detailed that are used for privacy-preserving processing of personal data. De-identification methods are applied on the personal data of a user with the goal to either hide his identity or any correlation between the personal data and the user. Therefore, others cannot have access to the identity of the user but cannot relate any personal information to it. Or personal information is available but cannot be correlated to the user. In general, it can be differentiated between anonymization and pseudonymization methods. Pseudonymization, the replacement of original values with a substitute value, enables the restoration of the original value if the correlation between the original value and substitute value is preserved. Anonymization, the partial hiding of original value, is not intended for the later restoration of the original value. While pseudonymization be part of an overall *GDPR* compliance strategy. In contrast to anonymization, pseudonymized data remains affected by *GDPR* obligations, because the link to the original value remains. Thus, companies should carefully balance the usage of pseudonymization and anonymization [133].

In the following, the notion of personal data is introduced from a computer science perspective before an overview on de-identification methods is given including anonymization, privacy models, pseudonymization and personal privacy.

3.1 PERSONAL DATA CATEGORIZATION

In general, data is classified in the privacy-preserving domain into four categories, i.e. privacy groups (see Table 3.1) [257] [266]:

- *Explicit Identifiers (*EI*)* defines attributes that identify a user uniquely. Examples for EI are the passport ID (identifier), or social security number.
- *Quasi-Identifiers (*QI*)*, as indicated before, defines attributes which in combination with other QI enable the identification of a user. Examples of QI are the IP-address, postal-code, birthday, age, gender and other demographic information. QI are often

publicly available for example in phone books, voters databases or other sources.

- *Sensitive Data (*SD*)* defines attributes which are confidential to the user. Examples for SD attributes are health data, financial data or other information that should not be correlated to the user at any cost depending on the purpose.

- *Non-Sensitive Data (*NSD*)* defines attributes which do not identify the user nor are sensitive for the user. Therefore, NSD attributes are all attributes which cannot be classified as any of the other data categories EI, QI, or SD.

The *GDPR* defines personal data as any information related to an identified or identifiable natural person, whereas an identifiable natural person is one who can be identified, directly or indirectly [110, Art. 4 (1)]. Furthermore, *GDPR* defines special categories of data which are by default prohibited from processing, unless special requirements are given, e.g., consent is given by the user or the processing is required for the public interest like research. The special categories of data cover for example personal data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, genetic and biometric data, health data or data concerning sexual orientation [110, Art. 9].

Comparing the legal definition of personal data [110, Art. 4 (1)] with the previously given classification of personal data, it can be observed that the *GDPR* addresses with its definition both EI and QI classified data. EI attributes correspond to personal data that identifies persons. Furthermore, QI attributes correspond to personal data that indirectly identifies persons. Special categories of data correlate to SD attributes, but may also be associated with QI attributes as they can be used to identify a person, e.g., the gender. Lastly, NSD attributes are not directly covered by *GDPR*, because they do not fall under this legal framework. Moreover, it is stated in the recitals that the principles of data protection do not affect anonymous information. Anonymous information is described as information that does not directly nor indirectly identify a natural person. Additionally, anonymous information is described as personal data that is altered in such a way that it can no longer be attributed to a person. According to the *GDPR*, this does not apply to pseudonymous data, if the mapping of the pseudonym and original value is preserved, because it is possibly identify the person with additional information [110, Recital 26].

Table 3.1: Example of a multidimensional data-set detailing various attributes classified as explicit identifier (EI), quasi-identifier (QI), sensitive data (SD) and non-sensitive data (NSD). Each row represents personal data of a user for a lottery. The NSD 'Lucky#' is randomly assigned to the person.

EI		QI			SD	NSD
ID	*Name*	*Sex*	*Age*	*Postal-Code*	*Salary*	*Lucky#*
1	Alice	F	27	94032	30.000	1234
2	Bob	M	33	94036	35.000	1337
3	Charlie	M	29	94405	28.000	404

Therefore, data-set scheme T is defined as a set of attributes.

$$T = (a_1, \cdots, a_n) \tag{3.1}$$

Each attribute can be assigned to one of the previously defined privacy groups, such that each privacy group is a set of attributes.

$$a_i \in (EI|QI|SD|NSD) \tag{3.2}$$

Therefore, an alternate representation of a data-set scheme is a tuple of all attributes of EI, QI, SD and NSD.

$$T = (EI, QI, SD, NSD) \tag{3.3}$$

Different kinds of data exist. In the following, different data types are presented alongside with with their properties that have to be considered during de-identification [266].

3.1.1 *Multidimensional Data*

Multidimensional data, also referred to as relational data, is the most common format of data. Considering Table 3.1 as an example for multidimensional data, the properties are that each record, i.e. row in the table, refers to one user. Each attribute, i.e. column of the table, is related to the user and is made up of attributes of the previously described privacy groups EI, QI, SD and NSD. Furthermore, it has to be considered that each record is independent of other records of the data-set. Additionally, the de-identification of an attribute of a record does not affect the other attributes of the record. Due to the common nature of multidimensional data it has been closely paid attention by researchers in privacy-preservation. The challenges have been hereby, the differentiation between QI and SD, the high dimensionality of the data, the clusters of sensitive data, and finding the trade-off between privacy and utility of the data [257] [177] [238] [4] [267] [5].

3.1.2 *Transaction Data*

Databases holding transactions of users, e.g., diagnosis results of patients in a hospital or sales of an online shop, can be referred to as holding transactional data. This type of data usually has high dimensionality and shows sparsity of data. High dimensionality can hereby be caused due to the various products an online shop may offer and sparsity is caused due to the customers only buying a small subset of the available goods (see Table 3.2). These properties prevent the application of privacy-preserving methods for multidimensional data [8] [186] [266].

Table 3.2: Example of data-set of transactional data for an online shop selling french food specialities. Each transaction is represented by a '1', while no transaction is denoted by a '0'. For example Alice bought 'Baguette' and 'Cheese'.

Name	*Baguette*	*Croissant*	*Cheese*	Coffee	*Wine*
Alice	1	0	1	0	0
Bob	0	0	0	0	1
Charlie	0	1	0	1	0

3.1.3 *Longitudinal Data*

Continuous studies are the main sources of longitudinal studies, which can be found in, e.g., the health care domain. For example diabetic patients have to measure their glucose level over a specific time frame for adjusting their insulin dose (see Table 3.3). Such data helps to identify changes in the behaviour of the patient or a disease process. A longitudinal data-set usually contains EI, QI, and SD attributes. The data is hereby correlated and clustered due to repeated measurements of the same patient, while temporally ordered. These characteristics have to be preserved during the de-identification process while disclosure of identity and attributes is prevented [107] [242] [266].

Table 3.3: Example of data-set of longitudinal data in the context of continuous blood glucose measurements. Each row represents a measurement for a patient representing date and time of the measurement, as well as the measured blood glucose level. For example, Bob measured his blood glucose level six times over the course of two days.

Name	*Date*	*Time*	*Diabetes*	*Blood Glucose mg/dL*
Bob	01.03.2019	10:00	Type 1	110
Bob	01.03.2019	14:00	Type 1	95
Bob	01.03.2019	19:00	Type 1	130
Alice	01.03.2019	20:00	Type 2	110
Bob	02.03.2019	11:00	Type 1	100
Bob	02.03.2019	14:00	Type 1	130
Bob	02.03.2019	18:00	Type 1	110

3.1.4 *Graph Data*

Graphs model relationships between different entities. A graph hereby consists of a set of vertices and a set of edges (pairs of vertices) denoting a relationship. Graphs can be found in many domains, e.g., networking, transportation, telecommunication and social networks. Social networks connect many users which share their personal data (see Figure 3.1). Thus, social networks are a valuable source of information for companies analysing user behaviour and preferences. Therefore, companies create own accounts on social networks to connect to users and receive feedback and opinions. The created data is used for data mining and analysis to provide insights. This poses a threat to the privacy of users, e.g., the data may be used to identify individuals. Therefore, graph data has to be de-identified. It has been shown that more complex graph data makes it easier to identify a user, which has to be considered creating de-identification methods for graph data. To preserve the privacy of graph data the following aspects have to be considered while the properties of the graph are preserved, e.g., path length, betweenness, and closeness. First, the identity of the user may be disclosed. Second, the link between users may be disclosed which is highly sensitive information as relationships between users are expressed. Lastly, each node in the graph representing a user has sensitive content assigned to it which can be used to identify the person, e.g., gender or other demographic information [244] [176] [281] [187] [283] [142].

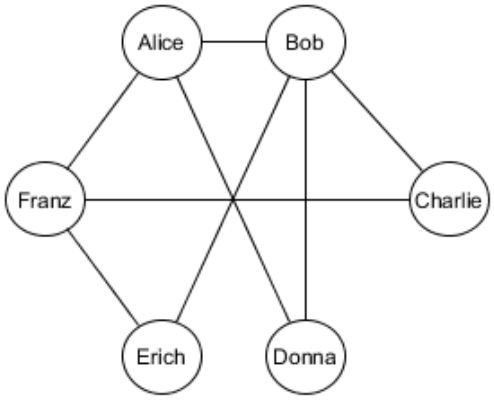


Figure 3.1: Example of graph data in the context of a social network. A 'friend' relationship is expressed between the different users.

3.1.5 *Time Series Data*

Time series data is the result of taking regular measurements, e.g., measure the temperature every day in different cities or weekly sales of a company. But also other domains like smart grid produce time series data. Compared to longitudinal data, which also have a temporal order, time series data is distinguished by the measurement of a single variable on a regular basis. Furthermore, time series data has a high dimensionality compared to longitudinal data, which is continuously growing. Assuming the regular measurement of the yearly income of users (see Table 3.4) it can be seen that in comparison to multidimensional data, the attributes of the record have a semantic relationship. For example the difference in salary, relative increase or decrease, is of interest. This dependency of attributes has to be considered for the de-identification process. This dependency is to retain the statistical properties of the data-set while privacy is preserved, for which the high dimensionality and possible large size of the data-set has to be considered [245] [144] [284] [104] [228] [105].

Table 3.4: Example of data-set of time series data regularly measuring the yearly salary of individuals. Each row represents the regular measurements for one person. The dimensionality of the data-set is growing regularly with each measurement, e.g., each year in this example.

ID	*Name*	*Postal-Code*	*Salary 2016*	*Salary 2017*	*Salary 2018*	*Salary 2019*
1	Alice	94032	25.000	27.000	29.000	30.000
2	Bob	94036	30.000	30.000	35.000	35.000
3	Charlie	94405	26.000	31.000	31.000	28.000

3.1.6 *Summary*

The classification of personal data in the privacy groups EI, QI, SD, and NSD has been presented. Furthermore, different typical types of data-sets have been presented showing that next to the preservation of privacy, the properties of the data-set have also to be preserved for further processing (see Table 3.5). For the remaining of this thesis, the focus is on multidimensional data, or relational data, as it is the most common type of data. Next, de-identification methods including anonymization, privacy models, and pseudonymization are detailed in the following.

Table 3.5: Summary of personal data-set categories.

Data Category	**Properties**
Multidimensional Data	Record Independence High Dimensionality Clustered Data
Transaction Data	High Dimensionality High Sparsity
Longitudinal Data	Temporal Order Correlated Data Clustered Data
Graph Data	Graph Properties
Time Series Data	High Dimensionality Attribute Dependency

3.2 ANONYMIZATION

The goal of anonymization is the prevention of the leakage of the identity of a user based upon personal data. In other words, the personal data of a user should not reveal the identity of the user. Processing of personal data is required in many domains, therefore multiple anonymization methods have been developed that intend to preserve the anonymity of the user while the processing of personal data remains possible.

The hereafter presented anonymization methods reduce the quality of the data, while its semantics is preserved for later processing.

3.2.1 *Suppression*

Suppression can be described as the replacement of parts of the original value with replacement characters. The replacement character as well as the replacement strategy have to be carefully chosen to

preserve the semantics of the original value in its specific domain. The replacement character has to be chosen appropriately to avoid any confusion in the interpretation of the value, e.g., assuming postal-codes only consist of numerical characters then no numerical replacement character should be chosen. The replacement strategy is important based on the structure and format of the original value, e.g., for a date in the German date format *dd.mm.yyyy* a replacement sequence that puts priority on the year could first replace the days *dd*, then month *mm* and then the year *yyyy* beginning from the last character *yyyy* [185] [258] [159] .

Assuming the previously detailed example data-set of multidimensional data (see Table 3.1), suppression is used for the anonymization of the postal-code of Bob. Therefore, the asterisk '$*$' is used for the replacement character. Furthermore, for the German postal-code a sequential replacement strategy beginning from the last character is used. Thus, the region covered by the postal-code is steadily increasing. This results in an anonymization hierarchy for the postal-code value '94032' resulting in five additional anonymized values as depicted in Figure 3.2. The original value is defined as anonymization level '0' and it is increased for each sequential anonymization step up to anonymization level 5. Continuing the example, suppression of the postal-code is assumed to the arbitrarily chosen anonymization level '2' resulting in following Table 3.6.

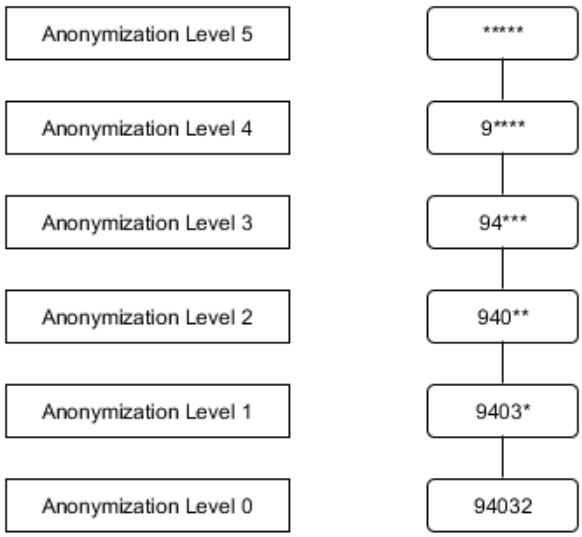


Figure 3.2: Anonymization hierarchy for the German postal-code for Passau '94032' based on suppressing first the last character of the postal-code using the replacement character '$*$'. The anonymization level is steadily increasing starting with level '0' and ending with level '5'.

Table 3.6: Suppression of the postal-code of Bob to anonymization level '2' based on the example multidimensional data-set in Table 3.1.

EI		**QI**			**SD**	**NSD**
ID	*Name*	*Sex*	*Age*	*Postal-Code*	*Salary*	*Lucky#*
1	Alice	F	27	94032	30.000	1234
2	Bob	M	33	**940****	35.000	1337
3	Charlie	M	29	94405	28.000	404

3.2.2 *Generalization*

Generalization can be described as the replacement of the original value with another value of a taxonomy denoting a more general description of the value. The creation of the taxonomy is hereby essential and cannot be chosen arbitrarily to preserve the semantic of the value which is processed in the later. Given the nominal domain of sex, a simple taxonomy can be given (see Figure 3.3).

Figure 3.3: Anonymization hierarchy for the domain 'sex'. The scope of original values at the anonymization level '0' is limited. The anonymization of the original value leads to the replacement of it with 'ANY' denoting a generic upper class for both 'M' for male and 'F' for female.

Furthermore, considering a cardinal domain like the age, intervals can be utilized for creating a taxonomy. The generation of the taxonomy heavily influences the quality of the de-identified data-set considering the intended usage. Thus, the taxonomy has to be carefully crafted with the intended usage in mind. For example, the anonymization of the location (longitude/latitude) has to be more fine-grained to enable navigation services, in contrast for the determination of points-of-interest a relatively rough location, i.e. the city, may be sufficient [185] [258] [195] [159] [127].

Assuming the previously suppressed example data-set for multidimensional data (see Table 3.6), generalization can be used for the anonymization of the sex and age of Bob. Hereby, the previously detailed taxonomies are used. The sex and age value is anonymized to anonymization level '1'. This results in the replacement of the value 'M' for the domain sex with the value 'ANY' and the replacement of the value '33' for the domain age with the value '25 – 37' resulting in following Table 3.7.

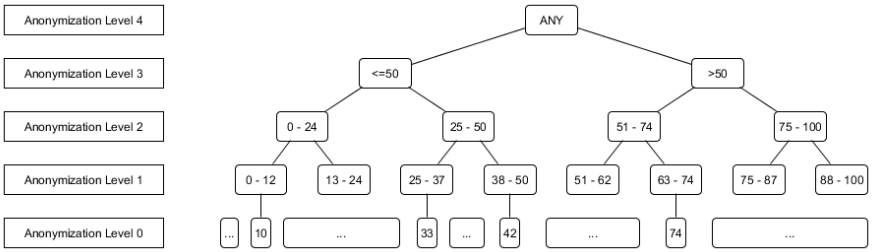


Figure 3.4: Anonymization hierarchy for the domain 'age'. The scope of original values at the anonymization level '0' has a broad range. Each subsequently higher anonymization level covers a sub-range of the original values. The generic value 'ANY' is used for the highest anonymization level.

Compared to suppression, taxonomies have to be defined beforehand with the intended usage in mind, while for suppression, the anonymization hierarchy can be calculated on-the-fly iff the suppression strategy and replacement value are given. Both generalization and suppression hide potentially important details in the QI that can influence further processing.

Table 3.7: Generalization of the sex and age of Bob to anonymization level '1' based on the previously suppressed data-set in Table 3.6.

EI		**QI**			**SD**	**NSD**
ID	*Name*	*Sex*	*Age*	*Postal-Code*	*Salary*	*Lucky#*
1	Alice	F	27	94032	30.000	1234
2	Bob	**ANY**	**25 - 37**	940**	35.000	1337
3	Charlie	M	29	94405	28.000	404

3.2.3 *Deletion*

Deletion is basically the extreme version of both *Suppression* and *Generalization*, leaving no trace of the original value. The probably easiest and straight-forward way to anonymize personal data is to delete it, which stretches the point of anonymization ad absurdum. Deletion of personal data has the advantage to remove any information from the data value that can identify an individual person. But because no semantic information is preserved, it may be argued against the classification of *Deletion* as an anonymization method. Nevertheless, *Deletion* is necessary to preserve the privacy of individuals.

$$\text{Deletion}(\text{Value}) \longrightarrow \varnothing \tag{3.4}$$

Assuming a data-set T with all privacy groups EI, QI, SD and NSD, both EI and QI have to be anonymized to preserve the identity of

the users. On QI either *Suppression* or *Generalization* can be applied to create QI′ in a way that the identity of each user is hidden. But EI, which explicitly identify a user, also have to be de-identified. Therefore, *Deletion* can be used, simply deleting each value that explicitly identifies a user. Therefore, the anonymized data-set $T'_{anonymized}$ is created, including generalization and suppression of QI [257] [258].

Assuming the previously suppressed and generalized example data-set for multidimensional data (see Table 3.7), deletion is used for the anonymization of the ID and name of Bob. The removal of the '2' for the ID and the name 'Bob' results in following Table 3.8.

Table 3.8: Deletion of both the ID and name of Bob based on the previously generalized data-set in Table 3.7.

EI		**QI**			**SD**	**NSD**
ID	*Name*	*Sex*	*Age*	*Postal-Code*	*Salary*	*Lucky#*
1	Alice	F	27	94032	30.000	1234
		ANY	25 - 37	940**	35.000	1337
3	Charlie	M	29	94405	28.000	404

The EI′ may be included in $T'_{anonymized}$ with empty values, but even the existence of EI attributes may allow attackers to gain information, e.g., the information that data-set contains an explicit identifier, therefore the complete removal of EI is assumed.

An alternate approach to de-identify EI values is suppression, which is shown in section 3.5.

3.2.4 *Summary*

In this section different strategies to anonymize personal data have been detailed (see Table 3.9). Although the presented anonymization methods hide information of original data values, privacy is not guaranteed by either of them if not applied considering the other records of the data-set. External knowledge can indeed be used to identify Bob in the data-set easily even though suppression, generalization and deletion have been applied on the EI and QI. For example, if the external knowledge exists that Bob is older than 30 years, then an attacker can derive from the de-identified data-set (see Table 3.8) that both Alice and Charlie are younger than 30. Thus, the second entry must be Bob and the attacker can learn sensitive information, e.g., the salary of Bob. To prevent such attacks, privacy models are introduced in the literature using the detailed basic anonymization methods. An overview on attacks models and privacy models mitigating them is given in the following.

Table 3.9: Summary of anonymization methods.

Anonymization Method	**Target**	**Strategy**
Suppression	EI, QI, SD	Replacement Strategy/Character
Generalization	QI, SD	Generalization Hierarchy
Deletion	EI	Value Removal

3.3 PRIVACY MODELS

Previously, anonymization methods have been introduced that alter an original value of personal data hiding information to preserve privacy. But anonymization methods are not sufficient to preserve the privacy of a data-set because users can be identified through unique properties of their record, especially when external knowledge is available. To tackle this challenge, privacy models have been proposed.

It has to be noted that usage of the term 'privacy model' is contested by the term 'confidentiality criteria'. The term 'confidentiality criteria' is based on the the work of Sicherman et al. [248] and is used in the works of Biskup and Bonatti [41] [43] [42] [44]. It is argued that 'privacy models' are defining privacy via a probabilistic indistinguishable definition, while 'confidentiality criteria' define privacy via precise indistinguishable definitions. The term 'privacy model' is used in the remaining of this work.

Privacy models define properties a de-identified data-set has to fulfil such that the privacy of individuals is guaranteed to a certain degree while the utility is preserved. Privacy models can be hereby classified according to their privacy guarantees, i.e. what attack models they mitigate.

To fulfil the properties of privacy models, previously introduced anonymization methods are applied. In general, an original data-set T is anonymized to $T'_{\mathtt{anonymized}}$. Typically, the EI attributes are completely removed, while QI attributes are anonymized to QI′. Sensitive attributes SD are usually preserved as they are required for further processing, e.g., data mining. Because NSD attributes have no impact on both utility or privacy, they are not further discussed within this section.

$$T'_{\text{anonymized}} = (\text{QI}', \text{SD}, \text{NSD}) \tag{3.5}$$

In general, four attack models are used in the literature to categorize privacy models – *Record Linkage*, *Attribute Linkage*, *Table Linkage* and a *Probabilistic Attack* [108]. Each attack model, as well as exam-

ples for privacy models mitigating them, is detailed in the following. Furthermore, the trade-off between privacy and utility is discussed.

3.3.1 *Record Linkage*

Record Linkage defines an attack pattern in which it is assumed that a data-set T is released. In this released data-set, a small number of records can be identified by a set of values corresponding to QI attributes. The set of records, with the same values for the QI attributes, is denoted as QI-group. If the values of a user, the victim, matches the values of the vulnerable QI-group, the linkage of the user to the small number of records in the QI-group is possible. An attacker can use this vulnerability with additional external knowledge to have the chance to uniquely identify the record of the user from the QI-group. Thus, anonymity can be broken [239] [240] [108].

Let us assume an accessible data-set from which EI attributes are removed, to prevent identification of the users, but QI and SD attributes are given. For example a table with the diseases of patients is given with additional demographic information about the sex, age and job of the patients (see Table 3.10). Furthermore, it is assumed that an attacker has access to external knowledge about patients that went to the hospital, e.g., the attacker is a hospital employee working at the hospital (see Table 3.11). The attacker can match the QI-groups of the patient table with the QI-groups of his external knowledge. This allows the attacker to identify individuals within the patient table. Within this example, the attacker is able to uniquely identify Bob in the patient table according to the matching age ('33'), sex ('M'), and job ('Engineer') and infer the sensitive information SD about his disease ('HIV').

The privacy model *k-Anonymity* has been proposed by Samarati and Sweeney to prevent *Record Linkage* [239] [240]. Hereby, *Record Linkage* is mitigated by requiring every record of a data-set to be in a QI-group with at least $k-1$ other records. Thus, each QI-group has the minimum size of k, so each record cannot be distinguished from at least $k-1$ other records considering the QI-group. This reduces the probability of an attacker to infer the identity of the user to at most $\frac{1}{k}$. Therefore, *k-Anonymity* does not prevent *Record Linkage* but reduces the risk for identification of a user. A data-set fulfilling this criteria is denoted as k-anonymous.

Table 3.10: Patient table with EI attributes deleted to prevent identification.

QI			SD
Job	*Age*	*Sex*	*Disease*
Engineer	35	M	HIV
Engineer	33	M	HIV
Lawyer	29	M	Flu
Lawyer	33	M	Flu
Writer	42	F	Cancer
Singer	45	F	Cancer
Writer	42	F	Cancer

Table 3.11: Available external knowledge with EI and QI attributes.

EI	QI		
Name	Sex	Age	Job
Alice	F	42	Writer
Bob	M	33	Engineer
Charlie	M	29	Lawyer
Donna	F	45	Dancer
Erich	M	37	Engineer
Franz	M	35	Lawyer
Gabi	F	44	Writer
Heidi	F	42	Dancer
Igor	M	32	Lawyer

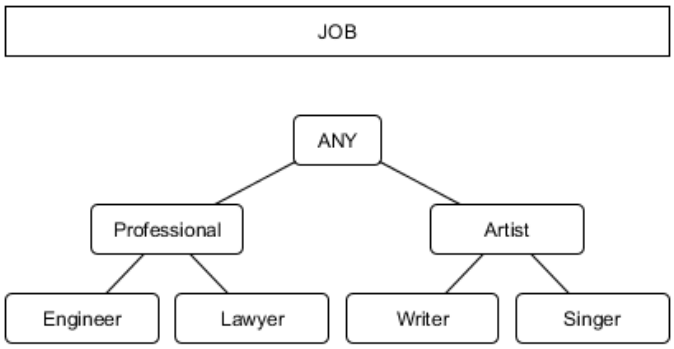


Figure 3.5: Anonymization hierarchy for the QI attribute job for the patient table (see Table 3.10).

Continuing the previously introduced patient data example, the patient table is anonymized to be 3-anonymous (see Table 3.12). To achieve the properties of the privacy model, the anonymization hierarchies given in Figures 3.3, 3.4 and 3.5 are used to anonymize QI attributes. This results in two QI-groups $('\texttt{Professional}','(25-37)','M')$, containing 4 records and $('\texttt{Artist}','(38-50)','F')$, containing 3 records. Thus, the resulting data-set is 3-anonymous, because each QI-group has at least 3 records assigned to it.

The attacker having the external knowledge (see Table 3.11) can no longer uniquely identify the individual Bob according to his QI-group $('\texttt{Engineer}','33','M')$, which now correlates to 4 records of the QI-group $('\texttt{Professional}','(25-37)','M')$. Thus, the property of *k-Anonymity*, that the probability of an attacker to infer the identity of the user is as most $\frac{1}{k}$, is fulfilled with the risk being $\frac{1}{4}$ in this case.

Other privacy models which mitigate *Record Linkage* can be found in the literature, e.g., *k-Map* [94], *MultiR k-Anonymity* [193], *(c, t)-Isolation* [66], and *Average Risk* [93].

Table 3.12: 3-anonymous patient table based on Table 3.10. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 3.4, and 3.5.

QI			**SD**
Job	*Age*	*Sex*	*Disease*
Professional	(25 - 37)	M	HIV
Professional	(25 - 37)	M	HIV
Professional	(25 - 37)	M	Flu
Professional	(25 - 37)	M	Flu
Artist	(38 - 50)	F	Cancer
Artist	(38 - 50)	F	Cancer
Artist	(38 - 50)	F	Cancer

3.3.2 *Attribute Linkage*

The *Attribute Linkage* attack considers that the attacker may not precisely identify the individual, but infer sensitive information about it using published data. Therefore, the attacker inspects the records of the QI-groups that the target belongs to correlating sensitive attributes SD to the target. If a sensitive value is predominating within the QI-group, then the attacker determine with a high probability that it correlates to the target. To mitigate this attack the correlation between QI-groups and SD attributes has to be decremented [108].

Considering the patient table (see Table 3.10) from previous example, the QI-group $(\text{'Writer'},\text{'42'},\text{'F'})$ indicates with a 100% probability that the individual has the disease 'Cancer'. Considering Table 3.11, the attacker can determine that Alice has cancer, assuming that the population of both tables is correlated.

To mitigate this attack, the diversity principle has been introduced by Machanavajjhala et al., and denoted as *l-Diversity* [177]. It requires, that each QI-group has at least l 'well-represented' sensitive values. It is noteworthy, that a similar idea was previously discussed by Ohrn and Ohno-Machado [203]. The term 'well-represented' can hereby be defined in different ways instantiating different definitions of *l-Diversity*, e.g., *distinct l-Diversity* [262], *entropy l-Diversity* [177], and *recursive (c, l)-Diversity* [177].

The definition of *distinct l-Diversity*, also known as *p-Sensitive k-Anonymity* [262], is used in the following. This definition states that each QI-group has to have at least l distinct values for the sensitive attribute, considering only one sensitive attribute. This definition of

l-Diversity also fulfils the properties of *k-Anonymity* if $k = l$, because each QI-group also contains at least l records. A data-set fulfilling these properties is denoted as *distinct l-diverse*.

Continuing the example, a *distinct 3-diverse* data-set is created (see Table 3.13) from Table 3.10. The QI of the data-set are anonymized such that only the QI-group $('ANY','(25-50)','ANY')$ exists for which 3 distinct sensitive values are present $\{'HIV','Cancer','Flu'\}$. Therefore, the data-set is *distinct 3-diverse* and the attacker can no longer infer with 100% certainty that Alice has cancer. Furthermore, it can be observed that the entries in Table 3.10 are very generic and therefore the utility, i.e. possible usage, of the anonymized data-set is limited.

Table 3.13: Distinct 3-diverse patient table based on Table 3.10. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 3.3, and 3.5.

QI			**SD**
Job	*Age*	*Sex*	*Disease*
ANY	(25 - 50)	ANY	HIV
ANY	(25 - 50)	ANY	HIV
ANY	(25 - 50)	ANY	Flu
ANY	(25 - 50)	ANY	Flu
ANY	(25 - 50)	ANY	Cancer
ANY	(25 - 50)	ANY	Cancer
ANY	(25 - 50)	ANY	Cancer

Considering the overall distribution of sensitive values with the local distribution of sensitive values within a QI-group, *l-Diversity* may not be sufficient to prevent *Attribute Linkage* attacks when the overall distribution is skewed. Assuming a patient table in which 90% of the patients have the flu and only 10% have HIV. Furthermore, assuming a QI-group that has a distribution of 50% HIV and flu patients. The data-set satisfies *2-Diversity*, but the mentioned QI-group poses a risk, because every record owner could be inferred with a 50% probability with HIV instead of 10% in the overall data-set. This is denoted as *skewness attack*, for which the privacy model *t-Closeness* has been developed for mitigation [167]. This privacy model requires the distribution of sensitive attributes in the overall table to be t close to the distribution in any QI-group, i.e. the distance between the distribution of a sensitive attribute and distribution of the attribute in the whole table is no more than the threshold t. The calculation of the closeness is based on the *Earth Mover Distance (EMD)* function. But using *t-Closeness* also has disadvantages regarding the utility of the resulting data-set. Furthermore, it lacks flexibility for

specifying different protection levels for different sensitive values, e.g., 'Flu' requires less protection than 'HIV'. Other privacy models mitigating *Attribute Linkage* are, e.g., *Confidence Bounding* [269], *(X,Y)-Privacy* [268], δ-*Disclosure* [58] and β-*Likeness* [64]. *Confidence Bounding* [269] is proposed to protect against threats based on data mining capabilities and limits the confidence of inferring sensitive, hereby the information is preserved such that it can be used for analysis but unwanted sensitive information is limited. *(X,Y)-Privacy* [268] is introduced to cope with sequential releases, i.e. publication, of data such that current release is anonymized such that it cannot be linked with the previous releases. *(X,Y)-Privacy* defines that value in X is linked to at least k distinct values on Y, whereas it is assumed that X and Y are disjoint sets of attributes that describe individuals and sensitive properties in any order. The authors of δ-*Disclosure* [58] question if anonymization of QI attributes is advantageous over trivial sanitization of the data-set. Therefore, they introduce the alternative δ-*Disclosure*, which defines that a data-set is δ-*disclosure* private, if the distribution of SD attributes within each QI-group is about the same as the distribution of SD attributes in the complete data-set. β-*Likeness* [64] assumes that SD attributes of the data-set are public knowledge. Therefore, β-*Likeness* assures that the relative distance between SD values to a given QI-group does not exceed the distance to the overall population of SD by the threshold β.

3.3.3 *Table Linkage*

In the *Table Linkage* attack, the goal of the attacker is to gain knowledge about the presence or absence of an individual within a data-set, which differs from the assumption of both *Record Linkage* and *Attribute Linkage*, which assume that the attacker already knows that the individuals record is within the data-set. But the presence (or absence) of a record within a data-set can reveal sensitive information. Assuming a hospital releases a data-set with a particular type of disease, e.g., HIV, then the attacker can reveal sensitive information about an individual if the presence of the individual in the data-set can be shown. Therefore, *Table Linkage* occurs if an attacker can infer the presence or absence of an individuals' record in the data-set with confidence.

For example let us assume the *3-anonymous* patient data-set (see Table 3.12). Furthermore, the related external knowledge data-set is *4-anonymous* is assumed (see Table 3.14). The probability that Alice is present in the patient data-set is $\frac{3}{4}$ because there are 3 records in the patient data-set with the QI-group $('\mathrm{F}', '(28-50)', '\mathtt{Artist}')$ and 4 records in the external knowledge data-set. Similarly, the probability that Bob is in the patient data-set is $\frac{4}{5}$.

The privacy model δ-*Presence* intends to prevent *Table Linkage* by bounding the probability of inferring the presence within a speci-

fied range $\delta = (\delta_{min}, \delta_{max})$ [194]. Assuming an external knowledge data-set EK and a private data-set T, with T being a subset of EK $T \subseteq EK$. The generalized data-set T' satisfies $(\delta_{min}, \delta_{max})$-*Presence* if $\delta_{min} \leqslant P(t \in T|T') \leqslant \delta_{max}$ for all $t \in EK$. Although δ-*Presence* is valid as-is, i.e. it assumes that the owner, e.g., company, of T has access to the same external knowledge EK as the attacker, which might not be true in practice. Other privacy models that mitigate *Table Linkage* are, e.g., *Distributional Privacy* [48], or ϵ-*Differential Privacy* [90]. This latter is detailed for the following attack model, the *Probabilistic Attack*. Whereas, *Distributional Privacy* [48] is stronger than ϵ-*Differential Privacy*.

Table 3.14: 4-anonymous external knowledge table based on Table 3.11. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 3.3, and 3.5.

EI	**QI**	**QI**	**QI**
Name	*Sex*	*Age*	*Job*
Alice	F	(38 - 50)	Artist
Bob	M	(25 - 37)	Professional
Charlie	M	(25 - 37)	Professional
Donna	F	(38 - 50)	Artist
Erich	M	(25 - 37)	Professional
Franz	M	(25 - 37)	Professional
Gabi	F	(38 - 50)	Artist
Heidi	F	(38 - 50)	Artist
Igor	M	(25 - 37)	Professional

3.3.4 *Probabilistic Attack*

Unlike *Record Linkage*, *Attribute Linkage* and *Table Linkage*, the *Probabilistic Attack* does not focus on what records can be linked to an individual, but on how the attacker would change his probabilistic beliefs on the sensitive information of an individual. The goal in preventing the *Probabilistic Attack* is to keep the difference between prior and posterior beliefs as small as possible [108]. In other words, it has to be prevented that an attacker gains information, which corresponds to the uninformative principle introduced by Machanavajjhala et al. [177].

Assuming statistical databases, an attacker should not gain any benefits from being able to access a published anonymized data-set. Therefore, the privacy model *(c, t)-Isolation* has been proposed by Chawla et al. [66]. This privacy model considers the distances between data records, which is suitable for numerical attributes statistical databases

but less suitable for multidimensional data. Therefore, ϵ-*Differential Privacy* is detailed, which has been proposed to compare the risk with and without the record owner's data in the published and anonymized data-set. Dwork [90] introduced a new privacy notion, that is very promising. It differs from the previous privacy notion comparing the prior and posterior probability before and and after accessing the data-set. Thus, ϵ-*Differential Privacy* has been developed to compare the risk with and without the addition of an individuals' record in the published data-set. Therefore, it has to be ensured that the addition or removal of a record does not significantly affect the outcome of any analysis. This allows for additional beneficial properties, e.g., the join of different data-sets does not increase the risk for privacy.

According to Dwork [90] [91] a randomized function K gives ϵ-*Differential Privacy* if for all data-sets D_1 and D_2 differing on at most one element, and all $S \subseteq Range(K)$,

$$Pr[K(D_1) \in S] \leqslant exp(\epsilon) \times Pr[K(D_2) \in S] \tag{3.6}$$

In practice, this notion of *Differential Privacy* often induces too much noise within the data-set, thus decreasing its utility. Therefore, its definition has been relaxed to *(ϵ, δ)-Differential Privacy* allowing for an error probability δ [168]. The relaxed version of *Differential Privacy* denotes, that a randomized algorithm A satisfies *(ϵ, δ)-Differential Privacy*, if for any pair of neighboring datasets D and D' and for any $O \subseteq Range(A)$ [92].

$$Pr[A(D) \in O] \leqslant e^{\epsilon} \times Pr\left[A(D') \in O\right] + \delta \tag{3.7}$$

The small error probability of *(ϵ, δ)-Differential Privacy* enables higher data quality in comparison to ϵ-*Differential Privacy* [40]. According to Fung et al. [108], *Differential Privacy* mitigates *Table Linkage* and the *Probabilistic Attack*.

3.3.5 *Privacy and Utility Trade-off*

If only privacy would have to be considered it could be stated that personal data shall just not be collected and processed nor existing personal data shall be deleted. But the processing of personal data is required for many processes to gain valuable insights, e.g., research, or marketing. Therefore, the usefulness has to be considered while the privacy requirements of individuals are met. In the following, it is detailed how the usefulness, or utility, of data can be quantified. The challenges for finding a trade-off between both privacy and utility are furthermore detailed.

3.3.5.1 *Utility*

The usefulness of a data-set for a specific purpose, e.g., data mining, is denoted as utility. Therefore, the measurement of utility requires a

context in the best case, which makes the measurement heavily related to the data-set, the de-identification process and the intended usage of the data, if specified. But this context can not always be determined beforehand. In the literature several aspects for measuring utility can be found, namely *Accuracy*, *Completeness* and *Consistency* [38].

Accuracy measures the proximity of a sanitized (de-identified) value to the original value. In other words, *Accuracy* measures the loss of information after the de-identification process has been applied.

The height of anonymization hierarchies was utilized as the basis for one of the first *Accuracy* focused utility metrics denoted as *Height Metric (HM)* [238]. For each anonymization of a data value, information loss is assumed. Hereby, height is measured by how often the original value has been anonymized, i.e. how many anonymization levels are iterated (see Equation 3.8). The fewer anonymizations are required the lesser information is lost. A downside of this metric is that the information loss of each anonymization level is considered equal.

$$HM(T') = \sum_{i=1}^{n} max(anonymizationLevels_i(T)) \tag{3.8}$$

The *General Loss Metric (LM)*, a successor to *HM*, intends to measure the average information loss of all data cells of an anonymized data-set T' [139]. In Equation 3.9 f is a function that based on a data cell value $T'[i][j]$ returns the number of distinct values that can be generalized to $T'[i][j]$, and g is a function that based on an attribute a_i, returns the number of distinct values of a_i. The proportion of f to g for each value is summed and put into relation with the data-set size to calculate the average information loss.

$$LM(T') = \frac{\sum_{i=1}^{n} \sum_{j=1}^{|T|} \frac{f(T'[i][j])-1}{g(a_i)-1}}{|T| \cdot n} \tag{3.9}$$

Various other *Accuracy* focused utility metrics have been introduced, e.g., the *Discernibility Metric (DM)* [29] which assigns a penalty to a tuple based on how many tuples in T' are indistinguishable from it, or for statistical-based perturbation which measure the loss of precision in T' [6].

Completeness evaluates the degree of missed data in T', while *Consistency* evaluates the internal properties and constraints of a data-set, e.g., relationships among different attributes of a record or among a group of records within a data-set. Both aspects are only rarely subject of utility metrics in the literature. Bertino et al. propose a utility metric that not only considers *Completeness* and *Consistency*, but also the relevance of the data as well as the structure of the database [37].

It can be concluded that the measurement of utility requires the consideration of *Accuracy*, *Completeness* and *Consistency*, while often only *Accuracy* is covered. Thus, utility metrics could rather be denoted as information loss metrics, because measuring the utility, i.e. the usefulness of a data-set for a specific purpose, is not always conducted.

3.3.5.2 *Privacy*

Privacy is measured quantitatively by the degree of uncertainty according to which personal data can be inferred [169] [171] [229]. As shown by the previously introduced attack models and privacy models, the notion of privacy is manifold. For example for *Record Linkage* the risk of identifying an individual within a data-set is mitigated by *k-Anonymity* such that it can be quantified to be $\frac{1}{k}$ at maximum. Similarly, the risk for a successful *Attribute Linkage*, *Table Linkage* or *Probabilistic Attack* can be quantified with other privacy models.

Although, privacy can be quantified, it is hard to determine which level of privacy is required in practice. Assuming *k-Anonymity*, the question is which value for k is sufficient to protect the privacy. Stating that all privacy requirements must be met would imply that k is maximized. Although this approach would preserve the privacy of the data-set, requiring a strong anonymization, a high penalty on its usability for further processing would be induced.

An important aspect for the anonymization process is to keep the data usable for companies while privacy is preserved. Therefore, privacy requirements, which require personal data to be hidden, and utility requirements, which require personal data to be available, have to be balanced.

A direct comparison of privacy and utility is flawed as shown by Brickell and Shmatikov, because it would directly indicate that a gain in privacy would equal loss in utility [58]. Both privacy and utility differ in their concepts and characteristics so much that a direct comparison is not possible. First, specific data, e.g., concerning a small group of individuals, has larger impact on privacy, while aggregated data has a larger impact on utility. Secondly, privacy is an individual concept which should be measured separately for every individual, while utility is an aggregated concept which should be measured accumulatively for all useful knowledge. Lastly, privacy loss can be caused by any information that deviates from the prior belief, false or true, but only true information contributes to utility [169].

Thus, an appropriate trade-off between utility and privacy has to be approximated, considering the privacy requirements of individuals while the data-set is still usable for the intended purpose of the company [178] [224].

3.3.6 *Reasoning on Privacy Models*

Comparing the before mentioned privacy models according to the mitigated attack models (see Table 3.15), it can be observed that no privacy model covers all attack models. Furthermore, it can be stated that the notion of privacy differs according to privacy models, even mitigating the same attack model, such that no "single privacy model fits all" option exists, especially if different scenarii and data-set types

have to be considered. Thus, a privacy language has to express privacy models in a generic and extensible approach to cover various scenarii.

Furthermore, a privacy language expressing privacy policies and thus both the privacy requirements of the user and the data processing requirements for specific purposes has to be considered. In other words, the company shall be able to express limitations for the de-identification of personal data for specific purposes. Hereby, it has to be considered that the measurement of utility seems not to be fully developed in the literature due to its focus on *Accuracy*. It has also to be considered that many privacy models use a derivative of other basic utility metrics, e.g., height metric, which leads to varying utility definitions and measurements. It can be boldly stated that for many privacy model approaches a specifically suitable utility metric is developed. Thus, as no uniform baseline is given, a privacy language has to consider both privacy and utility requirements.

Privacy models define properties a de-identified data-set has to fulfil, but the introduced privacy models do not consider an individuals' privacy requirements. Therefore, the personal privacy concept is detailed and discussed in the following.

Table 3.15: Non-exhaustive list of privacy models and their classification according to the attack models they mitigate.

Privacy Model	*Record Linkage*	*Attribute Linkage*	*Table Linkage*	*Probabilistic Attack*
k-Anonymity	x			
k-Map	x			
MultiR k-Anonymity	x			
Average Risk	x			
Distinct l-Diversity	x	x		
Entropy l-Diversity	x	x		
Recursive (c, l)-Diversity	x	x		
t-Closeness		x		x
Confidence Bounding		x		
(X, Y)-Privacy	x	x		
δ-Disclosure		x		x
β-Likeness		x		x
δ-Presence			x	
Distributional Privacy			x	x
ϵ-Differential Privacy			x	x
(ϵ, δ)-Differential Privacy			x	x
(c, t)-Isolation	x			x

3.4 PERSONAL PRIVACY

The concept of personal privacy, or personalized anonymity, has been detailed by Xiao and Tao [277]. Instead of assuming a consistent level of privacy protection for all users in a multidimensional data-set, personal privacy enables each user to specify the degree of privacy protection of his sensitive values SD. Therefore, the corresponding anonymization hierarchy has to be accessible by the user to express his anonymization level preference when his data is collected.

In the following, approaches for the implementation of the personal privacy concept are detailed. Based on the detailed technical approaches and the legal perspective on personal privacy, requirements for a holistic approach are detailed.

3.4.1 *Personal Privacy Approaches*

Only few works have been published on personal privacy anonymization. Both [230] and [21] address dynamically privacy concerns in the context of *Privacy-Preserving Data Publishing (PPDP)* taking into account personal privacy. Reddy et al. in [230] implement personal privacy at the record level only allowing the user to set a binary consent flag. If consent is not given by the user, the whole record is removed from the data-set during the anonymization process. For the anonymization process, multidimensional data-sets are considered for which common privacy models can be used, e.g., *k-Anonymity* [257] or *l-Diversity* [177]. This approach considers the consent of the user during the anonymization process, which may lead to a coarse loss of information because the binary consent flag affects the complete record.

Similarly, in the context of recommender systems, Saji et al. [237] propose a web search engine in which each user takes the binary decision to set his whole profile, that is utilized for queries, public or private. This allows the search engine to either consider the user profile for personalized query results or regular queries only using the users' input. This approach considers only the search engine, i.e. a single purpose, for the scenario limiting its scope. Consequently, there is no more general approach proposed in the literature.

Ma et al. [175] introduce a scheme for assessing personal privacy in the context of ubiquitous computing. Hereby, the user can set a threshold value determining the disclosure of his personal data in different scenarios, e.g., in scenario A only the name is provided but in scenario B, the e-mail is additionally provided. Depending on the scenario, i.e. the purpose of the processing, the user can define which of his data is processed expressing his personal privacy requirements. This work extends the work of Reddy et al. [230]. Therefore, instead of only considering binary consent for the record, a more fine-grained

decision can be made for each of the attributes. Although not explicitly stated, all attributes of the data-set are possibly affected, thus increasing the scope from only SD attributes as previously defined by Xiao and Tao [277] to possibly all EI, QI, SD, and NSD attributes. However, the user has full control over his personal data, this approach does not consider any follow up anonymization mechanisms.

Ashoka and Poornima [21] introduce *Sensitivity Flags* to be set by the users to indicate the minimum anonymization level for sensitive attributes SD. For each sensitive attribute, the user chooses the value that should be published from an anonymization hierarchy. The way the anonymization hierarchy is presented to the users, is hereby not detailed. To maintain a minimum level of data utility, the authors propose a *Stipulation-Based Anonymization Algorithm* that produces the data-set to be anonymized according to the data miners needs by reconsidering user anonymization choices for data that are not considered as sensitive by the company. This approach overrules the users' personal decision on the anonymization level, i.e. the *Sensitivity Flag*, to increase the utility for later processing, e.g., data mining. Therefore, the proposed solution encourages the users to decide on their individual privacy settings on the hand, but enables the processor of the data to override the aforementioned privacy settings. This approach is highly questionable, because the processing of personal data is valued higher than the privacy of users in this approach. Furthermore, this solution neither considers anonymization in the context of multiple purposes nor in the presence of different privacy models. Nevertheless, the approach can be seen as an improved approach for personal privacy compared to binary purpose decisions, because it allows the user to determine the privacy at the attribute-level, which is desirable for a fine-grained control of privacy.

Khan et al. [153] implement personal privacy as a variation of *k-Anonymity*. But instead of assuming a uniform privacy level k for the whole data-set, the privacy level is individual for each user u such that the privacy level is dependent on each user $k(u)$. This is denoted as adaptive anonymity. Adapting the definition of *k-Anonymity*, in order to fulfil the privacy guarantee for adaptive anonymity, the record of each user u cannot be distinguishable from the record of $k(u) - 1$ users in the data-set. This approach considers for each record individual privacy requirements in form of a user dependant k. Compared to previous approaches, it does not consider personal privacy on attribute-level, but used privacy model concepts to preserve the privacy of the complete data-set.

In the context of distributed systems, Neumann et al. [198] model personal privacy preferences of the user by including consent for services, attribute specific sensitivity, and overall sensitivity. Personal privacy preferences are analysed to identify privacy risks in the distributed system based upon the selected pseudonymization ap-

proaches. The personal privacy preferences are modelled based on the data flow between different systems and stakeholders generating a formal model of privacy. This model is then analysed to identify privacy risks, which are based upon the personal privacy preferences and the pseudonymization method choice. The approach is intended for usage by system designers and developers during the development of applications to support the understanding of privacy risks for an early mitigation. This system considers pseudonymization, commonly used to preserve privacy in the health care domain, instead of anonymization methods or privacy models to preserve the personal privacy of users.

3.4.2 *Reasoning on Personal Privacy*

In the following, the notion and different approaches to realizes personal privacy in the literature are positioned within the regulations of the *GDPR*. Considering the *GDPR*, the personal privacy preference, e.g., personal anonymization level, may be negotiated and expressed within the privacy policy, although *GDPR* does not require the specification of any de-identification mechanisms within the requirements for a privacy policy [110, Art. 12 - 14]. An argument for personal privacy settings in privacy policies is the notion of *Consent* enabling the user to control for which purposes his data is processed [110, Art. 7].

Following and extending the concept of consenting purposes, it can be argued for control over the quantity and quality of the processed data for each purpose by the user. Therefore, fine-grained control over the processing of personal data is given by the user.

Furthermore, this rationale is supported by the *Communication Privacy Management (CPM)* theory, opting for a dialectic approach, e.g., negotiation of consent and personal anonymization levels, expressing privacy as a range of complete openness to complete closeness [211]. Hereby, openness can be considered as the consent to the processing of personal data, while closeness can be considered as the rejection of the processing of personal data. The decision on the processing of personal data can hereby include the purpose, data recipient, data attributes and quality of the data, e.g. only parts of the address are provided.

Moreover, the utility requirements for processing have to be considered limiting the de-identification of certain attributes required for processing. Thus, an upper limit for the anonymization level is desirable, but this upper limit shall not override the preference of the user, i.e. in contrast to Ashoka and Poornima [21], but be considered during the negotiation of the privacy policy. The personalization should furthermore consider all aspects of the processing of personal data, including purpose-based consent as well as attribute-based per-

sonalization of privacy requirements (see Table 3.16). As previously detailed, personal privacy settings for SD attributes is considered by Ashoka and Poornima [21], while personal privacy settings for QI attributes is considered by Khan et al. [153]. Ma et al. [175] consider all attributes (EI, QI, SD and NSD), but the anonymization of NSD does not increase the privacy and EI attributes should commonly either be deleted or pseudonymized (see Section 3.5). Therefore, personal privacy settings, i.e. anonymization requirements for attributes, can in theory affect all attributes, but should only be considered for QI and SD attributes.

$$T'_{\text{personalPrivacy}} = (EI', QI', SD', NSD') \tag{3.10}$$

Next, an overview over pseudonymization methods and approaches is detailed.

Table 3.16: Summary of personal privacy approaches.

Work	User Decision
Xiao and Tao [277]	Anonymization Level for SD
Reddy et al. [230]	Consent Flag on Record
Saji et al. [237]	Consent via Profile
Ma et al. [175]	Threshold for EI, QI, SD, NSD
Ashoka and Poornima [21]	Sensitivity Flags for SD
Khan et al. [153]	Personal k for *k-Anonymity*
Neumann et al. [198]	Service Consent Attribute Sensitivity Overall Sensitivity

3.5 PSEUDONYMIZATION

As shown by Gerl and Bölz [113], various pseudonymization methods and variations exist. Pseudonymization replaces the original value with a replacement *pseudonym*. Pseudonymization is applied to make data useless outside its application scope, while it remains useful. Furthermore, authorized re-identification is possible. In general, an original data-set T is pseudonymized to $T'_{pseudonymized}$. It consists of previously mentioned privacy groups of data. In T′, attributes are replaced with uniquely identifiable pseudonyms.

$$T'_{\text{pseudonymized}} = (EI', QI, SD, NSD) \tag{3.11}$$

Usually, only EI attributes are pseudonymized to replace the identifying value with a pseudonym such that a person cannot be identified. Therefore, EI attributes are commonly pseudonymized, while on QI,

SD and NSD attributes usually no pseudonymization is applied. In the following, an overview of pseudonymization techniques is presented.

3.5.1 *Pseudonym Generation*

Pseudonymization (or tokenization) swaps distinct original values with a pseudonym (token). The generation of the pseudonym can hereby vary. It can be differentiated between one-way pseudonyms, that cannot be reversed, and reversible pseudonyms enabling re-identification. Reversible pseudonym generation, which is based on a function, requires that the parameters, i.e. the secret, of that function is kept secret. Thus, reversible pseudonyms require additional protection to prevent unauthorized identification of the user.

$$\text{Reversible Generator}(\text{Value}, \text{Secret}) \longrightarrow \text{Pseudonym} \tag{3.12}$$

$$\text{Re-identification}(\text{Pseudonym}, \text{Secret}) \longrightarrow \text{Value} \tag{3.13}$$

One-way pseudonym generation is usually based on hash functions, which are also used in other domains, e.g., cryptography, database indexing, or blockchains. A one-way hash makes it hard to determine the original value from the hash value within a reasonable amount of time, but allow a fast computation of the hash [201].

$$\text{One-Way Generator}(\text{Value}) \longrightarrow \text{Pseudonym} \tag{3.14}$$

$$\text{Re-identification}(\text{Pseudonym}) \nrightarrow \text{Value} \tag{3.15}$$

Furthermore, a distinction can be made between dependent and independent pseudonym generation. Dependent pseudonym generation retains a relationship of the pseudonym with the original data value. The original data value is used as an input for the generation of the pseudonym. In contrast, independent pseudonym generation selects the pseudonym randomly and matches it with the original value. Therefore, no correlation between pseudonym and original value can be monitored. As a result, pseudonyms based on independent tokens are more secure, because re-identification with given pseudonyms only is not possible. Thus, attack patterns like injection attacks can be prevented [57].

$$\text{Independent:Pseudonym Generator} \longrightarrow \text{Pseudonym} \tag{3.16}$$

$$\text{Dependent:Pseudonym Generator}(\text{Value}) \longrightarrow \text{Pseudonym} \tag{3.17}$$

In general, the generation of a token is based on random seeds, cryptographic methods or hashing [266]. An overview is given in the following.

3.5.1.1 *Random Seeds*

In health care scenarii, it is necessary to correlate data from different sources to the same patient without revealing the identity of the patient to unauthorized entities. Therefore, Lablans et al. [165] propose a RESTful pseudonymization interface which assigns new or existing pseudonyms to patient records (see Figure 3.6). The generation of pseudonyms is hereby based on the *Patient Identifier (PID)* generator of Faldum and Pommerening [101]. The *PID* generator is mainly based on random seeds allowing fast data processing. Furthermore, the hereby generated *PIDs* allow for error detection and correction.

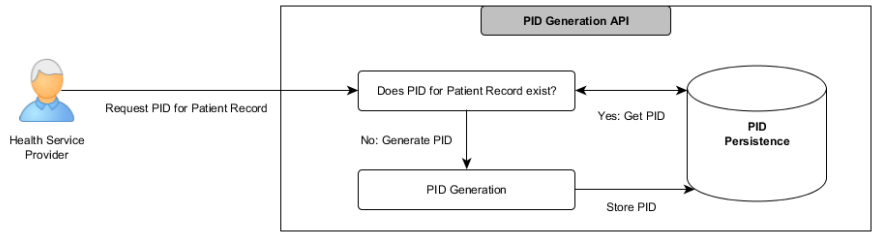


Figure 3.6: Example scenario for *Patient Identifier (PID)* generation based on Lablans et al. [165] using pseudonymization.

3.5.1.2 *Cryptographic Methods*

Cryptographic methods encrypt the original value to generate the pseudonym. Therefore, they can be classified as dependent pseudonym generators. Cryptographic methods themselves can be differentiated as symmetric or asymmetric approaches.

Asymmetric approaches utilize a public key for encryption and a private key for decryption. In the literature the usage of the *Rivest–Shamir–Adleman (RSA)* algorithm can be found for pseudonymization in a smart grid scenario. The data of meters is vital for the upkeep of the grid, but it also allows one to infer information about the activity of the observed household, e.g., when people go to work or if they are on a holiday trip. This private information has to be protected for which the meter information is pseudonymized considering time windows enabling third parties to process temporal sequentially correct data while the identity of users is protected [235].

Symmetric approaches utilize the same key for both encryption and decryption. Noumeir et al. [201] propose and discuss several architectures to enable de-identified clinical data to be exported to a research system based on the consent of the patient. Hereby, symmetric encryption utilizing *Data Encryption Standard (DES)* or *Advanced Encryption Standard (AES)* is used for pseudonymization to preserve the privacy of patients while data is used both by authorized medical personnel as well as researchers [132]. To secure the approaches the keys have to be securely stored [132]. In general, the disadvantage of

cryptographic methods is an increased computational cost compared to other approaches. To definitively prevent re-identification of the pseudonyms, the key used for decryption can be deleted. Furthermore, it has to be considered that the output of cryptographic methods can have a varying size, which has to be considered during the selection of the pseudonym generation method.

3.5.1.3 *Hashing*

Hash functions usually map the original value of an input set to a value of a smaller target set, creating dependent pseudonyms. Thus, hash functions are often not injective. However, due to the hash function's nature, collisions are possible allowing different original values to be assigned to the same pseudonym. Therefore, collision-resistant hash algorithms are preferred for pseudonymization. But contrary to cryptographic methods, the generated pseudonyms have the same size regardless of the size of the original value. Hashing methods can further be classified in keyed and non-keyed hashing.

Non-keyed hash functions have the disadvantage that the same original value is always assigned the same pseudonym, even across multiple systems using the same hashing function. This allows for rainbow table attacks, in which for a specific non-keyed hash function, e.g., *Message-Digest Algorithm (MD5)*, a finite amount of possible original value and hashed value combinations are calculated and stored. An attacker can use these stored combinations to look up possible original values for a hash and therefore approximate the original value. This attack can be mitigated by using hash functions wich have a very high number of possible hashed values [163].

Keyed hashing methods combine cryptographic methods with hashing utilizing a key. The exchange of the key results in different pseudonyms for the same original value. Thus, dictionary attack like the rainbow table attack for *MD5* algorithm are not applicable iff the key remains secret. Therefore, the key has to be stored securely. In the health care domain, medical records of patients used for research and personalized medicine. On the one hand, results should be offered to patients for clinical treatment requiring the re-identification of the individuals. On the other hand, research data must not be associated with the individual, which is a contradiction. This challenge is tackled by Aamot et al. [1] proposing an architecture using *Password-Based Key Derivation Function 2 (PBKDF2)* with a deterministic salt, derived from the *Hash Message Authentication Code (HMAC)* of the patient identifier, to preserve the privacy of the patient while the individual patient can be re-identified to receive feedback. The re-identification is hereby strongly regulated requiring the involvement and agreement of an ombudsman.

3.5.1.4 *Use Case Specific Techniques*

Further specializations for the generation of pseudonyms exist, which are tailored to specific use cases. Such specializations are intended to preserve specific operations on the de-identified data-set and therefore increase the utility.

For example, network traffic is essential for research of the Internet, but collecting and making available network traffic proses a privacy risk to the users. To tackle this challenges, IP-addresses have to be de-identified in such a way that privacy is preserved while IP-addresses can still be used for research. Fan et al. [102] propose a cryptographic scheme using a mix of pseudonymization and anonymization approaches to preserve the prefix of IP-addresses. Thus privacy of users can be preserved while prefixes of IP-addresses, required for research, are preserved in large scale distributed scenarii.

Furthermore, Kerschbaum [152] proposes a pseudonymization technique for timestamps preserving the distance, which is also applicable for network traffic. This distance-preserving pseudonymization technique is additionally applicable on two-dimensional spatial data and as such usable for different domains.

3.5.2 *Implementation Patterns*

To gain additional features, like an increased degree of privacy or the possibility of re-identification, the generation of the pseudonym can be coupled with additional implementation patterns.

To enable re-identification of the original value based on the pseudonym it is either possible to use a reversible pseudonym generator, e.g., based on RSA, or to implement bijective mapping with a one-way pseudonym generator, e.g., based on *MD5*. Bijective mapping stores and maps the generated pseudonym with the original value.

$$\text{Value} \longleftrightarrow \text{Pseudonym} \tag{3.18}$$

Therefore, bijective mapping enables re-identification of the one-way generated pseudonym. Assuming the previously suppressed and generalized example data-set for multidimensional data (see Table 3.7), deletion is used for the anonymization of the ID and name of Bob. If pseudonymization is used instead, the anonymity of the identity of Bob is preserved while authorized re-identification is enabled. Therefore, the usage of a one-way hash function is assumed creating a 5-digit hash for both the ID and the name of Bob, which 1) replaces the respective original value and 2) is stored for later authorized re-identification using bijective mapping. Therefore, the hash function generates the pseudonym '12345' for the ID '2' as well as the pseudonym '40404' for the name 'Bob'. The corresponding mapping is stored (see Table 3.17 and Table 3.18) and the original values replaced, thus resulting in Table 3.19.

Table 3.17: Mapping store for ID for the data-set in Table 3.1 in which the ID of the record of Bob is stored.

Mapping Store ID	
Original Value	*Pseudonym*
2	12345

Table 3.18: Mapping store for name for the data-set in Table 3.1 in which the name of the record of Bob is stored.

Mapping Store Name	
Original Value	*Pseudonym*
Bob	40404

Table 3.19: Pseudonymization of the ID and name of Bob replacing the corresponding original values based on the previously generalized data-set in Table 3.7.

EI		**QI**			**SD**	**NSD**
ID	*Name*	*Sex*	*Age*	*Postal-Code*	*Salary*	*Lucky#*
1	Alice	F	27	94032	30.000	1234
12345	**40404**	ANY	25 - 37	940**	35.000	1337
3	Charlie	M	29	94405	28.000	404

To prevent any unauthorized re-identification, it is necessary that the generated mapping is secured, e.g., by encryption [165]. For each entry to be pseudonymized, the original value can be looked up in the mapping store and, iff a corresponding pseudonym exists, returned. Otherwise, a new pseudonym is generated and appended to the mapping. As the performance of the pseudonym generator depends on the underlying algorithm, the lookup of the original value in the mapping store usually has run-time advantages compared to the generation of pseudonyms for each value. Additionally, this depends on the amount of distinct original values in the data-set.

To enable the processing of pseudonyms without alteration of the application, the format of the original value has to be preserved. Limited token generation mimics the original value format by exclusively calculating pseudonyms with the same character set and the same structure. For instance, an input date is also structured as a date after pseudonymization [266].

Additional privacy and security enhancing techniques can be combined with pseudonymization. Noise addition strengthens the pseudonym by adding pseudo-random noise to the generation in addition to the original value [17]. However, anomaly detection or similar techniques may counter the noise. Salting adds an entropy, i.e. salt, to the pseudonym generation process to mitigate dictionary attacks and therefore reduces the risk of re-identification [1]. The salt value can either be generated deterministically from a given value [1], or as a random value [235].

3.5.3 *Reasoning on Pseudonymization*

Summarizing, pseudonymization has several noteworthy benefits. First, data values are always transformed consistently, assuming the same method is used, allowing distributed generation of pseudonyms that are uniform. Pseudonyms can be generated such that their format is preserved and thus no adaptation of the applications is required to store and process pseudonyms. Furthermore, different data types are supported. Based on the chosen algorithm, the strength of pseudonymization can be high, especially when independent pseudonym generation is chosen. A great strength of pseudonyms lies in their ability to be reversible, due to bijective mapping or reversible pseudonym generation. Lastly, the generation of pseudonyms, especially in comparison to anonymization, is relatively simple demanding only a small CPU cost [251] [266].

The selection of the pseudonymization method is highly dependent on the intended use case. Several aspects have to be considered, including the format and length of the original data value, the intended usage of the pseudonym, the performance requirements, as well as the security and privacy requirements. If pseudonyms shall be reversible, the secure storage of the secret is essential to avoid unauthorized re-identification.

Similarly to anonymization and privacy models, no de-facto best pseudonymization method can be derived, because various options are available (see Table 3.20). Therefore, a privacy language expressing pseudonymization methods has to generically express several types, while the secure storage of the secrets or bijective mappings has to be handled by an overarching application. Next, related works on privacy languages are classified utilizing the previously detailed requirements for a privacy language (see Chapter 2).

Table 3.20: Summary of approaches for pseudonymization.

Pseudonym Generator Method		
Reversible Generation or One-Way Generation	Independent or Dependent	Random Seed or Cryptographic-Function or Hash-Function
Bijective-Mapping		

4

RELATED WORK

In this chapter, the derived set of requirements for a privacy language modelling privacy policies of Chapter 2 is used to to evaluate and classify state-of-the-art privacy languages. Hereby, a research gap of the integration of de-identification methodology in privacy languages is identified. Furthermore, the approach for the *Layered Privacy Language* is positioned within the state-of-the-art literature.

4.1 CLASSIFICATION OF PRIVACY LANGUAGES

Several privacy languages have been proposed in the literature, each with their own distinct focus. In this chapter, the classification for privacy languages is based on a broad literature research as well as the previously defined requirements (see Chapter 2).

Indeed existing classifications for privacy languages [164] [151] lack a focus on privacy but privilege a security focus.

Kumaraguru et al. [164] classify privacy languages according to their target use cases as *Sophisticated Access Control Languages*, *Web Privacy Policy Languages*, *Context Sensitive Languages*, and *Enterprise Privacy Policy Languages*. Furthermore, they create a sub-categorization for each of their categories, except *Enterprise Privacy Policy Languages*, denoting if the privacy language expresses the privacy requirements of the *User*, e.g., privacy preferences, or of the *Enterprise*, e.g., privacy policies.

Morel and Pardo [188] surveyed privacy languages and classified them according their *Features*, *Audience*, i.e. *Data Subject* or *Controller*, *Conditions*, i.e. if *Time* or *Space* constraints can be defined, and *Content*. For *Features* it is considered if the language is intended for *Usability*, if the*Syntax* is XML or a formal language is used, if *Enforcement* of the policy is defined, if the privacy language is *Implemented*, and which *Tools* are available.

Kasem-Madani and Meier [151] introduce a multidimensional categorization for security and privacy languages. The first category is the *Type* of the privacy language which considers the intended purpose of the privacy language in a similar way to the survey of Kumaraguru et al. [164]. They differentiate for the *Type* between *Security*, *Accountability*, *Availability*, *Privacy*, *Data carriage*, *Data usage control*,

and *Network and device management*. Compared to Kumaraguru et al. [164], they consider in the second category, i.e. the *Intention of Use*, not only the *User requirements* and the *Enterprise policies*, but also consider *Multiple parties interaction*, e.g., policies that are intended to be used from a user and company perspective. The category *Scope* represents the number and scope of actors of the use case, which has the sub-categories *Data exchange*, *Service requester/service provider*, *Agreement descriptions*, *Authorization*, *Access control*, and *Application Monitoring*. Lastly, they introduce sub-categories for *Design and Implementation Details* which are composed of *Usability*, *Context sensitivity*, *Syntax*, and *Extensibility* sub-category. It can be seen that privacy languages are more fine-grained classified in this approach, but the basic principle of classifying privacy languages according to their intended use case remains alongside with the focus on the user or the company.

This concept is followed but the overall complexity of the classification is reduced by considering only the intended usage and the previously introduced requirements for a privacy language (see Chapter 2). Therefore, a classification of privacy languages is developed according to their intended usage (see Figure 4.1). Hereby, it is differentiated between a focus on *Privacy* and *Security* of the privacy language based on Gerl et al. [121]. It is further differentiated according to the specific intended purpose of the privacy language. Privacy languages with a *Security*-focus are differentiated in *Access Policies* and *Service-Level-Agreement (SLA) Policies* (see Table 4.1). Privacy languages associated to *Access Policies* intend to prevent unauthorized access to data or files utilizing access control mechanisms. Although access control is required to ensure privacy, it is designated to the security focus, because it was developed more within the security domain. *SLA Policies* describe contracts and agreements for, e.g., B2B processes in order to ensure a reliable and secure environment. Hereby *SLA policies* can express various requirements both functional and non-functional.

Privacy languages with a focus on *Privacy* are differentiated according to their intended purpose in *Privacy Policy Transparency*, *Privacy Policy Preferences*, and *Privacy Policy Enforcement* (see Table 4.2).

In the following, noteworthy privacy languages are shortly presented and classified according to requirements for a privacy policy expressing privacy language from Chapter 2 while the privacy languages are differentiated based on their focus on privacy or security.

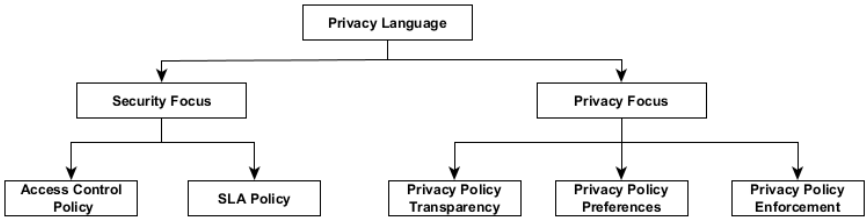


Figure 4.1: Classification of privacy languages in categories with *Security-Focus* and *Privacy-Focus* [121]. The classification is based on the works of Kumaraguru et al. [164] and Kasem-Madani and Meier [151].

4.1.1 *Access Control Policy*

Access Policies are categorized as a subset of privacy languages with a focus on security (see Table 4.1). Their main intention is to formulate rules that define which entities can access which resources, such that only authorized accesses are possible. Therefore, they mainly fulfil the *Privacy Policy Structure* and *Access Control* requirement.

Examples for *Access Control Policy* languages are *XACL* [128], *Ponder* [78], *Rei* [145], *Polymer* [28], *SecPAL* [30], *AIR* [156], *XACML* [96] and *ConSpec* [11], which are presented and discussed in the following according to the requirements for a privacy language.

The *XML Access Control Language (XACL)* specifies security rules to be enforced on specific XML documents. These security rules define CRUD (Create, Read, Update, Delete) actions on objects of the XML document. The core element is *ACL* defining access to *Objects*. An *XACL Policy* contains several *Rules* each holding several *ACL* elements. Because several *ACLs* can be defined for the same *Object*, conflict resolution is implemented [128].

Ponder defines role-based access control (RBAC) for networks and distributed systems. It supports obligation policies that are event triggered condition-action rules. Supported operations for policies are authorisation, delegation, information filtering and refrain. The base structure of *Ponder* consists of an *Operation*, e.g., the authorization given to a *Subject* on a *Target* with a corresponding *Action*. Roles are hierarchically structured and defined via a tree [78].

Contract Specification Language (ConSpec) allows fine-grained access control on sensitive resources of, e.g., mobile devices, and is based on an automata formalism. For the language design, a trade-off between clean semantics and expressiveness has been made. A *ConSpec* policy selects a set of acceptable executions from all defined executions and can therefore define different conditions under which sensitive resources can be accessed [11].

Rei is based on deontic logic, i.e. formalism for obligations and permission, for access control, which is defined with RDF (Resource

Description Framework). A policy is a tuple of *Action* and *Condition*, whereas *Action* and *Condition* are respectively a domain specific action and restriction on the actor or environment. The policy is referencing both a *Subject* and a *Policy Object* allowing RBAC on objects, each with a unique identifier. Additional constructs for *Rights*, *Prohibitions*, *Obligations* and *Dispensatios* exist [145].

Similar to *Rei*, *Accountability in RDF (AIR)* is a privacy language intended for the semantic web. Nested rules express the reasoning for actions to perform. Furthermore, *AIR* allows the explanation of inferred information and contextualized reasoning. AIR allows the specification of natural language explanations for rules. Lastly, *AIR* supports *Linked Rules*, so that rules can be reused in a manner similar to *Linked Data* [156] [146].

SecPAL is a logic-based privacy language which supports complex access control requirements of large scale distributed systems. Predicates are used to define policies and credentials as triples. Access requests are mapped to logical authorization queries, consisting of predicates and constraints combined by conjunctions, disjunctions, and negations. Thus, fine-grained access control to data can be defined, but purposes cannot be expressed [30].

The *eXtensible Access Control Markup Language (XACML)* is a generic privacy language for fine-grained policy-based access control, which is standardised. *XACML* policies can be defined and managed in a distributed manner, such that different individuals can manage separate parts of the policy, while a combined access decision can be found. In its core, *XACML* stipulates boolean expressions (*Conditions*) with corresponding *Effects* in the form of *Rule*-elements. *Rule*-elements are combined within a *Policy*-element to form access rules, which can be merged within a *Policy-Set*. *XACML Rules* operate on *Attributes*, which define known values by identifier, e.g., a document that should be accessed [96].

Polymer is different from other *Access Policies*. Instead of expressing access rules, it monitors actions with triggers. If an untrusted application intends to execute a secured action, it is trigerred and a suggestion for handling the incident is generated [28].

The presented privacy languages can be classified as *Access Policies*, because they allow the regulation of access to (personal) data in different domains, e.g., *XACL* protects access to XML documents, *Rei* is intended for the semantic web, while generic approaches exist, e.g., *XACML*. It can be observed that all presented languages, except *Polymer*, fulfil the *Access Control* requirement partly, whereas different access control schemes are used, e.g., RBAC is used by *Ponder* and *Rei*. The *Access Control* requirement is only fulfilled partially, because the legally required purpose is not considered for the access control decision.

Table 4.1: Overview of fulfilled requirements of related privacy languages of the *Access Control Policy* and *SLA Policy* category. An 'X' denotes the fulfilment of the requirement, an '(X)' denotes the partial fulfilment of the requirement, and a '-' denotes that the requirement is not fulfilled.

Category	Privacy Language	Privacy Policy Structure	Legal Compliance (GDPR)	Human-readability	Access Control	De-identification Capabilities	Provenance
Access Control Policy	XACL	X	-	-	(X)	-	-
	Ponder	(X)	-	-	(X)	-	-
	Rei	(X)	-	-	(X)	-	-
	Polymer	(X)	-	-	-	-	-
	SecPAL	(X)	-	-	(X)	-	-
	AIR	X	-	X	(X)	-	-
	XACML	X	-	-	(X)	-	-
	ConSpec	(X)	-	-	(X)	-	-
SLA Policy	SLAng	X	(X)	-	-	-	-
	USDL	(X)	(X)	X	-	-	-

The *Privacy Policy Structure* can be supported completely by *XACL*, *AIR*, and *XACML*, while the others can be interpreted on focusing only on data or purposes. The *R1 Privacy Policy Structure* requirement is fulfilled by several *Access Control Policy* languages, e.g. *AIR* or *XACML*, but others are missing either the explicit definition of purposes or attributes. The requirement *R3 Human-readability* is only considered by *AIR*, which allows for human-readable descriptions of its rules. The requirements *R5 De-identification Capabilities*, *R6 Provenance*, or *R2 Legal Compliance* in the context of the *GDPR* are not covered by any of the presented privacy languages. The *Access Control Policy* languages are specialized on in detail access control mechanisms and show that the *R4 Access Control* requirement is covered well in the literature (see Table 4.1). Especially in the emerging field of IoT, various promising approaches for fine-grained access control can be observed [204] [272] [282] [12]. Because this requirement is well covered in the literature, only basic purpose-based access control mechanisms are realized for LPL. The extension of LPL by additional access control elements is possible, e.g., integrating obligations, conditions, or effects.

4.1.2 *SLA Policy*

Service Level Agreement (SLA) Policy languages can be categorized as a subset of privacy languages with a focus on security (see Table 4.1). Their main intention is to formulate agreements or contracts for B2B processes. Therefore, they consider aspects of the *Privacy Policy Structure* and are focused on *Legal Compliance*.

Examples for *SLA Policies* are *SLAng* [166] [184] and *USDL* [202].

SLAng describes SLAs, which accommodate end-to-end quality of service (QoS), typically for B2B cloud use cases. QoS has many facets and requires complex agreements between network, storage and middle-ware services. This includes the definition of the retention of data, which is part of the requirements for legal compliance [166] [184].

The *Unified Service Description Language (USDL)* describes business, operational and technical parameters of services while context specific legal requirements, e.g., terms of use or copyright, are taken into account. Its *Legal Module* addresses the need for legal compliance in service networks and in trading services on marketplaces. *USDL* is designed to incorporate business processes that can be easily comprehended by its users [202].

Although, *SLA Policies* can be classified as privacy languages, their main focus is formulating inter business agreements instead of policies between users and companies. But, this is also essential in terms of the legal framework of *GDPR*, which states that technical and organizational measures have to be applied and documented to ensure

secure and private processing of personal data [110, Art. 32, Recital 78].

SLAng may be mapped to the structure of a privacy policy, whereas *USDL* fulfils this requirement only partially. Both, *SLAng* and *USDL* only partially cover legal compliance of privacy policies through their definition of data retention. *USDL* can be presented in an easily accessible human-readable format, which covers the human-readability requirement. Other requirements are not covered by either *SLAng* or *USDL* (see Table 4.1).

4.1.3 *Privacy Policy Transparency*

Privacy languages which specialize in informing about services' privacy conditions or privacy policy are classified as *Privacy Policy Transparency* (see Table 4.2). Therefore, they fulfil the *Privacy Policy Structure* and *Access Control* requirement, furthermore a focus on *Legal Compliance* and *Human-readability* is given.

The privacy languages *P3P* [273],and *CPExchange* [49] are presented for this category.

The privacy language *Platform for Privacy Preferences Project (P3P)* defined a protocol to inform users about the privacy policy of web-sites, e.g., online shops. *P3P* expresses its policies in the machine-readable XML format. Its elements, i.e. *Policy*, *Entity*, *Purpose* and *Data*, can cover the base structure of privacy policies. Legal compliance is partially covered by the *Retention* element. Access control mechanism are enabled by the *Access* element. Furthermore, human-readability is partially given, e.g., by the *CONSEQUENCE*-element, which optionally details the intention of the web-sites. But, *P3P* policies are mainly intended to be automatically compared to machine-readable privacy preferences of users. A separate human-readable privacy policy is required. Although *P3P* has been standardised and has been popular, it has several downsides including a fixed, thus limited, vocabulary, e.g., for purposes, a lack of formal semantics, and the possibility to define conflicting statements within a *P3P* policy [273] [260] [72] [280].

Customer Profile Exchange (CPExchange) is intended to exchange privacy information as meta-data attached to business data, whereas the privacy information is encoded as *P3P* policies. *CPExchange* provides a comprehensive view of the customer as an entity who interacts with multiple facets of a company instead of a user of a specific service of a company. A fine-grained authorization and privacy model supports the exchange of aggregated information of different data stores. Various profile data elements can be associated with multiple privacy policies. Next to the elements of *P3P*, additional information is stored to determine the partners for the exchange of the information as well as jurisdictional definitions. The jurisdictional information is intended

for use cases in which an exchange of data is realized which falls under different legal frameworks [49] [110, Art. 44 - 50].

Comparing the presented privacy languages, it can be observed that the basic structure of privacy policies is considered by all, but the legal compliance is only partially considered. Both *P3P* and *CPExchange* have been proposed before the enforcement of the *GDPR*, thus it is no surprise that its legal aspects could not be fully considered. Both *P3P* and *CPExchange* mostly rely on machine-readable policies with few human-readable information. Lastly, access control rules are considered by both presented privacy languages (see Table 4.2).

Table 4.2: Overview of fulfilled requirements of related privacy languages of the *Privacy Policy Transparency*, *Privacy Policy Preferences* and *Privacy Policy Enforcement* category. An 'X' denotes the fulfilment of the requirement, an '(X)' denotes the partial fulfilment of the requirement, and a '-' denotes that the requirement is not fulfilled.

Category	Privacy Language	Privacy Policy Structure	Legal Compliance (GDPR)	Human-readability	Access Control	De-identification Capabilities	Provenance
Privacy Policy Transparency	P3P	X	(X)	(x)	X	-	-
	CPExchange	X	(X)	(X)	X	-	-
Privacy Policy Preferences	APPEL	X	-	-	-	-	-
	XPref	X	-	-	-	-	-
	XPACML	X	(X)	-	X	-	-
	S4P	(X)	-	-	X	-	-
	YaPPL	(X)	(X)	X	X	-	-
Privacy Policy Enforcement	DORIS	(X)	-	-	X	-	-
	E-P3P	(X)	(X)	-	X	-	-
	EPAL	X	-	-	X	-	-
	PPL	X	(X)	X	X	-	(X)
	SPECIAL	X	(X)	X	X	(X)	(X)
	P2U	X	(X)	-	X	-	-
	PrivPolicy	X	(X)	-	X	(X)	-

4.1.4 *Privacy Policy Preferences*

Privacy languages, which specialize in the expression of the users' privacy preferences regarding services, are classified within the *Privacy Policy Preferences* category (see Table 4.2). Therefore, they focus on the *Privacy Policy Structure*, but also the *Legal Compliance*, *Human-readability*, and *Access Control* requirement is focused on by some privacy languages depending on their intended use.

The privacy languages *APPEL* [179], *XPref* [10], *XPACML* [34], *S4P* [32] [33], and *YaPPL* [264] are detailed and discussed.

A P3P Preference Exchange Language (APPEL) enables users to define privacy preferences. It is intended to support the decision-making if a *P3P* policy complies to a users' privacy preference. The concept of *APPEL* is that a user imports privacy preference rule-sets from third parties or shares its own privacy preference definition. The vocabulary of *APPEL* is limited to the scope of *P3P*. Furthermore, users can only specify what is not acceptable and not what is acceptable, which can be compared to a blacklist and limits the expressiveness of *APPEL*. Lastly, *APPEL* privacy preferences are error prone and hard to define [179] [10].

XPref is a successor to *APPEL* which intends to enhance the decision-making based on *P3P* policies while overcoming the downsides of *APPEL*. It is based on *XML Path Language (XPath)*, which is a query language for selecting nodes from an XML document as well as to compute values from the content of an XML document. This enables a faster matching between *P3P* policies and *XPref* privacy preferences, which are formulated as *Rules* of a *Ruleset*. Furthermore, *XPref* is as expressive as *APPEL* [10].

The *eXtensible Privacy Access Control Markup Language (XPACML)* expresses users' preferences and privacy terms of the service provider to enable negotiation between both. *XPACML* is a combination of the previously presented privacy languages *XACML* and *P3P*, thus inheriting the access control capabilities of *XACML* as well as the capabilities to represent privacy policies of *P3P*. But *XPACML* is lacking support for human-readability, e.g., the expression of the contents of a privacy policy. Furthermore, no mechanisms for de-identification nor provenance are considered [34].

Similar to *XPACML*, *S4P* expresses both users' preferences and privacy policies of the service. *S4P* is based upon the previously described *SecPAL*, which is used to express preferences and policies as assertions and queries written. Additionally, permissions and obligations are introduced. To verify if a users' preference is satisfied by a services' policy, queries are evaluated against the assertions. The abstract vocabulary and semantics of *S4P* enable its expressivness and application in various domains [32] [31] [33].

YaPPL is a privacy preference language which is designed for use in the IoT context to enable *GDPR*-compliant consent. For each preference rule, the recipient of the data as well as the purpose of the processing are expressed. Furthermore, the (temporal) validity of the processing of personal data is explicitly expressed and stored for accountability and archiving purposes. Several preference rules can be combined, which allows dealing with complex scenarii [264].

It can be observed that *APPEL*, *XPref*, and *XPACML* are either based on *P3P* or complementary to it, whereas *XPref* is a successor of *APPEL* and *XPACML* furthermore considers access control mechanisms due to its basis on *XACML*. *S4P* follows a similar concept like *XPACML*, both defining users' preferences as well as the privacy policy of the service. Among the introduced privacy languages, only *YaPPL* considers human-readability as a core feature. *YaPPL* requires human-readability, because it specializes in enabling *GDPR*-compliant consent in the context of IoT. But other legal requirements are also given by the *GDPR*, such that the legal compliance of *YaPPL* to *GDPR* is only classified as partially fulfilled. Neither de-identification nor provenance requirements are considered in any of the presented privacy languages classified under *Privacy Policy Preferences* (see Table 4.2).

4.1.5 *Privacy Policy Enforcement*

Privacy languages which specialize in the realization of privacy, either in terms of the users' privacy preferences or services' privacy policies are classified as *Privacy Policy Enforcement* (see Table 4.2). Therefore, they focus on the *Privacy Policy Structure* and *Access Control* mechanisms to realize privacy guarantees. The *Legal Compliance* and *Human-readability* requirement is focus of only a few *Privacy Policy Enforcement* privacy languages. But only in this category, privacy languages can be found that consider the *Provenance* and *De-identification Capabilities* requirements with basic approaches.

The privacy languages *DORIS* [45], *E-P3P* [19], *EPAL* [20], *PPL* [16], *P2U* [140] [141], *SPECIAL* [54] [53] [50], and *PrivPolicy* [270] are detailed and discussed.

Datenschutz-orientiertes Informationssystem (DORIS) is an information system with focuses on privacy. A data model and a data manipulation language are introduced, which are based on the right of informational self-determination, i.e. users have the authority to limit the extent of sharing information about his private life to others [126]. To the extent of our knowledge, *DORIS* is the first systematic approach to model a privacy language and use it within an information system. The concept considers only an application as its scope, in contrast to a scope that considers the exchange of personal data between various

applications. Hereby, the application details the persons and relationships. A person knows things about himself and his relationship to others. Furthermore, a person is acquainted with other persons representing a social environment. A person, which can have several roles, can furthermore interact with others by sending messages to query information or perform tasks. The privacy policy of *DORIS* models the rights of a person, which consists of acquaintances and role authorities and determines which information is obtained for a query. The privacy language describes persons and their interaction with others through several operations, e.g., *insert*, *tell*, or *revoke*. But the concept of the *Purpose* has not been introduced. Additionally, data protection officers, representing the legal responsibility, are also considered within the concept of *DORIS* [45].

The *Platform for Enterprise Privacy Practices (E-P3P)* defines fine-grained purpose-based access control for personal data. Unlike *P3P*, although the name indicates close similarity, *E-P3P* defines the privacy practices that are implemented inside a company. Because of companies' internal description of details is expressed, the syntax and semantics are more detailed than *P3P* allowing automatic enforcement and audit. *E-P3P* expresses the privacy policy as a set of terms and rules, which defines actions, obligations, and conditions on data groups for specific purposes and users. Although purposes are defined, only data groups instead of specific data items are addressed by *E-P3P*. *E-P3P* extends *P3P* mainly by access control rules to be enforced, such that an automatic translation of *E-P3P* policies to *P3P* policies is possible to inform users (externals of the companies) about the processing of personal data [150] [19]. *E-P3P* could also be considered as an *Access Control Policy* language, but the focus lies more on the enforcement of privacy policies, which includes access control.

The *Enterprise Privacy Authorization Language (EPAL)* intends to allow companies only to process data according to legal regulations. Therefore, actions on purposes are defined which the user has to consent to. *EPAL* includes authorization mechanisms, while both the data model and authentication are abstracted. Hereby, *EPAL* defines hierarchies of data, user and purposes which are used as the basis for access decisions. A peculiarity of *EPAL* is that the rules are sorted by descending precedence, such that more specific rules, e.g., for a single user, should be put before more general rules, e.g., for a department of a company, to realize exceptions [20] [255] [14].

The *PrimeLife Policy Language (PPL)* is an extension of *XACML*, thus inheriting its access control capabilities. *PPL* focuses on the description of the usage of personal data of users by companies. It tackles the following main challenges:

- Detection of inconsistent policies: The policy creator is warned about missing obligations, e.g., missing retention of personal data.

- Consideration and expression of users' privacy preferences: The user can agree to a full policy or consent to parts of the privacy policy by ticking check-boxes.
- Expression and enforcement of obligations: The policy denotes obligations to users, linked to their personal data, that have to fulfilled, thus are enforced.

When data is transferred to third parties, denoted as *downstream users*, the policy is attached to the personal data, thus enabling the enforcement of *PPL* during secondary use, which is only interpreted as a partial fulfilment of the provenance requirement, because the *Data Production* use case is not considered. The linkage of personal data with a corresponding policy is denoted in the literature as the *sticky* policy concept [207]. To enable the negotiation with users the *Send Data?* user interface has been introduced, visualizing the usage of personal data for different purposes in a layered approach [16] [261] [15].

The goal of the *Scalable Policy-aware Linked Data Architecture For Privacy, Transparency and Compliance (SPECIAL)* project is to address the contradiction between the processing of personal data to create innovative technology and the preservation of users' privacy. Trust in the processing of personal data should be enabled due to transparency while companies can gain insights from the processing of personal data. Within the *SPECIAL* project, a usage policy language [54] and a policy log vocabulary [53] are defined, which are based on *PPL*. Usage policies describe the general usage of data for a specific purpose, whereas the operation itself is described, where and how long the data is stored and which entities can access the data. Thus, the basic privacy policy structure is given. Furthermore, the processing of data can be expressed, e.g., the anonymization or transfer of data [51]. Therefore, the expression of de-identification of data is partially possible with anonymization, but no details or scope of possible anonymization methods is given. Furthermore, the expression of pseudonymization methods is missing. Lastly, the vocabulary of the usage policy language of SPECIAL does take into account consent, but fine-grained control on the quality, i.e. anonymization level, of attributes is not given (see Section 3.4).

Although compliance to *GDPR* is striven for, the usage policies express only basic information on the processing of personal data, e.g., the identity and the contact details of the processing company is not expressed [110, Art. 13(1), Art. 14(1)]. Furthermore, the processing of personal data is logged using the policy log vocabulary. This enables users to trace the processing of their personal data according to their consent decisions, while the company has a full documentation which can be used in possible reviews by supervisory authorities to prove the compliance to *GDPR*. Thus, *Provenance* is also partially fulfilled within

the scope of a *SPECIAL* enabled application because it can be tracked down from which individual personal data is processed for which purpose, but the exchange and processing of personal data beyond the application cannot be fully traced. Appropriate user interfaces for consent-management and transparency are introduced based on the privacy languages of *SPECIAL* [227] [54] [53] [50].

The *purpose-to-use (P2U)* privacy language, which is inspired by *P3P*, is designed for secondary data sharing of personal data. Therefore, its focus lies on the negotiation of the value or price of personal data between companies. The user is assumed to explicitly edit the policy which defines the sharing of his personal data for a specific purpose, with a retention period and price. The definition of retention is hereby interpreted as a partial fulfilment of the legal compliance requirement. This negotiation process is core of *P2U* for which an overarching framework is proposed [140] [141].

PrivPolicy [270] is a policy specification language inspired by *LEGALEASE* [243]. With *PrivPolicy* a set of *clauses* can be defined, each defining which personal data can be processed for which purpose by which data recipients. Instead of a purpose-based approach for defining the *clauses*, a data-based approach is used. Furthermore, in *PrivPolicy* it can be defined if the user is notified or consent is required, but no human-readable texts can be defined. Privacy guarantees, e.g., the use of a privacy model, can be defined by the *DECLASS* attribute. But, this only partially fulfils the *De-identification Capabilities* requirement, because 1) it is undefined which attributes are anonymized or pseudonymized and 2) personal privacy settings cannot be defined for specific attributes. *PrivPolicy* is used for privacy compliance checking and enforcement via an overarching framework *PrivGuard*.

It can be observed that *EPAL*, *PPL*, *SPECIAL*, *P2U*, and *PrivPolicy* fulfil the requirement set regarding the privacy policy structure. But *DORIS*, as the probably first privacy language, and *E-P3P* both lack structure. Expression of several legal frameworks is considered by *E-P3P*, *P2U* inherits the correlating properties of *P3P*, while *PPL*, *PrivPolicy* and *SPECIAL* consider some concepts of *GDPR* but not all. Furthermore *PPL* is considered for interaction with users through the *Send Data?* user interface; this capability is also inherited by the *SPECIAL* privacy languages. *SPECIAL* introduced user interfaces for consent-management and transparency. The usage of *sticky* policies is considered by *PPL* and *SPECIAL*, thus the provenance requirement is considered as partially fulfilled by these privacy languages. All presented privacy languages of the *Privacy Policy Enforcement* category utilize access control mechanisms for their enforcement. Only *SPECIAL* and *PrivPolicy* express the processing of personal data, i.e. anonymization of data. This is only a partial fulfilment of the requirement denoting the the enforcement of de-identification methods on

personal data (see Table 4.2), because it is undefined which anonymization (or de-identification) methods (see Chapter 3) can be expressed. Furthermore, it is unclear if the concept of personal privacy can be expressed. Thus, it can only be speculated which de-identification methods can be expressed with the privacy languages of *SPECIAL* and *PrivPolicy*.

4.1.6 *Discussion*

A classification of privacy languages in the categories *Access Control Policy*, *SLA Policy*, *Privacy Policy Transparency*, *Privacy Policy Preferences*, and *Privacy Policy Enforcement* was detailed. Furthermore, each related privacy language was categorized according to the previously given requirements for a privacy language expressing privacy policies in Chapter 2.

It can be observed that privacy languages not only focus on the modelling of privacy policy, but also on correlated properties like technical and organisational measures via *SLA Policies*. Furthermore, access control mechanisms, including authentication and authorization, are currently the state-of-the-art mechanism to preserve privacy within privacy languages. Hereby, various access control mechanisms have been proposed to enable the definition of rules expressing basically what data can be processed by which entity for which purpose.

Informing the individual transparently has been factored in by relatively few privacy languages in the past, but has become an essential feature for more current privacy languages like *PPL* or *SPECIAL*. Hereby, a strong effort is made in creating privacy languages that allow users to interact with the privacy policy to express their personal privacy requirements and to express consent, which is an essential requirement for legal compliance to *GDPR*.

The sharing and trading of personal data is furthermore considered by several privacy languages. *P2U* proposes a concept to enable the negotiation of a price for personal data trading, while *PPL*, *A-PPL* and *SPECIAL* consider the processing of personal data for secondary use. Hereby, the concept of *sticky* policies is introduced to link personal data with the corresponding data even after it has been transferred.

The legal compliance according to *GDPR* is partially fulfilled and partially pursued by several privacy languages, but especially regarding the privacy policy the presented privacy languages lack expressiveness to define all required information. For example, the possibility to define the responsible *Data Protection Officer* and *Controller* is missing, or the definition of automated decision-making is not possible.

Finally, all of the given requirements for a privacy language have been considered by a privacy language in the literature, but the introduction of de-identification capabilities for a fine-grained control over the quality of a data that is to be processed in a privacy-preserving

way is only schematically introduced by *SPECIAL* and *PrivPolicy*. This is identified as a research gap, which is addressed within this work in order to allow users not only to control access to their personal data but also to control the quality of the processed personal data. Considering this research gap, the approach of this work is positioned to the related works in the following.

4.2 POSITIONING

In the following, this work is positioned according to the privacy language requirements and related works. Requirements for a privacy language that models privacy policies according to *GDPR* are defined in Chapter 2. According to these requirements, a privacy language has to model the base structure of privacy policies (*R1 Privacy Policy Structure*). Furthermore, the privacy language has to comply with legal requirements, i.e. the *GDPR* (*R2 Legal Compliance*). To allow users to easily understand the privacy policy, the privacy language has to be human-readable, for which both a textual and visual representation of its contents is considered by the *R3 Human-readability* requirement. Furthermore, to enable the expression and enforcement of privacy guarantees, a privacy language has not only to be able to integrate *R4 Access Control* mechanisms, but also *R5 De-identification Capabilities* considering the personal privacy concept (see Section 3.4). Additionally, when data is transferred or processed, the *R6 Provenance* has to be maintained such that the user can identify himself as the owner of the personal data and claim his *Data Subject Rights*, while companies are accountable and can show their compliance to the legal regulations.

In this chapter, state-of-the-art privacy languages have been categorized and classified according to the introduced requirements of Chapter 2 and concluded that no privacy language fulfils all of them to the full extent. The requirements of the *Privacy Policy Structure* and *Access Control* have been fully covered by most privacy languages in the literature. Furthermore, *Legal Compliance* has been considered by several privacy languages while only few considered the legal framework of the *GDPR* due to its relatively novel enforcement. But, the legal requirements for the content of a privacy policy, have not been fully covered by any privacy language. Furthermore, the *Provenance* requirement is partially covered due to the usage of *sticky* policies. Lastly, the requirement *De-identification Capabilities* has not been covered by any privacy language to its full extent, which is identified as an important research gap.

This research gap is addressed by extending the shown approaches in the literature for privacy, which mainly focus on access control mechanisms, to express privacy requirements in a privacy policy with the definition of de-identification methods. Therefore, privacy is not only the definition of binary access decisions on personal data, i.e.

as it is realized in the literature, but also fine-grained control on the quality of information that is shared and processed. Hereby, a dialectic approach on privacy has to be considered taking into account the personal privacy requirements of the users and the data quality requirements for processing of personal data for specific purposes.

To preserve the individuals' privacy during processing, de-identification methods are applied. Furthermore, the concept of personal privacy, i.e. the possibility that the user can influence the quality of his personal data, complements the existing concept of consent management, i.e. the binary decision for processing, by additional fine-grained decisions (see Section 3.4). Hereby, the intention is that the user can directly influence the anonymization level of each of his attributes, thus denoting the quality of personal data that is processed. Taking into account the trade-off between privacy and utility, the business processes of a company may require a minimum data quality for processing. Therefore, the company has to be able to set a limit for the anonymization level to cap off the minimal required data quality for a specific purpose. By utilizing a privacy language for the expression of these personal privacy preferences, it can be assuming that the company first defines the privacy policy and sets limits for data quality. Then, users are presented the privacy policy and are allowed to negotiate, i.e. personalize, the privacy policy to express their personal privacy settings. If a user requires a higher anonymization level than allowed by the company, the user is free to decide not to consent to the purpose or even not to use the service of the company.

For example, considering the earlier introduced anonymization hierarchy for the postal-code (see Figure 3.2), the creator of the privacy policy would define the maximum anonymization level for the attribute such that it is still usable for the intended purpose, e.g., to level '3'. The user can then personalize his personal privacy settings for postal-code as long as it does not exceed the defined maximum level of '3'. Thus, a user defining his *Minimum Anonymization Level* to '1' would create a valid personalization (see Figure 4.2). Contrary, a user defining his *Minimum Anonymization Level* to '4' would break the policy requirements which would not lead to an agreement on the policy between the user and the company. The inclusive range between *Minimum Anonymization Level* and *Maximum Anonymization Level* expresses the range of possible values for the later anonymization of the value.

Furthermore, fine-grained personalization of privacy policies is considered, which bears challenges for the processing of personal data, especially the de-identification process. Assuming a data-set with a million records is processed for a specific purpose in which de-identification is required, each record has a personalized *sticky* policy expressing differing privacy guarantees that has to be considered

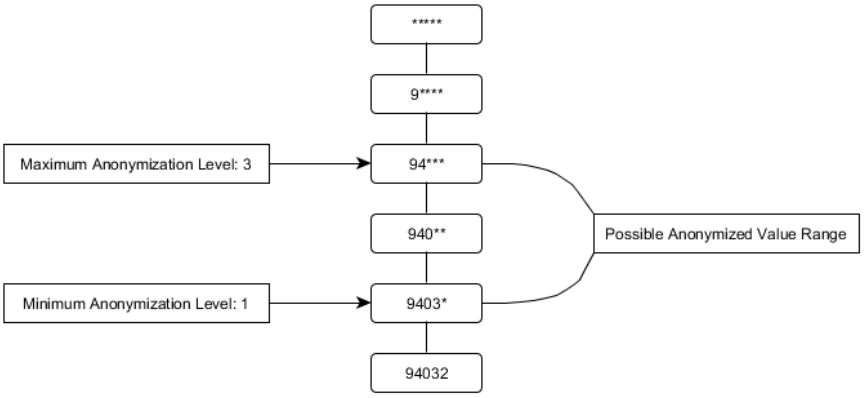


Figure 4.2: Anonymization hierarchy for the German postal-code for Passau '94032' (see Figure 3.2) with both the *Minimum Anonymization Level* '1' and *Maximum Anonymization Level* '3' detailed. Furthermore, the possible range for values during later anonymization is denoted.

during the de-identification process. But also the data quality limit set by the company for the process has to be considered. Therefore, several aspects for the guarantee of the individuals privacy have to be taken into account.

First, the combination and integration of de-identification processes has to be achieved in a way that they do not interfere with each other. This includes pseudonymization, personal privacy anonymization and application of privacy models. Especially, the combination of personal privacy anonymization and privacy models has to be carefully considered, because they influence the same data. In the same way, also pseudonymization, which replaces EI attributes with pseudonyms, and privacy models, which usually delete EI attributes, influence each other.

Second, the expression and control of personal privacy decisions, either binary consent or more fine-grained data quality control, in a machine-readable privacy language has to be achieved, such that the user can directly influence the processing of his personal data at any time. Therefore, classical assumptions that personal data is de-identified once for a defined privacy level are no longer applicable, because the privacy level can change at any time due to the direct influence of the user on it. Thus, the definition of privacy is dynamic over time for a data-set and has to be reconsidered each time the data-set is processed possibly altering the de-identified data-set T'. Furthermore, it has to be taken into account that the privacy and utility requirements can vary for different purposes.

Lastly, the execution of the de-identification of the whole data-set, while for each record an additional personalized privacy policy has to be achieved, which poses the risk of a significant execution time overhead compared to the fixed privacy definition approach,

e.g., usage of *3-Anonymity*. Hereby, both the processing of additional data, i.e. privacy policies expressed by the privacy policy, and the introduction of varying privacy definitions in the de-identification process may pose an exhausting processing overhead. Thus, the the influence of both the volume of privacy policies and the effect of personal privacy on the execution time of the de-identification process is evaluated to demonstrate the feasibility of this approach.

To enable such a policy-based de-identification process, a suitable privacy language is required. Therefore, the following chapter details the privacy policy expressing *Layered Privacy Language (LPL)* that defines privacy guarantees using de-identification methodology, while the *R1 Privacy Policy Structure*, *R2 Legal Compliance* to *GDPR*, *R3 Human-readability*, *R4 Access Control* and *R6 Provenance* are considered. The fulfilment of the requirements is either discussed in detail or demonstrated with proof-of-concept implementations. The realization of the *R5 De-identification Capabilities* requirement is covered by the *Policy-based De-identification* process in Chapter 6. With the fulfilment of the requirements for a privacy language (see Chapter 2), the first research question *RQ1* is addressed.

The second research question *RQ2* is addressed by the *Policy-based De-identification (PD)* process, which demonstrates how various individuals' privacy settings defined by LPL are used to de-identify a data-set in which each record has a linked LPL privacy policy. The efficiency of the PD process is quantitatively evaluated in comparison to the sole anonymization of a data-set to match the properties of a privacy model (see Chapter 7). Furthermore, the evaluation details the effects of various personal privacy settings on the run-time of the PD process.

5

LAYERED PRIVACY LANGUAGE (LPL)

In this chapter, the derived requirements for a privacy policy in Chapter 2 are leveraged to detail the reasoning for the proposed *Layered Privacy Language (LPL)*, the formalization of LPL and its intended life-cycle. Furthermore, detailed examples are given which show the fulfilment of the requirements of Chapter 2.

5.1 CONCEPTS

In Chapter 2, the requirements *R1 Privacy Policy Structure*, *R2 Legal Compliance*, *R3 Human-readability*, *R4 Access Control*, *R5 De-identification Capabilities*, and *R6 Provenance* for a privacy language that aims to express privacy policies have been introduced. In Section 4.1, it has been shown that other works in the literature, but the *R5 De-identification Capabilities* has been never the focus. Therefore, the focus is on this research gap, while considering the other requirements. Next, the core of the privacy language, the *R1 Privacy Policy Structure*, is detailed.

5.1.1 *Privacy Policy Structure*

In Section 2.1, the basic privacy policy structure has been introduced consisting of the *Privacy Policy*, *Data Source*, *Data Recipient*, *Purpose*, and *Data*. Naturally, this structure as the backbone for the LPL.

The root-element is the *LayeredPrivacyPolicy*, which corresponds to the *Privacy Policy* (see Figure 5.1). Furthermore, a *DataSource*-element and a set of *Purpose*-elements is defined to be referenced by the *LayeredPrivacyPolicy*. The *DataSource*-element can hereby represent the user, from whom the data originates, or a company which holds the data and shares it, e.g., with other entities. The *Purpose*-element represents a legal purpose, denoting a set of *DataRecipient*-elements and a set of *Data*-elements. A *Data*-element corresponds to exactly one attribute, e.g., a column in a relational database, such that for each *Purpose*, one can define which attributes are processed. The *DataRecipient*-element denotes the entities that have the right to process the *Data* for the specific *Purpose*, e.g., a company, department, or an individual person.

Thus, the *R1 Privacy Policy Structure* requirement is fulfilled. Next, the *R2 Legal Compliance* requirement is considered.

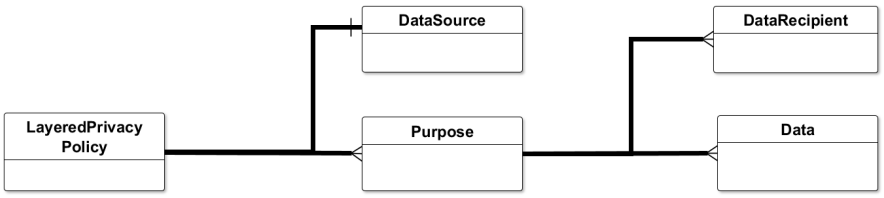


Figure 5.1: Core privacy policy structure of the Layered Privacy Language.

5.1.2 *Legal Compliance*

Fundamentally, the usage of privacy languages for the representation of privacy policies is allowed according to the *GDPR*, because it falls under the category of an electronic form [110, Art.12 (1) Sentence 2].

Moreover, to comply with the legal framework of the *GDPR*, several aspects have to be considered. In Section 2.2, the information that has to be provided according to Art. 13 and 14 is pointed out. To provide this information, elements are created according to the detailed requirements given in Table 2.1 and added to privacy language.

Moreover, to enable *Data Subject Rights*, which can be interpreted as actions performed upon the agreed privacy policy and provided personal data, is the second goal to achieve. It is necessary that the user can be identified and authorized to make use of his *Data Subject Rights*. For example, if a user wants to make use of his *Right of Access by the Data Subject*, then several details about the processing of the personal data of the user has to be provided in a human-readable format including the categories of data and the purposes. To answer such a request, LPL is intended to contain all necessary information.

A detailed analysis of the fulfilment of the *R2 Legal Compliance* requirement is given in Section 5.4.2.

5.1.3 *Human-readability*

To enable *R3 Human-readability* of a privacy language the introduction of human-readable texts and *Privacy Icons* is proposed (see Section 2.3).

To introduce human-readable text in an uniform way for elements, the abstract super-element *UIElement* is defined from which all other human-readable text requiring elements inherit, e.g., *LayeredPrivacyPolicy* or *Purpose*. To enable internationalization support, i.e. support for multiple languages, the *UIElement* denotes a set of *Header*- and *Description*-elements. The *Header*-element defines a human-readable header and the language of the header while the *Description*-element defines a human-readable description for the element as well as the language for the description. Thus, all essential information are provided with human-readable texts in order to enhance transparency of privacy policies. Naturally, the privacy language cannot guarantee

that the human-readable texts are produced in an easy to understand manner such that the target audience is sufficiently informed.

Furthermore, in LPL the expression of *Privacy Icons* is possible. Because no standardization of icons is achieved yet (see Section 2.3), a generic definition of *Privacy Icons* is realized in LPL through the introduction of the *Icon*-element. The positioning of the *Icon*-element within the LPL privacy language has to be decided on. *Privacy Icons* are intended to give meaningful overview of the intended processing [110, Recital 60]. Therefore, the *Icon*-element can be either referenced by the *Purpose*-element denoting the intended processing of the specific purpose or it can be referenced by the *LayeredPrivacyPolicy*-element denoting the intended processing of the policy. The reference of a set of *Icon*-elements by the *LayeredPrivacyPolicy*-element is used to give a distinct overview of the processing. Furthermore, the *Icon*-element inherits from *UIElement*, allowing for detailed description of their meaning in the context of the specific policy.

Lastly, the presentation of the privacy policy has to be considered for which the *Layered* approach has been suggested. A prototype implementations for user interface to create, present and negotiation LPL privacy policies is detailed in Section 5.6 to demonstrate the fulfilment of the *R3 Human-readability* requirement.

5.1.4 *Access Control*

In general, a privacy policy expresses what personal data can be used by which entities for which purpose. To ensure only authorized access, these entities, i.e. expressed via the *DataRecipient*-element, have to authenticate themselves. Similarly, the *DataSource* has to be able to perform the same actions to, e.g., make use of his *Data Subject Rights*. Furthermore, the *Controller*-element and *DataProtectionOfficer*-element are also concerned by access control mechanisms, e.g., the *Controller* can create a new privacy policy. Therefore, the *R4 Access Control* requirement affects several aspects of LPL.

The intention is to create LPL such that it is agnostic to the authentication method, e.g., private/public keys or user-name/password can be used, and authorization method, e.g., RBAC. In general, an identifier and a secret are required for authentication and authorization which is uniformly defined via the super-element *Entity*. Other elements – *DataSource*, *DataRecipient*, *Controller*, and *DataProtectionOfficer* – inherit from this element.

Additionally, to the defined entities, the *Purpose* and *Data* is considered during the authorization. Thus, only the specified set of *Data* can be processed for a specific *Purpose*. The integration of elements detailing obligations, conditions, or effects for access control are omitted for LPL, because they are covered by related works like *XACML* [96]. An

extension of LPL by such elements is possible. The fulfilment of the *R4 Access Control* requirement with LPL is shown in Section 5.5.

5.1.5 *De-identification Capabilities*

In Chapter 3, pseudonymization, anonymization, privacy models and personal privacy have been introduced. All these concepts are related to the *R5 De-identification Capabilities* requirement. Each of these de-identification methods has its distinct usage for preserving privacy, which are all intended to be expressed via LPL.

The concept of personal privacy allows the user to specify their own required privacy requirements, which can differ from other users. Therefore, for each user, a personal privacy policy has to be defined. In the literature, personal privacy has been either expressed by binary consent decisions for the whole record of the user or by allowing the user to specify a desired anonymization level for his sensitive attributes (SD).

Not only are both variants considered, but they are also extended to a holistic approach for personal privacy in the context of privacy policies. First, LPL allows the user to consent to the processing of his personal data based on the *Purpose* of the process, which corresponds to the notion of binary consent decisions. Hereby, the attribute `required` defines if the *Purpose*-element is necessary for the usage of the service or if it is optional and requires consent. This allows companies or service providers to distinguish between required and optional purposes of processing, while users are enabled to control for which purposes their data are processed. Furthermore, the personal privacy concept is adopted to the *Data*- and *DataRecipient*-elements. Thus, users can make fine-grained decisions on which of their personal data is processed for a specific purpose and by whom, provided that the *Data* or *DataRecipient* are optional.

Additionally, personalization of the anonymization level for each attribute is enabled and not only for SD attributes to enable the user to define for each attribute how it is processed. Therefore, for each *Data*-element, an *AnonymizationMethod*-element is referenced which defines how the value can be anonymized. Within the *AnonymizationMethod*-element both a *Minimum Anonymization Level* and *Maximum Anonymization Level* is specified. The *Maximum Anonymization Level* is set initially by the creator of the policy, e.g., a company, and defines the maximum anonymization level that can be set by the user. In other words, the minimal data quality level required for a specific purpose is defined by the *Maximum Anonymization Level*.

The user can set the *Minimum Anonymization Level* during the negotiation of the privacy policy in order to define his minimum privacy requirements. The *Minimum Anonymization Level* must not exceed the

Maximum Anonymization Level or otherwise the privacy policy cannot be agreed on.

Therefore, in LPL the policy can express the data requirements of the company for processing personal data while personal privacy requirements can be set by the user for *Purpose*, *DataRecipient*, *Data* including the anonymization level. Thus, fine-grained control over the privacy-preserving processing is enabled during the negotiation of the privacy policy.

Furthermore, LPL allows specifying pseudonymization and privacy models. In contrast to the definition of the *AnonymizationMethod*-element which targets a specific *Data*-element, the pseudonymization method and privacy model may affect the whole record. Considering a data-set with several records, pseudonymization replaces the original values of EI attributes with a pseudonym (see Section 3). Because both de-identification methods require a more global view on the data-set compared to the personal privacy anonymization, LPL integrates a set of *PseudonymizationMethod*-elements and a set of *PrivacyModel*-elements which are referenced by the *Purpose*-element.

Each of the elements expresses the respective de-identification method in a generic way, such that the various methods can be defined. The *PseudonymizationMethod*-element and *PrivacyModel*-element do not feature any possibility for the user to negotiate, because these methods require expert knowledge to be defined.

The efficient processing of various personalized privacy policies in combination with pseudonymization and privacy models for the de-identification of a requested data-set is detailed in Chapter 6.

5.1.6 *Provenance*

To allow users to assume their rights regarding their personal data, the personal data has to be linked to the user. Furthermore, it has to be ensured that for the processing of personal data, the agreed privacy policy is actually considered. These aspects are considered in the context of the *R6 Provenance* requirement.

As shown in Section 4.1, sticky policies are commonly utilized to create the link between the privacy policy and the related personal data. But this concept does not consider that the privacy policy is used as the basis for secondary processing. In the *Data Processing* use case, e.g., personal data may be transferred and processed by a third party only if additional stricter conditions are met, which are expressed by an additional policy. In the *Data Production* use case, e.g., personal data from several users is processed producing new data which is used for decision-making. To backtrack the source for the decision-making it is necessary to store all sources, i.e. users, of the original personal data within the processing policy of the newly generated data.

These two use cases are addressed in LPL through the introduction of a set of *UnderlyingPrivacyPolicy*-elements for each *LayeredPrivacyPolicy*-element. The *UnderlyingPrivacyPolicy*-element is hereby equal to the *LayeredPrivacyPolicy*-element. In other words, each privacy policy can reference several underlying privacy policies which it is based on. An underlying privacy policy serves hereby as the base for the top policy, whereas the top policy must be stricter or equally strict as the underlying privacy policies. Additional layers can be added.

The layering of privacy policies, which is unique to LPL, in combination with the well-known sticky policy concept, models and enforces *R6 Provenance*. This allows not only to backtrack to the origins of personal data, i.e. the user, but also to the previously agreed on privacy policies. An in depth example is discussed in Section 5.7.

5.1.7 *Naming*

Lastly, the name of the *Layered Privacy Language (LPL)* is elaborated. The name of the proposed privacy language, i.e. the term 'layered', is based upon two features that are enabled by LPL. On the one hand, the term 'layered' describes the concept of presenting privacy policies in several layers for transparency, i.e. the first layer informs the user about the key elements while the second layer covers details [18] [124] [99]. *Human-readability* is considered as one of the key requirements from a users' perspective, for which this *Layered* approach is suitable (see Section 2.3).

On the other hand, the term 'layered' is introduced as one unique feature of LPL, which allows to specify underlying privacy policies, i.e. privacy policies that the current privacy policy is based upon and has to be compliant with. As previously reasoned, this feature is introduced alongside the usage of the *sticky* policy concept to fulfil the *Provenance* requirement (see Section 2.6).

LPL is formalized in the following section, detailing all elements and attributes, before the concept of the life cycle of LPL is detailed.

5.2 FORMAL DEFINITION

In this section, the formal description of the *Layered Privacy Language (LPL)* is defined, which satisfies the requirements presented in Chapter 2. The structure, including elements and attributes, of LPL is depicted in Figure 5.2. All the elements presented in the diagram are detailed in the following subsections, whereas first *Super-elements* are introduced from which other elements inherit. For clarity of the description, Table 5.1 provides, for each element, notations that are used for a single element, a subset of elements and the complete set. The formalization of LPL is based on the works of Gerl et al. [121], Gerl and Pohl [119], Gerl [112], and Gerl and Bölz [113].

5.2.1 *Super-elements*

Super-elements from which other LPL elements inherit are introduced . None of the *Super-elements* are utilized standalone within LPL. On the one hand, the *UIElement* is introduced, which allows for human-readability of elements. On the other hand, *Entity* inherits attributes to represent and identify an individual, role, or organization like a company for access control purposes within LPL. Both are detailed in the following.

5.2.1.1 *UIElement*

The *UIElement* $\mathtt{ui}$ encapsulates human-readable text,

$$\mathtt{ui} = (\widehat{\mathtt{HEAD}}, \widehat{\mathtt{DESC}}) \tag{5.1}$$

is a tuple consisting of $\widehat{\mathtt{HEAD}}$ and $\widehat{\mathtt{DESC}}$. Both the set of *Header*-elements $\widehat{\mathtt{HEAD}}$ and set of *Description*-elements $\widehat{\mathtt{DESC}}$ are structured the same way,

$$\mathtt{head} = (\mathtt{lang}, \mathtt{value}) \tag{5.2}$$

$$\mathtt{desc} = (\mathtt{lang}, \mathtt{value}) \tag{5.3}$$

as a tuple with the following attributes:

- $\mathtt{lang}$: Defines the language for the human-readable text using the international standard ISO 639-1 utilizing the Alpha-2 code, representing languages with two letters [134]. For example $'\mathtt{en}'$ for English, $'\mathtt{de}'$ for German, or $'\mathtt{fr}'$ for French.
- $\mathtt{value}$: Denotes the human-readable text in the language specified by $\mathtt{lang}$. An understandable language, instead of a language requiring expert knowledge to understand, should be used to allow users to easily access information [110, Art. 12(1)].

Thus, an exemplary set of *Description*-elements stating 'Hello World' in English, German, and French would be defined as follows:

$$\widehat{\mathrm{DESC}} = \{(\text{'en'},\text{'Hello World'}), (\text{'de'},\text{'Hallo Welt'}), (\text{'fr'},\text{'Bonjour le monde'})\} \tag{5.4}$$

The set of all *Header*-elements, each denoting a concise header for the element, is denoted by HEAD and $\widehat{\mathrm{HEAD}}$ denotes a subset of HEAD. The set of all *Description*-elements is denoted by DESC and $\widehat{\mathrm{DESC}}$ denotes a subset of DESC.

5.2.1.2 *Entity*

The *Entity*-element e, representing persons, companies or any other entity having some processing rights on the data,

$$e = (\mathit{name}, \mathit{classification}, \mathit{authInfo}, \mathit{type}, \widehat{\mathrm{HEAD}}, \widehat{\mathrm{DESC}}) \tag{5.5}$$

is a tuple consisting of the following attributes:

- name: Used for authorization in access control.
- $\mathit{classification}$: Classifies the *Entity* in either 'Person' or 'Legal Entity'.
- $\mathit{authInfo}$: The challenge used for authentication of the *Entity*, e.g., a hashed password for which the original password has to be provided, or a public key for which a corresponding private key exists. Thus, LPL is agnostic of the authentication methodology.
- type: Specifies the type of entity, which can be either a 'DataSource', a 'DataRecipient', a 'Controller', or a 'DataProtectionOfficer'.

The *Entity*-element e inherits from *UIElement* and therefore references a set of *Header*-elements $\widehat{\mathrm{HEAD}}$ as well as a set of *Description*-elements $\widehat{\mathrm{DESC}}$ representing the human-readable information on the entity.

The set of all *Entity*-elements is denoted by E and $\widehat{\mathrm{E}}$ denotes a subset of E.

Table 5.1: Overview over all elements and their formal definition. Bold styled sets are tuples inheriting an order.

Element	Single Element	Subset of Elements	Set of Elements
UIElement	ui	$\widehat{UI}$	UI
Header	$head$	$\widehat{HEAD}$	$HEAD$
Description	$desc$	$\widehat{DESC}$	$DESC$
Entity	e	$\widehat{E}$	E
LayeredPrivacyPolicy	lpp	$\widehat{LPP}$	LPP
UnderlyingPrivacyPolicy	upp	$\widehat{UPP}$	UPP
Icon	i	$\widehat{I}$	I
DataSource	ds	$\widehat{DS}$	DS
Controller	c	$\widehat{C}$	C
DataProtectionOfficer	dpo	$\widehat{DPO}$	DPO
DataSubjectRight	dsr	$\widehat{DSR}$	DSR
LodgeComplaint	lc	$\widehat{LC}$	LC
Purpose	p	$\widehat{P}$	P
LegalBasis	lb	$\widehat{LB}$	LB
DataRecipient	dr	$\widehat{DR}$	DR
Safeguard	sg	$\widehat{SG}$	SG
AutomatedDecisionMaking	adm	$\widehat{ADM}$	ADM
Retention	r	$\widehat{R}$	R
PrivacyModel	pm	$\widehat{PM}$	PM
PrivacyModelAttribute	pma	$\widehat{PMA}$	PMA
PseudonymizationMethod	psm	$\widehat{PSM}$	PSM
PseudonymizationMethodAttribute	$psma$	$\widehat{PSMA}$	$PSMA$
Data	d	$\widehat{D}$	D
DataGroup	dg	$\widehat{DG}$	DG
AnonymizationMethod	am	$\widehat{AM}$	AM
AnonymizationMethodAttribute	ama	$\widehat{AMA}$	AMA
HierarchyEntry	he	$\widehat{\mathbf{HE}}$	$\mathbf{HE}$

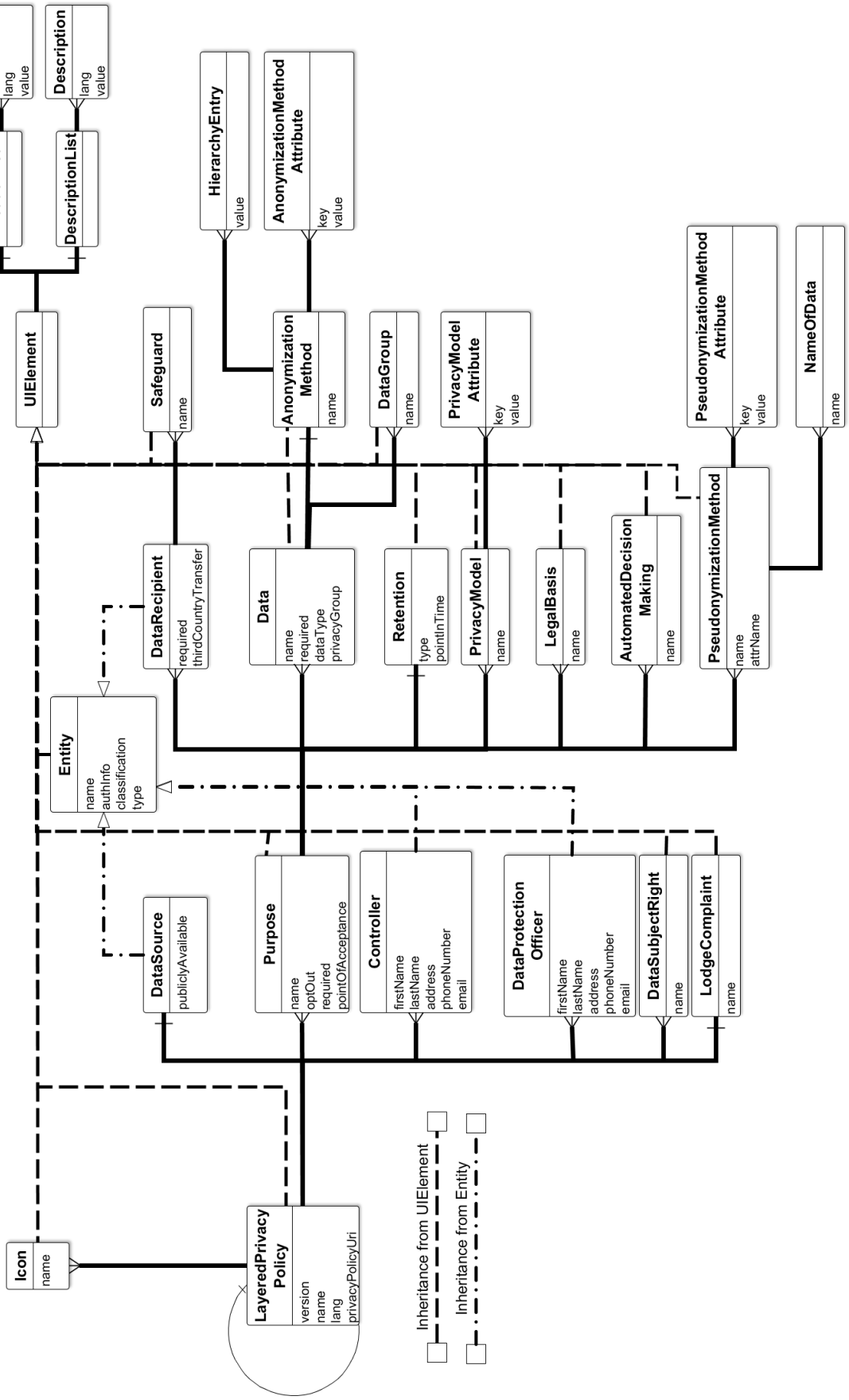


Figure 5.2: Entity-Relationship-Model showing the complete structure of the *Layered Privacy Language (LPL)*. All elements and attributes are shown. The inheritance of *UIElement* and *Entity* is denoted.

5.2.2 *Elements*

All elements from which LPL is composed of are detailed and formalized in the following. The basic privacy policy structure is given by the *LayeredPrivacyPolicy*-element referencing a *DataSource*-element and a set of *Purpose*-elements. Furthermore, the *Purpose*-element denotes a set of *DataRecipient*-elements and a set of *Data*-elements. Complementary elements are added to fulfil all requirements, e.g., *Legal Compliance* for *GDPR* or *De-identification Capabilities*.

5.2.2.1 *LayeredPrivacyPolicy*

The root-element of LPL is the *LayeredPrivacyPolicy*-element $\mathtt{lpp}$, which represents a legal privacy policy, e.g., between a user and a company. Only a single $\mathtt{lpp}$ is supposed to be defined for a LPL compliant file, e.g., privacy policy. A *LayeredPrivacyPolicy*-element

$$\begin{aligned}\mathtt{lpp} = (&\mathit{version}, \mathtt{name}, \mathtt{lang}, \mathtt{ppURI}, \widehat{\mathtt{HEAD}}, \widehat{\mathtt{DESC}}, \widehat{\mathtt{I}}, \mathtt{ds},\\ &\widehat{\mathtt{P}}, \widehat{\mathtt{C}}, \widehat{\mathtt{DPO}}, \mathtt{dsr}, \mathtt{lc}, \mathtt{upp})\end{aligned} \tag{5.6}$$

is a tuple consisting of the following attributes:

- $\mathit{version}$: Version number for future version management of LPL.
- $\mathtt{name}$: Identifier for the privacy policy.
- $\mathtt{lang}$: Defines the default language for the privacy policy, considering human-readable headers $\mathtt{HEAD}$ and descriptions $\mathtt{DESC}$ contained within the LPL privacy policy.
- $\mathtt{ppURI}$: An optional reference to a correlated common privacy policy, i.e. written document, to assure compliance with the current law, which is implemented as a static human-readable description of the privacy policy.

The *LayeredPrivacyPolicy*-element $\mathtt{lpp}$ references a set of *Header*-elements $\widehat{\mathtt{HEAD}}$ as well as a set of *Description*-elements $\widehat{\mathtt{DESC}}$ representing the human-readable information on the policy, e.g., for a short introduction of the policy. Furthermore, $\mathtt{lpp}$ references a set of *Icon*-elements $\widehat{\mathtt{I}}$, a *DataSource*-element $\mathtt{ds}$, a set of *Purpose*-elements $\widehat{\mathtt{P}}$, a set of *Controller*-elements, a set of *DataProtectionOfficer*-elements, a *DataSubjectRight*-element $\mathtt{dsr}$, and a *LodgeComplaint*-element $\mathtt{lc}$. Additionally, each *LayeredPrivacyPolicy* $\mathtt{lpp}$ can have a reference to an *UnderlyingPrivacyPolicy* $\mathtt{upp}$. Let an *UnderlyingPrivacyPolicy*-element be

$$\begin{aligned}\mathtt{upp} = (&\mathit{version}, \mathtt{name}, \mathtt{lang}, \mathtt{ppURI}, \widehat{\mathtt{HEAD}}, \widehat{\mathtt{DESC}}, \widehat{\mathtt{I}}, \mathtt{ds},\\ &\widehat{\mathtt{P}}, \widehat{\mathtt{C}}, \widehat{\mathtt{DPO}}, \mathtt{dsr}, \mathtt{lc}, \mathtt{upp}')\end{aligned} \tag{5.7}$$

where upp′ is another *UnderlyingPrivacyPolicy*-element denoting a previously consented privacy policy. The set of all *UnderlyingPrivacyPolicy*-elements is denoted by UPP and $\widehat{\text{UPP}}$ denotes a subset of UPP. This allows to create layers of privacy policies to satisfy the objective of being able to track privacy policies over multiple entities.

Let the deepest ('most underlying') *leaf-LayeredPrivacyPolicy*

$$\text{lpp}_{leaf} = (version, name, \text{lang}, \text{ppURI}, \widehat{\text{HEAD}}, \widehat{\text{DESC}}, \widehat{\text{I}}, \text{ds}, \widehat{\text{P}}, \widehat{\text{C}}, \widehat{\text{DPO}}, \text{dsr}, \text{lc}, \emptyset) \tag{5.8}$$

be the first privacy policy for which consent is given. In other words, the *LayeredPrivacyPolicy* with no *UnderlyingPrivacyPolicy* is the initial privacy policy, which is usually a consent between a user and a legal entity. If an additional privacy policy lpp_{new}, e.g., for a data-transfer to a third party, has to be added to an existing $\text{lpp}_{existing}$, then the $\text{lpp}_{existing}$ is wrapped by lpp_{new}. This results in

$$\text{lpp}_{existing} = (version, name, \text{lang}, \text{ppURI}, \widehat{\text{HEAD}}, \widehat{\text{DESC}}, \widehat{\text{I}}, \text{ds}, \widehat{\text{P}}, \widehat{\text{C}}, \widehat{\text{DPO}}, \text{dsr}, \text{lc}, \emptyset) \tag{5.9}$$

$$\text{lpp}_{new} = (version', name', \text{lang}', \text{ppURI}', \widehat{\text{HEAD}}', \widehat{\text{DESC}}', \widehat{\text{I}}', \text{ds}', \widehat{\text{P}}', \widehat{\text{C}}', \widehat{\text{DPO}}', \text{dsr}', \text{lc}', \text{lpp}_{existing}) \tag{5.10}$$

which is valid for each additionally added privacy policy lpp'_{new}. In this case, the data source (ds′) is the *Controller* (c or $\widehat{\text{C}}$) of the policy $\text{lpp}_{existing}$ and $\widehat{\text{P}}'$ can equal $\widehat{\text{P}}$ or a be subset of $\widehat{\text{P}}$. Naturally, the remaining elements and attributes may also differ, which are described in the following. An alternative approach to layering the *LayeredPrivacyPolicy*-elements lpp is to layer *Purpose*-elements, but this has the disadvantage that elements referenced by lpp cannot be layered, i.e. updated, without creating a new lpp, which has to take into account the predecessor lpp. The same argumentation can be made for any other element, thus *LayeredPrivacyPolicy*-elements are layered instead of other elements.

The set of all *LayeredPrivacyPolicy*-elements is denoted by LPP and $\widehat{\text{LPP}}$ denotes a subset of LPP.

5.2.2.2 *Icon*

The *Icon*-element i, representing a *Privacy Icon*,

$$\text{i} = (name, \widehat{\text{HEAD}}, \widehat{\text{DESC}}) \tag{5.11}$$

is a tuple consisting of the following attribute:

- name: The identifying name of the *Privacy Icon* to display the correct icon. To the current date no official standardization of *Privacy Icons* is given, thus no fixed set of values is given.

The *Icon*-element i references a set of *Header*-elements $\widehat{\mathrm{HEAD}}$ as well as a set of *Description*-elements $\widehat{\mathrm{DESC}}$ representing the human-readable information on the *Privacy Icon*.

The set of all *Icon*-elements is denoted by I and $\widehat{\mathrm{I}}$ denotes a subset of I.

5.2.2.3 *DataSource*

The *DataSource*-element ds inherits from the *Entity*-element e, whereas the type is set to $'\mathrm{DataSource}'$.

$$\begin{aligned}\mathrm{ds} = (&\mathrm{name}, \mathrm{classification}, \mathrm{authInfo}, \text{'DataSource'},\\ &\mathrm{publiclyAvailable}, \widehat{\mathrm{HEAD}}, \widehat{\mathrm{DESC}})\end{aligned} \tag{5.12}$$

Next to the attributes inherited by the *Entity*-element it has the additional attribute:

- $\mathrm{publiclyAvailable}$: A boolean defining if the data has been received from a publicly available source. This requirement is given by *GDPR* to allow the differentiation between personal information directly derived from a *Data Subject* and information that is already published [110, Art. 14(2) f)].

The *DataSource*-element describes the current authority granting data recipients the processing of data, based upon its own processing rights. For example, this can be the user (person) for whom the personal data is dedicated to or a company (legal entity) that has collected the personal data for a specific purpose. The set of all *DataSource*-elements is denoted by DS and $\widehat{\mathrm{DS}}$ denotes a subset of DS.

5.2.2.4 *Controller*

The *Controller*-element c inherits from *Entity*, whereas the type is set to $'\mathrm{Controller}'$.

$$\begin{aligned}\mathrm{c} = (&\mathrm{name}, \mathrm{classification}, \mathrm{authInfo}, \text{'Controller'},\\ &\mathrm{firstName}, \mathrm{lastName}, \mathrm{address}, \mathrm{phoneNumber}, \mathrm{email},\\ &\widehat{\mathrm{HEAD}}, \widehat{\mathrm{DESC}})\end{aligned} \tag{5.13}$$

Next to the attributes inherited by the *Entity*-element, it has the additional attributes:

- $\mathrm{firstName}$: First name of the representative of the *Controller*.
- $\mathrm{lastName}$: Last name of the representative of the *Controller*.
- $\mathrm{address}$: Address of the *Controller*. A further differentiation in e.g., postal-code, street and house number has been avoided for both clarity and generality.

- phoneNumber: Phone number of the *Controller*. For clarity international standards like the ITU E.123 [135] and E.164 [136] or national standards like the DIN 5002 [74] should be considered.
- email: E-mail address of the representative of the *Controller*. The format should comply with the standards denoted in RFC 5322 [205], RFC 5321 [158] and RFC 3696 [157].

Note that the attributes address, phoneNumber and email have been chosen as common ways of communication to allow a *Data Subject* to contact the *Controller* and could be further extended by other contact possibilities like a fax number or alternative contact possibilities.

The *Controller*-element c represents a legal *Controller*. This can be either a natural or legal person. The *Controller* determines the purposes and means of the processing of personal data. A *Joint Controller* [110, Art. 26] can be represented by a set of *Controller*-elements $\widehat{C}$.

The set of all *Controller*-elements is denoted by C and $\widehat{C}$ denotes a subset of C.

5.2.2.5 *DataProtectionOfficer*

The *DataProtectionOfficer*-element dpo inherits from *Entity*, whereas the type is set to 'DataProtectionOfficer'. The *DataProtectionOfficer*-element dpo represents a *Data Protection Officer (DPO)*.

$$\begin{aligned} \text{dpo} = (&\text{name}, \text{classification}, \text{authInfo},\\ &\text{'DataProtectionOfficer'}, \text{firstName}, \text{lastName},\\ &\text{address}, \text{phoneNumber}, \text{email}, \widehat{\text{HEAD}}, \widehat{\text{DESC}}) \end{aligned} \tag{5.14}$$

Next to the attributes inherited by the *Entity*-element, it has the additional attributes:

- firstName: First name of the DPO.
- lastName: Last name of the DPO.
- address: Address of the DPO. A further differentiation in e.g., postal-code, street and house number has been avoided for both clarity and generality.
- phoneNumber: Phone number of the DPO. For clarity international standards like the ITU E.123 [135] and E.164 [136] or national standards like the DIN 5002 [74] should be considered.
- email: E-mail address of the DPO. The format should comply with the standards denoted in RFC 5322 [205], RFC 5321 [158] and RFC 3696 [157].

Note that the attributes match the additional attributes of the *Controller*-element both in wording and function.

The *DataProtectionOfficer*-element dpo represents the *Data Protection Officer (DPO)* that is responsible for the privacy policy. This can be either a natural or legal person, e.g., a company acting as an external DPO [110, Art. 37]. The lpp allows the definition of several DPOs $\widehat{\text{DPO}}$, due to the possibility of one or several stand-ins.

The set of all *DataProtectionOfficer*-elements is denoted by DPO and $\widehat{\text{DPO}}$ denotes a subset of DPO.

5.2.2.6 *DataSubjectRight*

The *DataSubjectRight*-element dsr, representing the information for the *Data Subject* about his rights according to Art. 12 - 23 *GDPR* [110, Art. 12 - 23],

$$\text{dsr} = (name, \widehat{\text{HEAD}}, \widehat{\text{DESC}}) \tag{5.15}$$

is a tuple consisting of the following attribute:

- $name$: A textual representation of the identifying name, e.g., 'GDPR data subject rights'. A differentiation of *Data Subject Rights* is necessary due to the various readings of the law on international, national and intra-national level [162].

The *DataSubjectRight*-element dsr references a set of *Header*-elements $\widehat{\text{HEAD}}$ as well as a set of *Description*-elements $\widehat{\text{DESC}}$ representing the human-readable information on the rights of the *Data Subject*. Note that a set of *DataSubjectRight*-elements $\widehat{\text{DSR}}$ would have been an alternative option for modelling LPL, in which each element would specify one *Data Subject Right*. This alternative has not been chosen because the statement for *Data Subject Rights* is concise in practice.

The set of all *DataSubjectRight*-elements is denoted by DSR and $\widehat{\text{DSR}}$ denotes a subset of DSR.

5.2.2.7 *LodgeComplaint*

The *LodgeComplaint*-element lc, representing the information for the *Data Subject* about his right to lodge a complaint with a supervisory authority according to Art. 77 *GDPR* [110, Art. 77] and Art. 12 - 14 [110, Art. 12 - 14],

$$\text{lc} = (name, \widehat{\text{HEAD}}, \widehat{\text{DESC}}) \tag{5.16}$$

is a tuple consisting of the following attribute:

- $name$: A textual representation of the identifying name, e.g., 'GDPR lodge complaint'. A differentiation of lodge complaint statements is necessary because the supervisory authority differs for, e.g., different nations and states.

The *LodgeComplaint*-element lc references a set of *Header*-elements $\widehat{\text{HEAD}}$ as well as a set of *Description*-elements $\widehat{\text{DESC}}$ representing the human-readable information on the right of the *Data Subject* to lodge a complaint.

The set of all *LodgeComplaint*-elements is denoted by LC and $\widehat{\text{LC}}$ denotes a subset of LC.

5.2.2.8 *Purpose*

The *Purpose*-element p, representing the purpose of the processing,

$$\begin{aligned} p = (&\texttt{name}, \texttt{optOut}, \texttt{required}, \texttt{pointOfAcceptance}, \\ &\widehat{\text{HEAD}}, \widehat{\text{DESC}}, \widehat{\text{D}}, \widehat{\text{PM}}, \widehat{\text{PSM}}, \widehat{\text{DR}}, \widehat{\text{LB}}, \widehat{\text{ADM}}, r) \end{aligned} \tag{5.17}$$

is a tuple consisting of the following attributes:

- name: A textual representation of the identifying name, e.g., 'Marketing' or 'Research'. In the set of purposes there should be no duplicate names.
- optOut: A boolean defining if the *Purpose* is opt-out for *true* or opt-in for *false*. Opt-out implies that the user has to actively deny this purpose. In the opposite, opt-in implies that the user has to actively accept this purpose. Although *GDPR* only allows opt-in purposes, this option is included to 1) explicitly state how the purpose has to be consented to and 2) to support the opt-out option for possible compliance to other legal privacy frameworks.
- required: A boolean defining if the *Purpose* has to be accepted by the user. If the user does not accept a required *Purpose* then there cannot be a consent for the corresponding lpp.
- pointOfAcceptance: The date and time for when the purpose has been accepted (or consented to). The information is stored for accountability.

The *Purpose*-element specifies which *Data* $\widehat{\text{D}}$ can be processed for which *Purpose* p by which *DataRecipients* $\widehat{\text{DR}}$. Furthermore, it is specified which de-identification methods are applied, e.g. *PrivacyModels* $\widehat{\text{PM}}$ or *PseudonymizationMethods* $\widehat{\text{PSM}}$, and how long the data *Retention* r is.

Therefore, the *Purpose*-element p references a set of *Header*-elements $\widehat{\text{HEAD}}$ as well as a set of *Description*-elements $\widehat{\text{DESC}}$ representing the human-readable information on the purpose. Furthermore, p references a set of *Data*-elements $\widehat{\text{D}}$, a set of *PrivacyModel*-elements $\widehat{\text{PM}}$, a set of *PseudonymizationMethod*-elements $\widehat{\text{PSM}}$, a set of *DataRecipient*-elements $\widehat{\text{DR}}$, a set of *LegalBasis*-elements $\widehat{\text{LB}}$, a set of *AutomatedDecisionMaking*-elements $\widehat{\text{ADM}}$, and a *Retention*-element r all presented in the following paragraphs.

The set of all *Purpose*-elements is denoted by P and $\widehat{\mathsf{P}}$ denotes a subset of P.

5.2.2.9 *LegalBasis*

The *LegalBasis*-element $\mathtt{lb}$ denotes the legal basis for the processing of personal data, which can also be the requirement for consent [110, Art. 13(1)(c), Art. 14(1)(c), Art. 13(1)(d), Art. 14(2)(b)],

$$\mathtt{lb} = (\mathtt{name}, \widehat{\mathsf{HEAD}}, \widehat{\mathsf{DESC}}) \tag{5.18}$$

is a tuple consisting of the following attribute:

- $\mathtt{name}$: A textual representation of the identifying name for the legal basis. A differentiation legal basis is necessary because several laws and regulations may apply.

The *LegalBasis*-element $\mathtt{lb}$ references a set of *Header*-elements $\widehat{\mathsf{HEAD}}$ as well as a set of *Description*-elements $\widehat{\mathsf{DESC}}$ representing the human-readable information on the legal basis, e.g., how or why the law applies for a purpose.

The set of all *LegalBasis*-elements is denoted by LB and $\widehat{\mathsf{LB}}$ denotes a subset of LB.

5.2.2.10 *DataRecipient*

The *DataRecipient*-element $\mathtt{dr}$ inherits from *Entity*, whereas the $\mathtt{type}$ is set to $'\mathtt{DataRecipient}'$.

$$\begin{aligned}\mathtt{dr} = (&\mathtt{name}, \mathtt{classification}, \mathtt{authInfo}, \text{'DataRecipient'},\\ &\mathtt{required}, \mathtt{thirdCountryTransfer}, \widehat{\mathsf{HEAD}}, \widehat{\mathsf{DESC}}, \widehat{\mathsf{SG}})\end{aligned} \tag{5.19}$$

Next to the attributes inherited by the *Entity*-element it has the additional attributes:

- $\mathtt{required}$: A boolean defining if the *DataRecipient* has to be accepted by the user. If the user does not accept a required *DataRecipient* then there cannot be a consent for the corresponding $\mathtt{lpp}$. This attribute allows the user, if it is set as $'\mathtt{false}'$, to personalize the policy.
- $\mathtt{thirdCountryTransfer}$: A boolean defining if the represented data recipient is in a *Third Country* and therefore not under the jurisdiction of the *GDPR*. If this is the case, the *Data Subject* has to be informed about it [110, Art. 13(2)(a) and Art. 14(2)(a)], which is the rationale behind this attribute.

Furthermore, the *DataRecipient*-element references a set of *Safeguard*-elements SG, which have to be defined if the data recipient is located within a *Third Country*.

The *DataRecipient*-element represents the authority that gets specific processing rights (defined by the *Purpose*) granted. This can be a person or a legal entity. For example, given the *DataSource*-element representing the user (person) to whom the personal data is referring to, then this authority can grant the *DataRecipient* all processing rights. Assuming ds_C represents a *Controller C* that has collected the data from a user ds_U under specific processing rights $\widehat{P}_C$ and intends to grant a third party dr_T processing rights $\widehat{P}_T$, then ds_C can only grant dr_T the usage within the limits of its own processing rights $\widehat{P}_T \subseteq \widehat{P}_C$. It has to be noted that the processing rights of ds_C are a subset of the processing rights of the user, who has all the processing rights, $\widehat{P}_T \subseteq \widehat{P}_C \subseteq \widehat{P}_U$ This follows the principle of the right to informational self-determination [126]. The set of all *DataRecipient*-elements is denoted by DR and $\widehat{DR}$ denotes a subset of DR.

5.2.2.11 *Safeguard*

The *Safeguard*-element sg, representing the description of appropriate safeguards, i.e. an individual agreement to protect the privacy, for the transfer of personal data to *Third Countries* [110, Art. 46],

$$sg = (name, \widehat{HEAD}, \widehat{DESC}) \tag{5.20}$$

is a tuple consisting of the following attribute:

- name: A textual representation of the identifying name for the safeguard. A differentiation of safeguards is necessary.

The *Safeguard*-element sg references a set of *Header*-elements $\widehat{HEAD}$ as well as a set of *Description*-elements $\widehat{DESC}$ representing the human-readable information on the implemented safeguard. Safeguards have to be implemented if no *Adequacy Decision* for the *Third Country* is determined [110, Art. 46]. According to the European Commission lists, an *Adequacy Decision* exists only for a few *Third Countries* including Canada, Switzerland, and Israel. The United States of America are limited to the *Privacy Shield Framework* [97] [69] providing an *Adequate Decision* [97].

The set of all *Safeguard*-elements is denoted by SG and $\widehat{SG}$ denotes a subset of SG.

5.2.2.12 *AutomatedDecisionMaking*

The *AutomatedDecisionMaking*-element adm informs the *Data Subject* about automated decision-making and profiling. The *Data Subject* has the right to be not subject to automated decision-making, therefore such a processing is explicitly stated [110, Art. 22]. Therefore,

$$adm = (name, \widehat{HEAD}, \widehat{DESC}) \tag{5.21}$$

is a tuple consisting of the following attribute:

- $\mathtt{name}$: A textual representation of the identifying name for the automated decision-making. A differentiation of automated-decision making is necessary to allow the *Data Subject* to be informed about the possibly different processes.

The *AutomatedDecisionMaking*-element $\mathtt{adm}$ references a set of *Header*-elements $\widehat{\mathtt{HEAD}}$ as well as a set of *Description*-elements $\widehat{\mathtt{DESC}}$ representing the human-readable information on the automated decision-making processes. According to the *GDPR*, the *Data Subject* has to be informed about the existence of automated decision-making, including profiling, and gets provided with meaningful information about the logic, significance and envisioned consequences of the process [110, Art. 13(2)(f), Art. 14(2)(g)].

The set of all *AutomatedDecisionMaking*-elements is denoted by $\mathtt{ADM}$ and $\widehat{\mathtt{ADM}}$ denotes a subset of $\mathtt{ADM}$.

5.2.2.13 *Retention*

The *Retention*-element $\mathtt{r}$ defines when the described data has to be deleted [110, Art. 13(2)(a), Art. 14(2)(a)].

$$\mathtt{r} = (\mathtt{type}, \mathtt{pointInTime}, \widehat{\mathtt{HEAD}}, \widehat{\mathtt{DESC}}) \tag{5.22}$$

The element consists of the following attributes:

- $\mathtt{type}$: Describing the general condition of the retention. Possible values are *Indefinite*, *AfterPurpose* and *FixedDate*.
- $\mathtt{pointInTime}$: Textual representation describing the exact conditions for the execution of the retention.

Depending on the $\mathtt{type}$, the $\mathtt{pointInTime}$ has diverse meanings. The $\mathtt{type}$ *Indefinite* without a value for $\mathtt{pointInTime}$ defines that there is no time constrained for the deletion of the data. The $\mathtt{type}$ *AfterPurpose* defines that after the completion of the corresponding purpose $\mathtt{p}$, the data has to be deleted within the time-frame specified by $\mathtt{pointInTime}$. Lastly, the $\mathtt{type}$ *FixedDate* in combination with $\mathtt{pointInTime}$ explicitly defines the date for the deletion of the data within the corresponding $\mathtt{p}$.

The set of *Header*-elements $\widehat{\mathtt{HEAD}}$ as well as the set of *Description*-elements $\widehat{\mathtt{DESC}}$ represent the human-readable information on the retention. The set of all *Retention*-elements is denoted by $\mathtt{R}$ and $\widehat{\mathtt{R}}$ denotes a subset of $\mathtt{R}$.

5.2.2.14 *PrivacyModel*

The *PrivacyModel*-element $\mathtt{pm}$ specifies the privacy conditions that have to be fulfilled by the data-set for a specific purpose $\mathtt{p}$. This element can be specified but it is not mandatory. Due to the possibility

to apply more than one privacy model to a data-set [220], different privacy models may be defined. Alternatively, privacy can also be defined by *AnonymizationMethod*-element, defining personal privacy, by *PseudonymizationMethod*-element, or even omitted if not necessary.

$$\mathrm{pm} = (\mathit{name}, \widehat{\mathrm{PMA}}, \widehat{\mathrm{HEAD}}, \widehat{\mathrm{DESC}}) \tag{5.23}$$

The privacy model is defined by the name, e.g., *k-Anonymity* [240] [257] or *l-Diversity* [177]. Each privacy model can have a set of *PrivacyModelAttribute*-elements $\widehat{\mathrm{PMA}}$. The set of *Header*-elements $\widehat{\mathrm{HEAD}}$ as well as the set of *Description*-elements $\widehat{\mathrm{DESC}}$ represent the human-readable information on the privacy model. This is especially necessary, because privacy models are uncommon for regular users, so that additional information is required for a better understanding.

The set of all *PrivacyModel*-elements is denoted by PM and $\widehat{\mathrm{PM}}$ denotes a subset of PM.

5.2.2.15 *PrivacyModelAttribute*

A *PrivacyModelAttribute*-element pma, represents the configuration of a privacy model,

$$\mathrm{pma} = (\mathit{key}, \mathit{value}) \tag{5.24}$$

is a tuple of the following attributes:

- key: Definition of a variable that is required by the correlating pm, e.g., *k* for *k-Anonymity*.
- value: Definition of the actual variable content, e.g., for *k* the value '2', which describes that there have to be at least two records within the same QI-group values to preserve the required k-anonymity property [240] [257]

The set of all *PrivacyModelAttribute*-elements is denoted by PMA and $\widehat{\mathrm{PMA}}$ denotes a subset of PMA. The decision for utilizing $\widehat{\mathrm{PMA}}$ can be explained by the existence of privacy models that require more than one variable, e.g., *X,Y-Privacy* [108].

5.2.2.16 *PseudonymizationMethod*

The *PseudonymizationMethod*-element psm specifies a pseudonymization method that is applied on specific attributes of the data-set. This element can be given but it is not mandatory.

$$\mathrm{psm} = (\mathit{name}, \mathit{attrName}, \widehat{\mathrm{NOD}}, \widehat{\mathrm{HEAD}}, \widehat{\mathrm{DESC}}, \widehat{\mathrm{PSMA}}) \tag{5.25}$$

The element consists of the following attributes:

- $name$: Identifies the applied pseudonymization method, e.g., using the HMAC-SHA-1 algorithm with random seeds [165] or cryptographic methods using DES or AES [132] [201].
- $attrName$: Textual representation of the name for the resulting attribute. For instance, if the attributes $'prename'$ and $'surname'$ are jointly tokenized and the resulting attribute should be denoted as $'namePseudonym'$ with the $attrName$ attribute.

Furthermore, each *PseudonymizationMethod*-element defines a set of *NameOfData*-elements $\widehat{\mathrm{NOD}}$ and a set of *PseudonymizationMethodAttribute*-elements $\widehat{\mathrm{PSMA}}$. The set of *Header*-elements $\widehat{\mathrm{HEAD}}$ as well as the set of *Description*-elements $\widehat{\mathrm{DESC}}$ represent the human-readable information on the privacy model.

The set of all *PseudonymizationMethod*-elements is denoted by PSM and $\widehat{\mathrm{PSM}}$ denotes a subset of PSM.

5.2.2.17 *NameOfData*

A *NameOfData*-element nod represents a *Data*-element within the same *Purpose*-element,

$$nod = (name) \tag{5.26}$$

is a tuple consisting of the $name$ attribute. The intended purpose is to define all attributes that the pseudonymization method is applied to. Pseudonymization can be applied on a sole attribute or it can be applied on several attributes combined. If several attributes have to be pseudonymized independently, then several *PseudonymizationMethod*-elements have to be defined.

The set of all *PseudonymizationMethod*-elements is denoted by PSM and $\widehat{\mathrm{PSM}}$ denotes a subset of PSM.

5.2.2.18 *PseudonymizationMethodAttribute*

A *PseudonymizationMethodAttribute*-element $psma$, represents the configuration of a pseudonymization method,

$$psma = (key, value) \tag{5.27}$$

is a tuple of the following attributes:

- key: Definition of a variable that is required by the correlating psm, which may not even be necessary for some pseudonymization methods.
- $value$: Definition of the actual variable content.

The set of all *PseudonymizationMethodAttribute*-elements is denoted by PSMA and $\widehat{\mathrm{PSMA}}$ denotes a subset of PMA.

5.2.2.19 *Data*

The *Data*-element $\mathtt{d}$, representing a data field that is concerned by a purpose $\mathtt{p}$,

$$\mathtt{d} = (\mathtt{name}, \mathtt{dType}, \mathtt{required}, \mathtt{pGroup}, \widehat{\mathtt{DG}}, \mathtt{am}, \widehat{\mathtt{HEAD}}, \widehat{\mathtt{DESC}}) \tag{5.28}$$

is a tuple of the following attributes:

- $\mathtt{name}$: Distinct name for the stored data field. A duplicate $\mathtt{name}$ within a $\widehat{\mathtt{D}}$ of a *Purpose* is not allowed, because this could lead to discrepancies in the processing, while it is possible with P3P *DATA-Elements*. P3P indeed allows to define contrary rules for the same data element within one purpose. This makes the determination of the valid rule unfeasible [280].
- $\mathtt{dType}$: Defines the type of the data. Possible types are *Text*, *Number*, *Date*, *Boolean*, *Value Set* for a set of predefined values and *Other* for any data type that doesn't fit the aforementioned types.
- $\mathtt{required}$: A boolean defining if the data $\mathtt{d}$ is required for the purpose, or if the user can disagree with the processing of this attribute. If the user does not accept a required $\mathtt{d}$ then the corresponding $\mathtt{p}$ is not accepted. If the purpose $\mathtt{p}$ is required, then the whole privacy policy $\mathtt{lpp}$ is not accepted.
- $\mathtt{pGroup}$: This is the classification of the data field as *Explicit*, *QID*, *Sensitive* and *Non-Sensitive*. The processing of the data field by the privacy models is based upon this classification, e.g., for *k-Anonymity* the value of a data field which is classified *Explicit*, has to be deleted [257].

The *Data*-element $\mathtt{d}$ references a set of *Header*-elements $\widehat{\mathtt{HEAD}}$ as well as a set of *Description*-elements $\widehat{\mathtt{DESC}}$ representing the human-readable information on the data. Furthermore, a set of *DataGroup*-elements $\widehat{\mathtt{DG}}$ is referenced for the categorization of the data. Lastly, an *AnonymizationMethod*-element $\mathtt{am}$ is referenced which defines the minimum anonymization for the data in order to enable personal privacy anonymization. The set of all *Data*-elements is denoted by $\mathtt{D}$ and $\widehat{\mathtt{D}}$ denotes a subset of $\mathtt{D}$.

5.2.2.20 *DataGroup*

The *DataGroup*-element $\mathtt{dg}$, representing a group of data, that is used for categorizing *Data*-elements,

$$\mathtt{dg} = (\mathtt{name}, \widehat{\mathtt{HEAD}}, \widehat{\mathtt{DESC}}) \tag{5.29}$$

is a tuple consisting of the attribute:

- $name$: A textual representation of a logical data group. No predefined values are given. Data groups can be used to specify data in, e.g., a procedure directory, which usually does not refer to each data field but groups of it [110, Art. 30]. This enables to validate record of processing activities [110, Art. 30] with a privacy policy automatically or even to create one beforehand. For instance, data elements representing $'street'$, $'postal-code'$, $'city'$ of a person could be categorized as $'address'$.

The *DataGroup*-element dg references a set of *Header*-elements $\widehat{\mathrm{HEAD}}$ as well as a set of *Description*-elements $\widehat{\mathrm{DESC}}$ representing the human-readable information on the data group.

The set of all *DataGroup*-elements is denoted by DG and $\widehat{\mathrm{DG}}$ denotes a subset of DG.

5.2.2.21 *AnonymizationMethod*

The *AnonymizationMethod*-element am, represents the anonymization that is applied on a data,

$$am = (name, \widehat{\mathrm{AMA}}, \widehat{\mathbf{HE}}, \widehat{\mathrm{HEAD}}, \widehat{\mathrm{DESC}}) \tag{5.30}$$

is a tuple of the following attributes. The $name$ represents the chosen anonymization method. There are several methods available, for example *Deletion*, *Suppression* or *Generalization*. Additionally, each *AnonymizationMethod* has a set of *AnonymizationMethodAttributes*-elements $\widehat{\mathrm{AMA}}$ and an ordered set of *HierarchyEntry*-elements $\widehat{\mathbf{HE}}$.

The set of *Header*-elements $\widehat{\mathrm{HEAD}}$ as well as the set of *Description*-elements $\widehat{\mathrm{DESC}}$ represent the human-readable information on the anonymization method. This is especially necessary, because the way data is anonymized is not commonly known and therefore easily understandable information is required for users. The set of all *AnonymizationMethod*-elements is denoted by AM and $\widehat{\mathrm{AM}}$ denotes a subset of AM.

5.2.2.22 *AnonymizationMethodAttribute*

An *AnonymizationMethodAttribute*-element ama, represents the configuration of an anonymization method,

$$ama = (key, value) \tag{5.31}$$

is a tuple of the following attributes:

- key: Definition of a variable that is required by the correlating am.
- $value$: Definition of the actual variable content.

The definition of each an ama for a *Minimum Anonymization Level* and *Maximum Anonymization Level* is required. The *Minimum Anonymization Level* denotes the anonymization which has to be applied to comply with privacy requirements. The *Maximum Anonymization Level* denotes the anonymization level which must not be exceeded during the de-identification process, to preserve utility of certain attributes required for later analysis, e.g. the value must not exceed the anonymization level '2' (see Equation 5.32).

$$ama = (\text{'Maximum Anonymization Level'}, \text{'2'}) \tag{5.32}$$

If enabled by the user interface and if *Minimum Anonymization Level* is lower then the *Maximum Anonymization Level*, then the *Minimum Anonymization Level* may be adjusted by the user for personalization of the policy.

The set of all *AnonymizationMethodAttribute*-elements is denoted by AMA and $\widehat{AMA}$ denotes a subset of AMA.

5.2.2.23 *HierarchyEntry*

The *HierarchyEntry*-element he stores possible pre-calculated values for one data field and the correlating anonymization method. The hierarchy he is used during the de-identification process, e.g., for enabling personal privacy or anonymization for privacy models.

$$he = (value) \tag{5.33}$$

The attribute $value$ denotes a possible anonymization of the original value for an attribute, including the original value, i.e. the non-anonymized value. The tuple of all *HierarchyEntry*-elements is denoted by **HE** and $\widehat{\mathbf{HE}}$ denotes an ordered sub-tuple of **HE**. Considering the *Minimum Anonymization Level* and *Maximum Anonymization Level*, the elements are counted starting with level '0'. Thus, the first element of $\widehat{\mathbf{HE}}$ is specified as *Anonymization Level* 0, the second element is specified as 1, and so on.

Therefore, the definition of all LPL elements an attributes is completed. In the following, an overview over the life cycle of LPL is given before the fulfilment of the requirements for a privacy language (see Chapter 2) is detailed.

5.3 LIFE CYCLE

LPL is intended to cover the holistic life cycle of privacy policies from their creation to their realization as denoted by Gerl et al. [121]. To present the life cycle phases of LPL, following scenario is assumed:

A company e_{C1} intends to create an online shop, which collects and uses personal information for various purposes like billing, marketing

and research. Therefore, e_{C1} creates a privacy policy and presents it to the user e_{U1} via the online shop, e.g., at the registration time of the user. The user e_{U1} can freely decide if he agrees to the privacy policy or not. From a legal perspective, the company e_{C1} acts as the *Controller* and the user e_{U1} is the *Data Subject*. Considering the notation of LPL, the user is the *DataSource* ds_{U1} and the company acts as the *Controller* c_{C1}. Furthermore, the company may also be defined as the *DataRecipient* dr_{C1} for various purposes. Additionally, it is assumed that the data may be transferred to a third party c_{C2} for research. The data transfer between c_{C1} and c_{C2} is handled by an additional policy. In this case, this inter company transfer policy denotes ds_{C1} as the *DataSource* and c_{C2} as the responsible *Controller*, as well as *DataRecipient*.

$$ds_{U1} = (\text{'DS\_U1'}, \text{'Person'}, publicKey_{U1}, \text{'DataSource'}, false, \widehat{HEAD}_{U1}, \widehat{DESC}_{U1}) \tag{5.34}$$

$$c_{C1} = (\text{'C\_C1'}, \text{'Legal Entity'}, publicKey_{C1}, \text{'Controller'}, firstName_{C1}, lastName_{C1}, address_{C1}, phoneNumber_{C1}, email_{C1}, \widehat{HEAD}_{C1}, \widehat{DESC}_{C1}) \tag{5.35}$$

$$dr_{C1} = (\text{'DR\_C1'}, \text{'Legal Entity'}, publicKey_{C1}, \text{'DataRecipient'}, true, false, \widehat{HEAD}_{C1}, \widehat{DESC}_{C1}) \tag{5.36}$$

$$ds_{C1} = (\text{'DS\_C1'}, \text{'Legal Entity'}, publicKey_{C1}, \text{'DataSource'}, false, \widehat{HEAD}_{C1}, \widehat{DESC}_{C1}) \tag{5.37}$$

$$c_{C2} = (\text{'C\_C2'}, \text{'Legal Entity'}, publicKey_{C2}, \text{'Controller'}, firstName_{C2}, lastName_{C2}, address_{C2}, phoneNumber_{C2}, email_{C2}, \widehat{HEAD}_{C2}, \widehat{DESC}_{C2}) \tag{5.38}$$

$$dr_{C2} = (\text{'DR\_C2'}, \text{'Legal Entity'}, publicKey_{C2}, \text{'DataRecipient'}, false, false, \widehat{HEAD}_{C2}, \widehat{DESC}_{C2}) \tag{5.39}$$

Within this scenario, it is differentiated between the following phases (see Figure 5.3).

- *Creation*: Company e_{C1} (*Controller*) creates a raw LPL privacy policy lpp_{raw} for its service.
- *Negotiation*: User e_{U1} (*Data Subject*) is presented the raw LPL privacy policy. e_{U1} can personalize the privacy policy.

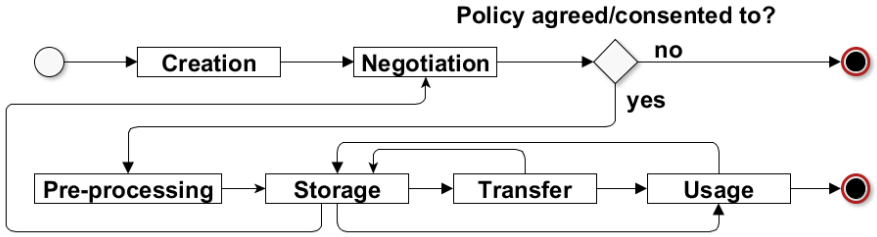


Figure 5.3: Life-cycle of LPL.

- *Pre-processing*: The personalized LPL privacy policy is verified, validated and anonymization hierarchies are pre-calculated.
- *Storage*: The personalized LPL policy is stored alongside the (personal) data (sticky policy).
- *Transfer*: The (personal) data and the sticky LPL policy are transferred to the third party e_{C2}.
- *Usage*: The personal data is requested for processing by e_{C1} and e_{C2} for which it is de-identified according to the LPL privacy policy.

These life cycle phases are not intended to be run through linearly, but may be repeated (see Figure 5.3). The *Negotiation* phase may be run through first by the user, e.g., during the registration process of a service. But the user is intended to revisit the privacy policy at any time so that he can change his personal privacy settings within the LPL privacy policy, which triggers the phases *Pre-processing* and *Storage* iff the policy is then updated in any way.

Furthermore, after the *Storage* phase, the personal data and LPL policies can be processed several times, such that the *Usage* phase is run through repeatedly. Therefore, it is considered that data is not processed once but several times for several purposes.

Similarly, the *Transfer* phase, which is optional, may be run through by e_{C1} and e_{C2} for various times. Depending on the use case, the data may be transferred to a third party and directly processed (*Usage*) or it may be stored before it is processed (*Storage*).

Next, every phase of the LPL life cycle is detailed.

5.3.1 *Creation*

During the *Creation* phase, the raw LPL privacy policy lpp_{raw} is created be the responsible entity, namely company e_{C1}. The creation can be based on an existing legal privacy policy or from scratch. This process may be facilitated via suitable tools, e.g., user interfaces (see Section 5.6.3). Hereby, the company defines all properties of the

privacy policy, e.g., which *Data* is processed for which *Purposes*. But a few attributes and elements cannot be detailed within this phase:

- *DataSource*-element ds: The user can only be identified and added to the policy after he agreed to it.
- pointOfAcceptance attribute of the *Purpose*-element p: This attribute expresses when the LPL policy has been agreed to by the user.
- HierarchyEntry: The anonymization hierarchy for each *Data*-element d can be pre-processed when an original value is given by the user.

Note that the *Minimum Anonymization Level* encapsulated within the *AnonymizationMethodAttribute*-element ama has to be set during the creation of the LPL privacy policy, but is ultimately intended to be altered by the user to define his personal privacy requirements.

Assuming the given scenario, a policy lpp_{raw} (see Equation 5.40) is created for the purpose 'Research'. Furthermore, the purposes 'Marketing' and 'Billing' would defined as *Purpose*-elements p, which are omitted for the scope of this example. Additionally, informative elements and attributes are omitted, e.g., *DataSubjectRight*-element dsr or human-readable texts.

The created lpp_{raw} expresses the processing of the postal-code $d_{postal-code}$ and salary d_{salary} for research $p_{research}$.

$$\begin{aligned} lpp_{raw} = (&version, \text{'LPP\_RAW'}, lang, ppURI, \\ &\widehat{HEAD}, \widehat{DESC}, \widehat{I}, \emptyset, \{p_{research}\}, \{c_{C1}\}, \\ &\widehat{DPO}, dsr, lc, \emptyset) \end{aligned} \tag{5.40}$$

$$\begin{aligned} p_{research} = (&\text{'Research'}, false, false, \emptyset, \\ &\widehat{HEAD}, \widehat{DESC}, \{d_{postal-code}, d_{salary}\}, \\ &\widehat{PM}, \widehat{PSM}, \{dr_{C1}, dr_{C2}\}, \widehat{LB}, \widehat{ADM}, r) \end{aligned} \tag{5.41}$$

Within the purpose $p_{research}$, it is furthermore specified that both the companies dr_{C1} and dr_{C2} are allowed to process the data. The second company dr_{C2} is hereby also defined as optional (see Equation 5.39).

The postal-code $d_{postal-code}$ is anonymized using *Suppression*, whereas the *Minimum Anonymization Level* is proposed with '1', which can be altered by the user up to the *Maximum Anonymization Level* of '3' (see Equation 5.42). Furthermore, it is specified that the purpose is optional with the attribute required set to 'false'.

$$\begin{aligned} d_{postal-code} = (&\text{'postal-code'}, \text{'Text'}, false, \text{'QID'}, \widehat{DG}, am_1, \\ &\widehat{HEAD}, \widehat{DESC}) \end{aligned} \tag{5.42}$$

$$am_1 = (\text{'Suppression'}, \{ama_1, ama_2, ama_3, ama_4\}, \emptyset, \widehat{HEAD}, \widehat{DESC}) \tag{5.43}$$

$$ama_1 = (\text{'Suppression Replacement'}, \text{'*'}) \tag{5.44}$$

$$ama_2 = (\text{'Suppression Direction'}, \text{'backward'}) \tag{5.45}$$

$$ama_3 = (\text{'Minimum Level'}, \text{'1'}) \tag{5.46}$$

$$ama_4 = (\text{'Maximum Level'}, \text{'3'}) \tag{5.47}$$

Also the postal-code $d_{postal-code}$ is optional, but the SD attribute salary d_{salary} is required. For salary, *Deletion* is specified as the anonymization method with a *Minimum* and *Maximum Anonymization Level* of '0', which specifies that the value for salary is not altered for the processing.

$$d_{salary} = (\text{'salary'}, \text{'Number'}, true, \text{'Sensitive'}, \widehat{DG}, am_2, \widehat{HEAD}, \widehat{DESC}) \tag{5.48}$$

$$am_2 = (\text{'Deletion'}, \{ama_5, ama_6\}, \emptyset, \widehat{HEAD}, \widehat{DESC}) \tag{5.49}$$

$$ama_5 = (\text{'Minimum Level'}, \text{'0'}) \tag{5.50}$$

$$ama_6 = (\text{'Maximum Level'}, \text{'0'}) \tag{5.51}$$

5.3.2 *Negotiation*

Within the *Negotiation* phase of the LPL life cycle, the raw LPL privacy policy lpp_{raw} is presented to the user e_{U1}, thus enabling an informed and voluntary consent (see Section 5.6.2).

Following the extension of the personal privacy concept, the user can at any time decide on the purposes, data, data recipients and anonymization level of personal data that is processed. Thus, the elements *Purpose*, *Data*, *DataRecipient* and *AnonymizationMethodAttribute* may be altered or removed from the policy. Assuming an optional *Purpose-*, *Data-*, or *DataRecipient*-element is rejected by the user, it is then removed from the policy. If the user alters the *Minimum Anonymization Level* for an attribute, then the corresponding *AnonymizationMethodAttribute*-element is updated.

The *Negotiation* phase of the life cycle is completed with the user's agreement to the personalized LPL privacy policy $lpp_{ds_{U1}\text{-}\{dr_{C1}, dr_{C2}\}}$. If the user e_{U1} agrees on the policy, then the personalized instance of the LPL policy is stored. Moreover, the *DataSource*-object ds_{U1} representing the user is set (see Equation 5.52).

$$lpp_{ds_{U1}\text{-}\{dr_{C1}, dr_{C2}\}} = (version, \text{'LPP\_U1'}, lang, ppURI, \widehat{HEAD}, \widehat{DESC}, \hat{I}, ds_{U1}, \{p'_{research}\}, \{c_{C1}\}, \widehat{DPO}, dsr, lc, \emptyset) \tag{5.52}$$

Furthermore, for each purpose, the attribute pointOfAcceptance is filled with the current date '[CURRENT_DATE]'. In the given scenario, it is assumed that the user accepts the processing of his personal data for research (see Equation 5.53).

$$\begin{aligned} p'_{research} = (&\text{'Research'}, false, false, \text{[CURRENT\_DATE]}, \\ &\widehat{HEAD}, \widehat{DESC}, \{d_{postal-code}, d_{salary}\}, \\ &\widehat{PM}, \widehat{PSM}, \{dr_{C1}, dr_{C2}\}, \widehat{LB}, \widehat{ADM}, r) \end{aligned} \tag{5.53}$$

Lastly, it is assumed hat the user agrees to the optional $d_{postal-code}$ and does not alter the *Minimum Anonymization Level*. Therefore, no additional changes have to be conducted on the policy.

If the user does not agree on the policy, the processing of his personal data is not allowed, e.g., the registration for an online shop cannot be completed.

5.3.3 *Pre-Processing*

The *Pre-processing* phase is conducted after each change of an instance of the LPL privacy policy in order to verify and validate the privacy policy. Additionally, the anonymization hierarchy is pre-processed for each *Data*-element.

The verification of the LPL structure guarantees the correctness of the syntax and also verifies that specific conditions of the privacy policy structure are met. For example, at least one *Safeguard*-element sg has to be specified for a *DataRecipient*-element dr if the thirdCountryTransfer attribute is set to 'true', or it has to be verified that at least one *Controller*-element c is specified.

The validation covers the comparison of the personalized LPL privacy policy $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$ to the raw LPL privacy policy lpp_{raw}. For example, it has to be verified that all required purposes in lpp_{raw} are also present in $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$, or that the *Maximum Anonymization Level* is not altered.

Lastly, the anonymization hierarchy has to be pre-processed and added via *HierarchyEntry*-elements to the corresponding *AnonymizationMethod*-elements. Hereby, the original values of the user are required. Within this scenario, the personal data of Alice (see Table 3.1) for ds_{U1} having the postal-code '94032' and a salary of '30.000' is assumed. Therefore, $\widehat{\mathbf{HE}}_{\mathbf{postal-code}}$ (see Equation 5.54) is added to the *AnonymizationMethod*-element am_1 for the postal-code of Alice.

$$\begin{aligned} \widehat{HE}_{postal-code} = \{&('94032'), ('9403*'), ('940**'), \\ &('94***'), ('9****'), ('*****')\} \end{aligned} \tag{5.54}$$

The anonymization hierarchy for the salary is added to the *AnonymizationMethod*-element am_2, but only consists of the original value and a place-holder character for its deletion (see Equation 5.55).

$$\widehat{HE}_{salary} = \{('30.000'), ('*')\} \tag{5.55}$$

This *Pre-processing* phase has to be repeated if either the privacy policy or the personal data of the user is updated or altered. The frequency hereby strongly depends on the specific use case.

5.3.4 *Storage*

After the personalized LPL privacy policy $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$ has been verified, validated and the anonymization hierarchies have been pre-processed, the policy instance is stored. The storage solution has to allow the linkage of the sticky policy to the personal data, such that a logical connection is supported at any time. Although the policy instance stores the original values of the personal data via the anonymization hierarchies, the LPL policy is not supposed to act as the storage solution for the personal data. For the storage of data, more reliable technologies should be used, i.e. relational databases.

5.3.5 *Transfer*

Personal data is not only stored within a service but it is also common that it is transferred intra- or inter-companies, e.g., for publishing, data trading, or third party processing. This *Transfer* of personal data requires the transfer of the corresponding personalized LPL privacy policies as required by the sticky policy concept. But a company may alter the conditions for processing the personal data by third parties to protect its users, e.g., it can require stricter privacy requirements or remove processable attributes from the data-set. Thus, a corresponding policy for the transfer is created. This new policy must be at least as strict as the previously defined policies between the users and the company (see Section 2.6). Therefore, a validation of the new and refined policy against the existing policies of the users has to be conducted.

Considering the *R6 Provenance* requirement, a refined policy incorporates the previous policies such that the source policy for personal data can be determined. This enables companies to show their compliance to individuals' consent and agreements, as they are accountable if the source of personal data cannot be presented.

For the given scenario, it is assumed that a policy between e_{C1} and e_{C2} is agreed upon. This secondary policy $lpp_{ds_{C1}\text{-}\{dr_{C2}\}}$ (see Equation 5.56) is based upon the original policy $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$, but only allows e_{C2} to process the postal-code $d_{postal-code}$ for research $p_{C2-research}$ (see Equation 5.57). The policy is altered, such

that the *DataSource* is ds_{C1} and the *Controller* is c_{C2} representing the change of responsibility for the policy. Additionally, the original policy $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$ is set as the *UnderlyingPrivacyPolicy* upp.

$$\begin{aligned} lpp_{ds_{C1}\text{-}\{dr_{C2}} = (&version, \text{'LPP\_C1'}, lang, ppURI, \\ &\widehat{HEAD}, \widehat{DESC}, \widehat{I}, ds_{C1}, \{p_{C2-research}\}, \\ &\{c_{C2}\}, \widehat{DPO}, dsr, lc, lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}) \end{aligned} \tag{5.56}$$

$$\begin{aligned} p_{C2-research} = (&\text{'Research'}, false, false, \\ &[\text{CURRENT\_DATE}], \widehat{HEAD}, \widehat{DESC}, \\ &\{d_{postal-code}\}, \widehat{PM}, \widehat{PSM}, \{dr_{C2}\}, \\ &\widehat{LB}, \widehat{ADM}, r) \end{aligned} \tag{5.57}$$

The validation and layering of policies has to be conducted for each record and policy that is transferred. Although this is a slight overhead, this *Transfer* phase enables *R6* and privacy-aware validation and usage of each individual record even after it has been transferred. The *Transfer* phase may be repeated several times creating a chain of layered privacy policies, each added policy at least as 'strict' as the underlying policy.

5.3.6 *Usage*

The main intention of collecting and storing personal data is of course to use it for various purposes to gain added value, e.g., research or data mining. This phase is denoted as *Usage*. If a company, i.e. any of its employees or services, intends to process personal data, it has to verify that this processing is allowed. Furthermore, it is possible that this personal data has to be protected by applying de-identification methods. This is especially challenging in situations for which high volumes of personal data records are processed, e.g., in a data-warehouse scenario, due to the possibility of varying personalized LPL privacy policies for each record.

Section 5.5 details how access control mechanism are realized by LPL. The derivation of a uniform privacy level from various LPL privacy policies and its application on the data-set are detailed in Chapter 6.

Thus, the fulfilment of the requirements for a privacy language of Chapter 2 is detailed. First, *R2 Legal Compliance* is detailed. Next, the previously mentioned *R4 Access Control* mechanisms are detailed. For the *R3 Human-readability* requirement, a proof-of-concept implementation of user interfaces for the *Creation* and *Negotiation* phase are detailed. The *R6 Provenance* requirement is detailed in Section 5.7. Lastly, the fulfilment of the *R5 De-identification Capabilities* requirement is detailed in the Chapter 6.

5.4 LEGAL COMPLIANCE

In Section 2.2, the requirement *R2 Legal Compliance* has been detailed, for which the *GDPR* is considered as legal framework. In the following, both the requirements for privacy policies according to Art. 12 - 14 *GDPR* as well as the *Data Subject Rights* are considered and used for the qualitative evaluation of LPL.

5.4.1 *Privacy Policy*

The main articles of the *GDPR* dealing with the requirements for privacy policies are Art. 12 - 14, which are compared to the capabilities of LPL as detailed by Gerl and Pohl [119] (see Table 5.2).

General provisions for the communication to the user, especially regarding transparency, are stated in Art. 12 *GDPR* [110, Art. 12]. It is stated that a privacy policy has to be provided in a clear and plain language [110, Art.12 (1) Sentence 1], which is enabled through the *UIElement* `ui`. This super-element provides all key elements with human-readable headers and descriptions. Hereby, the *R2 Legal Compliance* and *R3 Human-readability* requirement are complementarily considered. But, the legal requirement for a clear and plain language can not be guaranteed by LPL but has to be ensured by the responsible entity creating the policy, i.e. the *Controller*. Furthermore, the privacy policy can be provided in a written or electronic form [110, Art.12 (1) Sentence 2], under which LPL falls as an electronic format with its root-element *LayeredPrivacyPolicy* `lpp`. The last requirement derived of Art. 12 *GDPR* allows the usage of standardized icons [110, Art. 12 (7)], which are covered by LPL through the introduction of the *Icon*-element `i` for *Privacy Icons*, which is also considered to fulfil the *R3 Human-readability* requirement.

The Art. 13 and the Art. 14 have very similar content and are therefore compared combined in the following. Art. 13 details the information that has to be provided when personal data is collected from the user [110, Art. 13], i.e. after the privacy policy has been agreed/consented to, and Art. 14 describes the information that has to be provided when personal data is not directly collected from the user [110, Art. 14], i.e. when the personal data is collected by a third party. Both articles demand that the identity of the *Controller* and its contact details are provided [110, Art. 13 (1)(a), Art. 14 (1)(a)]. This is represented in LPL via a set of *Controller*-elements `c`, whereas the set is required to also represent *Joint Controllers* [110, Art. 26]. Furthermore, the contact details of the responsible *DPO* has to be provided [110, Art. 13 (1)(b), Art. 14 (1)(b)]. Hereby also several *DPOs* may be defined, which is covered by the set of *DataProtectionOfficer*-elements `dpo` referenced by the *LayeredPrivacyPolicy*-element `lpp`.

The purposes of the processing of personal data and their legal basis, including the legitimate interests [110, Art. 13 (1)(d), Art. 14 (2)(a)], have to be stated [110, Art. 13 (1)(c), Art. 14 (1)(c)]. Therefore, each *Purpose*-element p has a set of *LegalBasis*-elements lb. Furthermore, the user has to be informed about the collected data categories [110, Art. 14 (1)(d)] modelled by the *DataGroup*-element dg. Each *Data*-element d references a set of *DataGroup*-elements dg, such that a data can be associated with several groups. The required personal data has to be communicated [110, Art. 13(2)(e)], which is modelled by the *Data*-element d having the required attribute.

The data recipients for the personal data [110, Art. 13 (1)(e), Art. 14 (1)(e)] and, if the data is transferred in a third country, the applied safeguards [110, Art. 13 (1)(f), Art. 14 (1)(f)] have to be provided. In LPL this is modelled with a set of *DataRecipient*-elements dr which have the thirdCountryTransfer attribute indicating a third country transfer and a set of *Safeguard*-elements sg if the context requires it.

The storage period for the personal data has to be provided to the user [110, Art. 13(2)(a), Art. 14(2)(a)], which is modelled by the *Retention*-element r.

Furthermore, the user has to be informed about his *Data Subject Rights* [110, Art. 13(2)(b), Art. 14(2)(c)] and how to lodge a complaint [110, Art. 13(2)(d), Art. 14(2)(e)]. The *Data Subject Rights* are implemented by a *DataSubjectRights*-element dsr and a *LodgeComplaint*-element lc, which are referenced by lpp. These elements mainly define human-readable texts, i.e. *Header*- and *Description*-elements, to inform the users about their *Data Subject Rights* and how to lodge a complaint.

If personal data of a user is used for automated decision-making, the user has to be informed about it [110, Art. 13(2)(f), Art. 14(2)(g)]. In LPL, processing for automated decision-making is denoted for each purposes, such that each p optionally references an *AutomatedDecision-Making*-element adm.

The possibility to withdraw consent has to be communicated to the user [110, Art. 13(2)(c). Art. 14(2)(d)], which is implicitly modelled by the required attribute in the *Purpose*-element p based on this attribute. A suitable user interface (see Section 5.6.2) enables consent-management. This concept is further extended to the *Data*-element d and *DataRecipient*-element dr, thus enabling several personalization options. Therefore, the *R3 Human-readability* requirement also has to be considered [116] [115].

Lastly, the user, i.e. *Data Subject*, has to be informed about the source of personal data and if this source is publicly available [110, Art. 14(2)(f)], which is modelled by the *DataSource*-element with the public attribute indicating a public source.

Thus, LPL shows the capabilities to model all information required by Art. 12 - 14 *GDPR*. The information contained in LPL has to be processed and visualized in a suitable and human-friendly way (see Sec-

tion 5.6). Therefore, the *R2 Legal Compliance* and *R3 Human-readability* are closely related. The requirement stating the realization of *Data Subject Rights*, which implies actions on the data, cannot be covered by LPL, which only defines the structure of privacy policies. But the required information to perform the *Data Subject Rights* actions can be considered by LPL and is detailed in the following.

Table 5.2: Fulfilled legal requirements of Art. 12 - 14 [110, Art. 12 - 14] by LPL.

GDPR	
Requirement	**LPL**
Clear and Plain Language	*UIElement* `ui`
Written or Electronic Information	*LayeredPrivacyPolicy* `lpp`
Data Subject Rights Realization	Section 5.4.2
Standardised Icons	*Icon* `i`
Contact Details of Controller	*Controller* `c`
Contact Details of DPO	*DataProtectionOfficer* `dpo`
Purpose and Legal Basis	*Purpose* `p`, *LegalBasis* `lb`
Legitimate Interest	*LegalBasis* `lb`
Categories of Personal Data	*DataGroup* `dg`
Recipients of Personal Data	*DataRecipient* `dr`
Third Country Transfer	*DataRecipient* `dr`, *Safeguard* `sg`
Storage Period	*Retention* `r`
Information: Data Subject Rights	*DataSubjectRight* `dsr`
Information: Withdraw Consent	*Purpose* `p`
Information: Lodge a Complaint	*LodgeComplaint* `lc`
Information: Required Data	*Data* `d`
Source of Personal Data	*DataSource* `ds`
Automated Decision-Making	*AutomatedDecisionMaking* `adm`

5.4.2 *Data Subject Rights*

The realization of *Data Subject Rights* [110, Art. 12 (2)] requires various information, e.g., the processed personal data, available to answer the individual requests. In general, the response time to a *Data Subject Right* has to be within 1 to 3 months [110, Art. 12 (3)]. It can be assumed that the response time is so huge, because the process for answering a *Data Subject Right* request is commonly not supported by technical means in practice and therefore can require extensive manual labour to verify and answer the request. Furthermore, the *Controller*, e.g., the company, can refuse to answer excessive requests

[110, Art. 12 (5)], e.g., if continuous requests are made by a user. Both requirements can be achieved by technical means. Furthermore, the identity of the requesting user has to be verified, i.e. authenticated, to mitigate any misuse of the *Data Subject Rights*. The authentication may be conducted using the access control capabilities of LPL (see Section 5.5) or by other means. The validity of the request has to be checked before the response is generated. Hereby the response can be based on the individual's privacy policy and/or his personal data (see Figure 5.4). In the following, each of the *Data Subject Rights* is discussed.

First, the user has the *Right of Access by the Data Subject*, which allows him to request from the *Controller* if his personal data is processed. If personal data of the user is processed, then additional information can be requested. This includes purposes, personal data categories, data recipients, retention, information on the *Right to Rectification* and *Right to Erasure*, right to lodge a complaint, data source, information on automated decision-making, and safeguards [110, Art. 15 (1 - 2)]. Each of this information has already been noted within the analysis of Art. 12 - 14 *GDPR* and is integrated within LPL.

This information shall be provided to the user in a commonly used electronic form or via a physical copy [110, Art. 15 (3)], which is trivial to fulfil. Lastly, the rights and freedom of others should not be affected [110, Art. 15 (4)], which requires a distinction of this information based on the user's identity. This is achieved through individual LPL privacy policies for each user.

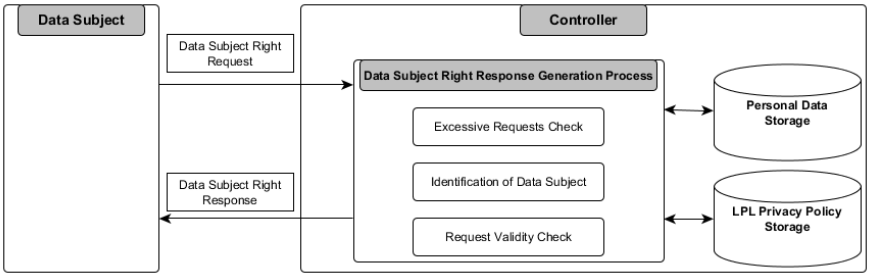


Figure 5.4: *Data Subject Right* scenario showing a schematic response generation based on the the individuals' personal data and LPL privacy policy.

Next, the user has the *Right to Rectification*, which gives him the right to request the correction of inaccurate or incomplete personal data [110, Art. 16]. Technically speaking, the user can request to update personal data. Therefore, it is necessary to identify and associate the data within the storage solution and update it. In LPL, both the user (*DataSource*) and data (*Data*) is uniquely identified, thus a mapping to the stored data values of the user is feasible and an update can be achieved.

The *Right to Erasure* or also denoted as *Right to be Forgotten* enables the user to request the deletion of his personal data under certain

conditions, e.g., when his consent is withdrawn. Note that exceptions for this right are defined, such that data is not erased, e.g., if it is required for reasons of public interest [110, Art. 16]. The validity of the reason for the request has to be decided involving the responsible entities, e.g., DPO and *Controller*. The technical realization is enabled by LPL due to the unique definition and association of personal data to the user via the *DataSource*- and *Data*-elements. The call of a deletion method with these parameters can be implemented directly deleting the data from the storage. However, it is common to use backup strategies, which redundantly store personal data. Thus, it is an issue that this right also affects backups, for which no standardized solutions are in place, which can deal with the various mediums and strategies used for backups. Furthermore, it is subject to an ongoing debate if backups may be an exception from this right [215] [214] [27].

Furthermore, the *Right to Restriction of Processing* allows the user to request a cease to the processing of his personal data. Again, a request requires a valid reason for the restriction, e.g., the processing is unlawful. The processing can only be continued if the user consents to it and the user has to be informed about the continuation of the processing [110, Art. 18]. Again it has to be first verified if the reason to restrict the processing is valid, for which the responsible entities are involved. The processing of personal data can be restricted utilizing the properties of LPL. Each processing is indeed expressed by a *Purpose*-element `p` defining the entities allowed to process the personal data, i.e. *DataRecipient*. The restriction of processing can be implemented via the deletion of the corresponding *DataRecipient* from a *Purpose*, which enables the partial restriction for a specific entity, or via the deletion of the corresponding *Purpose*-element from the policy, which completely prevents the processing. The restriction can only be lifted if consent is given, which can be expressed by either a *Purpose*-element or *DataRecipient*-element added to the policy with the attribute `required` set to `'false'`. Thus, an explicit action is required by the user. To inform the user about the lifted restriction, a simple notification about the purpose can be sent to him, whereas the user is identified via the *DataSource*-element. Thus, a technical realization is feasible based on LPL.

The *Data Subject Rights* for rectification, erasure and restriction are furthermore covered by the *Notification Obligation*, which requires the communication of the request to any other *Controller* processing the personal data. Furthermore, the user can request a list of recipients of his personal data [110, Art. 19]. In LPL, the *DataRecipient*-element expresses the entities receiving the personal data for processing, such that the information is accessible. Thus, this right can be fulfilled with the support of LPL.

The *Right to Data Portability* is a novelty within the *GDPR*, which is intended for transferring personal data between *Controllers*. Hereby,

two possibilities have to be considered. On the one hand, the user can request to receive personal data in a structured, commonly used and machine-readable format; in order to transmit it to another *Controller* himself. On the other hand, the user can request that one *Controller* directly transmits personal data to another *Controller* [110, Art. 20]. In both cases, LPL can serve in the identification of personal data and its values using the *Data*-elements as well as the first *HierarchyEntry*-element, which contains the original value. The processing of such data in a common machine-readable format and the transfer to the user is hereby trivial. But the direct transfer from one *Controller* to another is a hard challenge. Such enforcement of the law would require generic interfaces for personal data transfer between all possible *Controllers*. Thus, interpreting this law ad absurdum, a student could request his university to transfer his personal data, e.g., information on university degree, to a social media platform to inform his friends about his success. A limitation of the scope of this law, e.g., by the domain and size of the companies exchanging personal data, is therefore worth aspiring for [118]. The *Data Transfer Project (DTP)* [80] is an open-source, service-to-service data portability platform with the intention to tackle the transfer of personal data between several companies and services. Hereby, the company internal data format is translated into a common data model and transferred, afterwards the personal data is translated into the internal data format of the target company.

The *Right to Object* enables the user to object the processing of personal data. The user has to be informed about this right. The *Right to Object* states explicitly the right to object purposes which fall under the category of marketing, automated decision-making, and research [110, Art. 21]. Furthermore, the *Right to Object Automated Decision-making* redundantly defines the objection of processing of personal data for automated decision-making and its limitation [110, Art. 22]. Thus, a strong emphasis on automated decision-making is put by the *GDPR*. This objection can be expressed by LPL privacy policies in the same way as the restriction of processing through the deletion of the corresponding *Purpose*-element or *DataRecipient*-element. Furthermore, the user can be informed about the *Right to Object* via the *DataSubjectRight*-element.

Lastly, it is stated that Union or Member State law may restrict the *Data Subject Rights* [110, Art. 23]. Thus, national legal frameworks may limit and modify the *Data Subject Rights*.

In summary, this requirement analysis and its comparison to the capabilities of LPL shows that LPL is able to express all information necessary to fulfil *Data Subject Right* requests. However, the realization of such rights requires additional implementation of additional processes, e.g., for the notification of the user and other *Controllers* or for updating or deleting stored data. Furthermore, human-interaction is

required to decide if the conditions and reasons given by the user are valid.

It may be argued that the information about *Data Subject Rights* may not be expressed by only one *DataSubjectRight*-element, but a set of *DataSubjectRight*-elements for each individual right, for a clear distinction. This possibility is omitted because the information on *Data Subject Rights* consists in practice of standardized phrases for which a single element is sufficient. In conclusion, the *R2 Legal Compliance* requirement is fulfilled by LPL. Next, the realization of the *R4 Access Control* requirement is detailed utilizing LPL.

5.5 ACCESS CONTROL

The *R4 Access Control* requirement (see Section 2.4) states that a privacy language has to enable access control mechanisms. Access control requires to verify if an entity is authenticated and authorized to access data. Classical approaches include RBAC systems. For example a user has a role assigned, e.g., student, and the role is assigned the permission to read and write specified files. Therefore, the user has to be first authenticated before the authorization to access the files is verified as detailed by Gerl et al. [121] and Wilhelm and Gerl [276].

In the context of privacy policies, not only the user or his role have to be considered but also what data is allowed to be processed and for which purpose. The approach for LPL is inspired by the *Privacy-aware Role-based Access Control (P-RBAC)* of Ni et al. [200] and *Purpose-based Access Control (PBAC)* of Byun et al. [62], which extend classical RBAC approaches to support concepts of privacy policies. Furthermore, *Privacy-Aware Organization-Based Access Control (OrBAC)* **??** [73] is proposed, which enables the user to define its own purposes for which data is processed for a specific context. The context can consider temporal or spatial properties, the user defined purposes, as well as prerequisite and provisional contexts. LPL defines the access rules via its *Purpose-*, *Entity-* (*DataRecipient*-element), and *Data*-elements. Each *Purpose*-element can indeed be seen as a whitelist access rule. Thus, a *LayeredPrivacyPolicy*-element lpp contains a set of access rules that is specific for a users' data.

Within the *Usage* phase of the LPL life cycle, users' data is requested to be processed for a specific purpose. It is essential that access control is conducted to mitigate unauthorized access to personal data. Therefore, a *Policy-based Access Control (PAC)* mechanism based on LPL is proposed. PAC is intended to be not restricted to a single access control solution but to be integrated within various existing solutions. A PAC request to access data is defined as the following tuple:

$$\mathtt{request} = (\mathtt{userIdentifier}, \mathtt{credential}, \widehat{\mathrm{P}}_{req}, \widehat{\mathrm{D}}_{req}, \widehat{\mathrm{DS}}_{req}) \quad (5.58)$$

Where $\mathtt{userIdentifier}$ is the unique identifier of the requesting user, which corresponds to the attribute $\mathtt{name}$ of the *Entity*-element and $\mathtt{credential}$ defines the credential of the user for authentication. Furthermore, $\widehat{\mathrm{P}}_{\mathrm{req}}$ is the set of all requested purposes, which corresponds to the $\mathtt{name}$ of *Purpose*-elements. The requested attributes are defined by $\widehat{\mathrm{D}}_{\mathrm{req}}$, which corresponds to the $\mathtt{name}$ of the *Data*-elements. Lastly, the set of records, i.e. users, from which data is requested is defined by $\widehat{\mathrm{DS}}_{\mathrm{req}}$.

Based upon this request, *Entity-Authentication*, *Purpose-Authorization*, *Entity-Authorization*, and *Data-Authorization* are conducted in the given order to verify that the requesting entity is authorized to access the personal data for the specified purposes from the set of users (see Figure 5.5).

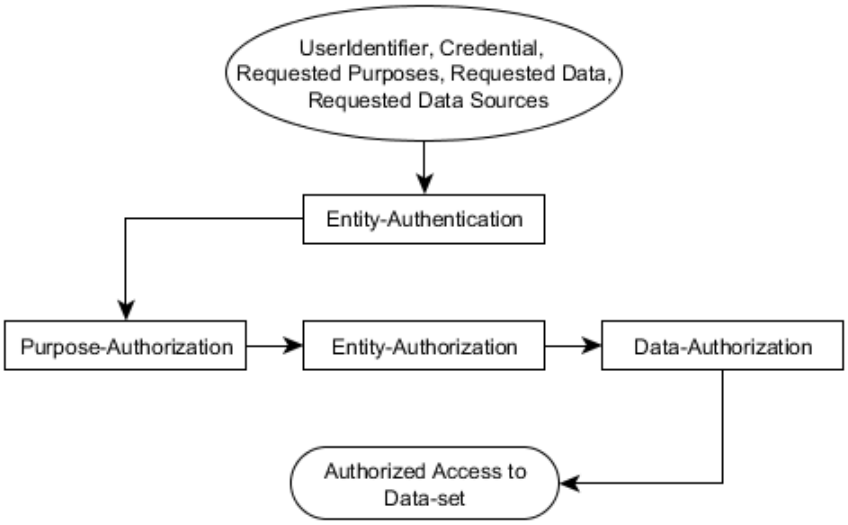


Figure 5.5: Process chain for the *Policy-based Access Control (PAC)*.

Before each of the PAC processes is detailed, several required support structures are detailed in the following.

5.5.1 *Support Structures*

LPL privacy policies should be considered as the rule sets for individuals' personal data access. In the *Transfer* phase of the LPL life-cycle, it is defined that the privacy policy may be refined to be stricter for third parties. Within companies a more fine-grained access control is required. Privacy policies and data recipients are indeed defined in a generic and high-level way. To meet the users' and companies' requirements, more fine-grained definitions are required. Therefore, the *Purpose-Hierarchy* and *Entity-Hierarchy* are introduced. Furthermore, the *Entity-Lookup Table* is introduced as a central register for entities. The *Entity-Lookup Table* and *Entity-Hierarchy* are used for *Entity-*

Authentication and *Entity-Authorization*, while the *Purpose-Hierarchy* is used for the *Purpose-Authorization* process.

5.5.1.1 Entity-Hierarchy

The *Entity-Hierarchy* allows to define *Child-Entities* that inherit the rights of the *Parent-Entities*. An entity can hereby define an individual or groups of individuals (roles).

Assuming a privacy policy, the data recipients are often defined in a generic way, e.g., a company may specify 'trusted third parties' or 'advertisement agencies' as recipients of personal data. This makes sense from a privacy policy point of view, because these entities might be replaced or extended. Thus, a generic definition allows to alter business partners while the privacy policy stays valid. These generic definitions of entities have then to be refined to match the intended access rules of the company, e.g., it has to be defined which companies fall under 'trusted third parties'. The same is applicable for internal company access control to personal data, e.g., definition of individual departments or individuals. This is enabled by *Entity-Hierarchy*, which is defined as a lattice.

Hereby, only the unique name is needed within the *Entity-Hierarchy* (see Figure 5.6). Assuming a user agrees to the processing of his personal data by dr_{C1} 'DR_C1', then this can be assumed as a *Parent-Entity* for which more fine-grained *Child-Entities* can be added to the *Entity-Hierarchy*, e.g., employees 'DR_EMP1' and 'DR_EMP2'. It is also possible that a *Child-Entity* inherits from two *Parent-Entities*, e.g., the employee 'DR_EMP2'. This represents the use case when several users allow a company to use their data. If the processing of a *Parent-Entity* is agreed to, then the processing of each of its *Child-Entities* is allowed, otherwise not.

5.5.1.2 Entity-Lookup Table

The *Entity-Lookup Table* is introduced to provide a centralised structure to look up entities. It is intended as a hash-map to allow a fast lookup of entities without traversing the *Entity-Hierarchy*, e.g., during the *Entity-Authentication* process.

The *Entity-Lookup Table* is the set of all entities (ds and dr) that exist within all stored lpp and additionally all entities that are defined in *Entity-Hierarchy*. Each entry of the *Entity-Lookup Table* defines the name and authInfo attribute. The authInfo defines the value that is authenticated against, e.g., the public key if a public/private key authentication is used or the hashed password if a username/password authentication is used.

Considering the entities defined in Section 5.3, each distinct entity, defined by the name, is added to the *Entity-Lookup Table* (see Table 5.3).

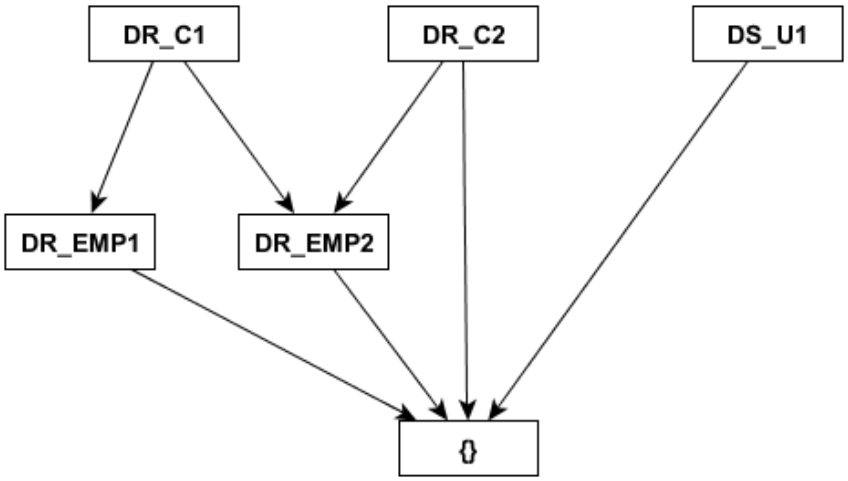


Figure 5.6: Possible structure of *Entity-Hierarchy* based on the scenario in Section 5.3). The {}-node is a default node with no access rights assigned.

Table 5.3: General structure of *Entity-Lookup Table* shown for the scenario introduced in Section 5.3 as an example.

Entity-Lookup Table	
name	*authInfo*
DS_U1	$\texttt{publicKey}_{U1}$
C_C1	$\texttt{publicKey}_{C1}$
DR_C1	$\texttt{publicKey}_{C1}$
DS_C1	$\texttt{publicKey}_{C1}$
C_C2	$\texttt{publicKey}_{C2}$
DR_C2	$\texttt{publicKey}_{C2}$

5.5.1.3 Purpose-Hierarchy

The *Purpose-Hierarchy* is introduced for the *Purpose-Authorization* process. Its intention is similar to the *Entity-Hierarchy*. In privacy policies, usually only generic high level purpose are defined, e.g., 'Research' or 'Marketing'. To allow the company to define fine-grained purposes that can be assigned to third party companies or internal departments, the *Purpose-Hierarchy* is introduced.

The *Purpose-Hierarchy* is a forest that consists of several hierarchical trees of purposes. For each purpose, it is possible to define *Child-Purposes* that inherit from the *Parent-Purpose* (see Fig. 5.7). The unique `name` of the *Purpose*-element `p` defines each element of the *Purpose-Hierarchy*. Assuming a user agrees to the processing of his personal data for the purpose 'Marketing', then this can be assumed as a *Parent-Entity* for which fine-grained *Child-Entities* can be added to

the *Purpose-Hierarchy*, e.g., the purpose 'Direct Marketing'. It is not possible that a *Child-Purpose* inherits from two *Parent-Purposes*. If the processing of a *Parent-Purpose* is agreed to, then the processing of each of its *Child-Purposes* is allowed, otherwise not.

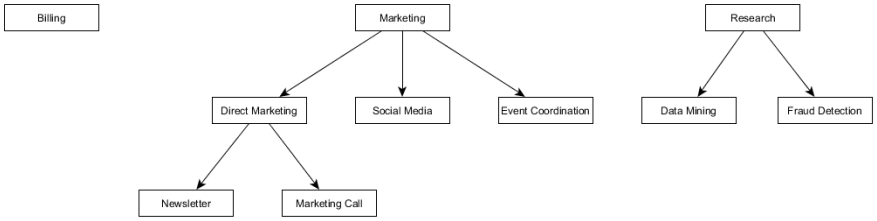


Figure 5.7: *Purpose-Hierarchy* showing a possible inheritance hierarchy for purposes based upon the online shop scenario of Section 5.3.

Next, the PAC processes are detailed, which utilize the *Entity-Hierarchy*, *Entity-Lookup Table* and *Purpose-Hierarchy*.

5.5.2 *Entity-Authentication*

Entity-Authentication is necessary to identify the entity e_{req} that requests the usage of data. Assuming the *Entity-Lookup Table* from Table 5.3, every entity and its $\mathtt{authInfo}$ is accessible. In the following, the usage of $\mathtt{publicKey}$ and $\mathtt{privateKey}$ for the authentication is assumed. During the *Usage* step, an entity e_{req} requests data protected by the privacy policy $\mathtt{lpp}_{\mathtt{ds}_{U1}\text{-}\{\mathtt{dr}_{C1}\}}$. To confirm the identity of e_{req}, it has to be authenticated. Therefore, it is assumed that an employee of company e_{C1} requests data protected by $\mathtt{lpp}_{\mathtt{ds}_{U1}\text{-}\{\mathtt{dr}_{C1}\}}$ (see Equation 5.59).

$$\begin{aligned}\mathtt{request}_{C1} = (&\text{DR\_C1}, \mathtt{privateKey}_{C1}, \{\text{Data Mining}\},\\ &\{\text{postal-code}, \text{salary}\}, \{\mathtt{ds}_{U1}\})\end{aligned} \tag{5.59}$$

Hereby, $\mathtt{request}_{C1}$ defines 'DR_C1' as the $\mathtt{userIdentifier}$ and $\mathtt{privateKey}_{C1}$ as the credential. Hereby, $\mathtt{privateKey}_{C1}$ matches to $\mathtt{publicKey}_{C1}$.

In general, the following authentication process is assumed for the scenario. The $\widehat{\mathtt{E}}_{look-up}$ is traversed to identify matching entities to the requesting entity by its $\mathtt{userIdentifier}$ to the corresponding $\mathtt{name}$ of the *Entity-Lookup Table*, i.e. 'DR_C1'. If a match is found, the $\mathtt{credential}$ is used to authenticate the entity via the $\mathtt{authInfo}$ of the *Entity-Lookup Table*.

Assuming public/private keys for the authentication in this scenario, the following process can be used for authentication. If a matching entity e_{match} is found, then the $\mathtt{publicKey}_{match}$ is used to encrypt a $\mathtt{nonce}$ and send it to the requesting entity.

$$\begin{aligned}\mathtt{encryptedMessage}_{match} = \mathtt{encrypt}(&\mathtt{nonce},\\ &\mathtt{publicKey}_{match})\end{aligned} \tag{5.60}$$

To successfully authenticate the requesting entity, the computed encrypted message has to be decrypted with the $privateKey_{requesting}$ and sent back.

$$decryptedMessage = decrypt(encryptedMessage_{match}, privateKey_{requesting}) \tag{5.61}$$

The requesting entity is authenticated if the $decryptedMessage$ equals the $nonce$.

$$nonce == decryptedMessage \tag{5.62}$$

Considering the scenario, the requesting entity provides the identifying $userIdentifier$ 'DR_C1', that is matched against the *Entity-Lookup Table* of Table 5.3. Therefore, the $authInfo$ $publicKey_{C1}$ can be identified and matched with the credential of $request_{C1}$ $privateKey_{C1}$ (see Equation 5.63).

$$nonce = decrypt(encrypt(nonce, publicKey_{C1}), privateKey_{C1}) \tag{5.63}$$

The authentication fails if either the $userIdentifier$ cannot be found in the *Entity-Lookup Table* or the $authInfo$ and $credential$ do not match. For this scenario, it is assumed that the *Entity-Authentication* process results in a successful authentication for $request_{C1}$.

Furthermore, it is verified if the requesting entity matches a *DataRecipient*-element of the corresponding policy $lpp_{ds_{U1}\text{-}\{dr_{C1}\}}$ in the *Entity-Authorization*. But the *Purpose-Authorization* is conducted beforehand, following the overall purpose-based structure of LPL.

In practice, asymetric authentication protocols or challenge-response authentication methods should be used [86] [231] [265]. Examples for well-known asymetric authentication protocols are the Needham-Schroeder protocol [192] with Lowe's [173] correction or Kerberos [197].

5.5.3 *Purpose-Authorization*

The *Purpose-Authorization* process verifies if the $request$ has an authorized purpose, once the requesting entity has been authenticated. Therefore, the requested purposes $\widehat{P}_{req}$ have to be matched with the purposes of the LPL privacy policies. Hereby, not only a direct match is considered, but also matches to *Child-Purposes*, which are stored in the *Purpose-Hierarchy*. The request is rejected if no purpose can be authorized.

The policy $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$ from Equation 5.52 is assumed to be subject of the request. Moreover, the purpose 'Research' is available in the *Purpose-Hierarchy*, as well as its *Child-Purposes* 'Data Mining' and 'Fraud Detection' (see Figure 5.7).

```
1
 method: authorizePurpose
 in: requestedPurpose; lpp //requested purpose for the request; LPL policy
 out: P //set of authorized purposes

6    //initialize authorizedPurposes
     authorizedPurposes = {};

     //receive set of possible purposes
     for p : lpp.P
11       possiblePurposes = purposeHierarchy.getChildPurposes(p);

         //verify if purpose matches at least one p of lpp
         for possibleP : possiblePurposes

16           if match(possibleP, requestedPurpose)
                 authorizedPurposes.add(p);

     return authorizedPurposes;
```

Listing 5.1: Pseudocode describing the authorization of purposes of a lpp utilizing *Purpose-Hierarchy*. The *Entity-Hierarchy* is assumed to be accessible within the method.

Assuming $\mathrm{request}_{C1}$, the purpose 'Data Mining' is given. This purpose, by itself, does not match any of the purposes given in $\mathrm{lpp}_{\mathrm{ds}_{U1}\text{-}\{\mathrm{dr}_{C1},\mathrm{dr}_{C2}\}}$, i.e. 'Research'. Because a *Purpose-Hierarchy* is assumed in which every *Child-Purpose* of the authorized purpose is also authorized, a set of *Authorized Purposes* $\widehat{\mathrm{P}}_{\mathrm{auth}}$ can be derived. Therefore, the purpose 'Research' is identified in the *Purpose-Hierarchy* and all its *Child-Purposes* as well. $\widehat{\mathrm{P}}_{\mathrm{possible\text{-}research}}$ is returned as Equation 5.64:

$$\widehat{\mathrm{P}}_{\mathrm{possible\text{-}research}} = \{\text{'Research', 'Data Mining', 'Fraud Detection'}\} \tag{5.64}$$

The requested purpose 'Data Mining' is therefore authorized, because the requested purpose 'Data Mining' is present within the possible authorized purposes $\widehat{\mathrm{P}}_{\mathrm{possible\text{-}research}}$.

In general, the authorization process for a purpose requires the name of the purpose and the corresponding lpp instance. The authorization is successful if the name of the requested purpose matches any of the p of the lpp or any of the corresponding *Authorized Entities*. The *Purpose-Authorization* process has the task of gathering all authorized *Purposes* for a specific request, whereas each requested purpose of $\widehat{\mathrm{P}}_{\mathrm{req}}$ is processed individually (see Listing 5.1).

5.5.4 *Entity-Authorization*

After all authorized purpose have been identified, it is verified if the requesting entity is eligible to process the data for the authorized

```
1
 method: authorizeEntity
 in: userIdentifier; p //userIdentifier of requesting entity; authorized
     purpose
 out: boolean //true, iff requesting entity is authorized for a purpose,
     otherwise false

6 boolean authorizeEntity(userIdentifier, p):

    //for each data recipient of the authorized purpose
    for dr : p.DR
        //receive set of authorized entities
11      possibleDR = entityHierarchy.getChildEntities(dr).

        if possibleDR != null
            //verify if userIdentifer matches at least one dr of p
            for dr : possibleDR
16              if match(userIdentifer, dr.name)
                    return true;

    return false;
```

Listing 5.2: Pseudocode describing the authorization of a requesting entity for an authorized purpose utilizing *Entity-Hierarchy*. The *Entity-Hierarchy* is assumed to be accessible within the method.

purposes. Hereby, the userIdentifier is matched with the *DataRecipient*-elements dr of the authorized purposes. This *Entity-Authorization* process follows the same basic principles as the *Purpose-Authorization* process. The *Entity-Hierarchy* is utilized hereby in a similar manner.

A requesting entity, identified by the userIdentifier, is authorized iff the name of a dr for an authorized purposes is matched. The $request_{C1}$ is assumed for the policy $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$ for which $p_{research}$ is identified as an authorized purpose. If the userIdentifier 'DR_C1' of $request_{C1}$ matches the name of dr_{C1}, it is authorized.

However, employees of the company may be defined that have more restricted roles. Assuming the userIdentifier 'DR_EMP2' from the *Entity-Hierarchy* of Figure 5.6 does not match dr_{C1} nor dr_{C2}. Considering the *Entity-Hierarchy*, 'DR_EMP2' can be identified as a *Child-Entity* of dr_{C1} and dr_{C2}. Therefore, also an entity with the userIdentifier 'DR_EMP2' is eligible for $p_{research}$.

In general, the authorization process for an entity requires the userIdentifier of the requesting entity as well as the purpose for which it should be verified against (see Listing 5.2). The authorization is successful iff the userIdentifier matches any of the name of dr of the purpose or any of its *Child-Entities*. This process has to be conducted after the *Purpose-Authorization*. If the authorization fails for a purpose, then this purpose is removed from the authorized purposes for the requesting entity.

```
1
 method: authorizeData
 in: requestedD; P //set of requested data; set of authorized purposes
 out: D //set of authorized data

6    //initialize authorizedData
     authorizedData = {};

     //check for each requested data if it is authorized
     for d : requestedD
11       //several authorized purposes are possible
         for p : P
             for dAuthorized : p.D
                 if match(d.name, dAuthorized.name)
                     authorizedData.add(d);
16
     return authorizedData;
```

Listing 5.3: Pseudocode describing the authorization of data from authorized purposes.

5.5.5 *Data-Authorization*

Lastly, the *Data-Authorization* process is conducted. It is intended to verify that the requested data $\widehat{D}_{req}$ matches the defined data defined in the authorized purpose. It must indeed be prevented that more data than allowed in lpp can be requested.

Unlike in the *Purpose-* and *Entity-Authorization*, no additional support structures are needed. This process basically implements a direct match between the requested data and the data for which the authorized purpose is defined. Hereby, the requested data $\widehat{D}_{req}$ has to match or be a subset of the defined set of *Data*-elements d of the purpose p. If the verification is not successful, then the query is rejected.

Assuming the $request_{C1}$ requests the data $'postal-code'$ and $'salary'$. For each of the requested data, it has to be verified if it is contained within $p_{research}$. In this scenario, $p_{research}$ contains d_{salary} and $d_{postal-code}$ which matches the requested data $'postal-code'$ and $'salary'$ when comparing it to the $name$ of the corresponding d.

If the data that is requested is not within the authorized purpose, e.g., $'age'$, then the *Data-Authorization* fails for the specific authorized purpose. The *Data-Authorization* requires the $name$ of the data d of the corresponding purpose p (see Listing 5.3). The authorization is successful if the $name$ of the requested data matches any of d from any *AuthorizedPurpose*.

When the $request$ is run through the *Entity-Authentication*, *Purpose-Authorization*, *Entity-Authorization*, and *Data-Authorization*, it is stated that the requesting entity is eligible to process the data for the specified purpose according to the LPL privacy policy lpp. Therefore, the *R4*

Access Control requirement is fulfilled. This process is the basis for the *Policy-based De-identification* process, which is detailed in Chapter 6.

In the following section, user interfaces for the creation, presentation and negotiation of LPL privacy policies are proposed.

5.6 PRIVACY POLICY USER INTERFACE

As previously detailed, the *R3 Human-readability* requirement is considered during the creation of LPL. Within the life cycle of LPL (see Section 5.3), human interaction with the privacy policy is indeed essential during the *Creation* and *Negotiation* phase. During the *Creation* phase, the privacy policy is created, for which it is essential to provide suitable user interface to support the process. The user shall hereby be informed about the contents of the privacy policy to comply with the *GDPR*, thus also the *R2 Legal Compliance* requirement has to be considered during the creation of the user interface. Additionally, the user can personalize the privacy policy according to his privacy requirements.

In the following, a proof-of-concept implementations for both use cases is proposed to demonstrate the capability of LPL for fulfilling the *R3 Human-readability* requirement [120]. The general layout of the user interface is detailed, which is utilized for both the *Negotiation View* and the *Creation View*. Furthermore, different negotiation scenarii are detailed for the *Negotiation View*. Lastly, the *Creation View* is detailed, which enables the creation of LPL privacy policies.

A web-based environment, e.g., an online shop, is assumed for the proof-of-concept implementations of the user interfaces. Thus, web-technologies are used for the realization of the following user interfaces [116] [115] [114].

5.6.1 *User Interface Layout*

In Section 2.3, several transparency key dimensions for privacy policies as proposed by Bertino et al. [36] are introduced. The detailed key dimensions – *Record Transparency*, *Use Transparency*, *Disclosure and Data Provisioning Transparency*, *Algorithm Transparency*, and *Law and Policy Transparency* – can be covered by the presentation of the corresponding elements of LPL, e.g., *Use Transparency* is covered by the *Purpose*-element `p`.

Furthermore, the *Layered* approach is detailed for the presentation of privacy policies, which suggests a separation and prioritisation of the content of the privacy policy, such that the user gets informed about the most important content first and then additional information through interaction[18].

This concept corresponds to the *Visual Information Seeking Mantra (VISM)*, which essentially states: *Overview* first, *Zoom* and *Filter*, then

Details-on-demand [247]. VISM defines seven tasks in total, for which an overview is given in the following.

- *Overview*: The user gains an overview of the content.
- *Zoom*: The user zooms on elements of interest.
- *Filter*: The user removes uninteresting elements.
- *Details-on-demand*: The user selects element(s) to get details.
- *Relate*: The user can view relationships between elements.
- *History*: A history of actions is kept to enable undo- and replay-functions, or progressive refinement of the selection.
- *Extract*: Selected elements can be extracted.

VISM is intended for information exploration, for which tasks like *Filter*, *History*, or *Extract* are useful [247]. Considering privacy policies, which contain a lot of information, not all VISM tasks are applicable. Thus, the focus lies only on a subset of VISM tasks. *Overview* is suitable to give the user an overview over the most important information of the privacy policy 'at a glance'. *Zoom* enables the user to display only information that is of interest for him. Lastly, *Details-on-demand* enables the user to inform himself about specific details of the privacy policy. These tasks are considered in the layout, which is visualized on the example of the *Negotiation View* in Figure 5.8.

5.6.1.1 *Policy Header*

The *Policy Header* incorporates the title of the policy, localization settings, i.e. drop-down menu for available languages, and a link to common legal privacy policy. The title of the policy is taken from the *Header*-element `head` referenced by the *LayeredPrivacyPolicy*-element `lpp`.

The localisation of the user interface, as well as the content of the privacy policy, is derived from the web-browser settings of the user. If the localisation of the web-browser is not available within the LPL privacy policy, the default localization defined within the `lang` attribute of `lpp` is used. Furthermore, the user can change the localisation manually using the corresponding drop-down menu.

Lastly, a link to the common privacy policy is given to comply with the current standard for representing privacy policies. This might be omitted, if privacy language based privacy policies are commonly accepted.

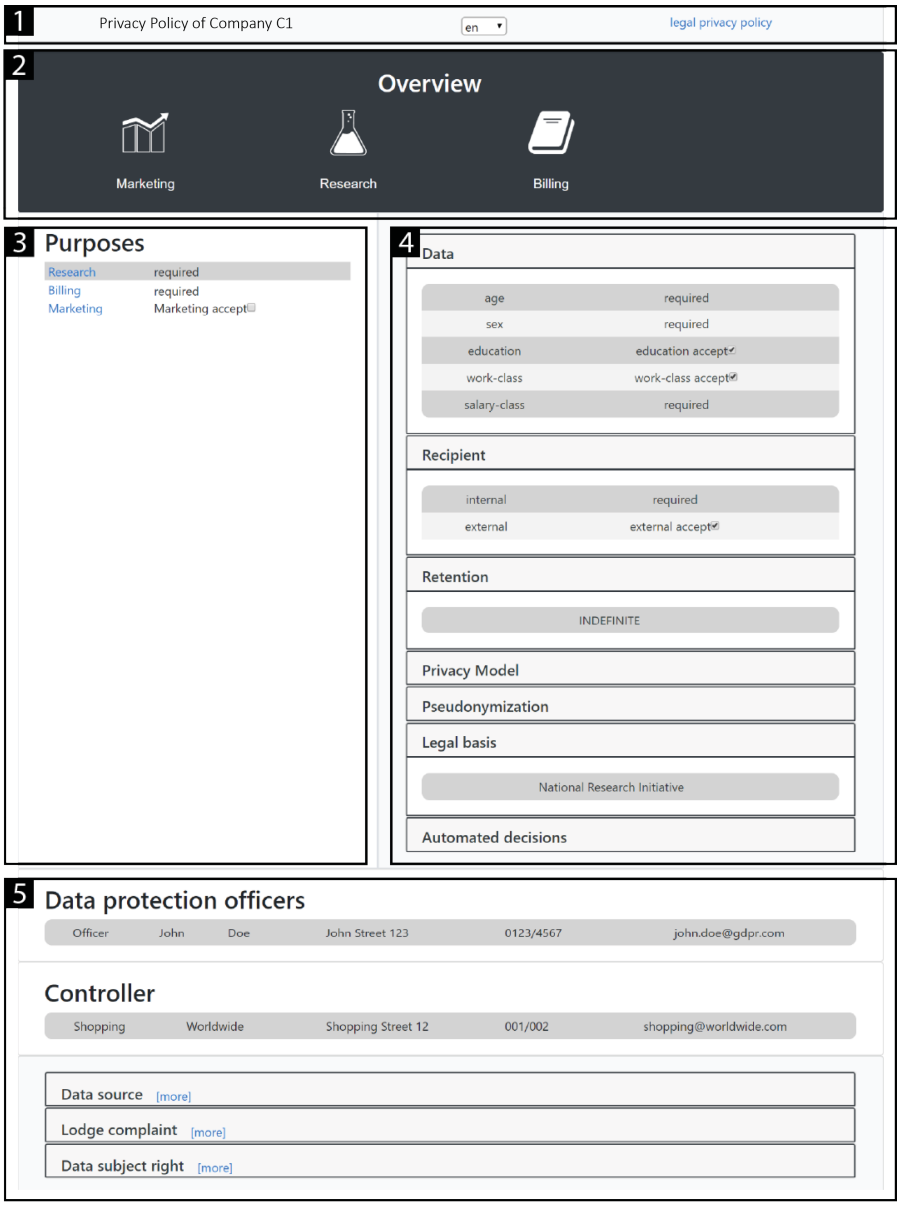


Figure 5.8: *Negotiation View* for a LPL privacy policy stating the processing of personal data for marketing, research, and billing. The layout elements based on the *Negotiation View* are highlighted as follows: 1: *Policy Header*, 2: *Privacy Icon Overview*, 3: *Purpose Overview*, 4: *Purpose Detail*, and 5: *Policy Information*.

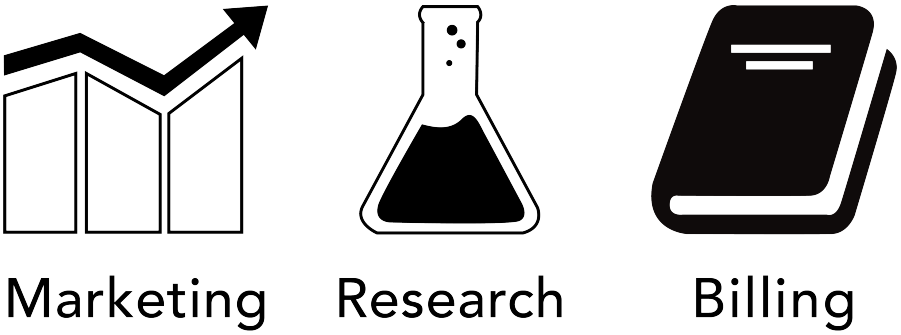


Figure 5.9: Place-holder *Privacy Icons* indicating the processing of personal data for marketing, research, and billing purposes.

5.6.1.2 *Privacy Icon Overview*

The *Privacy Icon Overview* is intended to follow the VISM *Overview* task giving an overview over the processing of the personal data, i.e. purposes. The intention is that users can identify the core purposes of the processing 'at a glance' via meaningful pictograms [112].

Currently, no *Privacy Icon* set has been standardised (see Section 2.3), therefore place-holder *Privacy Icons* are introduced (see Figure 5.9). Such *Privacy Icons* have to be evaluated, which is out of the scope of this thesis.

The *Privacy Icons* are globally defined for a LPL privacy policy by a set of *Icon*-elements. Unlike other concepts for *Privacy Icons* [213] [234] [189] [183], the intention is to only express the purpose of the processing of personal data. Therefore, the user shall be able to identify if the processing of his personal data is done for purposes that he is not willing to allow. If this is the case, the user can furthermore inspect the privacy policy beginning with the *Purpose Overview*.

5.6.1.3 *Purpose Overview*

The *Purpose Overview* lists all purposes of the privacy policy following the *Overview* concept of VISM. Hereby, the set of *Purpose*-elements `p` is utilized, from which the *Header*-element `head` is used to display the human-readable name of the purpose.

Furthermore, it is indicated if a purpose is required or optional for which the `required` attribute is utilized. An optional purpose is indicated by the addition of a check-box, which is pre-ticked depending on the `optOut` attribute.

Given the example in Figure 5.8, the purposes `'Research'` and `'Billing'` are required, and the purpose `'Marketing'` is optional and has to be consented to.

Thus, the *Purpose Overview* allows the user to *Zoom* on the contents of the purpose via a click on the corresponding purpose. Hereby, the contents of the purpose are detailed in *Purpose Details*.

5.6.1.4 *Purpose Details*

Within the *Purpose Details* section of the layout an overview over all details of the purpose is given. The user is hereby presented information on the processed data (*Data*-element `d`), the corresponding recipients for the data (*DataRecipient*-element `dr`), the retention period of personal data (*Retention*-element `r`), the privacy models (*PrivacyModel*-element `pm`) and pseudonymization method (*PseudonymizationMethod*-element `psm`), the legal basis of the processing (*LegalBasis*-element `lb`), and if automated decision-making is conducted (*AutomatedDecisionMaking*-element `adm`).

Similar to purposes, data and recipients may be required or optional, which is defined by the `required` attribute of the *Data*- or *DataRecipient*-element. The presentation follows the same principle as for the the purposes, where the term '`required`' indicates a required element and a check-box indicates an optional element that may be (de-)selected by the user.

By default, the contents of each of the information domains, e.g., data or recipients, is not visible to avoid to overwhelm the user with information. The user can reveal the information by interacting with the corresponding header. This corresponds to the *Details-on-demand* task of VISM.

5.6.1.5 *Policy Information*

Within the *Policy Information* section of the layout, general information on the policy is given. This includes the display of the responsible DPO (*DataProtectionOfficer*-element `dpo`) and *Controller* (*Controller*-element `c`). In both cases, contact details are detailed to allow the user to contact them. Furthermore, information on the data source (*DataSource*-element `ds`), how to lodge a complaint (*LodgeComplaint*-element `lc`), and *Data Subject Rights* (*DataSubjectRight*-element `dsr`) is given. This content is not visible by default, but can be accessed interacting with the corresponding header, thus the *Details-on-demand* task of VISM is represented again. Therefore, a user can access the information on how to lodge a complaint when it is necessary for him.

5.6.2 *Negotiation View*

The *Negotiation View* (see Figure 5.8) is intended to inform the user about the contents of the privacy policy in an easy and comprehensive way for a better transparency utilizing the previously introduced layout.

During the *Negotiation* phase of the life cycle, the user gets the privacy policy presented via the *Negotiation View* (see Figure 5.8). On the one hand, the user is informed about the contents of the privacy policy. On the other hand, the user can customize the privacy settings

defined by LPL. Note that the privacy policy can be negotiated during the first presentation of the privacy policy, or at any given time after the policy has been agreed on. Hereby, the intention is to empower the user to personalize his policy in detail instead of making only binary consent decisions on purposes. With LPL, the user can act on the *Purpose*-, *Data*-, and *DataRecipient*-element changing the `required` attribute which are the basis for personalization as detailed by Gerl et al. [117]. Furthermore, the *Minimum Anonymization Level* may be defined by the user for single attributes. The enabled personalization possibilities are discussed in the following sub-sections.

5.6.2.1 *Purpose Consent*

Consent management based on the purpose is commonly used in practice. For example, a user has to agree on the processing of his e-mail for receiving newsletters as part of marketing campaigns. This purpose requires explicit consent because it is not covered by any legal basis nor is part of a legitimate interest of a company, i.e. the business model.

In the *Negotiation View* example (see Figure 5.8), the purpose `'Marketing'` is optional and can be consented to by ticking the corresponding check-box. Hereby, it has to be considered that only opt-in consent is legal [110, Art. 7]. Opt-in means that the purpose has to be actively selected and therefore is not pre-selected. If a purpose is consented to, then the attribute `pointOfAcceptance` of the corresponding *Purpose*-element `p` is set with the current data and time such that the *Controller* can demonstrate when the consent has been given. This is required for accountability reasons, e.g., towards supervisory authorities.

Furthermore, the withdrawal of consent has to be as easy as to give the consent to a purpose [110, Art. 7]. Within the *Negotiation View* this is as simple as to remove the tick of a check-box. A purpose, which is not consented to, is omitted from the LPL privacy policy. However, it can be added again based upon the raw privacy policy. Thus, the *Negotiation View* uses the raw privacy policy, defining possible negotiations, and the personalized privacy policy.

5.6.2.2 *Data Negotiation*

Data Negotiation is denoted as a negotiation action that can be performed by the user. For optional *Data*-elements `d`, it is assumed that these are de-selected by default; in other words opt-in actions are assumed for data. The same assumption is made for optional *DataRecipient*-elements `dr`. A user is assumed to consent to the optional purpose `'Marketing'`. If data and data recipients require opt-in decision, then for each element an additional action is required. This is intended to raise awareness for the processed personal data of the

user. Thus, a user can restrict his consent on a fine-grained level for optional personal data. For example, the *Negotiation View* in Figure 5.8 enables the user to accept or reject the processing of 'education' and 'work − class' for the purpose 'Research'.

5.6.2.3 *Data Recipient Negotiation*

Similar to *Data Negotiation*, the user can also negotiate the recipients for his personal data. This can have various applications in different domains. For example in social media, the user may define that his personal data is shared with specific user groups, e.g., data is shared with 'family' but not with 'friends'. Considering the given example of the *Negotiation View* in Figure 5.8, the user cannot prevent the processing of his personal data by the service provider 'internal' which is required, but can decide if 'external' entities can access his personal data. Thus, the user has fine-grained control over the processing of his personal data by different recipients, while the *Controller* can state what purposes, data and data recipients are required.

5.6.2.4 *Minimum Anonymization Negotiation*

Next to binary decisions on purposes, data and recipients, LPL enables the user to directly influence the quality of the data, i.e. the anonymization level, that is processed for a specific purpose. For each *Data*-element d, an *AnonymizationMethod*-element am is specified which holds the *Minimum Anonymization Level* and the *Maximum Anonymization Level* via two *AnonymizationMethodAttribute*-elements ama. The *Maximum Anonymization Level* is set during the *Creation* phase of the LPL life cycle. It represents the required data quality for the processing of the data for the specific purpose. Thus, the requirements for the *Controller*, i.e. company, are defined. However, the *Minimum Anonymization Level* can be negotiated during the *Negotiation* phase of the LPL life cycle. The *Minimum Anonymization Level*, a numerical value, has to be lesser or equal to the *Maximum Anonymization Level*. Thus, the negotiation of the *Minimum Anonymization Level* can be omitted if the *Maximum Anonymization Level* is set to '0'. Otherwise, the user can adjust the anonymization level accordingly to match his personal privacy requirements.

For example, a user could agree to the processing of his personal data, i.e. postal-code '94032', for a research purpose. Moreover, assume the user does not want to state his exact postal-code in order to preserve his privacy. Thus, the user can adjust the *Minimum Anonymization Level* such that only the anonymized postal-code '9403∗' is used for research. As detailed before, this requires that the *Maximum Anonymization Level* is set to at least '1' in this case.

This scenario is enabled by defining the anonymization method 'Suppression' for the attribute postal-code with the *Maximum Ano-*

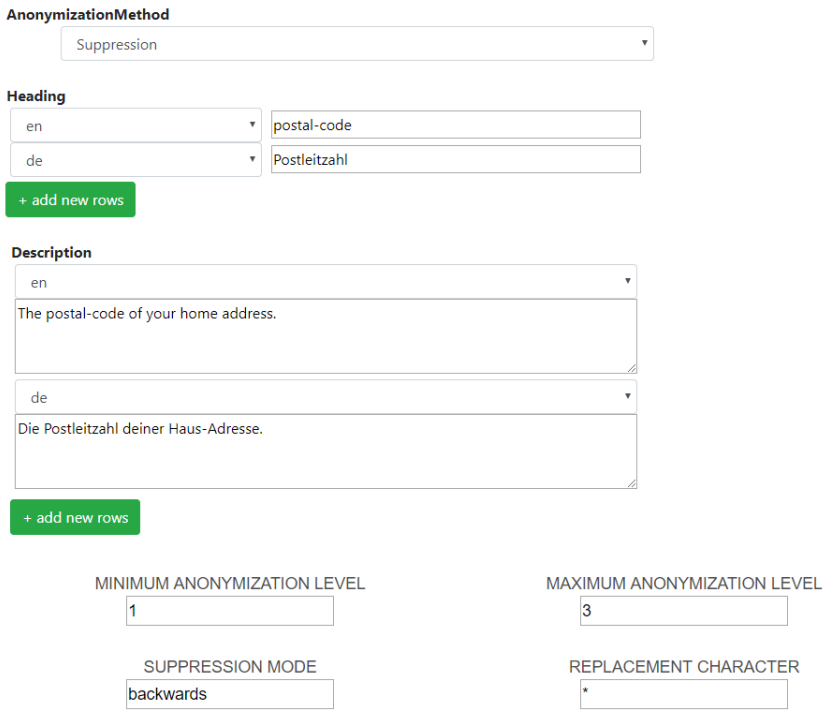


Figure 5.10: Anonymization method settings for the attribute `postal-code` using the*Creation View*.

nymization Level set to '3' and the *Minimum Anonymization Level* set to '1' by default (see Figure 5.10). The *Minimum Anonymization Level* could be set higher, but the *Privacy by Default* principle is interpreted such that data is protected, e.g. de-identified, by default [110, Art. 25 (2)]. Therefore, a company should define that data is used anonymized by default. Thus, the user may even reduce the *Minimum Anonymization Level*.

This allows the user to potentially specify the anonymization level of each attribute (EI, QI, SD, and NSD). Thus, fine-grained negotiation is enabled, including consent management based on purpose, data and recipients as well as the quality of the data via the *Minimum Anonymization Level*.

5.6.3 *Creation View*

The LPL privacy policy has to be created before the user is able to give consent to the processing of personal data or to negotiate his personal privacy requirements. Due to the vast amount of requirements given by the *GDPR* (see Section 2.2), the creation of a common privacy policy requires expert knowledge on both the legal requirements and the details of the corresponding service the policy is created for [116] [115].

The overall layout of the *Creation View* (see Figure 5.11) corresponds with the layout of the *Negotiation View*, but the intended usage dramatically varies. While the *Negotiation View* enables users to inform themselves about the privacy policy and negotiate it, the *Creation View* is intended for use by the *Controller* to realize raw LPL privacy policies lpp_{raw}.

The creation of raw LPL privacy policies lpp_{raw} can be conducted based on an existing legal privacy policy. This requires a matching of the legal text to LPL elements. Apparently, it can be realized from scratch. In this context, the *Creation View* offers functionality to create, update, or delete LPL elements.

In the *Policy Header* section, controls for uploading an existing LPL policy, resetting the whole policy or to add additional layers (*UnderlyingPrivacyPolicy* upp) are given. Thus, an existing LPL policy can be edited or a new policy can be created.

Within the *Privacy Icon Overview*, the user can add new *Privacy Icons* from a set of pre-defined icons. The *Purpose Overview* and *Purpose Details* sections enable the creation of various purposes for the policy. Furthermore, all required information for the privacy policy as well as the de-identification settings are detailed. Lastly, general information on the policy can be defined within the *Policy Information* section (see Figure 5.11).

5.6.4 *Discussion*

To demonstrate the fulfilment of the *R3 Human-readability* requirement, proof-of-concept user interfaces for the creation and negotiation of LPL privacy policies have been realized and detailed. Furthermore, the *R2 Legal Compliance* requirement has been considered, such that all required information is accessible via the user interfaces. In the following paragraphs, possible improvements and extensions for the presented user interfaces are discussed considering both requirements *R2* and *R3*. For the creation of the user interfaces VISM is used, whereas only *Overview*, *Zoom*, and *Details-on-demand* is considered. The remaining design principles have been excluded for the scope of the proof-of-concept prototypes, but possible implementation is discussed in the following. The *Filter* design principle could find application in the *Negotiation View* allowing the user to search and filter for specific groups of personal data, e.g., health data, that is of relevance for the user. *View Relationships* could visualise how data fields are used within automatic decision-making [110, Art. 13 No. 2]. A *History* of accepted privacy policies could be made available for the user to recall his decisions, which would also require the user to have the possibility to *Extract* the privacy policy contents. In the same way, a *History* of negotiations and consent given via the privacy policy

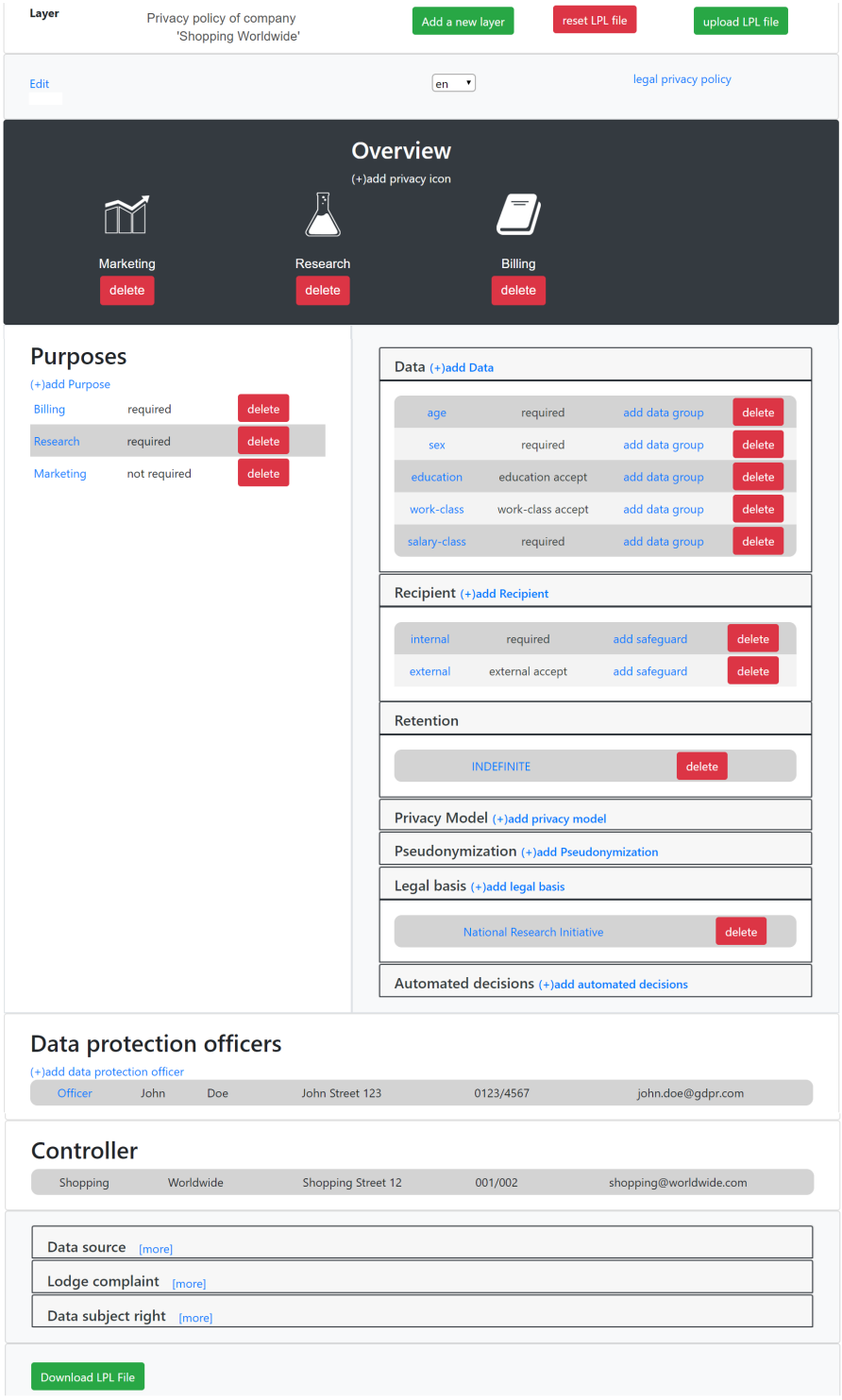


Figure 5.11: *Creation View* for a raw LPL privacy policy lpp_{raw} stating the processing of personal data for marketing, research, and billing.

could be visualized or even a *History* of the processing of personal data such that the user is completely and transparently informed.

The *Negotiation View* prioritises the presentation of the *Data*-element `d` over the *DataGroup*-elements `dg`, which can be considered problematic in the context of the *GDPR* requirements. In the *GDPR*, it is indeed stated that the user has to be informed about the collected data categories [110, Art. 14 (1)(d)]. Thus, this requirement is not directly fulfilled by the current version of the *Negotiation View*, but this could be changed in future versions.

During the creation of raw LPL privacy policies, it is a hard challenge to determine which privacy model, anonymization and pseudonymization is sufficient for a purpose, since it requires extensive expert knowledge. To facilitate the selection of appropriate de-identification methods a wizard or questionnaire may be introduced. Within the wizard, a questionnaire could be used to determine the general risk and attacks that the data-set has to be protected from and therefore suitable de-identification methods could be proposed [225].

Not only the definition of de-identification methods but also their presentation is an open challenge. Assuming *5-Anonymity* is defined for an optional purpose publishing the personal data, then the decision to consent to this purpose is strongly influenced on how the personal data is protected. From the point of view of the user, *5-Anonymity* is probably unknown or even if known, the user should understand what the guarantees of this privacy model are, which is not plausible. Thus, de-identification methods have to be presented in an understandable way for common users. One approach could be the presentation of the risk of identification via a percentage, e.g., 20% for *5-Anonymity*. Another approach would be to use coloured indicators (red for high risk, yellow for medium risk, and green for low to no risk).

Similarly, a suitable selection of the *Minimum Anonymization Level* is subject to future work and has to be evaluated. Considering the *Privacy by Default* principle [110, Art. 25 (2)] several pre-defined risk profiles may be proposed to the user.

Concerning the overall design of the user interface, different user groups have to be considered. The usage of a privacy language enables the alteration of the user interface for different contexts. Therefore, different user interfaces may be implemented for LPL targeting different user groups, e.g., children, elderly people, or people with disabilities.

5.7 PROVENANCE

In Section 2.6, the *R6 Provenance* requirement is detailed. The use cases *Data Processing* and *Data Production* are considered, which are discussed in the following for LPL.

Provenance can be covered by applying the sticky policy concept, which states that the policy should always be linked to the personal

data (see Section 4.1). Therefore, the origin of the personal data can be identified assuming the source of the personal data is specified within the considered privacy language.

Provenance of data access control is extensively covered in the literature by various works due to its importance for critical systems. For example, in the medical domain it is necessary that data records are properly archived to be audited [129]. Research efforts have been made for the management of provenance [76] [249] [131] [60] [123] and securing provenance [129] [130] [174] [199]. Provenance of data is denoted as the documentation of messages, operations, actors, preferences and the context that constitute the data [199].

However, *Provenance* can not only be considered for the origin of the data (source) but also for the privacy policy, such that it can be back-traced what the conditions for the processing of the personal data have been and how they have been refined. Therefore, accountability for companies can be monitored, i.e. it can be validated according which privacy policy data is transferred to a third party.

Within LPL, the *UnderlyingPrivacyPolicy*-element upp is introduced for *Privacy Policy Provenance*. Hereby, refined privacy policies incorporate the privacy policies that they are based upon as denoted by Gerl et al. [121]. This enables to distinguish between different privacy policies, their origin, and the validation of refined privacy policies against their previous policies. This is illustrated for *Data Processing* and *Data Production* use cases in the following paragraphs.

5.7.1 *Data Processing*

In the *Data Processing* use case, it is assumed that data and policies are transferred to third parties, e.g., for processing or trading. This scenario corresponds to the *Transfer* phase of the LPL life cycle (see Section 5.3.5).

Hereby, it is assumed that the user ds_{U1} agrees upon a policy with company e_{C1} to process his postal-code and salary for research purposes ('P_Research'). The policy between the user and the company e_{C1} is hereby denoted as $lpp_{ds_{U1}\text{-}\{dr_{C2},dr_{C2}\}}$ (see Equation 5.52).

Although, the policy $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$ allows a secondary company e_{C2} to process the data as well, company e_{C1} restricts the processing of the personal data. According to the agreement between e_{C1} and e_{C2}, company e_{C2} is only allowed to process postal-code for the purpose 'P_Research'. As this restriction is not expressed within the existing policy, a new LPL policy $lpp_{ds_{C1}\text{-}\{dr_{C2}\}}$ is defined. The new policy has to be validated to ensure that the personal data is processed accordingly to the previous agreed upon policy $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$. Without going into the details of the validation process, it has to be ensured that the new policy is at least as 'strict' as the previously agreed policy. For the given case, this is fulfilled because the purposes

match, the set of data recipients is a subset of the previous set, and the processed data attributes are also a subset of the data defined in $\mathrm{lpp}_{\mathrm{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}}$. The validation would fail if, e.g., the data is processed for a different purpose (taking into account the *Purpose-Hierarchy*), as data is processed which has not been specified for the purpose before, or de-identification methods are removed or weakened. Thus, the semantics of the policy have to be taken into account during the validation.

Iff the validation is successfully performed, the previously agreed upon LPL privacy policy $\mathrm{lpp}_{\mathrm{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}}$ is set as the *UnderlyingPrivacyPolicy*-element upp of $\mathrm{lpp}_{\mathrm{ds_{C1}\text{-}\{dr_{C2}\}}}$ (see Figure 5.12). Thus, a chain of policies is created which can be re-evaluated by the user, company or supervisory authorities to show compliance to the privacy regulations.

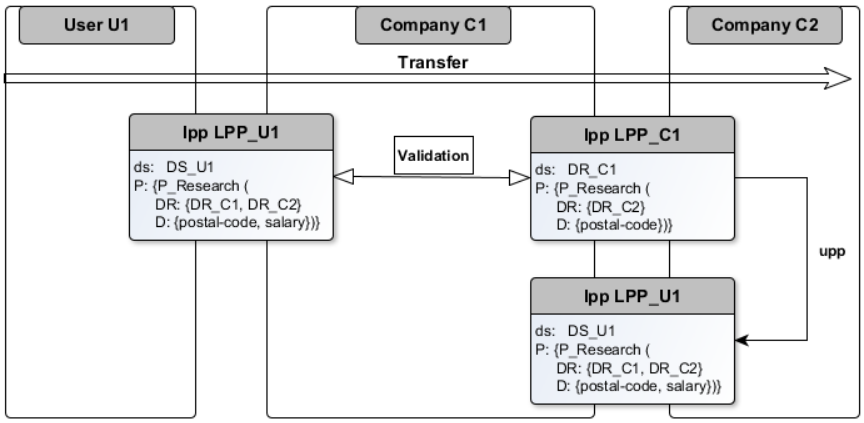


Figure 5.12: *Data Processing* use case based upon the *Transfer* phase scenario. Data is transferred from user $\mathrm{ds_{U1}}$ to company $\mathrm{c_{C1}}$ under the LPL privacy policy $\mathrm{lpp}_{\mathrm{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}}$. The data is furthermore transferred to $\mathrm{c_{C2}}$ under the policy $\mathrm{lpp}_{\mathrm{ds_{C1}\text{-}\{dr_{C2}\}}}$, which has to be validated against the previous policy.

The sticky policy concept in combination with *UnderlyingPrivacyPolicy*-elements upp allows furthermore to identify the source of personal data or a specific attribute (see Listing 5.4).

5.7.2 *Data Production*

The *Data Production* use case considers that personal data is processed to derive new data or information from it, e.g., in machine learning or data mining. The derived data is hereby based upon the personal data. It can be either classified again as personal data with a direct relation to a distinct user, or no longer classified as personal data, e.g. if the data is anonymized. Moreover, this derivative may be used for critical decisions, e.g., in health care, disaster prediction, or fraud detection. For critical use cases, it is crucial to be able to verify and backtrack the

```
method: determineSource
in: lpp; data //LPL privacy policy; data element to identify origin source of
out: e //source of data

    dataSource = null; //entity e

    //if data can be found in any purpose
    for p : lpp.P
        for d : p.D
            //match data according to name
            if match(data, d)
                dataSource = lpp.ds;

    //recursively iterate over all upp
    for upp : lpp.UPP
        if upp != null
            temp = determineSource(lpp.upp, data);

            //if dataSource is found
            if temp != null
                dataSource = temp;

    return dataSource;
```

Listing 5.4: Pseudocode to determine the origin source of a specific data attribute.

decision-making, e.g., to make transparent decisions, to verify results, or for fault checking.

The following *Data Production* scenario is based upon the scenario in Section 5.3. A company e_{C1} has collected personal data of several users ds_{U1}, ds_{U2}, and ds_{U3} with the corresponding privacy policies $lpp_{ds_{U1}\text{-}\{dr_{C1},dr_{C2}\}}$, $lpp_{ds_{U2}\text{-}\{dr_{C1},dr_{C2}\}}$, and $lpp_{ds_{U3}\text{-}\{dr_{C1},dr_{C2}\}}$ (see Figure 5.13). The LPL privacy policies only differ with respect to the different users, but not due to their description of the processing of the personal data. It is defined that the personal data can be processed for research, which is done by the company e_{C1} under consideration of each of the policies.

Thanks to the research process, the company can derive their most profitable $sales\ regions$ based upon the $postal-codes$ of the users. The company e_{C1} defines for this newly generated data a new policy stating that it can be only processed within the company for further research purposes. The new policy is denoted as $lpp_{ds_{C1}\text{-}\{dr_{C1}\}}$. The LPL privacy policies of the users are hereby set as *UnderlyingPrivacyPolicy*-elements upp to document the origin for the $sale\ regions$ process. The new LPL privacy policy is linked to the derived data to ensure a compliant handling of the personal data.

Thus, *Data Production* use cases can be covered by LPL using the *LayeredPrivacyPolicy*-element.

It is demonstrated how LPL can be used for the *Data Processing* and *Data Production* use cases to cover the *R6 Provenance* requirement.

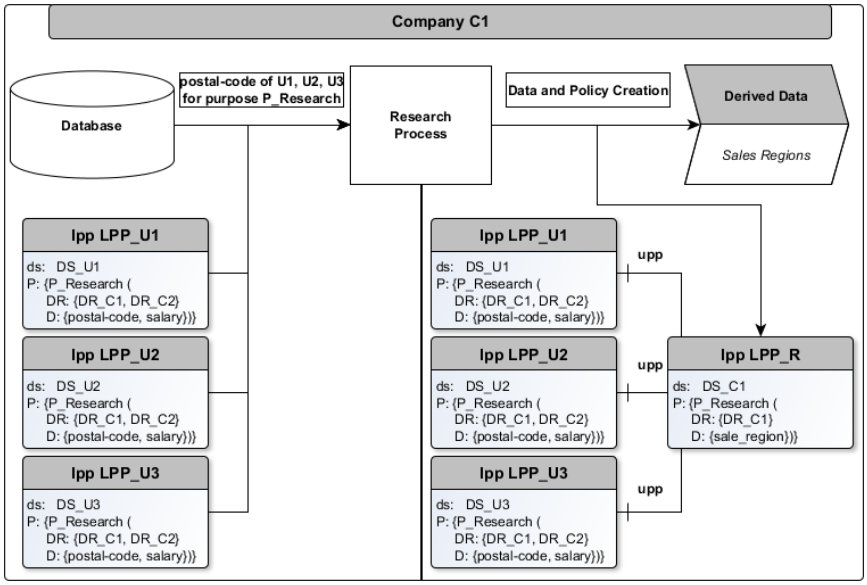


Figure 5.13: *Data Production* use case, demonstrating the production of new data based upon the personal data of several users. The derived data is associated with a new LPL privacy policy incorporating the distinct privacy policies of the users as underlying privacy policies for enforcing *Provenance* and credibility.

Thanks to the layering of LPL privacy policies, it is possible not only to backtrack the source of personal data, i.e. the user, but also the original agreement (or consent) to the processing of personal data. This enables to validate the authorization to process personal data and to review the personal data used to produce new data.

Thus, a chain or graph of policies is generated which can be leveraged to transparently show the flow of personal data and its corresponding policies.

Only the creation of new LPL privacy policies is discussed using layering for *Provenance* indeed, but more dynamic scenarios have to be considered. Users may indeed alter their personal privacy requirements within the policy, object the processing of their personal data, or delete their account and corresponding privacy policy. Companies may update privacy policies concerning their users, e.g., add new purposes or data to be processed, or alter the transfer conditions of personal data to third parties. Therefore, not only the creation but especially the update and the deletion of LPL privacy policies within the policy graph have to be considered.

Assume a scenario in which a user withdraws his consent to process personal data for research by a third party, not only the local LPL privacy policy has to be updated, but the changes have to be cascaded to the third party. Similarly, if a company adds another mandatory purpose to its privacy policy, this change has to be transparently presented to the user, e.g., via notifications. Furthermore, such a dis-

tributed scenario requires also a protocol to authenticate and authorize entities, purposes, and other LPL elements across companies to allow processing and validation of privacy policies. With LPL it is possible, because LPL contains all necessary information that is required to add additional functionality, e.g. notifications. These challenges are not covered within this thesis, but are subject to future work (see Section 8.2).

5.8 DISCUSSION

This chapter detailed the rationale behind the *Layered Privacy Language (LPL)* and gave a formalization of its elements. Based on the formalized LPL, the intended life-cycle has been detailed in Section 5.3. Furthermore, the fulfilment of requirements for a privacy language introduced in Chapter 2 is shown to answer the first research question *RQ1*: 'How to represent legal privacy policies in a machine-readable format which complies to the legal requirements of the General Data Protection Regulation in the EU while privacy guarantees are defined?'.

Therefore, the fulfilment of the *R1 Privacy Policy Structure* requirement is given by the core structure of LPL denoted by the *LayeredPrivacyPolicy*-element lpp, *DataSource*-element ds, *Purpose*-element p, *Data*-element d and *DataRecipient*-element dr. The fulfilment of the *R2 Legal Compliance* requirement is shown by an in detail comparison of the legal requirements and their fulfilment by LPL in Section 5.4. The fulfilment of the *R3 Human-readability* requirement is discussed according to proof-of-concept user interfaces for privacy policy creation, presentation and negotiation, whereas also *R2 Legal Compliance* has been considered in Section 5.6. For the fulfilment of the *R4 Access Control* requirement, the algorithms for PAC are detailed in Section 5.5). Lastly, it is shown how LPL privacy policies can be utilized for *Data Processing* and *Data Production* use cases due to the layering of LPL privacy policies to fulfil the *R6 Provenance* requirement.

Therefore, it has been shown that LPL can express all information required by the discussed legal regulations of the *GDPR*, while a structured and transparent visualization of LPL privacy policies is possible, i.e. due the usage of human-readable texts and *Privacy Icons*. Provenance is given in LPL due to the layering of privacy policies, while the well-known sticky policy concept is assumed. LPL privacy policies express access rules to personal data, which take into account the purpose, data recipient and data. Therefore, the fulfilment of requirements for a privacy language introduced in Chapter 2 is shown, except the fulfilment of the *R5 De-identification Capabilities* requirement, to answer the first research question *RQ1*.

The fulfilment of the *R5 De-identification Capabilities* requirement is shown in the next chapter detailing the *Policy-based De-identification (PD)* process to demonstrate how LPL can express privacy guarantees

to completely answer *RQ1*. Furthermore, the PD process is introduced to answer the second research question *RQ2*: 'How can machine-readable privacy policies, expressing privacy-preserving methods, be utilized to efficiently preserve the privacy of individuals when a set of users' personal data is requested for processing?'. The PD process is evaluated in Chapter 7 with respect to its efficiency to answer the second research question *RQ2*.

6

POLICY-BASED DE-IDENTIFICATION

This chapter details the *Policy-based De-identification (PD)* process, which demonstrates the realization of the *R5 De-identification Capabilities* requirement of Section 2.5. This process concerns the *AnonymizationMethod*-element $\mathtt{am}$, *PrivacyModel*-element $\mathtt{pm}$, and *PseudonymizationMethod*-element $\mathtt{psm}$ of LPL.

A data-warehouse scenario is assumed as basis for the PD process, because in a data-warehouse (personal) data is regularly collected, updated, and processed for various purposes by the company to gain added value. The data can be sourced from various origins and is integrated into a common scheme. Depending on the purpose, personal data has to be de-identified. If such a data-warehouse scenario with multidimensional data is considered, manifold challenges arise.

Personalized LPL privacy policies, i.e. LPL policies with varying privacy and utility configurations, have to be assumed for each record, which makes this process particularly complex. Thus, for the PD process, not only a large number of data, but also an equally high volume of possibly personalized LPL privacy policies has to be taken into account. Indeed, each LPL privacy policy instance defines the privacy requirements for a user. Hereby, it is required to efficiently determine and apply a uniform level of privacy, i.e. a consensus on the de-identification approach has to be reached, for the whole data-set. Furthermore, the privacy requirements of the users is contradicted by the utility needs of the company. The different methods for privacy preservation – *Pseudonymization*, *Personal Privacy Anonymization*, and *Privacy Models* – have to be carefully considered and combined to mitigate any unnecessary negative interference on either privacy or utility [121] [113].

In the following, the concept of *Policy-based De-identification (PD)* is based and illustrated on a data-warehouse scenario. Privacy and utility requirements are highlighted. Next, each essential process step is detailed for *Pseudonymization*, *Personal Privacy Anonymization*, and *Privacy Model*. Lastly, the PD process is discussed before it is evaluated in Chapter 7.

6.1 OVERVIEW

The *Layered Privacy Language (LPL)* is capable to express de-identification methods. These latter can be seen as privacy guarantees given by the company to the user regarding the processing of personal data for a specific purpose. Attribute-based personalization of the users' privacy requirements is considered in LPL through the adjustment of the *Minimum Anonymization Level* during the *Negotiation* phase, which has a direct impact on the processed data quality. The negotiation of consent for *Purpose*-, *Data*-, and *DataRecipient*-elements has only an indirect impact on the de-identification process, because it is considered during the *Policy-based Access Control* process, whereas whole records or single values of a record can be removed from the data-set.

The utility requirements of the company is considered through the *Maximum Anonymization Level*, which is set during the *Creation* phase of the LPL life cycle. Finally, next to personal privacy anonymization, also pseudonymization methods and privacy models can be used on the data-set to mitigate the risks of re-identification. Therefore, a high variety of privacy policies is possible, which in the following is illustrated by a data-warehouse scenario.

6.1.1 *Data-warehouse Scenario*

Consider a data-warehouse scenario in which the data-warehouse DW1 is regularly, e.g., daily, updated with the records of two different services S1 and S2 (see Figure 6.1). Although the services differ in their functionality, related privacy policies both define a common purpose 'Research'. For this purpose, data may be processed by the respective services, e.g., S1 or S2, and the data-warehouse DW1. The assumption is that both services collect and provide the same types of personal data: name, age, and postal-code.

Service S1 defines that the personal data is de-identified using pseudonymization on the name as well as *3-Anonymity* on the data-set. Service S2 may define different de-identification conditions to protect personal data, e.g., *2-Diversity* instead of *3-Anonymity*, or none. Therefore, the de-identification requirements for the same purpose, i.e. 'Research', may differ due to the various origins of the policy. Also the utility requirements can differ, which are defined by the *Maximum Anonymization Level* for each *Data*-element d.

The possibility of personal privacy policies introduces an additional variability of the privacy requirements by altering *Minimum Anonymization Levels* for each *Data*-element d. Thus, the PD process has to consider various privacy and utility requirements, which may change over time as a user or a company updates the policy. Considering the data-warehouse, a pre-processing of de-identified data-sets can

be done after each regular refresh, if known requests exist. Moreover, the PD process is considered, in general, to be executed on-the-fly and requests are also applicable for production databases that are continuously altered.

Fabian and Göthling [100] compared different strategies for the anonymization of personal data in a data-warehouse scenario. In general, the possibilities for the point of de-identification in a data-warehouse scenario can vary from de-identification the personal data at their source, i.e. service S1 and S2, and transfer it to the data-warehouse to a query-based anonymization, which de-identifies the personal data after it is requested. Fabian and Göthling [100] quantitatively evaluated several scenarios measuring the utility of the anonymized data-set. The evaluation did not include the query-based anonymization scenario, because no suitable implementation was available at the time. Nevertheless, the results indicate that an as-late-as-possible de-identification, i.e. anonymization, is overall advantageous for the utility, because the data-set can be anonymized according to the privacy and utility requirements of the specific purpose.

The PD process can be classified close to the query-based anonymization approach, because it determines and applies de-identification methods on-the-fly. The PD process differs from the query-based de-identification due to missing query capabilities, e.g., selection or filter. Although, basic query capabilities can be assumed in the *Policy-based Access Control (PAC)* processes, which are intended to select attributes of records from a selected set of data sources. Next, the interplay of PAC and PD is detailed as well as privacy and utility challenges that are considered for the PD process.

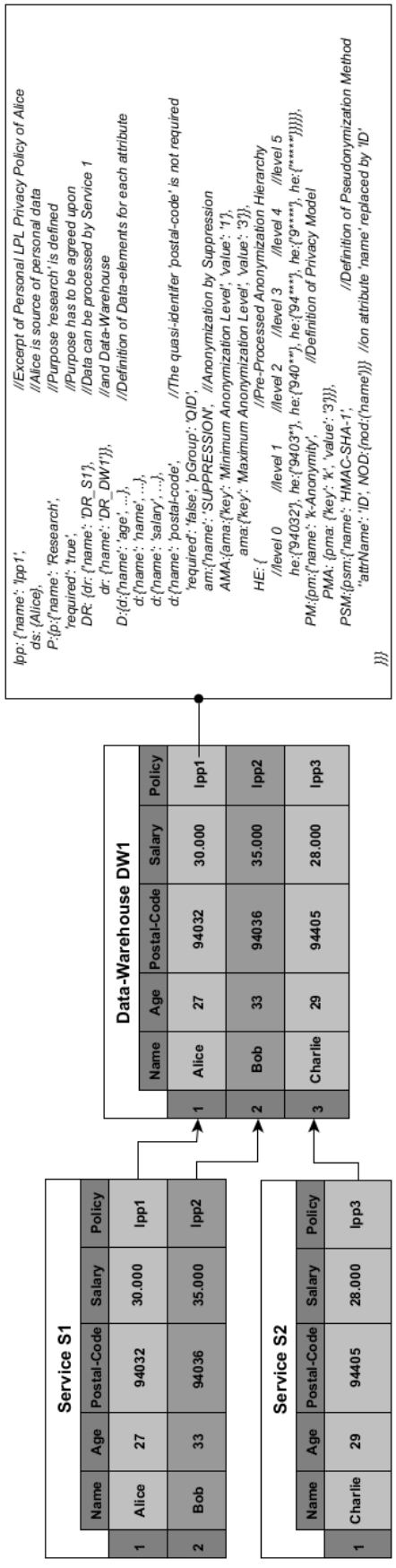


Figure 6.1: Data-warehouse scenario in which records with personal data from different sources are collected for further processing. Each record is linked to a (personalized) LPL privacy policy. A LPL privacy policy is detailed in excerpts.

6.1.2 *Process Sequence*

Because the PD process only focuses on the de-identification of a data-set, it has to be verified beforehand if the requesting entity is authorized to access the personal data. This is done by the PAC process (see Figure 6.3).

The PD process includes *Pseudonymization*, *Personal Privacy Anonymization*, and *Privacy Models* (see Chapter 3). *Personal Privacy Anonymization* considers the anonymization of individual attributes of a record according to the *Minimum Anonymization Level*.

In the PD process, privacy for a data-set is guaranteed due to a combination of all the individual de-identification methods, while utility is considered as an anonymization limit defined by the *Maximum Anonymization Level* for each attribute. The sequence of the de-identification methods is crucial to avoid any negative effect on both privacy and utility.

In Chapter 3, the classification of data into the privacy groups *Explicit Identifier* EI, *Quasi-Identifier* QI, *Sensitive Data* SD, and *Non-Sensitive Data* NSD is detailed. EI attributes identify a user uniquely. QI attributes identify a user in combination with other QI attributes. SD attributes do not identify a user, but are valuable information on the user. NSD attributes do not identify a user and do not have valuable information on the user. Furthermore, it is explored how the different de-identification methods affect the privacy groups in a data-set. To match the properties of *Privacy Models*, QI attributes are anonymized and EI are deleted (see Section 3.3). *Personal Privacy Anonymization* can be conducted on all attributes, but commonly only QI and SD attributes are anonymized (see Section 3.4). *Pseudonymization* commonly only affects EI attributes (see Section 3.5).

PSEUDONYMIZATION AND PRIVACY MODELS *Pseudonymization* and *Privacy Models* both affect EI attributes. *Privacy Models* commonly require the deletion of EI attributes from the data-set and *Pseudonymization* is usually applied on EI attributes to replace the original values with a pseudonym. Therefore, to preserve the best possible utility of the data-set, the *Pseudonymization* has to be conducted before *Privacy Models* are used. Otherwise, if *Privacy Models* would be used before *Pseudonymization*, the EI value would by deleted to comply with the privacy model, before it can be used to generate a pseudonym. Thus, *Pseudonymization* has to be used before *Privacy Models*.

Furthermore, the privacy group of the attribute, i.e., the EI attribute, has to be altered to NSD after *Pseudonymization*. The privacy group of the EI attribute has to be altered to NSD, because otherwise the values of the pseudonymized EI attribute would be deleted to comply with the *Privacy Model*. This alteration of the privacy group is valid, because pseudonyms are no longer sensitive or able to identify a user,

directly or indirectly, without authorized access to a bijective mapping table (see Section 3.4).

PSEUDONYMIZATION AND PERSONAL PRIVACY ANONYMIZATION
Pseudonymization affects EI attributes and *Personal Privacy Anonymization* affects commonly QI and SD attributes, but it is possible that also EI attributes can be affected. Assuming, *Personal Privacy Anonymization* is used before *Pseudonymization*, EI attributes would be first anonymized before they are pseudonymized. Thus, an anonymized value would be used for the pseudonym generation, which can negatively affect the utility.

Therefore, similar to the process sequence rationale of *Pseudonymization* and *Privacy Models*, *Pseudonymization* has to be applied before *Personal Privacy Anonymization*. In fact, the EI attribute values will be pseudonymized, such that any *Personal Privacy Anonymization* of the EI values is obsolete. Therefore, *Pseudonymization* has to be conducted before *Personal Privacy Anonymization*.

PERSONAL PRIVACY ANONYMIZATION AND PRIVACY MODELS
As previously detailed, *Privacy Models* require the anonymization of QI attributes and the deletion of EI attributes. *Personal Privacy Anonymization* is commonly applied on QI and SD attributes, but can also be applied on EI attributes. Thus, those processes have in common that EI and QI attributes are altered. In general, *Personal Privacy Anonymization* defines the minimal privacy requirements of a user for each attribute of the users' record and *Privacy Models* define properties of a data-set to mitigate privacy attacks, whereas anonymization is used to achieve such properties.

Considering EI attributes, if *Privacy Models* are used before *Personal Privacy Anonymization* the EI attributes would be deleted to achieve the privacy model and any personal privacy anonymization is obsolete, thus privacy is guaranteed. Otherwise, the EI attributes would be first anonymized during *Personal Privacy Anonymization* and then deleted to achieve the privacy model. Thus, the alteration of the EI attribute is not decisive for the process sequence of *Privacy Models* and *Personal Privacy Anonymization*.

Considering QI attributes, if *Privacy Models* are used before *Personal Privacy Anonymization*, the data-set would first be anonymized to match the requirements of the *Privacy Model* before single records are additionally anonymized to match the individuals' privacy requirements. Therefore, the properties of the data-set that match the privacy model can be invalidated. For example, assume *3-Anonymity* is applied on a data-set, QI-group with at least 3 records are created due to anonymization. Thus, the data-set is *3-anonymous* (see Table 6.1). Table 6.1 is *3-anonymous*, because each QI group has at least 3 records, e.g., the QI-group $('\text{Artist}','(38-50)','\text{F}')$ has exactly 3

records. Assuming that the attribute age of only one of the three records requires the *Minimum Anonymization Level* '2' instead of the current anonymization level '1' (see Figure 6.2), then one record has to be additionally anonymized. This creates the additional QI-group ('Artist','(25 – 50)','F') with 1 record (see Table 6.2). Furthermore, the QI-group ('Artist','(38 – 50)','F') has only 2 records, thus invalidating the properties of the *3-anonymous* data-set. Therefore, privacy is at risk because is no longer mitigated.

Table 6.1: 3-anonymous patient table based on Table 3.10. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 6.2, and 3.5.

QI			**SD**
Job	*Age*	*Sex*	*Disease*
Professional	(25 - 37)	M	HIV
Professional	(25 - 37)	M	HIV
Professional	(25 - 37)	M	Flu
Professional	(25 - 37)	M	Flu
Artist	(38 - 50)	F	Cancer
Artist	(38 - 50)	F	Cancer
Artist	(38 - 50)	F	Cancer

Table 6.2: Result patient table for the usage of *Personal Privacy Anonymization* on a 3-anonymous patient table (see Table 6.1. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 6.2, and 3.5.

QI			**SD**
Job	*Age*	*Sex*	*Disease*
Professional	(25 - 37)	M	HIV
Professional	(25 - 37)	M	HIV
Professional	(25 - 37)	M	Flu
Professional	(25 - 37)	M	Flu
Artist	(38 - 50)	F	Cancer
Artist	(38 - 50)	F	Cancer
Artist	(25 - 50)	F	Cancer

If the *Personal Privacy Anonymization* is conducted before *Privacy Models* are used, the QI attributes are first anonymized to meet the individuals' privacy requirements before the data-set is further anonymized to meet the criteria of the*Privacy Model*. For example, the same personal privacy requirement for the last record is assumed,

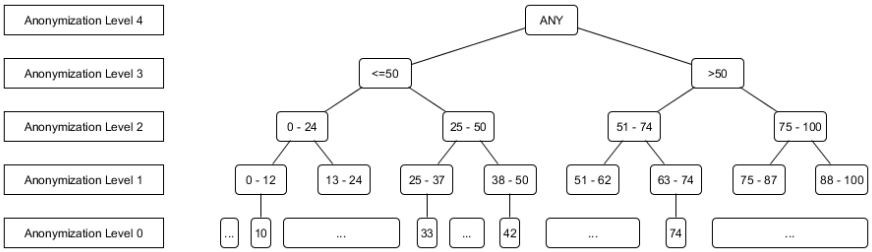


Figure 6.2: Anonymization hierarchy for the domain 'age'. The scope of original values at the anonymization level '0' has a broad range. Each subsequently higher anonymization level covers a sub-range of the original values. The generic value 'ANY' is used for the highest anonymization level.

i.e. the *Minimum Anonymization Level* for the attribute age is set to'2'. Therefore, the result of the *Personal Privacy Anonymization* is Table 6.3. This table is furthermore anonymized to meet the criteria of *3-Anonymity*, thus the QI attributes are anonymized such that each QI-group has at least 3 records. This results in the QI-group $('Artist','(38-50)','F')$ with 3 records and $('Professional','(25-37)','M')$ with 4 records (see Table 6.4). This de-identified data-set fulfils the privacy requirements of the individuals and the privacy model, but it has to be noted that the utility is reduced compared to the previous example, i.e. the anonymization level '2' is used for 4 records, instead of only 1 record (see Tables 6.2 and 6.4).

Therefore, the usage of the *Personal Privacy Anonymization* before *Privacy Models* is necessary to preserve the privacy, although the utility may be negatively affected by the personal privacy settings. Therefore, the individuals' privacy requirements are applied for each attribute of the record according to the *Minimum Anonymization Level*. Then, the data-set is further anonymized to meet the requirements of the *Privacy Model* step. Thus, privacy can be preserved considering both individuals' privacy requirements during the *Personal Privacy Anonymization* for each attribute and privacy requirements for the data-set using *Privacy Models*.

In summary (see Figure 6.3), *Pseudonymization* is applied first on EI attributes replacing them with pseudonyms and therefore changing their *privacy group* to NSD. Next, *Personal Privacy Anonymization* is applied on QI and SD attributes (doing it on EI and NSD is technically possible but uncommon). Lastly, *Privacy Models* are applied, altering QI attributes and deleting EI attributes resulting in a de-identified data-set T_{PD} based on the privacy requirements defined in LPL.

$$T_{PD} = (QI', SD', NSD') \tag{6.1}$$

Table 6.3: Result patient table for the usage of *Personal Privacy Anonymization*. The value of the attribute age is anonymized to meet the individuals privacy requirements. Anonymization is conducted with the anonymization hierarchy shown in Figure 6.2.

QI			**SD**
Job	*Age*	*Sex*	*Disease*
Engineer	35	M	HIV
Engineer	33	M	HIV
Lawyer	29	M	Flu
Lawyer	33	M	Flu
Writer	42	F	Cancer
Singer	45	F	Cancer
Writer	(25 - 50)	F	Cancer

Table 6.4: Result *3-anonymous* patient table for the usage of *Personal Privacy Anonymization* before *Privacy Models*. Anonymization is conducted with the anonymization hierarchies shown in Figures 3.3, 6.2, and 3.5.

QI			**SD**
Job	*Age*	*Sex*	*Disease*
Professional	(25 - 37)	M	HIV
Professional	(25 - 37)	M	HIV
Professional	(25 - 37)	M	Flu
Professional	(25 - 37)	M	Flu
Artist	(25 - 50)	F	Cancer
Artist	(25 - 50)	F	Cancer
Artist	(25 - 50)	F	Cancer

6.1.3 *Data Structure – DataWrapper*

The *DataWrapper*-object dw acts as a container for the data of each record as well as the corresponding authorized purpose which is derived after the individuals' LPL privacy policy successfully runs through the PAC process. A set of *DataWrapper*-elements DW represents therefore a data-set, that is processed during PD. A *DataWrapper*-object consists of the following attributes:

$$dw = (\mathtt{dataList}, p) \tag{6.2}$$

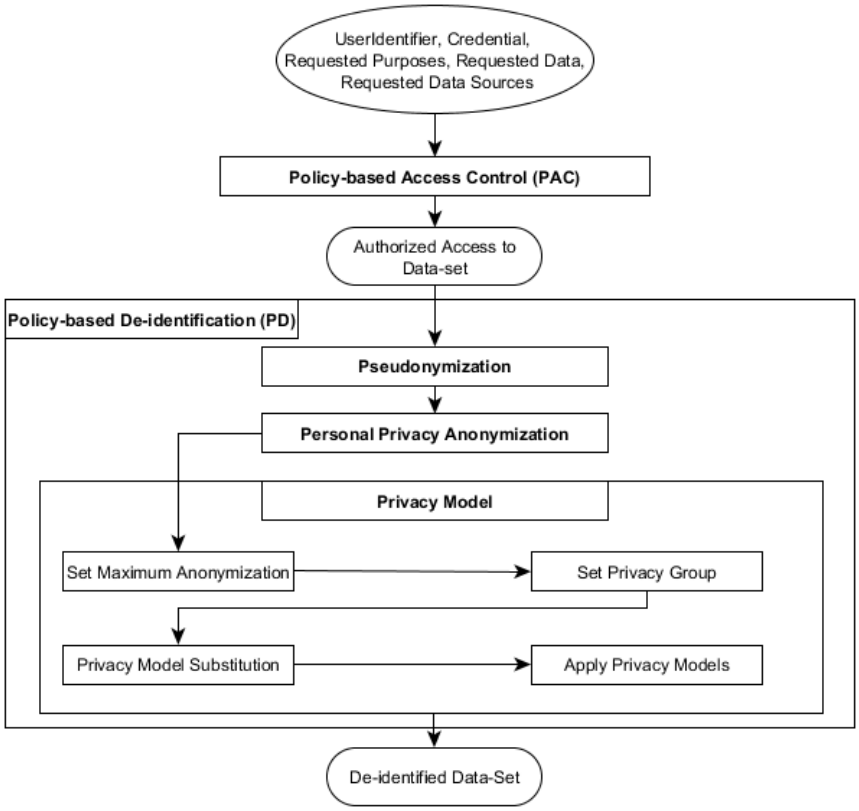


Figure 6.3: *Policy-based De-identification (PD)* process step sequence after the *Policy-based Access Control (PAC)* has been completed for a request.

- $\mathtt{dataList}$: A hashmap with the attribute name as $\mathtt{key}$ and the attribute value as the $\mathtt{value}$ representing the data values for a record (see Equation 6.3).
- $\mathtt{p}$: After PAC has been conducted successfully, an authorized *Purpose*-element $\mathtt{p}$ is derived for the data record and added to the *DataWrapper*. It contains all sub-elements required for PD including the set of *Data*-elements $\widehat{\mathtt{D}}$ with the corresponding *AnonymizationMethod*-element $\mathtt{am}$, the set of *Pseudonymization-Method*-elements $\widehat{\mathtt{PSM}}$, and the set of *PrivacyModel*-elements $\widehat{\mathtt{PM}}$, e.g. the purpose $\mathtt{p}_{\mathtt{Research}}$ for Alice in Figure 6.1).

$$\begin{aligned}\mathtt{dw}_{\mathtt{Alice}}.\mathtt{dataList} = \{&(\text{'name'}, \text{'Alice'}), (\text{'age'}, \text{'27'}),\\ &(\text{'postal-code'}, \text{'94032'}), (\text{'salary'}, \text{'30.000'})\}\end{aligned} \tag{6.3}$$

In the following, the individual algorithms necessary for the PD process are detailed. Hereby, the focus lies on the efficient determination of a uniform level of privacy and utility requirements that are applied on the data-set based on various different LPL privacy policies. The uniform level incorporates, e.g., the anonymization level for single values, privacy model configuration, or *Maximum Anonymization Level* for attributes.

6.2 PSEUDONYMIZATION

After the PAC process a multidimensional data-set is assumed. For each record of the data-set, an authorized *Purpose* p is assumed detailing the privacy requirements. The first step of the PD process is the pseudonymization of EI attributes. The *PseudonymizationMethod*-element psm defines the pseudonymization method that is to be applied on the data-set. Furthermore, additional options are defined via *PseudonymizationMethodAttribute*-elements psma [113].

As introduced in Section 3.5, various options for the generation of the pseudonym exist including hashing- and encryption-based pseudonym generators, dependant and independent pseudonym generation, or the usage of bijective mapping. LPL is not intended to store or provide any secret used for the generation of the pseudonym. Furthermore, a secure and separate storage for bijective mapping tables is required to avoid unauthorized re-identification [110, Recital 29].

Within the data-warehouse scenario, as many individuals and data sources are involved, not only one but several policies have to be considered with possible varying specifications for pseudonymization Therefore, different pseudonymization methods may be defined on the same attribute. This challenge is addressed by determining and applying all distinct pseudonymization methods on the data-set. This has the advantage that a data-set can be generated for a specific purpose with all pseudonym attributes, but the pseudonyms can only be re-identified by the authorized entities. The disadvantage of this approach is that some pseudonyms may be generated that are unnecessary. This does not affect the resulting privacy or utility of the resulting de-identified data-set, but only the run-time of the PD process. This is an edge case, because it can be usually assumed that purposes that define *PseudonymizationMethod*-elements are use-case specific. Thus, the probability of different pseudonymization method definitions for the same attribute and purpose is low, but still this challenge is addressed within the PD process.

Thus, the *Pseudonymization* is done by inspecting each authorized purpose p of the *DataWrapper* to determine a set of distinct *PseudonymizationMethod* psm (see Listing 6.1). The set of distinct pseudonymization methods is then used on the data-set generating pseudonyms. The pseudonyms are added to the dataList of each *DataWrapper* dw.

Considering the efficiency of this algorithm, the determination of the set of distinct pseudonymization methods that has to be applied on the data-set requires the iteration of all *DataWrapper* dw, i.e. each *Purpose* p (see Listing 6.1). Furthermore, each *PseudonymizationMethod* psm has to be iterated. Therefore, this algorithm is linearly dependent on the data-set size, i.e. the number of *DataWrappers* dw each having one *Purpose* p, and the number of *PseudonymizationMethod*-elements

defined. Furthermore, the generation of the pseudonyms is highly dependent on the defined pseudonymization methods (see Section 3.5).

```
1
 method: determineDistinctPseudonymizationMethods
 in: DW //DataWrapper list containing all records with corresponding purpose
 out: PSM //set of distinct pseudonymization methods to be applied

6    distinctPSM; //initialize empty PSM list

     //iterate all DataWrapper
     for (dw : DW)
         for (psm : dw.p.PSM) //access PSM of authorized purpose
11
         if(!distinctPSM.contains(psm)) //if psm not in distinct PSM list
             distinctPSM.add(psm); //add psm

     return distinctPSM;
```

Listing 6.1: Pseudocode of the determination of pseudonymization methods that have to be applied on the data-set.

Considering the previously introduced data-warehouse scenario (see Figure 6.1), assume an employee of the data-warehouse 'DR_DW1' to requesting all available data ($name$, age, $postal\text{-}code$, and $salary$) from all data sources ('Alice', 'Bob', 'Charlie') for the purpose 'Research'. Furthermore, assume that the PAC process successfully authorizes the employee to access the requested data-set (see Table 6.5), which has to be de-identified according to the corresponding LPL privacy policies.

Table 6.5: Requested data-set for the data-warehouse scenario depicted in Figure 6.1 which is represented by DW_{req}.

EI	**QI**		**SD**	**LPL**
Name	*Age*	*Postal-Code*	*Salary*	*Policy*
Alice	27	94032	30.000	lpp1
Bob	33	94036	35.000	lpp2
Charlie	29	94005	28.000	lpp3

According to Figure 6.1, Alice's record has a sticky LPL privacy policy that defines for the purpose 'Research' that the attribute $name$ is replaced by a pseudonym generated via *HMAC-SHA-1*. The $attrName$ for the pseudonym is hereby defined as 'ID'. Furthermore, it is assumed that the usage of a bijective mapping table is defined via $psma_{bijective}$. The LPL privacy policies of Bob and Charlie define the same pseudonymization method.

$$\begin{aligned} psm_{Alice} = (&\text{'HMAC-SHA-1'}, \text{'ID'}, \{\text{'name'}\}, \\ &\widehat{HEAD}, \widehat{DESC}, \{psma_{bijective}\}) \end{aligned} \tag{6.4}$$

$$psma_{bijective} = (\text{'bijective\_mapping'}, \text{'true'}) \tag{6.5}$$

Therefore, the data-set encapsulated in DW_{req} is pseudonymized to DW_{psm} replacing the attribute $name$ with ID holding the pseudonyms (see Table 6.6). Furthermore, the pseudonyms and original values are stored in a separate bijective mapping table (see Table 6.7). Thus, the *Pseudonymization* process is completed and the *Personal Privacy Anonymization* is applied on the data-set next.

Table 6.6: Pseudonymous data-set for the data-warehouse scenario depicted in Figure 6.1 which is represented by DW_{psm}.

NSD	**QI**		**SD**	**LPL**
ID	*Age*	*Postal-Code*	*Salary*	*Policy*
80085	27	94032	30.000	lpp1
81183	33	94036	35.000	lpp2
40440	29	94005	28.000	lpp3

Table 6.7: Bijective mapping table for the pseudonymized data-set of the data-warehouse scenario.

Mapping Store Name	
Original Value	*Pseudonym*
Alice	80085
Bob	81183
Charlie	40440

6.3 PERSONAL PRIVACY ANONYMIZATION

To achieve each individuals' privacy requirement, i.e. the *Minimum Anonymization Level* for each value, the *Personal Privacy Anonymization* process is run. This process essentially considers for each attribute of the record the corresponding *Data*-element d and its privacy definition contained by the *AnonymizationMethod*-element am. The *Minimum Anonymization Level* is defined as an *AnonymizationMethodAttribute*-element ama. It defines which *HierarchyEntry*-element he replaces the original value.

The *Minimum Anonymization Level* can be altered by the user during the *Negotiation* phase up to the *Maximum Anonymization Level* (see Section 5.3.2). In the *Personal Privacy Anonymization* process, the selection of the anonymization, i.e. the *HierarchyEntry*-element he representing the anonymized value, is key. Therefore, two alternative algorithms are proposed – *Minimum Anonymization (AM)* and *Global Minimum Anonymization (GMA)* – that both enforce individuals' privacy constraints,

but have a potentially different impact on the run-time performance of PD and the utility of the de-identified data-set.

Considering the previously introduced data-warehouse scenario, let's assume that only for the attribute `postal-code` the *Minimum Anonymization Level* has been altered from the default value '0'. Therefore, the *Minimum Anonymization Level* for the `postal-code` is set to '1' by Alice (see Figure 6.1), to '2' by Bob, and to '1' by Charlie. Based on this scenario, the *Minimum Anonymization (MA)* and *Global Minimum Anonymization (MA)* algorithms are detailed in the following.

6.3.1 *Minimum Anonymization*

The *Minimum Anonymization (MA)*-algorithm directly realizes the individuals' privacy requirements by replacing each original value with the anonymized value specified by the *Minimum Anonymization Level*. Therefore, the *MA*-algorithm iterates over all *DataWrapper*-objects. For each *DataWrapper*-object `dw`, the *Minimum Anonymization Level* is identified for each attribute. The corresponding *HierarchyEntry*-element is selected accordingly and used to replace the original value in the `dataList` of `dw` (see Listing 6.2).

Considering the efficiency of this algorithm, the *Personal Privacy Anonymization* of the data-set requires the iteration of all *DataWrapper* `dw`, i.e. each *Purpose* `p`, for each attribute (see Listing 6.2). Therefore, this algorithm is linearly dependent on the data-set size, i.e. the number of *DataWrappers* `dw`, and the number of attributes.

```
method: minimumAnonymization
in: DW //DataWrapper list containing all records with corresponding purpose
out: DW //updated DW with anonymized d according to Minimum Anonymization
     Level

//determine miminum-anonymization level from ama for each d
for (dw : DW)
  for (data : dw.dataList)
    //determine Minimum Anonymization Level
    minLevel = getMinimumLevel(dw.p.D̂(data.key).am.ÂMÂ);

    //overwrite value with value of hierarchy at Minimum Anonymization Level
    data.value = dw.p.D̂(data.key).am.ĤÊ(minLevel);
    //update anonymization hierarchy
    //first element of ĤÊ) has to match data.value
    updateHE(data.value, dw.p.D̂(data.key).am.ĤÊ);

return DW;
```

Listing 6.2: Pseudocode of the *Minimum Anonymization*-algorithm.

Considering the previously introduced scenario, the original value of each record is replaced by exactly the value of *HierarchyEntry*-element `he` specified by the *Minimum Anonymization Level*.

For the attribute postal-code of Alice the *Minimum Anonymization Level* of '1' is specified. Furthermore, the set of *HierarchyEntry*-elements is defined as $\widehat{\mathbf{HE}}_{\mathbf{Alice}}$.

$$\widehat{\mathrm{HE}}_{Alice} = \{('94032'), ('9403*'), ('940**'), ('94***'), ('9****'), ('*****')\} \tag{6.6}$$

The *MA*-algorithm determines that the value '9403∗' corresponds to the *Minimum Anonymization Level* and replaces the original value with it. The anonymization hierarchy $\widehat{\mathbf{HE}}_{\mathbf{Alice}}$ is updated, because privacy models technically require that 1) an anonymization hierarchy with the first value equalling the processed value such that the original value can be assigned to an anonymization hierarchy and 2) all anonymization hierarchies have to have the same size. The same size is required to avoid that an anonymization level is determined for the attribute which can not be provided for some values, because the anonymization hierarchy size is too small. Therefore, all he elements with a lower *Anonymization Level* than the *Minimum Anonymization Level* are replaced with the determined value. Thus, $\widehat{\mathbf{HE}}_{\mathbf{Alice}}$ is replaced with $\widehat{\mathbf{HE}'}_{\mathbf{Alice}}$ (see Equation 6.7). Note, that the hierarchy size remains such that the *Maximum Anonymization Level* references the same he as before.

$$\widehat{\mathrm{HE}'}_{Alice} = \{('9403*'), ('9403*'), ('940**'), ('94***'), ('9****'), ('*****')\} \tag{6.7}$$

Similarly, the value '940 ∗ ∗' is determined for Bob and the value '9440∗' is determined for Charlie (see Figure 6.4).

The *MA*-algorithm follows exactly the specification of the individuals' privacy requirement. The downside of the *MA*-algorithm is that it can introduce various additional distinct values to the de-identified data-set, i.e. anonymized values. If no privacy model is specified, the data recipients receive possibly a data-set that consists of a subset of values that are anonymized and a subset of original values. If a privacy model is used, the increased number of distinct values and anonymization hierarchies to be considered can negatively affect the run-time (see Chapter 7).

Therefore, an alternative algorithm is proposed – *Global Minimum Anonymization* – below, which is evaluated against this algorithm in Section 7.2.3.

6.3.2 *Global Minimum Anonymization*

The *Global Minimum Anonymization*-algorithm does not consider the individuals' privacy requirement for each attribute directly, but derives a global unified privacy requirement for each attribute. This global privacy requirement is derived as the maximum of all *Minimum*

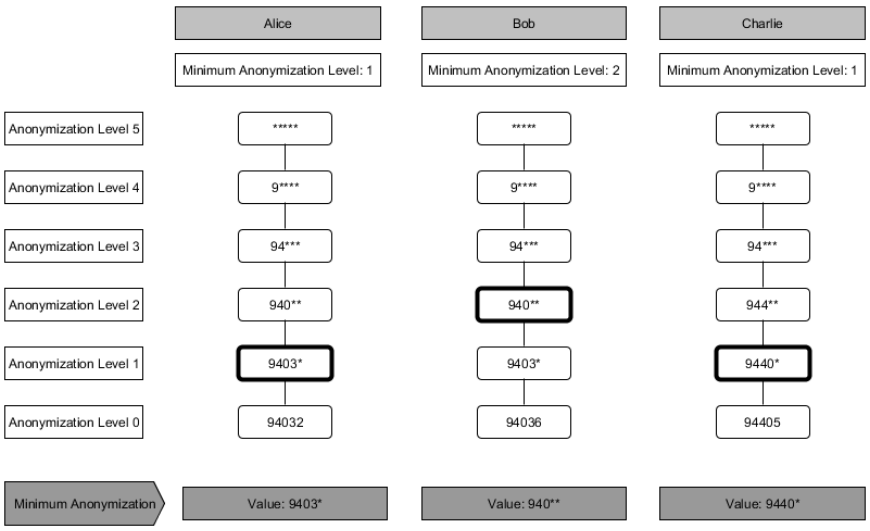


Figure 6.4: *Minimum Anonymization* example for the attribute postal-code.

Anonymization Levels. The maximum of all *Minimum Anonymization Levels* guarantees that all individuals' *Minimum Anonymization Levels* are met or exceeded.

Therefore, the *GMA*-algorithm requires two iterations over all *DataWrapper*-objects DW. In the first iteration, the maximum of the *Minimum Anonymization Levels* is determined and temporarily stored. In the second iteration, each *DataWrapper*-object dw is iterated again, and the maximum of the *Minimum Anonymization Levels*, determined in the first iteration, is used to replace the original value in the dataList of dw (see Listing 6.3).

Considering the efficiency of this algorithm, the *Personal Privacy Anonymization* of the data-set requires two iterations of all *DataWrappers* dw (see Listing 6.3). Although this algorithm requires two iterations, it remains linearly dependent on the data-set size and the number of attributes. But, an increased run-time compared to the *Minimum Anonymization*-algorithm (see Section 6.3.1) is expected.

Consider the previously introduced scenario again now with the *GMA*-algorithm. The maximum of the *Minimum Anonymization Levels* is determined for the attribute postal-code. The *Maximum Anonymization Level* for Alice is '1', '2' for Bob, and '2' for Charlie. In the second iteration, the original values are replaced with the respective *HierarchyEntry*-elements at the *Anonymization Level* '2', which corresponds to '940 $**$' for Alice, '940 $**$' for Bob, and '944 $**$' for Charlie (see Figure 6.5).

Furthermore, each $\widehat{\mathbf{HE}}$ has to be adapted. In the case of Alice, $\widehat{\mathbf{HE}}_{\mathbf{Alice}}$ is shortened to $\widehat{\mathbf{HE}''}_{\mathbf{Alice}}$ (see Equation 6.8).

$$\begin{aligned}\widehat{HE''}_{Alice} = \{&('940^{**}'),\\ &('94^{***}'), ('9^{****}'), ('^{*****}')\}\end{aligned} \tag{6.8}$$

```
method: globalMinimumAnonymization
in: DW //DataWrapper list containing all records with corresponding purpose
out: DW //updated DW with anonymized d according to global Minimum
     Anonymization Level

maxLevel; //map storing the global maximum of the Minimum Anonymization Level
      for each attribute

//determine miminum-anonymization level from ama for each d
for (dw : DW)
  for (data : dw.dataList)
    //determine Minimum Anonymization Level
    minLevel = getMinimumLevel(dw.p.D̂(data.key).am.ÂMÂ);

    //determine global maximum Minimum Anonymization Level for each attribute
    if (maxLevel(data.key) < minLevel)
      maxLevel.put(data.key, minLevel);

for (dw : DW)
  for (data : dw.dataList)
    //overwrite value with value of hierarchy at the global maximum of the
         Minimum Anonymization Level
    data.value = dw.p.D̂(data.key).am.ĤÊ(maxLevel(data.key)));
    //update anonymization hierarchy
    updateHE(data.value, dw.p.D̂(data.key).am.ĤÊ);

return DW;
```

Listing 6.3: Pseudocode of the *Global Minimum Anonymization*-algorithm.

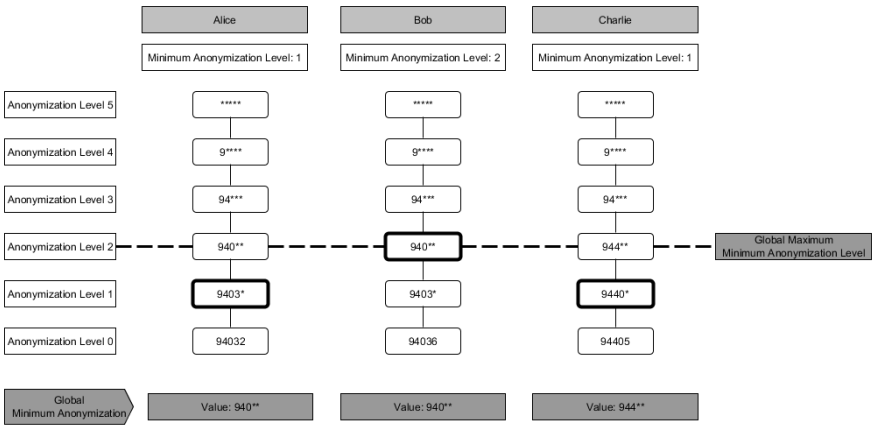


Figure 6.5: *Global Minimum Anonymization* example for the attribute postal-code.

Because all records are anonymized equally, the $\widehat{\mathbf{HE}}$ can be shortened for all affected records equally. Thus, all anonymization hierarchies have the same size for each attribute. Note, the *Maximum Anonymization Level* may be adapted in this case to match the same value in the updated $\widehat{\mathbf{HE}}$ than the original $\widehat{\mathbf{HE}}$.

The *GMA*-algorithm fulfils the personal privacy requirements of each user, but may also exceed them for records in order to have a

uniform *Anonymization Level*. Such excessive anonymization, generally speaking, has a negative impact on the utility of the de-identified data-set, because valuable information is redacted that would be otherwise available. Because the personal privacy approach should always be used in combination with a privacy model, it can be argued that the various anonymized values from the *Personal Privacy Anonymization* would probably be anonymized anyway to meet the privacy models requirements, e.g., to create QI-groups with k records in case of *k-Anonymity*. Therefore, the possible negative impact of the *GMA* on the utility may be neglect-able in the end.

Considering the data-warehouse scenario, the usage of the *MA*-algorithm on the already pseudonymized data-set results in Table 6.8, while the usage of the *GMA*-algorithm results in Table 6.9. It can be clearly observed that in Table 6.9 the values for the `postal-code` are more anonymized for the records of Alice and Charlie compared to Table 6.8, thus more utility is lost.

Table 6.8: Data-warehouse scenario data-set pseudonymized and anonymized using the *Minimum Anonymization*-algorithm.

NSD	**QI**		**SD**	**LPL**
ID	*Age*	*Postal-Code*	*Salary*	*Policy*
80085	27	9403*	30.000	lpp1
81183	33	940**	35.000	lpp2
40440	29	9440*	28.000	lpp3

Table 6.9: Data-warehouse scenario data-set pseudonymized and anonymized using the *Global Minimum Anonymization*-algorithm.

NSD	**QI**		**SD**	**LPL**
ID	*Age*	*Postal-Code*	*Salary*	*Policy*
80085	27	940**	30.000	lpp1
81183	33	940**	35.000	lpp2
40440	29	944**	28.000	lpp3

Consider furthermore that the data-set has to comply to the properties of *3-Anonymity* in the next step of PD, then both data-sets have to be additionally anonymized which results in the same de-identified data-set (see Table 6.10). Thus, use cases exist in which utility is not negatively affected by the *Global Minimum Anonymization*-algorithm compared to the *Minimum Anonymization*-algorithm.

Table 6.10: Data-warehouse scenario data-set which fulfils the properties of *3-Anonymity*. The data-set is pseudonymized and anonymized using either the *MA*- or *GMA*-algorithm for *Personal Privacy Anonymization*.

NSD	**QI**		**SD**	**LPL**
ID	*Age*	*Postal-Code*	*Salary*	*Policy*
80085	25 - 37	94***	30.000	lpp1
81183	25 - 37	94***	35.000	lpp2
40440	25 - 37	94***	28.000	lpp3

Considering edge cases in which the overall distribution of records with personal privacy or outliers in the *Minimum Anonymization Level* definitions are present, the negative impact on the utility of the de-identified data-set can be significant. For example, if only one record out of a million records in the data-set sets the *Minimum Anonymization Level* higher than every other record, then all other records will be set to the same *Anonymization Level* and an excessive anonymization of the attribute will be performed on all records.

The advantage of the *GMA*-algorithm is that the number of distinct values as well as the size of the corresponding anonymization hierarchies is potentially decreased. This can have a positive impact on the run-time performance of the privacy models, which is essential for the usage of the PD process on-the-fly (see Chapter 7).

In summary, the *Global Minimum Anonymization*-algorithm preserves the individuals' privacy, may have a negative impact on the utility depending on the use case, but it may reduce the overall number of distinct values and the size of anonymization hierarchies. If the *GMA*-algorithm is evaluated against the *MA*-algorithm according to the run-time performance and discussed in Chapter 7. Next, the application of the privacy model on the data-set is discussed for the PD process.

6.4 PRIVACY MODEL

After *Pseudonymization* and *Personal Privacy Anonymization* have been carried out, the data-set is further anonymized to meet the conditions of privacy models in order to mitigate privacy attacks.

Consider the data-warehouse scenario, various definitions of privacy models have to be considered. To overcome the trivial solution to use every distinct privacy model sequentially on the data-set, which would induce a significant run-time overhead, the *Privacy Model Substitution* process is proposed. This process is intended to determine a minimal set of privacy models that has to be applied on the data-set in order to meet the privacy requirements of all LPL privacy policies.

Furthermore, the *Maximum Anonymization Level* is used to limit the anonymization of attributes. Therefore, the *Maximum Anonymization Level* can be seen as the utility requirement, which is set by the corresponding policy creator, e.g., company, to limit the anonymization of attributes.

All process steps, which are based on LPL privacy policies, required to preserve the privacy with privacy models and to preserve the utility requirements, are detailed in the following.

6.4.1 *Set Maximum Anonymization*

During the *Creation* phase of the LPL life-cycle, the creator of the policy, e.g., the company, sets the *Maximum Anonymization Level* for each attribute for each purpose. The *Maximum Anonymization Level* defines the limit for the anonymization within the PD process and limits the *Minimum Anonymization Level* during the *Negotiation* phase. The *Maximum Anonymization Level* is specified by an *AnonymizationMethodAttribute*-element ama for each *AnonymizationMethod*- am and respective *Data*-element d.

Consider the QI attributes age and postal-code. The preservation of the postal-code is assumed to be more valuable for a research purpose. Therefore, the policy creator can define a *Maximum Anonymization Level* that takes into account this utility preference.

In the data-warehouse scenario, different LPL privacy policies are considered. Therefore, the *Maximum Anonymization Level* can differ for the same attribute due to different LPL privacy policies. To determine the potentially best utility requirement for the attribute, the minimum of the *Maximum Anonymization Levels* is determined for each attribute in the *Set Maximum Anonymization* process and set as the limit for anonymization (see Listing 6.4).

Considering the efficiency of this algorithm, the determination of the maximum of the *Minimum Anonymization Levels* requires the iteration of all *DataWrappers* dw (see Listing 6.4). Furthermore, each attribute has to be iterated. Therefore, this algorithm is linearly dependent on the data-set size and the number of attributes.

For example, in the data-warehouse scenario, the *Maximum Anonymization Level* for the attribute postal-code is set to '3'. Because the policy of Bob originates also from service S2, it is assumed that Bobs' policy is based upon the same raw LPL privacy policy and therefore also has the *Maximum Anonymization Level* '3', but for Charlie the *Maximum Anonymization Level* '4' is assumed (see Figure 6.6). Thus, the minimum *Maximum Anonymization Level* is '3' and therefore is set for all records.

In this way, the utility requirements are integrated within the PD process. Next, the privacy groups are defined for each attribute.

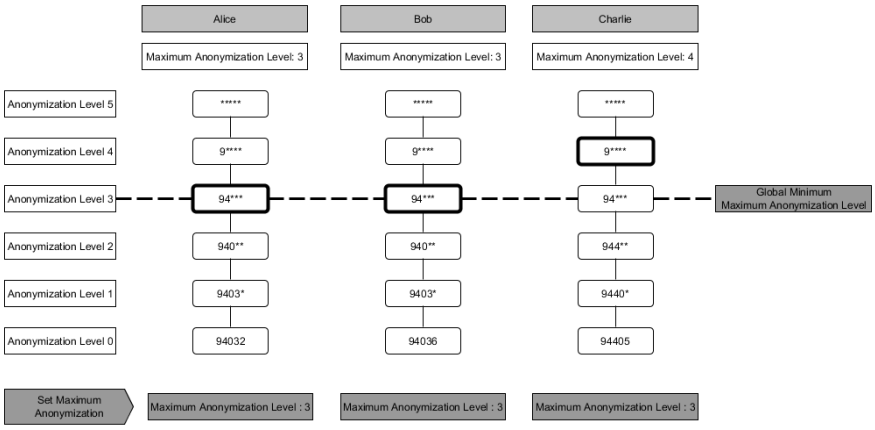


Figure 6.6: Example for *Set Maximum Anonymization* process for the attribute postal-code.

```
method: setMaximumAnonymization
in: DW //DataWrapper list containing all records with corresponding purpose
out: void //nothing is returned

minLevel; //map storing the global minimum of the Maximum Anonymization Level
      for each attribute

//determine Maximum Anonymization Level from ama for each d
for (dw : DW)
  for (data : dw.dataList)
    //determine Maximum Anonymization Level
    maxLevel = getMaximumLevel(dw.p.D̂(data.key));

    //storage of global minimum Maximum Anonymization Level for each
         attribute
    if (minLevel(data.key) > maxLevel)
      minLevel.put(data.key, maxLevel);

// set the minimum of Maximum Anonymization Level as the anonymization limit
setAnonymizationLimit(minLevel);
```

Listing 6.4: Pseudocode for the *Set Maximum Anonymization*.

6.4.2 *Set Privacy Group*

To use privacy models on the data-set, it is required that each attribute is assigned a privacy group, which is subject to the *Set Privacy Group* process. For each *Data*-element d, the attribute pGroup defines that the corresponding attribute is either an EI, QI, SD or NSD attribute. The classification of an attribute in privacy group is not trivial and may be done based on the context of the purpose of the processing. For example, the attribute room temperature could be classified as NSD because it is not related to any individual. But, if the increase or decrease of the room temperature is used to detect the presence of persons in a room it might be classified as a QI attribute, because it

could be used to identify the location of individuals if it is combined with other attributes, e.g., individuals' working period. Thus, the classification of attributes is strongly use case dependent.

In the data-warehouse scenario, different privacy policies are assumed, therefore it is possible that the attributes are assigned different privacy groups by different policies. To use the privacy model, a unique privacy group for each attribute has to be found in case of heterogeneous settings in different privacy policies. For attributes, privacy groups can be seen as privacy risk and utility indicators, e.g., EI attributes have the highest risk for identifying an individual and SD have no risk to identify an individual but are valuable information. Thus, they can be assigned an order (see Equation 6.9).

$$\mathrm{EI} > \mathrm{QI} > \mathrm{SD} > \mathrm{NSD} \tag{6.9}$$

The strictest privacy group is *Explicit Identifier* followed by *Quasi-Identifier*, *Sensitive Attribute*, and *Non-Sensitive Attribute*. Because only one privacy group can be assigned to one attribute, the algorithm determines the strictest privacy group for each attribute utilizing the order (see Listing 6.5).

To determine the strictest privacy group for the whole data-set, a list of all attributes and privacy groups (*pGroupList*), defining the currently strictest privacy group for each attribute, is stored. For each *Data*-element d of the *DataWrapper*, the privacy group is compared against the corresponding privacy group of the attribute from *pGroupList*. If the value of the privacy group is stricter than the entry in the *pGroupList*, the privacy group of the attribute is updated in *pGroupList*. Thus, after all *DataWrapper*-objects are iterated, the strictest privacy group is determined for each attribute and can be added to the configuration of the privacy model.

Considering the efficiency of this algorithm, the determination of the privacy group for each attribute requires the iteration of all *DataWrappers* dw (see Listing 6.5). Furthermore, each attribute has to be iterated. Therefore, this algorithm is linearly dependent on the data-set size and the number of attributes.

For example, assume the policy of Alice defines the attribute postal-code as a QI and the policy of Charlie defines it as a SD. According to the order of Equation 6.9 QI is stricter then SD, therefore QI is set for the attribute postal-code.

6.4.3 *Privacy Model Substitution*

In the previous process steps, all information required to use privacy models on the data-set is prepared. What is missing is the privacy model (or set of privacy models). Following the previous argumentation regarding the data-warehouse scenario, it is possible that varying

```
1
  method: setPrivacyGroup
  input: DW //DataWrapper list containing all records with corresponding
       purpose
  output: pGroupList //array with the privacy group for each attribute

6 pGroupList; //pGroupList array

  //for each record
  for (dw : DW)
    int i = 0; //attribute counter
11  //determine the strictest privacy group for each d
    for (d : dw.p.D̂)
      if (d.pGroup < pGroupList[i])
        pGroupList[i] = d.pGroup;
      i++;
16
  return pGroupList;
```

Listing 6.5: Pseudocode for the *Set Privacy Group*.

privacy models are defined. Thus, the strictest set of privacy models has to be derived from all corresponding LPL privacy policies.

The selection of a privacy model for a use case is challenging by itself, because it requires expert knowledge. Psaraftis et al. [225] proposed a question-based recommendation system to tackle this challenge. The *Customized Recommendation System for Optimum Privacy Model Adoption (CRPM)* determines a suitable privacy model based upon a decision-tree using 15 distinct questions. These questions consider the properties of privacy models and the processed data-set to recommend one suitable privacy model. But the questions themselves require expert knowledge in the privacy domain, e.g., the user of CRPM has to know about QI-groups and privacy groups. Furthermore, the CRPM approach only considers the recommendation of a privacy model, but it does not recommend the configuration of the privacy model, e.g. the value of k for *k-Anonymity*. For these reasons, the CRPM approach cannot be directly used for this problem statement. However, the properties of privacy models can be considered to determine the strictest set of privacy models from various LPL privacy policies.

Prasser and Kohlmayer [223] [159] introduce in the ARX framework another approach. In the ARX framework, the configuration of the de-identification process is optimized considering the properties and configuration of privacy models. For example, if *k-Anonymity* and *l-Diversity* are configured, both defining the size of QI-groups, the maximum value of k and l is derived and used for the configuration. Thus, privacy models are not individually used on the data-set, but the properties required for the data-set are derived from all privacy models and applied. Therefore, performance is optimized. Furthermore, ARX verifies that no conflicting or duplicate privacy models are

defined [221]. Therefore, ARX prevents misconfiguration of privacy models, optimizes the de-identification process due to consideration of overlapping privacy model properties, and considers the configuration, i.e. parameters, of privacy models.

Based on both approaches, the *Privacy Model Substitution* is proposed to identify the strictest set of privacy models. The term 'strictest' concerns the privacy properties, i.e. the risk of re-identification and which privacy attacks are mitigated by the privacy model. For this purpose, the classification of privacy models according to the mitigated attack models in Table 6.11 is used.

Table 6.11: Reduced list (see Table 3.15) of privacy models and their classification according to the attack models they mitigate.

Privacy Model	*Record Linkage*	*Attribute Linkage*	*Table Linkage*	*Probabilistic Attack*
k-Anonymity	x			
Recursive (c, l)-Diversity	x	x		
t-Closeness		x		x

Considering the comparison of two privacy models, following substitution scenarios are possible (see Figure 6.7).

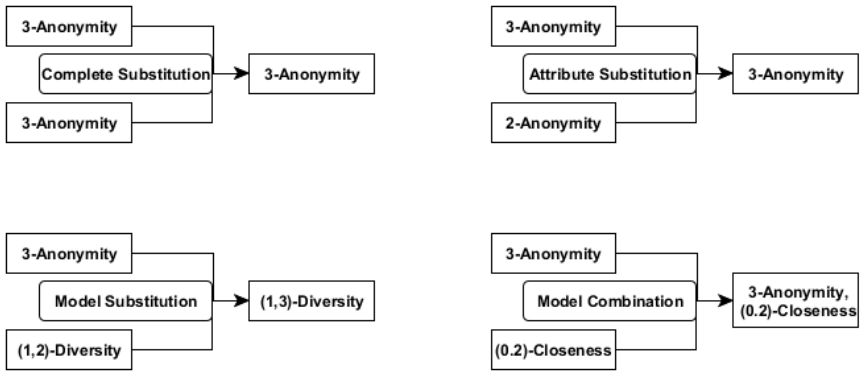


Figure 6.7: Examples for *Privacy Model Substitution* showing *Complete Substitution*, *Attribute Substitution*, *Model Substitution*, and *Model Combination*.

- *Complete Substitution:* If two privacy models are equal including the same configuration, then one privacy model is completely substituted by the other. For example, if the first privacy model is *3-Anonymity* and the second privacy model is *3-Anonymity*, then they are substituted, i.e. reduced, to *3-Anonymity*.
- *Attribute Substitution:* If two privacy models are equal but have different configurations, then the result is the privacy model with

the strictest configuration, i.e. lower the risk for re-identification. For example, if the first privacy model is *3-Anonymity* and the second privacy model is *2-Anonymity*, then the same privacy model is used with the strictest configuration, i.e. $k = 3$, such that the set of privacy models is substituted to *3-Anonymity*.

- *Model Substitution:* If the two privacy models are different, then the result consists of one of the given privacy models, while the configuration depends on both given privacy models. For example, if the first privacy model is *3-Anonymity* and the second privacy model is *(1,2)-Diversity*, then the mitigated privacy attacks are considered for the selection of the substitute privacy model. Thus, the attack model *Record Linkage* is mitigated by both *k-Anonymity*, and *(c,l)-Diversity*. But, *(c,l)-Diversity* also mitigates *Attribute Linkage* (see Table 3.15) [108]. Thus, *(c,l)-Diversity* is used as substitute privacy model. Next, the configuration for the substitute privacy model is derived. For this purpose the configurations of the initial privacy models are considered. The parameters l and k denote how many records are required for each QI-group, the greater the value for k or l the lower the re-identification risk [177]. Therefore, the maximum of both l and k is used to determine the substitute configuration. Furthermore, the c of the given privacy model has no comparable variable in *k-Anonymity*, thus it is used as is for the substitute configuration. Thus, *(1,3)-Diversity* is the substitute privacy model.

- *Model Combination:* If the privacy models cannot be substituted by another single privacy model, e.g., because there is no privacy model covering their properties, the process result is the original set of privacy models. For example, if the first privacy model is *3-Anonymity* and the second privacy model is *(0.2)-Closeness*, then a privacy model is required which mitigates *Record Linkage*, *Attribute Linkage*, and *Probabilistic Attacks*. Considering the overview of mitigated attacks by privacy models in the *Privacy Model Substitution Table* (see Table 6.11), no privacy model can be found fulfilling all these requirements. Thus, both privacy models have to be defined.

It can be observed that not only the attack models are used for creating the substitution rules, e.g., for *Model Substitution*, but also expert knowledge on the impact of each privacy model configuration and their interplay.

Instead of crafting a decision-tree like Psaraftis et al. [225], I propose to formalize comprehensive rules that can be extended. A *Privacy Model Substitution Table* is introduced holding such rules. Each rule takes a pair of privacy models as an input and defines a substitute privacy model and rules for deriving the substitute privacy model configuration using basic operations (see Table 6.12). Rules are created

for each possible privacy model pair. Considering only *k-Anonymity*, *(c,l)-Diversity*, and *t-Closeness*, this results in six rules. For each privacy model, one rule for matching the same privacy model, e.g., $\{k_{i1}\text{-Anonymity}, k_{i2}\text{-Anonymity}\}$, results in the same substitute privacy model, e.g., $\{k_{r1}\text{-Anonymity}\}$, for which the handling of the configuration is detailed, e.g., using the maximum of the *k* parameters $\{(k_{r1}, max(k_{i1}, k_{i2}))\}$.

Pairing distinct privacy models requires expert knowledge of each privacy model. Because the combination of *k-Anonymity* with *l-Diversity* and *k-Anonymity* with *t-Closeness* was already detailed, let's focus on the remaining pair – *l-Diversity* and *t-Closeness* ($\{(c_{i1}, l_{i1})\text{-Diversity}, t_{i2}\text{-Closeness}\}$). Inspecting the mitigated privacy attacks for each privacy model, i.e. *Record Linkage* and *Attribute Linkage* for *l-Diversity*, and *Attribute Linkage* and *Probabilistic Attacks* for *t-Closeness*, no substitute privacy model can be found that covers the given attack models. Therefore, the *Model Combination* case applies with the set ($\{(c_{r1}, l_{r1})\text{-Diversity}\}$, $\{t_{r2}\text{-Closeness}\}$) with the same configurations ($\{(c_{r1}, c_{i1}), (l_{r1}, l_{i1}), (t_{r2}, t_{i2})\}$). Adding a new privacy model to the *Privacy Model Substitution Table* requires to pair it with each other privacy model in the table. Therefore, the size of the *Privacy Model Substitution Table* for n privacy models is $\frac{n(n+1)}{2}$, which requires more effort and expert knowledge for each additional privacy model.

With the *Privacy Model Substitution Table* accessible, each record, represented by its corresponding *DataWrapper* dw, is iterated sequentially, thus substituting the privacy models of the record (see Listing 6.6). For each iteration, the *strictestPrivacyModelList* is derived through substitution, which is used as the basis for the next iteration. When DW is completely iterated, the *strictestPrivacyModelList* contains the strictest privacy models that cover each privacy requirement given by every records' LPL privacy policy. Therefore, the privacy guarantees of several privacy models can be applied on the data-set with the *Policy-based De-identification* process. Assuming the previous example with *l-Diversity* and *t-Closeness*, the mitigation of *Record Linkage*, *Attribute Linkage Probabilistic Attacks* will be given. Note that although *Attribute Linkage* is mitigated by both privacy models, strategies of both *l-Diversity* and *t-Closeness* will be used to protect the data-set from the privacy attack.

Considering the efficiency of this algorithm, the determination of the set of privacy models used on the data-set requires the iteration of all *DataWrappers* dw (see Listing 6.6). Furthermore, each *PrivacyModel*-element pm has to be iterated and substituted. The substitution process itself requires the iteration of the *Privacy Model Substitution Table* to first determine and then apply the appropriate substitution rule. Therefore, this algorithm is linearly dependent on the data-set size and

the number of *PrivacyModel*-elements. Additionally, the substitution process has to be taken into account.

This allows the reduction of privacy models, which have a relatively high computational cost, used on the data-set, while the minimal required privacy settings (considering the whole data-set) are applied. Therefore, privacy is preserved and the overall performance is potentially increased.

Table 6.12: Example *Privacy Model Substitution Table* for the privacy models *k-Anonymity*, *(c,l)-Diversity* and *t-Closeness*.

Privacy Model Set	**Substitution Privacy Model**	**Substitution Privacy Model Attribute**
$\{k_{i1}\text{-Anonymity}, k_{i2}\text{-Anonymity}\}$	$\{k_{r1}\text{-Anonymity}\}$	$\{(k_{r1}, max(k_{i1}, k_{i2}))\}$
$\{(c_{i1}, l_{i1})\text{-Diversity}, (c_{i2}, l_{i2})\text{-Diversity}\}$	$\{(c_{r1}, l_{r1})\text{-Diversity}\}$	$\{(c_{r1}, min(c_{i1}, c_{i2})), (l_{r1}, max(l_{i1}, l_{i2}))\}$
$\{t_{i1}\text{-Closeness}, t_{i2}\text{-Closeness}\}$	$\{t_{r1}\text{-Closeness}\}$	$\{(t_{r1}, min(t_{i1}, t_{i2}))\}$
$\{k_{i1}\text{-Anonymity}, (c_{i2}, l_{i2})\text{-Diversity}\}$	$\{(c_{r1}, l_{r1})\text{-Diversity}\}$	$\{(c_{r1}, c_{i2}), (l_{r1}, max(k_{i1}, l_{i2}))\}$
$\{k_{i1}\text{-Anonymity}, t_{i2}\text{-Closeness}\}$	$\{k_{r1}\text{-Anonymity}\}, \{t_{r2}\text{-Closeness}\}$	$\{(k_{r1}, k_{i1}), (t_{r2}, t_{i2})\}$
$\{(c_{i1}, l_{i1})\text{-Diversity}, t_{i2}\text{-Closeness}\}$	$\{(c_{r1}, l_{r1})\text{-Diversity}\}, \{t_{r2}\text{-Closeness}\}$	$\{(c_{r1}, c_{i1}), (l_{r1}, l_{i1}), (t_{r2}, t_{i2})\}$

```
method: privacyModelSubstitution //run for each dw of DW
input: privacyModelList //PM^ from a p of dw merged with previous
      strictestPrivacyModelList
output: strictestPrivacyModelList //substituted PM

strictestPrivacyModelList;
iterator = privacyModelList.iterator();

// add initial privacy model
strictestPrivacyModelList.add(iterator.next());

//for each privacy model currentPM
while (iterator.hasNext()) {
  currentPM = iterator.next();
  tempPMList = strictestPrivacyModelList;

  //substitute currentPm with each pm of strictestPrivacyModelList
  for (pm : strictestPrivacyModelList)
    //substitution according to Privacy Model Substitution Table
    substitutedPM = substitutePM(pm, currentPM);
    // replace pm with substituted pm
    temporaryPrivacyModelList.replace(pm, substitutedPrivacyModels);
  //updated strictestPrivacyModelList
  strictestPrivacyModelList = temporaryPrivacyModelList;

return strictestPrivacyModelList;
```

Listing 6.6: Pseudocode for the *Privacy Model Substitution*.

Consider the data-warehouse scenario. The policy of Alice denotes that *3-Anonymity* has to be applied, which is also assumed for Bob as his record originates from the same service. Because Charlies' record originates from services S2, a different privacy model – *2-Anonymity* – is assumed. Sequentially processing the records of Alice, Bob, and Charlie, the first pair of privacy models that is compared is $\{3\text{-Anonymity}, 3\text{-Anonymity}\}$ (see Figure 6.8). According to the *Privacy Model Substitution Table* (see Table 6.12), this results in $\{3\text{-Anonymity}\}$ matching the *Complete Substitution* case. This substituted privacy model is then paired with the privacy model defined by Charlie: $\{3\text{-Anonymity}, 2\text{-Anonymity}\}$. This matches the *Attribute Substitution* case for which the same rule of the *Privacy Model Substitution Table* is applicable. The rule denotes that *k-Anonymity* is the substitute privacy model with the maximum $k(\{(k_{r1}, \max(k_{i1}, k_{i2}))\})$ resulting in $\{3\text{-Anonymity}\}$. The algorithm terminates with all records iterated, resulting in the definition of *3-Anonymity* for the data-set as the strictest privacy model.

6.4.4 *Apply Privacy Models*

After the completion of the *Set Maximum Anonymization*, *Set Privacy Group*, and *Privacy Model Substitution* processes, all necessary information is available to apply the privacy models, such that the data-set

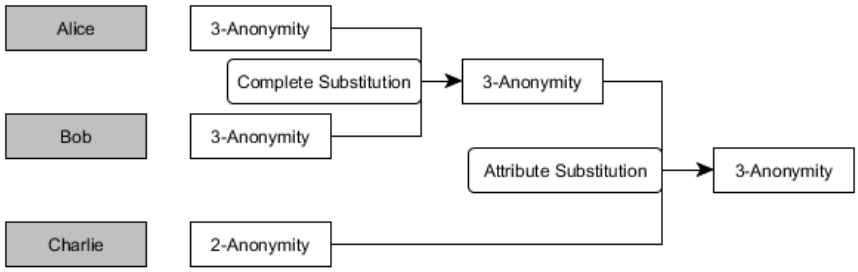


Figure 6.8: Example for *Privacy Model Substitution* process.

is anonymized to fulfil the requirements of the privacy models. The minimum necessary set of privacy models is configured resulting in a *De-identified Data-set*, which is pseudonymized and anonymized according to individuals' personal privacy requirements. Furthermore, the utility requirements given by the policy creators in form of anonymization limitation is considered and fulfilled. Assuming the data-warehouse scenario, this results in Table 6.10. The efficiency of this process step highly depends on the set of privacy models, i.e. the efficiency of each privacy model itself.

6.5 DISCUSSION

Within this chapter, the *Policy-based De-identification (PD)* process is detailed which combines *Pseudonymization*, *Personal Privacy Anonymization*, and *Privacy Models*. This process aims at de-identification of multidimensional data-sets. Furthermore, utility requirements,i.e. the *Maximum Anonymization Level* set by the policy creator, are taken into account. Thus, a trade-off between privacy and utility based upon LPL privacy policies is achieved. The possibility of multiple privacy policies is introduced. This can be introduced by personalized policies or different origin of LPL privacy policies. Therefore, the challenge of unifying the privacy and utility requirements for each distinct request of personal data is raised. With the combined usage of more than one privacy model, the privacy guarantees of all privacy models apply, i.e. privacy models are complementary such that the combined strictest privacy guarantees are given. This challenge is tackled by several approaches, most noteworthy are the *Privacy Model Substitution Table* or different algorithms for the *Personal Privacy Anonymization*. Each approach is carefully designed to preserve the individuals' privacy and the data-sets utility. Furthermore, the complexity of each algorithm is discussed. In the following, several aspects of the PD process are detailed that are subject to discussion or improvement in future work.

During the *Pseudonymization* process, it is assumed that all distinct pseudonymization methods are applied on the data-set, which could introduce unnecessary overhead. This issue can be tackled by, for ex-

ample, extending the request by specifying the requested pseudonym attribute, i.e. attrName of the *PseudonymizationMethod*-element psm. Therefore, the corresponding pseudonymization is only done if the attribute representing the pseudonym is requested, but this requires also the requesting entity to be aware of such virtual pseudonym attributes that are only calculated on-the-fly and are not persisted by default.

In the data-warehouse scenario, it is assumed that the anonymization hierarchies, represented by $\widehat{\mathbf{HE}}$, are pre-processed and stored in the LPL privacy policy. Considering the data-warehouse scenario, it is possible that the anonymization hierarchies for the same original value differs, e.g., due to different origins. Therefore, the anonymization process for *Personal Privacy Anonymization* and *Privacy Models* may require additional correction mechanisms to unify heterogeneous anonymization hierarchies. Another approach would be to process the anonymization hierarchies on-the-fly instead of pre-processing them (see Section 5.3.3). Therefore, generalization hierarchies or suppression strategies would have to be provided (see Section 3.2). But this approach could be exploited to tamper with the *Minimum Anonymization Levels* of individuals by introducing unnecessary hierarchy levels without any actual anonymization. Thus, use cases and appropriate solution approaches have to be researched to tackle this challenge.

The *GMA*-algorithm for the *Personal Privacy Anonymization* may introduce high utility loss in the case of an outlier *Minimum Anonymization Levels*. Therefore, appropriate mechanisms to detect such outliers have to be put in place to preserve the utility. One possible approach can be envisioned which analyses first DW to determine the minimum, maximum and mean *Minimum Anonymization Level* for each attribute. If the distance of the minimum to the mean or mean to maximum exceeds some pre-defined thresholds then countermeasures can be put into place to mitigate information loss. Such countermeasures can include to switch to the *MA*-algorithm or to alter DW removing all outlier dw while preserving the overall properties of the data-set. The feasibility and effects of personal privacy outlier detection and mitigation strategies is therefore subject to future research.

To define a set of privacy models that fulfils the requirements of a set of privacy models, the *Privacy Model Substitution Table* has been introduced alongside a complementary algorithm. Hereby, the rules for the substitution are defined based on the privacy guarantees, i.e. mitigated privacy attacks, as well as expert knowledge to unify the configuration. The generation of the rules could be further extended by considering additional properties of privacy models, e.g., performance, utility-preservation or compatible data-set formats. Furthermore, the extension of the *Privacy Model Substitution Table* becomes infeasible if a high number of privacy models has to be included, because each additional privacy model has to be matched against all existing privacy

models. This could be tackled by an algorithmic solution automatically deriving the optimal rules based upon the properties of each privacy model, such as only n entries are required instead of $\frac{n(n+1)}{2}$.

Lastly, the PD process is only designed for multidimensional data-sets, but other types of data-sets have to be privacy-preserving processed, e.g., transactional data-sets or longitudinal data-sets (see Section 3.1). Although the overall design of the PD process is applicable to other data-sets, the de-identification methods as well as LPL would need to be extended, e.g., to incorporate location-based privacy. On the one hand, the current location of the user may be factored in as a restriction during the definition of the purpose, i.e. an extension for LPL. For example, if the user is located within his work area his movements can be processed, otherwise the movements of the user are not allowed to be tracked. On the other hand, the PD process may be extended to incorporate methods to preserve the privacy of location data. A sequence of locations of a user can be used to infer movement patterns, i.e. to identify the homes of users [22], or infer sensitive information, e.g. the age, work frequency, or if the user is a smoker or coffee drinker [180]. Various methods are proposed to counter such inferences, i.e. by degrading the spatial and temporal data [161] [279]. The integration of such location-based privacy approaches in LPL and PD will require extensive literature research, use case definitions, as well as quantitative and qualitative evaluations.

In summary, the PD process demonstrate how a privacy language can be utilized to preserve the privacy when a set of users' personal data is requested for processing. This fulfils the last requirement for a privacy language – *R5 De-identification Capabilities* – introduced in Chapter 2. Therefore, the first research question *RQ1* 'How to represent legal privacy policies in a machine-readable format which complies to the legal requirements of the General Data Protection Regulation in the EU while privacy guarantees are defined?' is answered.

Furthermore, the second research question *RQ2* 'How can machine-readable privacy policies, expressing privacy-preserving methods, be utilized to efficiently preserve the privacy of individuals when a set of users' personal data is requested for processing?' is partially answered, since it was shown how a machine-readable privacy policy, i.e. LPL, can be utilized to preserve the individuals' privacy while a data-set is requested. Therefore, not only the *Policy-based De-identification (PD)* is essential, but also the *Policy-based Access Control (PAC)* (see Section 5.5). What remains unanswered of *RQ2* is the efficiency of the PD process, which is evaluated in the next chapter via multiple experiments.

7

EVALUATION

After detailing the *Layered Privacy Language (LPL)* and the *Policy-based De-identification (PD)* process, the open question regarding *RQ2* is if the processing of LPL policies in addition to the de-identification process can be done efficiently. In this chapter, we introduce the *Policy-based De-identification Benchmark Framework (PDBF)* which is used for the following experiments. The used data-sets, privacy model framework, and hardware specification are furthermore detailed. After presenting the set-up, several aspects of the PD process are evaluated. First, the PD process run-time is compared to the run-time of privacy models to validate the overall feasibility of the PD process. Next, the impact of personal privacy settings on the run-time of the PD process is evaluated for which the *Minimum Anonymization*- and *Global Minimum Anonymization*-algorithm are compared for the *Personal Privacy Anonymization*. Lastly, the results are summarized and discussed.

7.1 POLICY-BASED DE-IDENTIFICATION BENCHMARK FRAMEWORK

To evaluate the PD process, not only data-sets have to be provided, but also LPL privacy policies have to be provided for each record of the data-set. The LPL privacy policies offer various possibilities to define de-identification including pseudonymization, personal privacy settings, i.e. setting the *Minimum Anonymization Level*, and privacy models, which can vary in each policy. Thus, various configurations are possible. Furthermore, run-time measurements have to be taken throughout the PD process and conditioned for evaluation. The *Policy-based De-identification Benchmark Framework (PDBF)* is proposed by Gerl and Becher [111] to fulfil those requirements (see Figure 7.1). PDBF uses a specification language to define the properties of the benchmark, e.g., the data-sets, level of anonymity and points for run-time measurements. Based upon the configuration, the *Data Provider* and *Policy Provider* generate the necessary inputs, which are then processed using the PD process. The generated run-time measurements and de-identified data-sets are used for evaluation.

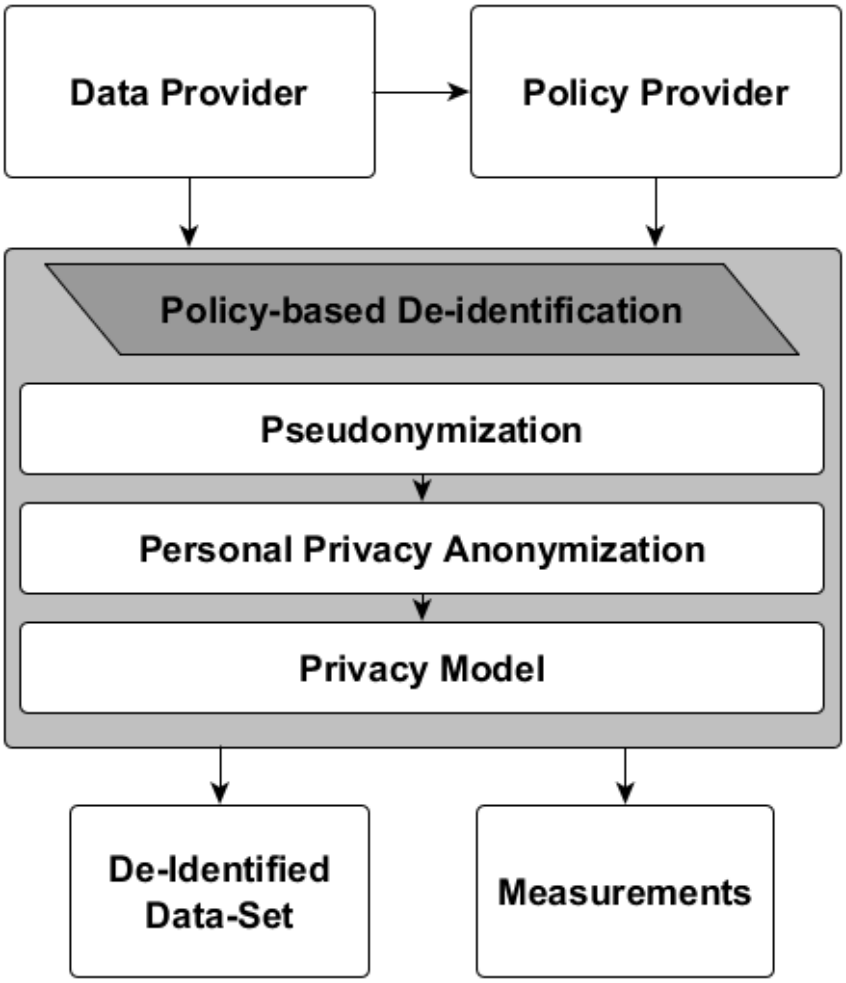


Figure 7.1: Framework architecture for *Policy-based De-identification* benchmark evaluation.

7.1.1 *Data Provider*

The *Data Provider* defines the data, which is de-identified by the PD process. Therefore, the data is served as an input to the *Policy Provider* as well as the PD process.

To conduct the experiments, common data-sets for evaluating privacy models were used, i.e. *1994 US census database (ADULT)* [88], *1998 KDD data mining competition (CUP)* [137], *US NHTSA crash statistics (FARS)* [191], *American Time Use Survey (ATUS)* [263], and *Integrated Health Interview Series (IHIS)* [47] (see Table 7.1). These data-sets have been provided by Prasser and Kohlmayer [223] [159] with corresponding anonymization hierarchies for each attribute.

All data-sets have a different number of records, ranging from around thirty thousand to over one million records. Each data-set has 8 to 9 attributes. The anonymization hierarchies for the attributes have

a size from 2 up to 6. It is noteworthy, that the number of distinct values for all attributes in the data-set for *CUP*, i.e. with over 14,000 distinct values, highly exceeds other data-sets, which have up to 305 distinct values.

Table 7.1: Overview of used data-sets for the evaluation. *#Attr* denotes number of attributes. *Hierarchy Size* defines for each attribute the depth of the hierarchy. *#Records* denotes the number of records. *#DV* denotes the number of distinct attribute values.

Data-set	**#Attr**	**Hierarchy Size**	**#Records**	**#DV**
ADULT	9	(2,5,2,3,4,3,3,2,3)	30,162	166
CUP	8	(6,5,2,3,2,5,5,5)	63,441	14,407
FARS	8	(6,3,4,4,2,3,3,4)	100,937	238
ATUS	9	(3,6,2,3,3,3,3,3,4)	539,253	305
IHIS	9	(6,3,3,4,5,3,2,2,2)	1,193,504	186

The *Data Provider* can condition the data-sets such that it is randomized, i.e. shuffling records, or shortened, i.e. a subset of records is used. Thus, comparable subsets of a data-set can be configured.

7.1.2 *Policy Provider*

The *Policy Provider* creates a corresponding LPL privacy policy lpp with exactly one *Purpose*-element p for each record of the data-set. It is assumed, that p resulting from the PAC process for each record (see Section 5.5). The LPL privacy policy lpp, i.e. the *Purpose* p, is generated only with the necessary elements and attributes required for the PD process. Thus, elements like the *Controller* c or *DataProtectionOfficer* dpo are omitted while for the PD process relevant elements are included, e.g., *Data* d or *Privacy Model* pm.

For each attribute of the record, a corresponding *Data*-element d is generated with an *AnonymizationMethod*-element am and set of *HierarchyEntry*-elements for the anonymization hierarchies. Furthermore, the pGroup is defined for each d, such that the attributes are classified as either EI, QI, SD, or NSD. Independent from the data-set, the remaining de-identification methods are defined for the purpose p.

- *PseudonymizationMethod*-element psm: A set of psm defines the pseudonymization applied on the data-set. Required *PseudonymizationMethodAttribute*-elements psma are generated for each psm to complete the configuration.
- *PrivacyModel*-element pm: A set of pm defines the privacy models for the data-set. Required *PrivacyModelAttribute*-elements pma are generated for each pm to complete the configuration.

- *AnonymizationMethodAttribute*-element ama: For each attribute, the *Minimum Anonymization Level* is specified representing the personal privacy requirements of the individual. Furthermore, for each attribute the *Maximum Anonymization Level* is specified representing the utility requirements of the policy creator, i.e. the company.

Therefore, the *Data Provider* defines the properties of the policy, pseudonymization methods, personal privacy anonymization, and privacy models, which can all be configured and customized. The created LPL privacy policy lpp is correlated to the data-set records. Thus, the *Policy Provider* specifies a personalized sticky policy for each record of the provided data-set.

7.1.3 *Policy-based De-identification*

The *Policy-based De-identification* receives the data-set from the *Data Provider* and the LPL privacy policies from the *Policy Provider*. Based on this, for each record and LPL privacy policy pair, a *DataWrapper*-object dw is created. The attributeList of dw is based upon the attribute name for the key and the value of the record is set as the value. Furthermore, the *Purpose*-element p of the generated LPL privacy policy lpp is set. Based on the set of *DataWrapper*-objects DW, the PD process is performed including *Pseudonymization*, *Personal Privacy Anonymization*, and *Privacy Models* as detailed in Chapter 6. All algorithms are implemented with *Java*, except privacy models for which the ARX framework is utilized. An overview of implemented algorithms and selection of the ARX framework is detailed in the following.

For *Pseudonymization*, several algorithms are implemented for both *Hashing* and *Cryptographic* approaches. For non-keyed *Hashing*, the MDX and SHA-X algorithms are used, such that their variations MD2 [147], MD5 [232], SHA-1, SHA-256, SHA-384, and SHA-512 are supported [254] [79] [57] [201]. For keyed *Hashing*, HMAC is implemented using MDX and SHA-X algorithms, e.g., HMAC-MD5 or HMAC-SHA-512 [35] [1]. The usage of salts is supported by the PBKDF2 algorithm in combination with HMAC functions, e.g., PBKDF2 with HMAC-SHA-256 [148] [1]. Lastly, for the *Cryptographic* approach the algorithms AES [75] [250], DES [85], and Blowfish [241] are supported.

Both the *MA*- and *GMA*-algorithms are implemented for the *Personal Privacy Anonymization*. The anonymization is based upon *Generalization* and *Suppression*, but only the pre-processed values in the *HierarchyEntry*-elements are used to replace the original values.

Several privacy-preserving frameworks, providing privacy models implementations, were surveyed to select the most suitable for PDBF. Therefore, several criteria were defined that the desired framework has to meet.

- *Currentness:* The privacy-preserving framework should be actively maintained and developed with a recent last update.
- *Number of Privacy Models:* Various privacy models, and their variations, should be provided. At least one variation of *k-Anonymity*, *l-Diversity*, *t-Closeness* and δ-*Presence* should be provided.
- *Benchmark:* The framework should provide a benchmark or possibilities to quantify the de-identified data-set such that the results can be used as a baseline for the implementation, e.g., run-time, utility, or privacy.

Considering these criteria, several frameworks were compared, including *UTD Anonymization Toolkit (UTD)* [149], *Cornell Anonymization Toolkit (CAT)* [125], *Tool for Interactive Analysis of Microdata Anonymization Techniques (TIAMAT)* [77], *System for Evaluating and Comparing Anonymization Algorithms for Relational, Transaction, and R-T Data-sets (SECRETA)* [218], *Open Anonymizer* [252], *Anon-Tool* [87], μ-*ARGUS* [208], *sdcMicro* [259] and *ARX* [223]. Table 7.2 summarizes the results of the comparison, showing that *sdcMicro*, μ-*ARGUS*, and ARX are the most current frameworks. Furthermore, it can be observed that most frameworks only support 1 to 3 privacy models, except *ARX* which supports 21 variations of privacy models including the desired privacy models *k-Anonymity*, *l-Diversity*, *t-Closeness* and δ-*Presence*. Lastly, a *Benchmark* is only supported by *CAT*, *SECRETA*, and *ARX*, whereas *ARX* offers exceptional functionality to measure the run-time, utility and privacy. Based on this comparison, the *ARX* framework fulfils all criteria. Additionally, it is free to use under the Apache License 2.0 and well documented. Thus, it was selected as the basis for the PD process in PDBF. Because, ARX is chosen, additional algorithms had to be integrated in the PD process (see Annex A).

In summary, the PDBF supports various methods for *Pseudonymization*, *Personal Privacy Anonymization*, and *Privacy Models* that can be freely combined for the PD process. Furthermore, the usage of various data-sets and varying LPL privacy policies for each record is supported, such that personal privacy settings can be simulated.

Table 7.2: Overview of privacy-preserving frameworks providing privacy model implementations.

Framework	Currentness	Privacy Models	Benchmark
UTD	2012	3	-
CAT	2014	2	X
TIAMAT	2009	1	-
SECRETA	2014	1	X
Open Anonymizer	2015	2	-
Anon-Tool	2014	1	-
μ-ARGUS	2018	-	-
sdcMicro	2018	2	-
ARX	2018	21	X

7.1.4 *Benchmark Configuration*

The run-time for all following experiments is measured with nanosecond precision. The measurements use the arithmetic mean of 10 consecutive runs. Initial experiments showed that no warm-up runs are required for the measurements, therefore they are omitted.

Within *ARX*, several parameters can influence the overall performance (run-time, memory consumption) while the privacy model is applied on the data-set [222]. For ARX the following default configuration is used, which affects the performance, i.e. run-time and memory consumption, of the anonymization for the privacy model:

- *historySize = 200*: The maximum number of snapshots stored in the buffer.
- *maximumSnapshotSizeSnapshot = 0.8*: The maximum relative size of a snapshot compared to the dataset.
- *maximumSnapshotSizeDataset = 0.2*: The maximum relative size of a snapshot compared to its predecessor.

The anonymization process, in terms of privacy and utility, is not affected by these parameters. To have comparable configurations, common privacy models in the literature are selected. Their configuration is furthermore chosen according to the configuration used in the respective works:

- *k-Anonymity* (k = 5) [257]
- *(c,l)-Diversity* (c = 4) and (l = 3) [177]

- *t-Closeness* (t = 0.2) [167]
- δ-*Presence* (δ = (0.05, 0.15)) [194]

The experiments are run on a 64-bit Windows 7 desktop computer, which is stocked with an Intel Core i7-6700HQ processor and 32 GB of RAM. The 64-bit JVM (1.8.0_91) is given a heap size of 12.5 GB *-Xmx12500m*. To achieve more stable run-times the concurrent mark sweep (CMS) garbage collector is used (*-XX:+UseConcMarkSweepGC*).

Based on the introduced *Policy-based De-identification Benchmark Framework*, several experiments are detailed in the following to evaluate the run-time efficiency of the PD process.

7.2 EXPERIMENTS

The PD process has been introduced to answer how machine-readable privacy policies can be utilized to preserve the privacy of individuals, thus partially answering the second research question *RQ2*. Indeed, what has not been covered yet of *RQ2* is how efficient the PD process is. Therefore, the focus of this chapter lies on the *Personal Privacy Anonymization* and *Privacy Model* sub-processes of PD. Experiments on the efficiency of the *Pseudonymization* sub-process are omitted, because their efficiency highly relies on the method for the pseudonym generation. For example, the usage of cryptographic methods, e.g., AES, requires more run-time than the usage of hashing methods, e.g., MD5. However, the run-time of the remaining PD processes is subject to the evaluation as well as the influence of personal privacy settings. In both cases, it is required to not only process the dataset but also the corresponding LPL privacy policy. Therefore, the viability of the PD process compared to the sole application of state-of-the-art privacy models is evaluated. Furthermore, the addition of personal privacy settings, i.e., LPL privacy policies with varying de-identification requirements, is subject to the evaluation. Lastly, the *MA*- and *GMA*-algorithms are compared according to their influence on the run-time of the PD process while personal privacy settings are given.

7.2.1 *Policy-based De-identification Overhead*

To evaluate the viability of the overall *Policy-based De-identification* approach, the run-time of the processes introduced by the PD process are compared to the sole application of privacy models. For a fair comparison, no pseudonymization methods nor records with personal privacy settings are used, such that only *Personal Privacy Anonymization* and *Privacy Model* processes of PD are taken into account. A detailed analysis of the run-time performance of the sub-processes highlights the impact of each algorithm on the overall run-time. It is expected

that the overall PD process run-time exceeds the run-time of the sole application of privacy models.

7.2.1.1 *Experiment Set-up*

Each privacy model (*k-Anonymity*, *(c,l)-Diversity*, *t-Closeness*, *δ-Presence*) is used on each data-sets (*ADULT*, *CUP*, *FARS*, *ATUS*, *IHIS*). All the attributes of the data-sets are used. No personal privacy settings or pseudonymization is applied. The *Minimum Anonymization* is used for the *Personal Privacy Anonymization*. The run-time of the overall PD process, as well as each sub-process is measured. This results in 20 different configurations for which the results are shown in Table 7.3. This experiment simulates a scenario in which all records have the same privacy policy configuration considering only one privacy model. Therefore, it extends the sole application of an privacy model on a data-set by the PD process.

The determination of the run-time difference is conducted based on the measures, which have been conducted with nanosecond precision, of each sub-process and the overall run-time. Therefore, the run-time measures r_{MA}: *Minimum Anonymization*, r_{max}: *Set Maximum Anonymization*, r_{group}: *Set Privacy Group*, r_{sub}: *Privacy Model Substitution*, r_{pm}: *Apply Privacy Models*, and $r_{overall}$: *Overall Run-time* (see Equation 7.1) are introduced.

$$r_{overall} = r_{MA} + r_{max} + r_{group} + r_{sub} + r_{pm} \tag{7.1}$$

The Δ_{time} is calculated as the sum of the introduced sub-process of the PD algorithms to apply *Personal Privacy Anonymization* and *Privacy Models*.

$$\Delta_{time} = r_{MA} + r_{max} + r_{group} + r_{sub} \tag{7.2}$$

For the calculation of $\Delta_{percent}$, the overall adjusted run-time $r_{overall}$ is put into relation to the sole run-time of the application of privacy models r_{pm} (see Equation 7.3).

$$\Delta_{percent} = ((\frac{r_{overall}}{r_{pm}}) * 100) - 100 \tag{7.3}$$

7.2.1.2 *Individual Process Run-time Impact*

In the following, each process is analysed and discussed in terms of its impact on the overall run-time performance.

MINIMUM ANONYMIZATION The results are similar within each data-set group, i.e. for the same data-set size. This is to be expected considering the theoretical complexity analysis (see Section 6.3.1). The results within a data-set group vary, this can be tracked back to measurement errors which is also shown by a relative high deviation

of run-times even for the same configuration. The percentage of the run-time of the *Minimum Anonymization*-algorithm to the overall run-time of the PD process is calculated (see Equations 7.4.

$$\%_{MA} = (\frac{r_{MA}}{r_{overall}}) * 100 \tag{7.4}$$

Considering $\%_{MA}$, it can be observed that the impact of the *MA*-algorithm on the overall run-time $r_{overall}$ is small for configurations with big data-sets, i.e. *IHIS*, or run-time heavy privacy models, i.e. *(0.05, 0.15)-Presence* (see Table 7.3). The range of $\%_{MA}$ lies between 0.11% and 10.27%.

Although no personal privacy settings are given, the algorithm iterates over each *Data*-element, or more specifically its *AnonymizationMethod*- and *AnonymizationMethodAttribute*-elements, to determine the *Minimum Anonymization Level*. Then, the *Minimum Anonymization Level* is used to replace the original value with the corresponding value of the *HierarchyEntry*-element. Within, this experiment the *Minimum Anonymization Level* is always set to '0', therefore the original value is used for each value. Therefore, each attribute for each record in the data-set is iterated. Thus, the number of attributes and the number of records both linearly influences the run-time of *MA*.

SET MAXIMUM ANONYMIZATION The *Set Maximum Anonymization*-algorithm determines the *Maximum Anonymization Level* of each attribute of each record, then calculates the minimum of the *Maximum Anonymization Level* for each attribute and sets it as a limit for the anonymization. For this experiment, the default *Maximum Anonymization Level* is used, which is the size of the hierarchy for each anonymization hierarchy (see *Hierarchy Size* in Table 7.1). Therefore, the *Set Maximum Anonymization*-algorithm depends on the number of records and number of attributes (see Section 6.4.1). This low complexity of the algorithm, is represented by the results. The results show constant measurements below 0.00 seconds for each configuration, indicating that the impact on the overall run-time is neglect-able even for big data-sets, i.e. *IHIS*.

SET PRIVACY GROUP The *Set Privacy Group*-algorithm determines the privacy group for each attribute and sets it for the application of the privacy model. Similar to the *Set Maximum Anonymization*-algorithm, it depends on the number of records and number of attributes (see Section 6.4.2). This low complexity of the algorithm, is represented by the results. The results show constant measurements below 0.00 seconds for each configuration, indicating that the impact on the overall run-time is also neglect-able.

Table 7.3: Benchmark results for the *Policy-based De-identification Overhead* evaluation. All measurements are displayed in seconds. Calculations of the Δ_{time}, $\Delta_{percent}$, $\%_{MA}$, $\%_{sub}$, and $\%_{pm}$ and have been conducted with nanosecond precision measures, therefore deviations of displayed values and calculations can occur. The column *Overall*, representing the full run-time and *PM*, representing the run-time of the privacy model application, are used for the overhead calculation. Labels have been shortened according to following scheme: (*MA*: Minimum Anonymization, *Max*: Set Maximum Anonymization, *Group*: Set Privacy Group, *Sub*: Privacy Model Substitution, *PM*: Apply Privacy Models, *Overall*: Overall Execution Time).

Configuration		*Policy-based De-identification Process*						$\%_{MA}$	$\%_{sub}$	$\%_{pm}$	Δ	
Privacy Model	Data-set	MA	Max	Group	Sub	PM	Overall				Time	Percent
5-Anonymity	ADULT	0.09	0.00	0.00	0.04	1.07	1.20	7.50	3.33	89.17	0.13	10.83
(4,3)-Diversity	ADULT	0.07	0.00	0.00	0.05	1.33	1.45	4.83	3.45	91.72	0.12	9.02
0.2-Closeness	ADULT	0.07	0.00	0.00	0.03	2.98	3.08	2.27	0.97	96.75	0.10	3.36
(0.05, 0.15)-Presence	ADULT	0.06	0.00	0.00	0.05	6.45	6.56	0.91	0.76	98.32	0.11	1.71
5-Anonymity	CUP	0.15	0.00	0.00	0.07	1.24	1.46	10.27	4.79	84.93	0.22	17.74
(4,3)-Diversity	CUP	0.18	0.00	0.00	0.09	3.45	3.72	4.84	2.42	92.74	0.27	7.83
0.2-Closeness	CUP	0.23	0.00	0.00	0.04	62.37	62.64	0.37	0.06	99.57	0.27	0.43
(0.05, 0.15)-Presence	CUP	0.19	0.00	0.00	0.10	177.98	178.27	0.11	0.06	99.84	0.29	0.16
5-Anonymity	FARS	0.23	0.00	0.00	0.09	3.61	3.93	5.85	2.29	91.86	0.32	8.86
(4,3)-Diversity	FARS	0.22	0.00	0.00	0.11	4.12	4.45	4.94	2.47	92.58	0.33	8.01
0.2-Closeness	FARS	0.20	0.00	0.00	0.11	9.15	9.46	2.11	1.16	96.72	0.31	3.39
(0.05, 0.15)-Presence	FARS	0.27	0.00	0.00	0.13	30.79	31.19	0.87	0.42	98.71	0.40	1.30
5-Anonymity	ATUS	1.83	0.00	0.00	0.33	26.47	28.63	6.39	1.15	92.46	2.16	8.16
(4,3)-Diversity	ATUS	1.86	0.00	0.00	0.46	27.42	29.74	6.25	1.55	92.20	2.32	8.46
0.2-Closeness	ATUS	1.87	0.00	0.00	0.37	39.41	41.65	4.49	0.89	94.62	2.24	5.69
(0.05, 0.15)-Presence	ATUS	1.84	0.00	0.00	0.52	122.99	125.35	1.47	0.41	98.12	2.36	1.92
5-Anonymity	IHIS	3.76	0.00	0.00	0.74	117.03	121.53	3.09	0.61	96.30	4.50	3.85
(4,3)-Diversity	IHIS	3.42	0.00	0.00	1.03	109.74	114.19	3.00	0.90	96.10	4.45	4.06
0.2-Closeness	IHIS	3.33	0.00	0.00	0.84	169.17	173.34	1.92	0.48	97.60	4.17	2.46
(0.05, 0.15)-Presence	IHIS	4.40	0.00	0.00	1.21	563.73	569.34	0.77	0.21	99.02	5.60	1.00

PRIVACY MODEL SUBSTITUTION The *Privacy Model Substitution*-algorithm determines a set of privacy models that the de-identified data-set has to comply to. The measurement results are similar within each data-set group. For the experiment, each LPL privacy policy has exactly one *PrivacyModel*-element pm specified. Although the privacy model does not differ within each configuration, each pm has to run through the substitution process (see Section 6.4.3). Furthermore, it can be observed that the substitution of the privacy models has an impact on the overall run-time due to the noticeable run-time increase compared to *Set Maximum Anonymization* and *Set Privacy Group*. The percentage of the run-time of each the *Privacy Model Substitution*-algorithm to the overall run-time of the PD process is calculated (see Equations 7.5.

$$\%_{sub} = (\frac{r_{sub}}{r_{overall}}) * 100 \tag{7.5}$$

Considering $\%_{sub}$, it can be observed that the impact of the *Privacy Model Substitution*-algorithm on the overall run-time $r_{overall}$ is very small, especially for configurations with big data-sets, i.e. *IHIS*, or run-time heavy privacy models, i.e. *(0.05, 0.15)-Presence* (see Table 7.3). The range of $\%_{sub}$ lies between 0.06% and 4.79%.

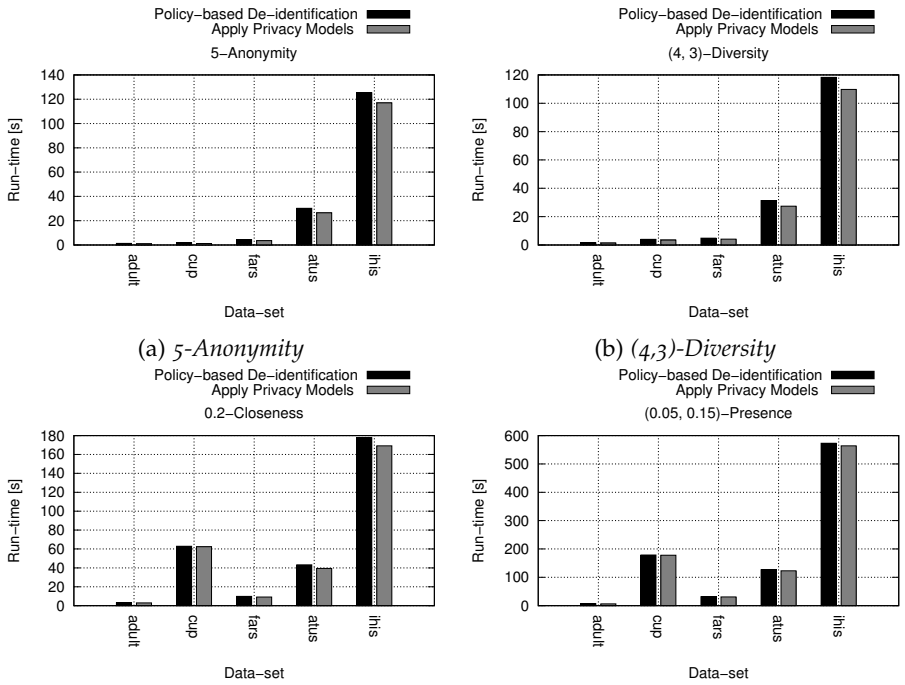
Considering the configurations for the *IHIS* data-set, it can be observed that for *(0.05, 0.15)-Presence* the run-time of 1.21 seconds exceeds the run-time of privacy models with less configuration parameters, e.g., *5-Anonymity* with a run-time of 0.74 seconds. Therefore, it can be concluded that also the number of *PrivacyModelAttribute*-elements pma impacts the run-time. Thus, the run-time of this algorithm is influenced by the number of records, number of defined privacy models pm, and number of privacy model parameters pma.

APPLY PRIVACY MODELS It can be observed that the run-time to apply the privacy model, i.e. anonymize the data-set to match the defined properties of the privacy model, highly varies for each configuration (see Section 6.4.4). Additionally, it can be observed that the size of the data-set has considerable impact on the run-time. Also the number of attributes, i.e. EI, QI and SD attributes, which are concerned by privacy models, influences the run-time. Thus, the data-set size, number of EI, QI and SD attributes, and privacy model itself have an impact on the run-time. The percentage of the run-time of the *Apply Privacy Models*-algorithm to the overall run-time of the PD process is calculated (see Equations 7.6.

$$\%_{pm} = (\frac{r_{pm}}{r_{overall}}) * 100 \tag{7.6}$$

Considering $\%_{pm}$, it can be seen that the run-time of the privacy model highly exceeds the run-time of all other algorithms, e.g., for the application of *(0.05, 0.15)-Presence* on IHIS the *MA* requires 4.40

seconds while the run-time of the privacy model *(0.05, 0.15)-Presence* r_{pm} requires 563.73 seconds. This is a $\%_{pm}$ of 99.02%. In general, the range of $\%_{pm}$ lies between 84.93% and 99.84%. Thus, the run-time of the privacy model is identified with the highest impact factor on the run-time (see Figure 7.2).



(a) *5-Anonymity* (b) *(4,3)-Diversity*

(c) *0.2-Closeness* (d) *(0.05, 0.15)-Presence*

Figure 7.2: Run-time comparison of *Policy-based De-identification* to *Apply Privacy Models* for different privacy models.

SUMMARY It can be concluded that the algorithms *Set Maximum Anonymization* and *Set Privacy Group* have only a minor impact on the run-time, because they only depend on the number of records and number of attributes and require no complex logic. The processes *Minimum Anonymization* and *Privacy Model Substitution* also depend on the number of records and number of attributes, but have more complex functionality. Therefore, they have a small but noticeable impact on the run-time considering $\%_{MA}$ and $\%_{sub}$. Lastly, the experiment has shown that the run-time of the privacy model has the highest impact on the overall run-time, which is further detailed in the following comparison of the run-time difference of the PD process to the sole application of privacy models.

7.2.1.3 *Policy-based De-identification Run-time*

To validate the efficiency of the presented PD process, Δ_{time} and $\Delta_{percent}$ are calculated and considered. Hereby, the actual run-time difference (*Time*) and the percentage difference (*Percent*) for each con-

figuration is calculated (see Table 7.3). It can be observed that the run-time increases overall linearly for each privacy model and data-set, i.e. number of records (see Figure 7.3). Minor variations can be observed, which can be tracked back to measurement errors. Due to the variation in performance of the privacy model and the rather similar run-time of the remaining algorithms for configurations with the same data-set (see Table 7.3), it can be observed that the Δ_{time} run-time mainly depends on the privacy model.

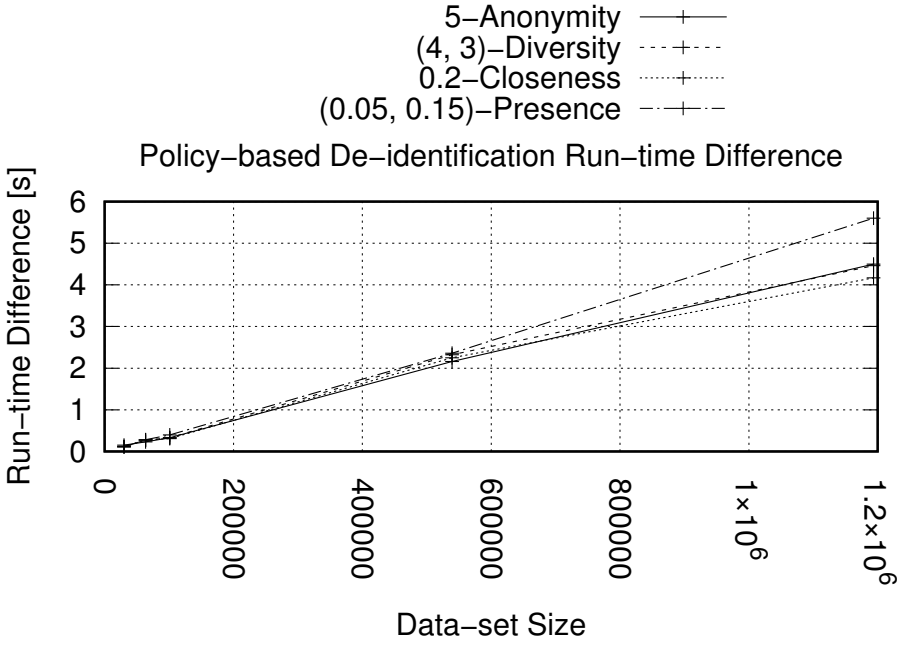


Figure 7.3: Run-time difference for all configurations from Table 7.3 of the *Policy-based De-identification Overhead* experiment. All data-sets are used and visualized in the graph via their size, i.e. number of records (see Table 7.1).

Considering the percentage difference of the PD process to the sole application of privacy models, several observations can be made (see Figure 7.4). First, the percentage $\Delta_{\mathrm{percent}}$ run-time varies for each privacy model asserting the previously derived observation that the privacy model itself has a significant impact on the run-time. Second, the overall run-time overhead decreases with an increasing size of the data-set.

Lastly, abnormalities in the $\Delta_{\mathrm{percent}}$ run-time can be observed for the *CUP* data-set for each privacy model. While, the $\Delta_{\mathrm{percent}}$ is increased exponentially for *5-Anonymity*, it is decreased for the remaining privacy models compared to the assumed linear growth of $\Delta_{\mathrm{percent}}$. Inspecting the run-time measurements of the individual processes, it can be observed that r_{pm} is increased, thus the run-time for the privacy model is causing this abnormality. A closer inspection of the data-sets shows that *CUP* has a significant higher number of distinct values (see Table 7.1), which can have an impact on the run-

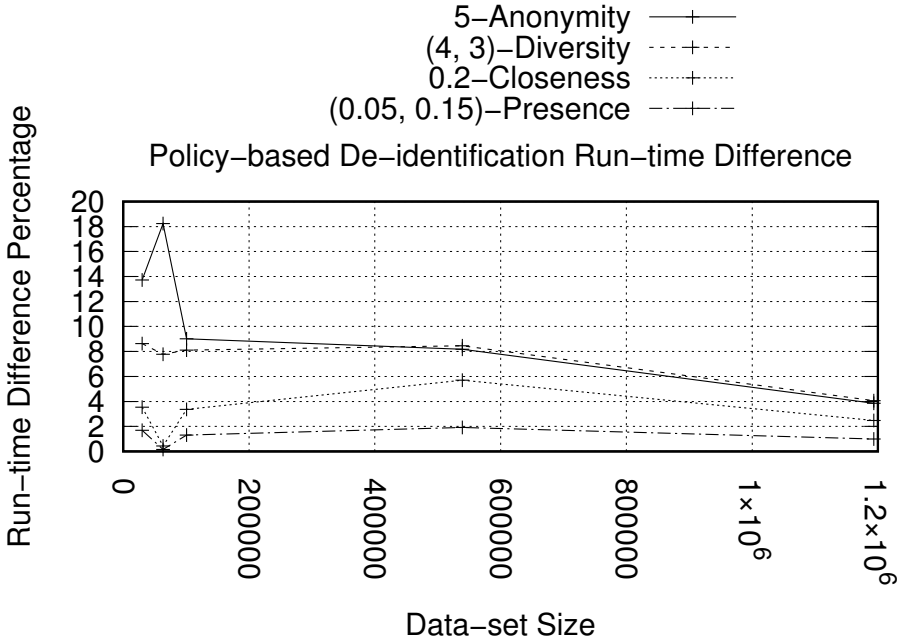


Figure 7.4: Percentage run-time difference for all configurations from Table 7.3 of the *Policy-based De-identification Overhead* experiment. All data-sets are used and visualized in the graph via their size, i.e. number of records (see Table 7.1).

time of the privacy models. Additional experiments revealed that also *Differential Privacy* or *k-Map* show the same effect as *k-Anonymity* if a data-set with a high number of distinct values (#DV) is de-identified. Thus, this abnormality can be tracked back to the increased number of distinct values in *CUP*, which has a varying impact on the run-time of some privacy models.

Concluding the results for the data-set groups, the average run-time difference of the overall process is 6.23% for all configurations of *ADULT*, 6.54% for *CUP*, 5.39% for *FARS*, 6.06% for *ATUS*, and 2.84% for *IHIS*. Therefore, the overhead has a tendency to decrease for an increased number of records.

Considering the results of the privacy model groups, the average run-time difference of the overall process is 9.89% for all configurations with *5-Anonymity*, 7.48% for *(4,3)-Diversity*, 3,07% for *0.2-Closeness*, and 1.22% for *(0.05, 0.15)-Presence*. This confirms the strong dependency of the run-time overhead to the used privacy model.

Furthermore, it can be concluded that the PD approach is suitable for data-warehouse scenarios (millions or billions of data records) as the percentage $\Delta_{\mathtt{percent}}$ run-time decreases with the size of the data-set. In other words, the PD approach is especially suitable for large data-sets. For example, for the data-set *IHIS* (1,193,504 records) and *(00.5, 0.15)-Presence* the overhead is only 1.00%. Therefore, if the de-identification of a big data-set in a data-warehouse environment is required, the usage of the PD process (with LPL privacy policies)

imposes only a small run-time overhead compared to the usage of privacy models. But the PD process has various advantages. First, not only anonymization of the data-set can be done to fulfil the privacy requirements of privacy models, but also the privacy requirements of individuals, i.e. personal privacy settings, are taken into account. Second, the PD process automatically determines the required privacy settings for the de-identification from various LPL privacy policies, thus diverse requirements can be unified automatically. Furthermore, this is done dynamically for each request and the decision-making is accountable. Without machine-readable privacy policies, this process would require manual, or semi-automatic, inspection of privacy policies and purpose-based decision-making, which may be error-prone. Lastly, the PD process differentiates the utility requirements, i.e. anonymization limit for attributes, for each purpose. Therefore, not only the privacy of individuals is taken into account, but also the utility requirements of the processing entity, e.g., company.

All this features are provided by the PD process for a relative small run-time overhead for big data-sets, which is advantageous for data-warehouse scenarii in which personal data originates from various sources with varying processing policies.

7.2.2 *Personal Privacy – Minimum Anonymization*

In the previous experiment, the run-time efficiency of the PD process has been analysed without any personal privacy settings. Within this experiment, the impact of the personal privacy on the run-time is evaluated using the *MA*-algorithm for *Personal Privacy Anonymization*. The experiment varies both the number of records with personal privacy settings (PP) as well as the number of attributes with personal privacy settings (#Attr).

7.2.2.1 *Experiment Set-up*

In contrast to the previous experiment, the data-set and privacy model is not altered. An initial set of experiments showed that the impact of personal privacy settings on the run-time is not influenced by the data-set or the applied privacy model. Therefore, *IHIS* is used, because it is has the highest volume of records, and *k-Anonymity* is used, because it has a relatively small run-time compared to the other privacy models and therefore the experiment is run faster overall. To determine the impact of the data-set size 10,000, 100,000, and 1,000,000 records are randomly chosen of *IHIS* for each run with the Fisher–Yates shuffle algorithm [106] to prevent any dependencies on the data. All attributes of the data-set are used.

The number of records with personal privacy (PP) is altered, i.e. how many users personalize their privacy policy, and the number of attributes (#Attr) for each record that are used for personal privacy

settings are altered. PP is altered proportional from none '0.0' to all '1.0' in steps of 10% ('0.0', '0.1', '0.2', '0.3', '0.4', '0.5', '0.6', '0.7', '0.8', '0.9', '1.0').

The *IHIS* data-set has 9 attributes, therefore the number of attributes with personal privacy (#Attr) is varied from none '0' to '9' ('0', '1', '2', '3', '4', '5', '6', '7', '8', '9'). Personal privacy is applied on an attribute, i.e. *Data*-element, by specifying the *Minimum Anonymization Level* randomly between '0' (default) and an exclusive upper limit. The exclusive upper limits have been chosen accordingly to the size of the given hierarchy of the attribute, representing a mediocre anonymization level (see Table 7.4).

The usage of '0' attributes or '0.0' records with personal privacy denotes the baselines in which no personal privacy has to be considered. This results in 331 different configurations for which the run-time is measured.

Table 7.4: Overview of IHIS attributes, the size of the corresponding anonymization hierarchies, and the exclusive upper *Minimum Anonymization Levels* used in the experiments.

IHIS Attribute	*Minimum Anonymization Exclusive Upper Limit*	*Anonymization Hierarchy Size*
year	4	6
quarter	2	3
region	2	3
pernum	3	4
age	3	5
marstat	2	3
sex	1	2
racea	1	2
educ	1	2

7.2.2.2 *Variation of Personal Privacy Percentage*

For the evaluation of the influence of the number of records in the data-set with personal privacy settings (varying from the default privacy policy) the results of the experiment using all attributes for personal privacy ('9') are used. The proportion of records with and without personal privacy settings (PP) is hereby varied for comparison (see Table 7.5).

These results show a linear growth of the run-time with regards to the data-set size, which was expected based on previous experiment (see Figure 7.5). Comparing the baseline measurements (PP '0.0') for 10,000, 100,000, 1,000,000 records with the respective results of the remaining measurements, it can be observed that for some

configurations with 100,000 and 1,000,000 records in the data-set the overall run-time is increased. For example, the baseline measurement (PP '0.0') for 1,000,0000 records has a run-time of 98.46 seconds. If records with personal privacy settings are introduced then the overall run-time is altered, e.g., if 40% percent of the the records use personal privacy settings (PP '0.4') then 116.39 seconds are required for the overall run-time. It can be observed, that the effect on the run-time can vary for 10,000 and 100,000 records (see Table 7.5). For example, the processing of 10,000 records with PP '0.1' requires 0.49 seconds compared to the baseline of 0.6, which is a decrease of 0.11 seconds. The processing of 100,000 records with PP '0.2' requires 6.66 seconds which is a decrease of 0.28 seconds to the baseline of 6.94 seconds, but the measurement of '0.8' has an increased run-time with 0.41 seconds.

Considering 1,000,000 records, the run-time is increased for each configuration using records with personal privacy settings (PP '0.1' to '1.0'). For example, the configuration with PP '0.3' requires a run-time of 112.59 seconds which is an increase of 14,13 seconds compared to the baseline of 98.46 seconds.

Table 7.5: Overall run-time of the PD process using the *Minimum Anonymization*-algorithm for varying number of records with personal privacy settings (PP). The number of attributes used for personal privacy settings (#Attr) is fixed to '9'.

Overall Run-Time [s]		**IHIS Data-set Size**		
		10,000	*100,000*	*1,000,000*
PP	*0.0*	0.6	6.94	98.46
	0.1	0.49	6.71	103.52
	0.2	0.51	6.66	108.15
	0.3	0.54	6.80	112.59
	0.4	0.53	7.06	116.39
	0.5	0.53	6.94	119.07
	0.6	0.52	6.95	122.45
	0.7	0.56	7.37	123.71
	0.8	0.57	7.35	124.51
	0.9	0.57	7.42	124.68
	1.0	0.55	7.39	124.70

A closer inspection of the individual run-time measurements of the sub-processes reveals that the run-time is relatively constant (considering measurement errors) for all sub-processes, e.g., *Personal Privacy Anonymization*, but varies for *Apply Privacy Models*. Considering the measurements for 100,000 records, it can be observed that the run-times of *Minimum Anonymization* and *Privacy Model Substitution* vary within the margin of measurement error (see Table 7.6). Therefore, the

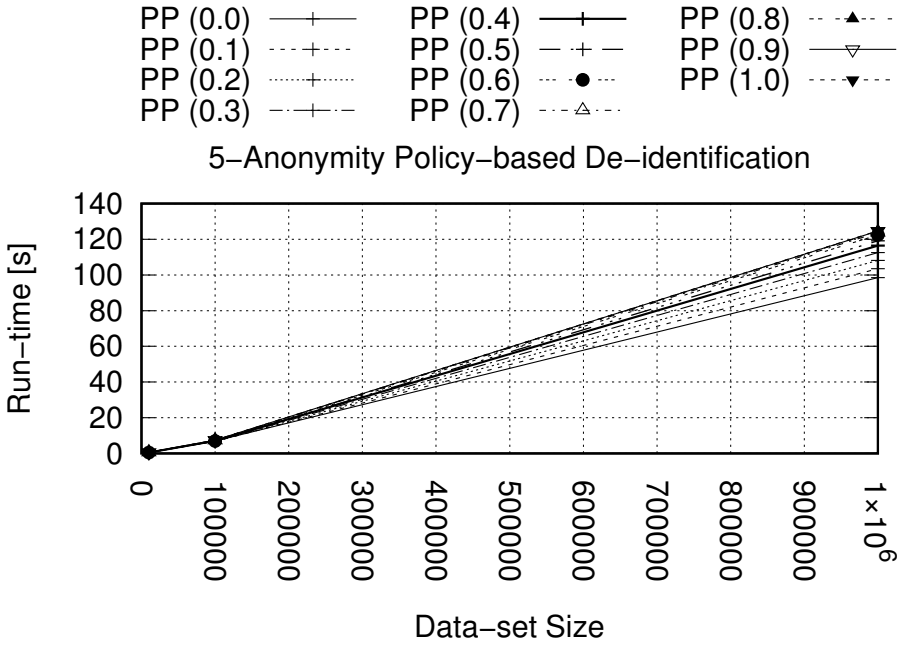


Figure 7.5: Line graph showing the overall run-time in seconds for different number of records with personal privacy (PP) on all attributes '9' using the *Minimum Anonymization*-algorithm. *5-Anonymity* is applied on different data-set sizes of *IHIS*.

variation of the proportion of records with personal privacy settings (PP) within the data-set does not have a significant impact on those processes.

Apparently, run-time of the privacy model, i.e. *5-Anonymity*, is affected by the variation of PP, which is ambiguous for data-sets with 10,000 and 100,000 records. But for data-sets with 1,000,000 records, i.e. large data-sets, it can be stated that the variation of the proportion of records with personal privacy settings (PP) within the data-set has a negative effect on the run-time, but this effect has no direct correlation to the number of records with personal privacy. In both cases, the main influencing factor on the overall run-time is the *Apply Privacy Models* sub-process of PD, which is affected by varying PP.

With the variation of PP, some values are anonymized during *MA*, which can alter the properties of the data-set, i.e. number of distinct values (#DV) and the correlated number of hierarchy elements (#HE). Depending on an increase or decrease of #DV and #HE, the run-time of the privacy model, i.e. *k-Anonymity*, is increased or decreased (see Section 7.2.3.4).

Table 7.6: Detailed run-time measurements for each individual process of PD using *5-Anonymity* on the *IHIS* data-set with 100,000 records while varying the number of records with personal privacy settings PP. Labels have been shortened according to following scheme: (*MA*: Minimum Anonymization, *Max*: Set Maximum Anonymization, *Group*: Set Privacy Group, *Sub*: Privacy Model Substitution, *PM*: Apply Privacy Models.

Configuration				*Policy-based De-identification Process*				
Privacy Model	*Data-set Size*	*PP*	*#Attr*	*MA*	*Max*	*Group*	*Sub*	*PM*
5-Anonymity	100,000	0.0	9	0.26	0.00	0.00	0.09	5.81
5-Anonymity	100,000	0.1	9	0.31	0.00	0.00	0.10	5.54
5-Anonymity	100,000	0.2	9	0.41	0.00	0.00	0.11	5.43
5-Anonymity	100,000	0.3	9	0.31	0.00	0.00	0.10	5.60
5-Anonymity	100,000	0.4	9	0.29	0.00	0.00	0.10	5.83
5-Anonymity	100,000	0.5	9	0.30	0.00	0.00	0.11	5.68
5-Anonymity	100,000	0.6	9	0.33	0.00	0.00	0.10	5.71
5-Anonymity	100,000	0.7	9	0.33	0.00	0.00	0.09	6.13
5-Anonymity	100,000	0.8	9	0.33	0.00	0.00	0.09	6.10
5-Anonymity	100,000	0.9	9	0.31	0.00	0.00	0.09	6.17
5-Anonymity	100,000	1.0	9	0.32	0.00	0.00	0.09	6.13

7.2.2.3 *Variation of the Number of Personal Privacy Attributes*

For the evaluation of the influence of the number of attributes with personal privacy in the data-set, the number of attributes (#Attr) as well as the number of records with personal privacy (PP) is varied for *IHIS*. The PP is varied, because of the previous rationale showing an unpredictable influence on the run-time. Previous results show a possible significant influence of the number of records with personal privacy. Therefore, the influence of #Attr is discussed based on a sub-set with PP of '0.4' (see Table 7.7). The discussion of other configuration sets, e.g., PP of '0.6', is omitted because it leads to the same conclusions.

The baseline measurements with '0' attributes used for personal privacy settings indicate a linear growth of the overall run-time reassuring previous results. Comparing the baseline measurements (#Attr '0') for 10,000, 100,000, 1,000,000 records with the respective results of the remaining measurements, it can be observed that for some configurations with 10,000, 100,000 and 1,000,000 records in the data-set the overall run-time is increased (see Table 7.7). Considering 1,000,000 records, the run-time is increased for each configuration (#Attr '1' to '9'). For example, the configuration with #Attr '5' requires a run-time of 114.22 seconds which is an increase of 16.55 seconds compared to the baseline of 97.67 seconds (see Figure 7.6).

In a detailed inspection of the individual PD process run-times, it can also be stated that the run-time of the privacy model, i.e. *5-Anonymity*, is affected by the variation of #Attr, which is ambiguous for data-sets with 10,000 and 100,000 records. But for large data-sets, i.e. 1,000,000 records, the number of attributes used for personal privacy settings has a negative effect on the run-time.

With the variation of #Attr, a sub-set of attributes (compared to the previous experiment) is anonymized during *MA*. Thus, the effect of altered properties of the data-set, i.e. number of distinct values (#DV) and the correlated number of hierarchy elements (#HE), is weaker. Therefore, depending on the increase or decrease of #DV and #HE, the run-time of the privacy model, i.e. *k-Anonymity*, is increased or decreased (see Section 7.2.3.4).

In fact, with the usage of the *MA*-algorithm, the percentage of records with personal privacy settings (PP) in the data-set and the number of attributes used for personal privacy settings (#Attr) affects the run-time of privacy models. The run-time of the privacy model can be either increased or decreased, with a tendency for an increase of the run-time for bigger data-sets. The remaining processes of PD are not significantly affected by either PP or #Attr.

Table 7.7: Overall run-time of the PD process using the *Minimum Anonymization*-algorithm for varying number of number of attributes used for personal privacy settings (#Attr). The proportional number of records with personal privacy settings (PP) is fixed to ′0.4′.

Overall Run-Time [s]		**IHIS Data-set Size**		
		10,000	*100,000*	*1,000,000*
#Attr	*0*	0.61	6.96	97.67
	1	0.63	7.37	107.76
	2	0.55	7.26	110.35
	3	0.53	7.09	110.73
	4	0.56	7.04	113.00
	5	0.52	7.06	114.22
	6	0.52	6.92	115.89
	7	0.54	6.98	115.78
	8	0.53	6.91	117.47
	9	0.53	7.06	116.39

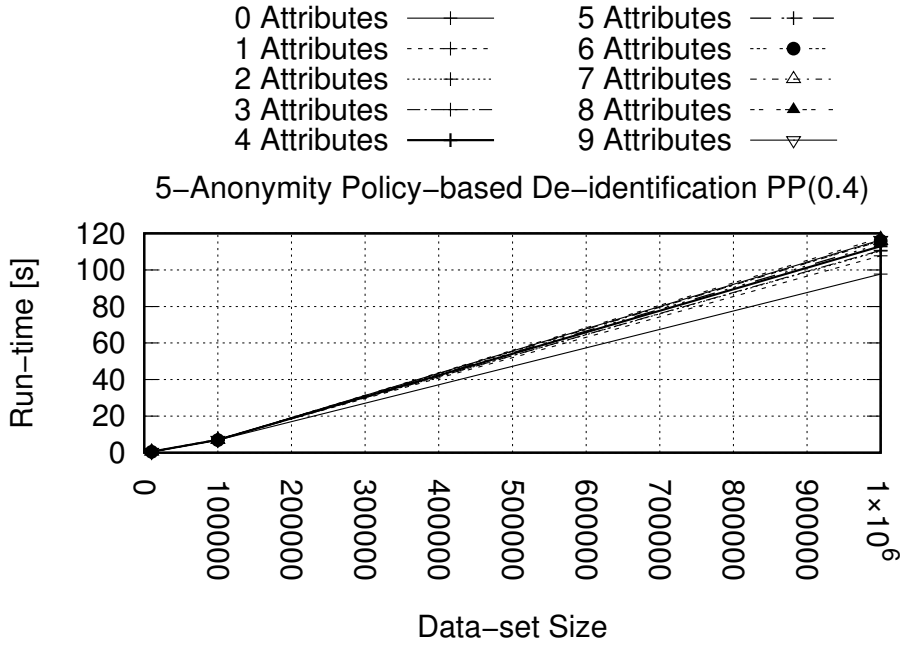


Figure 7.6: Line graph showing the overall run-time in seconds for different numbers of attributes with personal privacy using the *Minimum Anonymization*-algorithm. *5-Anonymity* is used on different data-set sizes of *IHIS*. The number of records with personal privacy (PP) ′0.4′ is constant.

7.2.3 *Personal Privacy – Global Minimum Anonymization*

In the previous experiment the impact of personal privacy on the run-time is evaluated using the *MA*-algorithm. In contrast, within this experiment the impact of personal privacy on the run-time is evaluated using the *GMA*-algorithm for *Personal Privacy Anonymization*. This algorithm is intended to improve the run-time of privacy models, if personal privacy settings are used within the processed data-set. Therefore, the experiment varies both the number of records with personal privacy settings (PP) as well as the number of attributes with personal privacy settings (#Attr).

7.2.3.1 *Experiment Set-up*

The experiment is conducted with the same configuration as before for the *MA*-algorithm in Section 7.2.2, except that the *GMA*-algorithm is used for *Personal Privacy Anonymization*. The run-time of the PD process, as well as each sub-process is measured. This results in 331 different configurations. In the following, the results are first detailed and then compared to the results using the *MA*-algorithm.

7.2.3.2 *Variation of Personal Privacy Percentage*

For the evaluation of the influence of the number of records in the data-set with personal privacy settings (varying from the default privacy policy) the results of the experiment using all attributes for personal privacy ('9') are used. The proportion of records with and without personal privacy settings is hereby varied for comparison (see Table 7.8). These results show a linear growth of the run-time regarding the data-set size, which is expected based on previous experiments. Comparing the baseline measurements (PP '0.0') for 10,000, 100,000, 1,000,000 records with the respective results of the remaining measurements, it can be observed that for each configuration the overall run-time is decreased significantly. For example, the baseline measurement (PP '0.0') for 1,000,0000 records has a run-time of 99.24 seconds. If records with personal privacy settings are introduced then the overall run-time is altered, e.g., if 40% percent of the the records use personal privacy settings (PP '0.4') then 13.95 seconds are required for the overall run-time, which is a decrease of 85.29 seconds.

It has to be noted that the baseline measurements slightly vary from the baseline measurements of the experiment using the *MA*-algorithm, which can be traced back to measurement errors. Therefore, the introduction of records with personal privacy settings in the data-set using the *GMA*-algorithm has a significant impact on the run-time (see Figure 7.7).

Table 7.8: Overall run-time of the PD process using the *Global Minimum Anonymization*-algorithm for varying number of records with personal privacy settings (PP). The number of attributes used for personal privacy settings is fixed to '9'.

Overall		**IHIS Data-set Size**		
Run-Time [s]		*10,000*	*100,000*	*1,000,000*
PP	*0.0*	0.61	7.11	99.24
	0.1	0.16	1.58	13.82
	0.2	0.18	1.68	14.86
	0.3	0.18	1.64	13.98
	0.4	0.17	1.56	13.95
	0.5	0.17	1.58	14.14
	0.6	0.15	1.56	14.13
	0.7	0.16	1.57	13.93
	0.8	0.15	1.73	15.49
	0.9	0.17	1.81	15.94
	1.0	0.19	1.55	13.91

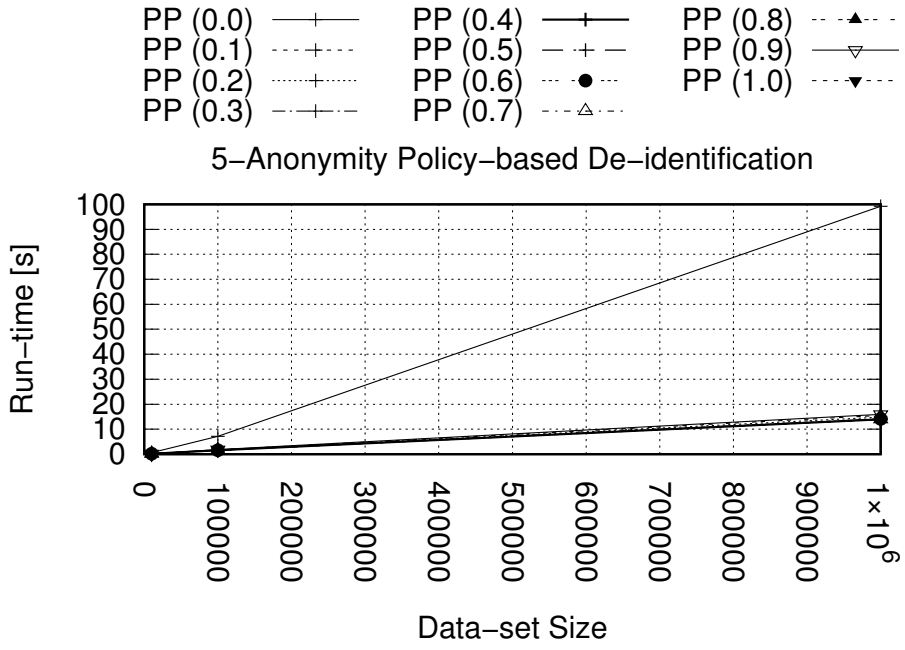


Figure 7.7: Line graph showing the overall run-time in seconds for different number of records with personal privacy (PP) on all attributes '9' using the *Global Minimum Anonymization*-algorithm. *5-Anonymity* is is on different data-set sizes of *IHIS*.

A closer inspection of the individual run-time measurements of the sub-processes reveals that the run-time of the *GMA*-algorithm is greater than the run-time of the *MA*-algorithm, e.g., 0.52 seconds

compared to 0.31 seconds for PP of '0.3' (see Table 7.9). This can be traced back to the additional iteration required on the data-set to determine the maximum of the *Minimum Anonymization Level* (see Section 6.3.2).

Furthermore, it can be observed that the run-time of *Apply Privacy Models* is significantly reduced from 5.81 seconds of the baseline, e.g., 0.16 seconds for PP of '0.3'. The run-time of the privacy model, i.e. *5-Anonymity*, is affected by the introduction of records with personal privacy settings using the *GMA*-algorithm for PD. It can be observed that the variation of PP between '0.1' and '1.0' does vary the run-time of the privacy model (PM). This variation can be caused by different data-set instances of *IHIS* and personal privacy settings, which can affect the #DV and #HE in the data-set, thus varying the run-time of the privacy model. Moreover, measurement errors and a high deviation in the run-time for privacy models can cause this variation.

In general, the reduction of the run-time is caused by the *GMA*-algorithm, which reduces the number of #DV and #HE in the data-set before the privacy model is used, thus decreasing the run-time of the privacy model (see Section 7.2.3.4). Moreover, the run-time difference within the range of PP '0.1' to '1.0' is small, because the *GMA*-algorithm determines a global anonymization level for each attribute according which each value is anonymized. Therefore, it is expected that attributes are anonymized to similar levels among different configurations, thus a low difference in the run-time can be observed. The baseline configuration with PP '0.0' is unaffected, because no personal privacy settings are given, thus a big run-time difference to the remaining configurations can be observed.

Thus, it can be stated that the main influencing factor of the PD process, i.e. *Apply Privacy Models*, has a significantly reduced run-time if records with personal privacy settings are present within the data-set and the *GMA*-algorithm is used, because the number of #DV and #HE is decreased by the *GMA*-algorithm.

Table 7.9: Detailed run-time measurements for each individual process of PD using *5-Anonymity* on the *IHIS* data-set with 100,000 records while varying the number of records with personal privacy settings PP. Labels have been shortened according to following scheme: (*GMA*: Global Minimum Anonymization, *Max*: Set Maximum Anonymization, *Group*: Set Privacy Group, *Sub*: Privacy Model Substitution, *PM*: Apply Privacy Models.

Configuration				*Policy-based De-identification Process*				
Privacy Model	*Data-set Size*	*PP*	*#Attr*	*GMA*	*Max*	*Group*	*Sub*	*PM*
5-Anonymity	100,000	0.0	9	0.41	0.00	0.00	0.09	5.81
5-Anonymity	100,000	0.1	9	0.48	0.00	0.00	0.10	0.15
5-Anonymity	100,000	0.2	9	0.50	0.00	0.00	0.09	0.18
5-Anonymity	100,000	0.3	9	0.52	0.00	0.00	0.10	0.16
5-Anonymity	100,000	0.4	9	0.47	0.00	0.00	0.09	0.16
5-Anonymity	100,000	0.5	9	0.45	0.00	0.00	0.10	0.16
5-Anonymity	100,000	0.6	9	0.48	0.00	0.00	0.10	0.16
5-Anonymity	100,000	0.7	9	0.48	0.00	0.00	0.09	0.17
5-Anonymity	100,000	0.8	9	0.54	0.00	0.00	0.10	0.20
5-Anonymity	100,000	0.9	9	0.56	0.00	0.00	0.09	0.20
5-Anonymity	100,000	1.0	9	0.46	0.00	0.00	0.10	0.14

7.2.3.3 *Variation of the Number of Personal Privacy Attributes*

For the evaluation of the influence of the number of attributes with personal privacy in the data-set, #Attr and PP is varied. For comparison, the influence of #Attr is discussed based on the sub-set with a PP of '0.4' (see Table 7.10). The discussion of other configuration sets, e.g., PP of '0.6', is omitted because it leads to the same conclusions.

Table 7.10: Overall run-time of the PD process using the *Minimum Anonymization*-algorithm for varying number of number of attributes used for personal privacy settings (#Attr). The proportional number of records with personal privacy settings (PP) is fixed to '0.4'.

Overall Run-Time [s]		**IHIS Data-set Size**		
		10,000	*100,000*	*1,000,000*
#Attr	*0*	0.64	7.34	101.06
	1	0.38	4.31	49.16
	2	0.29	3.30	33.63
	3	0.24	2.52	26.97
	4	0.18	2.02	17.89
	5	0.19	1.72	15.13
	6	0.17	1.64	14.16
	7	0.18	1.56	14.02
	8	0.15	1.56	13.91
	9	0.17	1.56	13.95

The baseline measurements with '0' attributes used for personal privacy settings indicate a linear growth of the overall run-time reassuring previous results. Comparing the baseline measurements (#Attr '0') with the remaining measurements, it can be observed that if more attributes with personal privacy settings are introduced then the overall run-time is decreased. For example, if '5' attributes of the record are used for personal privacy settings then 15.13 seconds are required for the overall run-time compared to the baseline of 101.06 seconds for 1,000,000 records. Furthermore, it can be observed that the run-time is increased for each configuration (#Attr '1' to '9') for each data-set size (see Figure 7.8).

In a detailed inspection of the individual PD process run-times, it can also be seen that the run-time of the privacy model, i.e. *5-Anonymity*, is significantly affected by the variation of #Attr confirming the rationale for PP.

With the variation of #Attr, a sub-set of attributes is anonymized during *GMA*. Thus, the effect of altered properties of the data-set, i.e. number of distinct values (#DV) and the correlated number of hierarchy elements (#HE), increases for each additional attribute with

personal privacy settings. In other words, the higher the number of attributes with personal privacy settings #Attr the lower the run-time for the privacy model (PM). This can be observed in Table 7.10, except for #Attr '8' and '9' for which the run-time of '9' is slightly higher. This can be caused by differing sub-sets of the data-set or measuring errors.

Thus, the usage of the *GMA*-algorithm, the introduction of records with personal privacy settings (PP or #Attr) decreases the run-time of the privacy models. The remaining processes of PD are not significantly affected by either PP or #Attr.

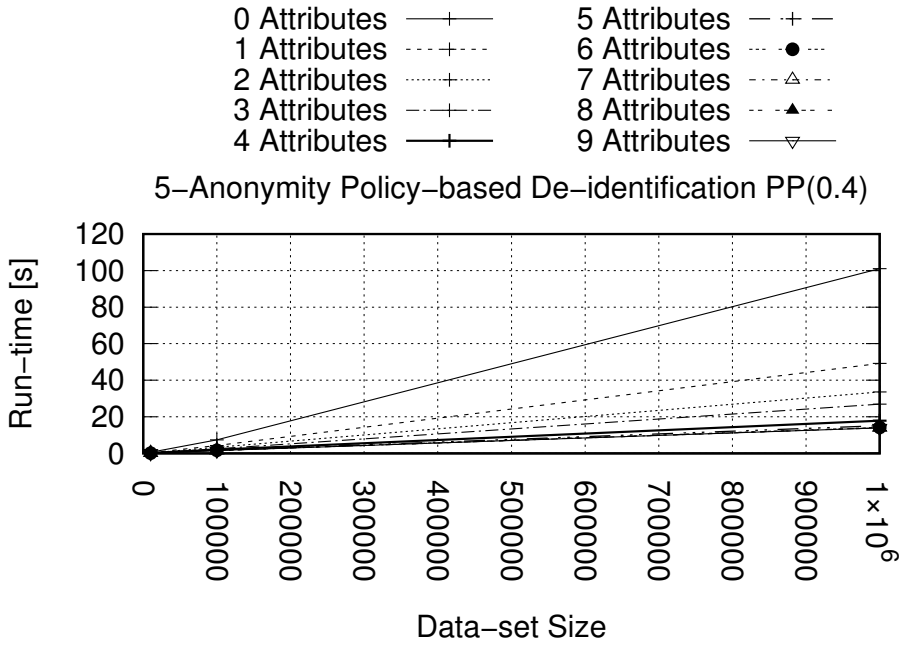


Figure 7.8: Line graph showing the overall run-time in seconds for different numbers of attributes with personal privacy using the *Global Minimum Anonymization*-algorithm. *5-Anonymity* is used on different data-set sizes of *IHIS*. The number of records with personal privacy (PP) is set to '0.4'.

7.2.3.4 *Personal Privacy Anonymization Algorithm Comparison*

In the following, the run-time difference of the *Minimum Anonymization*-algorithm and *Global Minimum Anonymization*-algorithm is detailed. It has already been shown that the variation of PP and #Attr has a minor impact on the run-time of the *GMA*-algorithm. This discussion focuses on the differentiation of the run-time difference for small and big data-set sizes of 10,000 and 1,000,000 records, while PP is fixed to '0.4' and #Attr varies from '0' to '9'.

For the run-time difference calculation, the run-time of the PD process with *MA* ($r_{PD\text{-}MA}$) and PD process with *GMA* ($r_{PD\text{-}GMA}$) is

used. The time difference (see Equation 7.7) and percentage difference (see Equation 7.8) is used for the following comparison.

$$\Delta_{\text{time-difference}} = r_{\text{PD-MA}} - r_{\text{PD-GMA}} \tag{7.7}$$

$$\Delta_{\text{percent-difference}} = \left(\frac{r_{\text{PD-MA}} - r_{\text{PD-GMA}}}{r_{\text{PD-MA}}}\right) * 100 \tag{7.8}$$

Considering the smaller data-set with 10,000 records (see Table 7.11), it can be observed that the baseline run-time (#Attr '0') varies by 0.03 seconds, which is a $\Delta_{\text{percent-difference}}$ of 4.92%. Similarly, the baseline run-time varies for the data-set with 1,000,000 records for 3.47% (see Table 7.13). On the one hand, this difference is introduced by the additionally required run-time of the *GMA*-algorithm. On the other hand, this difference is based on measuring errors.

The $\Delta_{\text{time-difference}}$ for the small data-set with 10,000 records indicates a better run-time for the *GMA*-algorithm if records with personal privacy settings are present within the data-set. The run-time is reduced on average, excluding the baseline measurements, by 0.33 seconds with 0.25 seconds as the minimal run-time difference and 0.38 seconds as the maximal run-time difference. This corresponds to a $\Delta_{\text{percent-difference}}$ between 39.68% and 71.70%, which averages to 60.73%.

Similar results can be observed for the data-set with 100,000 records (see Table 7.13). The $\Delta_{\text{time-difference}}$ indicates a better run-time for the *GMA*-algorithm if records with personal privacy settings are present within the data-set. The run-time is reduced on average, excluding the baseline measurements, by 91.42 seconds with 58.60 seconds as the minimal run-time difference and 103.56 seconds as the maximal run-time difference. This corresponds to a $\Delta_{\text{percent-difference}}$ between 54.38% and 88.16%, which averages to 80,26%.

Considering the data-sets with 10,000 and 1,000,000 records, the *GMA*-algorithm improves the run-time of the PD process significantly. The rationale for this improvement is the reduction of the number of distinct values and anonymization hierarchies during the *Personal Privacy Anonymization*. The reduced volume of inputs for the privacy models, also reduces their run-time. For example *k-Anonymity* requires the calculation of QI-groups, which are based upon the distinct values and anonymization hierarchies. Therefore, if fewer distinct values are available also fewer QI-groups have to be considered.

This can be observed by relating the number of distinct values (#DV) and number of distinct anonymization hierarchy elements (#HE) of the data-set before it is anonymized by the privacy model (Pre-PM Data-set) to $\Delta_{\text{percent-difference}}$ (see Table 7.12 and Table 7.14). The #DV and #HE are measured for each configuration for the input data-set as well as the data-set before it is anonymized to comply with the privacy model (Pre-PM Data-set). The baseline configuration has no personal privacy settings defined for the LPL privacy policies (#Attr is set to

'0'), thus the *MA*-algorithm does not affect the data-set. Considering the remaining configurations, it can be observed that the number of distinct values (#DV) is reduced. The percentage reduction of #DV ($\Delta_{\#DV_{\#Attr}}$) is calculated on the baseline #DV in relation to the #DV of each configuration (see Equation 7.9). Considering $\Delta_{\#DV_{\#Attr}}$, the #DV is reduced up to 59.49% for the data-set size of 10,000 (see Table 7.12) and up to 56.18% for the data-set size of 1,000,000 (see Table 7.14).

$$\Delta_{\#DV_{\#Attr}} = \left(\frac{\#DV_0 - \#DV_{\#Attr}}{\#DV_0}\right) * 100 \tag{7.9}$$

Furthermore, it can be observed that the number of hierarchy elements (#HE) is reduced. The percentage of the #HE reduction ($\Delta_{\#HE_{\#Attr}}$) is calculated on the baseline #HE in relation to the #HE of each configuration (see Equation 7.10). Considering $\Delta_{\#HE_{\#Attr}}$, the #HE is reduced up to 80.06% for the data-set size of 10,000 (see Table 7.12) and up to 80.03% for the data-set size of 1,000,000 (see Table 7.14).

$$\Delta_{\#HE_{\#Attr}} = \left(\frac{\#HE_0 - \#HE_{\#Attr}}{\#HE_0}\right) * 100 \tag{7.10}$$

For both data-set sizes, it can be observed that an increase of $\Delta_{\#DV_{\#Attr}}$ or $\Delta_{\#HE_{\#Attr}}$ correlates to an increased $\Delta_{\text{percent-difference}}$ (see Table 7.12 and Table 7.14).

Thus, a significant reduction of the overall run-time can be achieved using the *GMA*-algorithm if records with personal privacy settings are present. In contrast, the *MA*-algorithm potentially increases the overall run-time of PD, because the number of distinct values (#DV), i.e. anonymized values, are added to the data-set as well as additional anonymization hierarchies (increased number of distinct hierarchy elements #HE).

Table 7.11: Run-time difference of the *Policy-based De-identification* (PD) comparing the *Minimum Anonymization (MA)*-algorithm with the *Global Minimum Anonymization (GMA)*-algorithm. The time difference $\Delta_{\texttt{time-difference}}$ and percentage difference $\Delta_{\texttt{percent-difference}}$ is calculated. The *IHIS* data-set with 10,000 records is used. 5-*Anonymity* is used as the privacy model.

Data-set Size	*PP*	*#Attr*	*PD-MA in [s]*	*PD-GMA in [s]*	$\Delta_{\texttt{time-difference}}$	$\Delta_{\texttt{percent-difference}}$
10,000	0.4	0	0.61	0.64	-0.03	-4.92
10,000	0.4	1	0.63	0.38	0.25	39.68
10,000	0.4	2	0.55	0.29	0.26	47.27
10,000	0.4	3	0.53	0.24	0.29	54.72
10,000	0.4	4	0.56	0.18	0.38	67.86
10,000	0.4	5	0.52	0.19	0.33	63.46
10,000	0.4	6	0.52	0.17	0.35	67.31
10,000	0.4	7	0.54	0.18	0.36	66.67
10,000	0.4	8	0.53	0.15	0.38	71.70
10,000	0.4	9	0.53	0.17	0.36	67.92

Table 7.12: Measurement of distinct values #DV and number of anonymization hierarchy elements #HE for the processed data-set at the beginning of the PD process (Input Data-set) and before *Apply Privacy Models* (Pre-PM Data-set). $\Delta_{\#DV_{\#Attr}}$ and $\Delta_{\#HE_{\#Attr}}$ are put into relation to $\Delta_{\text{percent-difference}}$ is calculated. The *IHIS* data-set with 10,000 records is used. *GMA* is used for *Personal Privacy Anonymization* and*5-Anonymity* is used as the privacy model.

Data-set Size	*PP*	*#Attr*	*Input Data-set*		*Pre-PM Data-Set*		$\Delta_{\#DV_{\#Attr}}$	$\Delta_{\#HE_{\#Attr}}$	$\Delta_{\text{percent-difference}}$
			#DV	*#HE*	*#DV*	*#HE*			
10,000	0.4	0	158	636	158 (+/- 0)	697 (+/- 0)	0.00	0.00	-4.92
10,000	0.4	1	158	638	147 (- 11)	625 (- 72)	6.96	10.30	39.68
10,000	0.4	2	159	639	146 (- 12)	615 (- 82)	7.59	11.76	47.27
10,000	0.4	3	159	639	144 (- 14)	603 (- 94)	8.86	13.49	54.72
10,000	0.4	4	157	635	146 (- 12)	563 (- 134)	7.59	19.23	67.86
10,000	0.4	5	159	640	72 (- 86)	166 (- 531)	54.43	76.18	63.46
10,000	0.4	6	159	640	66 (- 92)	143 (- 554)	58.23	79.48	67.31
10,000	0.4	7	158	636	65 (- 93)	141 (- 556)	58.86	79.77	66.67
10,000	0.4	8	158	638	64 (- 94)	139 (- 558)	59.49	80.06	71.70
10,000	0.4	9	159	639	66 (- 92)	143 (- 554)	58.23	79.48	67.92

Table 7.13: Run-time difference of the *Policy-based De-identification* (PD) comparing the *Minimum Anonymization (MA)*-algorithm with the *Global Minimum Anonymization (GMA)*-algorithm. The time difference $\Delta_{\texttt{time-difference}}$ and percentage difference $\Delta_{\texttt{percent-difference}}$ is calculated. The *IHIS* data-set with 1,000,000 records is used. *5-Anonymity* is used as the privacy model.

Data-set Size	*PP*	*#Attr*	*PD-MA in [s]*	*PD-GMA in [s]*	$\Delta_{\texttt{time-difference}}$	$\Delta_{\texttt{percent-difference}}$
1,000,000	0.4	0	97.67	101.06	-3.39	-3.47
1,000,000	0.4	1	107.76	49.16	58.60	54.38
1,000,000	0.4	2	110.35	33.63	76.72	69.52
1,000,000	0.4	3	110.73	26.97	83.76	75.64
1,000,000	0.4	4	113.0	17.89	95.11	84.17
1,000,000	0.4	5	114.22	15.13	99.09	86.75
1,000,000	0.4	6	115.89	14.16	101.73	87.78
1,000,000	0.4	7	115.78	14.02	101.76	87.89
1,000,000	0.4	8	117.47	13.91	103.56	88.16
1,000,000	0.4	9	116.39	13.95	102.44	88.01

Table 7.14: Measurement of distinct values #DV and number of anonymization hierarchy elements #HE for the processed data-set at the beginning of the PD process (Input Data-set) and before *Apply Privacy Models* (Pre-PM Data-set). $\Delta_{\#DV_{\#Attr}}$ and $\Delta_{\#HE_{\#Attr}}$ are put into relation to $\Delta_{percent\text{-}difference}$ is calculated. The *IHIS* data-set with 1,000,000 records is used. *GMA* is used for *Personal Privacy Anonymization* and5-*Anonymity* is used as the privacy model.

Data-set Size	*PP*	*#Attr*	*Input Data-set*		*Pre-PM Data-Set*		$\Delta_{\#DV_{\#Attr}}$	$\Delta_{\#HE_{\#Attr}}$	$\Delta_{percent\text{-}difference}$
			#DV	*#HE*	*#DV*	*#HE*			
1,000,000	0.4	0	160	641	160 (+/- 0)	746 (+/- 0)	0.00	0.00	-3.47
1,000,000	0.4	1	160	642	149 (- 11)	674 (- 72)	6.88	9.65	54.38
1,000,000	0.4	2	160	641	147 (- 13)	666 (- 80)	8.13	10.72	69.52
1,000,000	0.4	3	160	642	145 (- 15)	662 (- 84)	9.38	11.26	75.64
1,000,000	0.4	4	160	640	152 (- 8)	576 (- 170)	5.00	22.79	84.17
1,000,000	0.4	5	160	641	75 (- 85)	173 (- 573)	53.13	76.81	86.75
1,000,000	0.4	6	160	640	69 (- 91)	149 (- 597)	56.88	80.03	87.78
1,000,000	0.4	7	160	642	69 (- 91)	149 (- 597)	56.88	80.03	87.89
1,000,000	0.4	8	160	644	69 (- 91)	149 (- 597)	56.88	80.03	88.16
1,000,000	0.4	9	160	643	69 (- 91)	149 (- 597)	56.88	80.03	88.01

7.3 CONCLUSION

To evaluate the efficiency of the *Policy-based De-identification* process, the *Policy-based De-identification Benchmark Framework* is proposed in this chapter and used for three different experiments. The PD process considers for each record a corresponding LPL privacy policy (see Chapter 6).

First, the run-time overhead introduced by the PD processes compared to the sole application of privacy models is evaluated. For this experiment no personal privacy settings are introduced. The detailed investigation of the run-time for each PD sub-process shows that they have only a minor impact on the overall run-time, except *Apply Privacy Models* which has the highest impact. Therefore, it can be stated that the PD sub-processes are efficient. Furthermore, it is concluded that the PD approach is suitable for data-warehouse scenarios (millions or billions of data records) as the percentage Δ_{percent} run-time decreases with the size of the data-set and run-time intensive privacy models. For example, for the data-set *IHIS* (1,193,504 records) and *(00.5, 0.15)-Presence* the run-time overhead is only 1.00%.

Next, the impact of personal privacy settings on the run-time of the PD process is evaluated using the *MA*-algorithm for *Personal Privacy Anonymization*. The variation of the proportional number of records with personal privacy settings (PP) and number of attributes with personal privacy settings (#Attr) is investigated using different numbers of records of *IHIS*. The results show that the run-time of *Apply Privacy Models* sub-process is mainly influenced by the introduction of personal privacy settings (PP or #Attr). The results show that the run-time can be either increased or decreased, while the results of the big data-set with 1,000,000 records consistently has increased run-times (see Table 7.5). An in detail inspection revealed, that the introduction of personal privacy settings alters the properties of the data-set after the *MA*, such that #DV and #HE is increased or decreased and as a consequence the run-time of the privacy model is respectively increased or decreased. The remaining sub-processes are not significantly affected by the introduction of personal privacy settings (see Table 7.6).

Lastly, the experiment has been repeated with the *GMA*-algorithm and compared to the results using the *MA*-algorithm. The results show that the *Personal Privacy Anonymization* has a slightly increased run-time and a significantly decreased run-time for the privacy models if personal privacy settings (PP or #Attr) are introduced (see Tables 7.8 and 7.9). The increased run-time for the *GMA*-algorithm can be traced back to the additional iteration required for the determination of the maximum *Minimum Anonymization Level* for each attribute (see Section 6.3.2). The significantly decreased run-time for *Apply Privacy Models* is caused by the reduction of the number of distinct values #DV and number of hierarchy elements #HE processed.

The direct comparison of the results for the PD process with *MA* and the PD process with *GMA* showed, that the usage of *GMA* decreases the run-time significantly, e.g., up to 88.16% for the presented results (see Tables 7.11 and 7.13). Moreover, the correlation of a decreased #DV and #HE to a decreased privacy model run-time, i.e. *k-Anonymity*, is demonstrated (see Tables 7.12 and 7.14). But this improvement of the run-time comes with the cost of possible loss of utility, because the anonymization of all values to the maximum *Minimum Anonymization Level* anonymizes some values more than required (see Section 6.3.2). Thus, valuable data may be lost depending on the specific use case.

To answer the second research question *RQ2*, the *Policy-based De-identification* process was introduced to preserve the privacy of individuals considering pseudonymization, personal privacy settings, i.e. *Personal Privacy Anonymization*, and privacy models which are expressed via LPL (see Chapter 6). Furthermore, the evaluation demonstrated that the PD processes are efficient considering the run-times of the individual processes in comparison to the run-time of privacy models. Additionally, the impact of personal privacy on the run-time has been evaluated comparing the *Minimum Anonymization*- and *Global Minimum Anonymization*-algorithm. Although both algorithms preserve the privacy according to the individuals requirements, the run-time and utility properties differ. On the one hand, the *MA*-algorithm only anonymizes data according to the *Minimum Anonymization Levels* defined by the corresponding LPL privacy policy, but the run-time of the privacy model may be increased or decreased. The increase/decrease of the run-time is correlated to a increase/decrease of #DV and #HE in the data-set after *MA*, because some values are anonymized altering the number distinct values in the data-set. On the other hand, the *GMA*-algorithm anonymizes all values of an attribute to the same anonymization level, i.e. the maximum of the *Minimum Anonymization Levels* for each attribute, hence #DV and #HE is decreased after *GMA*. This significantly decreases the run-time of the privacy model. On the downside, the *MA*-algorithm may decrease the utility of the resulting data-set. Thus, a trade-off between run-time efficiency and utility has to be made depending on the use case.

8

CONCLUSION AND FUTURE WORK

This chapter concludes the work of this thesis and gives and outlook for future works.

8.1 CONCLUSION

The goal of this thesis is to formalize privacy policies in machine-readable format integrating legal requirements of the *GDPR* and privacy-preserving technologies. Furthermore, an efficient enforcement of policy-based and user-guided processing of personal data has to be enabled.

To achieve this goal the research questions *RQ1* and *RQ2* are formalized and answered. To answer the first research question *RQ1* – 'How to represent legal privacy policies in a machine-readable format which complies to the legal requirements of the General Data Protection Regulation in the EU while privacy guarantees are defined?' – a set of requirements is derived considering the legal and technical point of view on privacy (see Chapter 2):

- *R1 Privacy Policy Structure*
- *R2 Legal Compliance* (to *GDPR*)
- *R3 Human-readability*
- *R4 Access Control*
- *R5 De-identification Capabilities*
- *R6 Provenance*

Background information on the variety of de-identification methods that are considered is given in Chapter 3. An extensive literature research has been conducted comparing and classifying related privacy languages according to this set of requirements for a privacy language expressing privacy policies (see Chapter 4). The literature research revealed, that several privacy languages have been proposed each with a distinct focus objective. Most of the surveyed privacy languages have a *R1 Privacy Policy Structure* and focus on the realization of privacy

guarantees using *R4 Access Control*. The remaining requirements are only partially fulfilled by other privacy languages. *R2 Legal Compliance* to *GDPR* is considered by several privacy languages, but the explicit integration of the requirements for a privacy policy (Art. 12 - 14 *GDPR*) or *Data Subject Rights* (Art. 12 - 23 *GDPR*) in a privacy language is not realized. *R3 Human-readability* is essential for transparency, but is often decoupled from the privacy-preserving rules. Only a few privacy languages in the literature provide or consider mechanisms for *R6 Provenance*. Lastly, expressing *R5 De-identification Capabilities*, i.e. pseudonymization, anonymization and privacy models, and their realization have not been the main focus of any privacy language in the literature. Therefore, a research gap is identified for a privacy language expressing privacy policies and using de-identification methodology to realize privacy-preserving processing rules while all given requirements are fulfilled.

To answer research question *RQ1* the *Layered Privacy Language (LPL)* is proposed for which each of the requirements *R1*, *R2*, *R3*, *R4*, *R5*, and *R6* are fulfilled. The core structure of LPL is modelled in such a way to integrate the *R1 Privacy Policy Structure* requirement. Therefore, LPL consists of a set of purposes that furthermore define which attributes are processed by which data recipients. Additionally, the data source is denoted.

The *R2 Legal Compliance* requirement is realized by a broad legal requirement analysis based on the *GDPR* from which various elements are derived which have to be modelled within LPL. For example, the *Controller* and *Data Protection Officer* have to be expressed by the privacy policy to inform the user about the responsible authorities. Furthermore, requirements of the *Data Subject Rights* are considered, such that the fulfilment of *Data Subject Right* requests can be eased based on the information available in LPL privacy policies. The fulfilment of the *R2 Legal Compliance* requirement is shown via a detailed qualitative evaluation of each legal requirement.

The *R3 Human-readability* requirement is essential for a privacy language such that the contents of the policy can be transparently communicated to the user, which is also essential for *R2 Legal Compliance*. This *R3 Human-readability* requirement is considered within LPL via the addition of human-readable headers and descriptions for elements as well as the introduction of a distinct element for expressing *Privacy Icons*. The fulfilment of the requirement is shown via a proof-of-concept implementation of user interfaces allowing the creation, presentation, and negotiation of LPL privacy policies. Hereby, also the *R2 Legal Compliance* requirement has been considered, such that all required information is presented using a *Layered* approach.

The *R4 Access Control* requirement is realized in LPL by enabling entities, e.g., the processing entities, to be authenticated and authorized. Furthermore, it is assumed that a request for data also denotes

the purpose, attributes and individuals for which personal data is processed. Therefore, the LPL privacy policy acts as a rule-set against which a request is validated. The fulfilment of the *R4 Access Control* requirement is discussed according to the introduced *Policy-based Access Control (PAC)* process for which the pseudocode is detailed.

The *R5 De-identification Capabilities* requirement is realized by several elements in LPL. Therefore, LPL can express for each purpose a set of privacy models and pseudonymization methods that have to be applied on the data-set if requested. Furthermore, for each attribute of the purpose a *Minimum Anonymization Level* is defined, which can be altered by the user to define in which quality his personal data will be used for the purpose. Moreover, the creator of the LPL privacy defines the *Maximum Anonymization Level*, which defines the limit for the anonymization, thus utility requirements are defined. This extensive definition of de-identification requirements is unique to LPL and allows a fine-grained definition of privacy and utility requirements. The fulfilment of the *R5 De-identification Capabilities* requirement is shown by the introduction of the *Policy-based De-identification (PD)* process, which utilizes LPL privacy policies to de-identify on-the-fly requested data-sets. The combination of the PAC and PD process enables a purpose-based restricted access to personal data, which is furthermore de-identified if required.

The *R6 Provenance* requirement is realized via the introduction of an identifiable data source, i.e. the user, to the LPL privacy policy, which is also subject to the *R4 Access Control* requirement. Therefore, the source of the personal data can be identified. Furthermore, a LPL privacy policy is designed to reference other LPL privacy policies. This enables the refinement of privacy policies, e.g., more fine-grained access control or de-identification requirements, for the sharing of personal data with third parties. On the one hand, this allows the validation of an LPL privacy policy against an underlying LPL privacy policy, thus proof for compliance to the *GDPR* can be given. On the other hand, this allows to trace back the source of personal data, e.g., allowing users to make use of their *Data Subject Rights* for data that is transferred to third parties. Lastly, the processing of personal data produces also new data, e.g., survey results, for which it can be crucial to trace back the original data sources to demonstrate that the original data has not been tampered with. This scenario can also be expressed with the underlying privacy policies. The fulfilment of the *R6 Provenance* requirement is discussed according to use case scenarios.

Therefore, it was shown that all requirements for a privacy language expressing privacy policies are integrated in LPL and their application discussed. The first research question *RQ1* is answered with the *Layered Privacy Language (LPL)*, for which compliance to legal requirements of

the *GDPR* is shown. This enables the expression of privacy guarantees via access control and de-identification rules.

Considering the big picture, privacy policies are intended to transparently inform users about the processing of their personal data and their rights. Thus, users can freely agree and consent to the processing of their personal data. The personal privacy concept denotes that users can influence, i.e. negotiate, the processing of their personal data, thus for each user a personal privacy policy is defined. During the processing of personal data the corresponding individuals' personal privacy policies have to be considered distinctively such that each user's privacy is preserved. Similarly, when personal data is transferred, e.g., sold or shared, to third parties, the sticky policy concept is considered such that the personal data record is always linked to the corresponding privacy policy. Besides, the entity, i.e. the company, that issues the privacy policy can express it's requirements that have to be considered during the de-identification process for a specific purpose. These utility requirements, i.e. how much a value can be anonymized to be still useful, are defined within the policy and have to realized during the de-identification process.

Considering a data-warehouse, containing personal data from various sources, the privacy requirements for processing the personal data for a specific purpose have to be derived from all corresponding privacy policies. Because users can have personalized privacy policies and privacy policy can origin from various sources, different privacy and utility requirements are possible for the de-identification process. Additionally, the de-identification process has to be conducted efficiently for each request, because the privacy policies of some records can be altered at any time, i.e. withdrawal of consent by the user. These additional challenges are subject to the second research question *RQ2* – 'How can machine-readable privacy policies, expressing privacy-preserving methods, be utilized to efficiently preserve the privacy of individuals when a set of users' personal data is requested for processing?' – which can be split into two problem statements. First, the variety of users' personal privacy policies, expressed by a privacy language, has to be considered during the de-identification process, thus complexity has to be handled. Second, the de-identification process has to be efficient in regard to its run-time to be a viable option in practice compared to common de-identification approaches, i.e. use privacy models on a data-set.

To tackle the first challenge, handling varying privacy and utility requirements, is possible thanks to the *Policy-based De-identification*, which is proposed to determine a common or unified level of privacy and utility from varying LPL privacy policies. For example, varying definitions of privacy models are tackled by the introduction of the *Privacy Model Substitution*-algorithm, which determines a set of privacy models that fulfils all privacy requirements of the given pri-

vacy models. Therefore, the *Privacy Model Substitution Table* is used as a basis for the decision-making, which is created using expert knowledge on privacy models. Similarly, mechanisms for determining the set of pseudonymization methods, privacy groups, *Maximum Anonymization Level* and *Minimum Anonymization Level* for attributes are introduced. Furthermore, two alternative algorithms for *Personal Privacy Anonymization* are introduced, for which the *Minimum Anonymization (MA)*-algorithm is intended to exactly realize the privacy requirements of each individual. In contrast, the *Global Minimum Anonymization (GMA)*-algorithm determines a unified anonymization level for each attribute. This reduces the number of distinct values within the data-set which is intended to improve the overall efficiency, i.e. run-time, of the PD process.

This leads to the second challenge, the PD process should be efficient. In other words, the additional processing of possible millions or billions of personalized LPL privacy policies for a data-warehouse scenario should not significantly increase the overall run-time. Therefore, a quantitative evaluation is conducted based on three experiments. First, the overall overhead of the PD processes compared to the sole application of privacy models on a data-set is evaluated, showing an overhead of 5.58% on average. Next, the run-time impact of personal privacy settings on the PD process is evaluated by comparing the usage of the *MA*- and the *GMA*-algorithm. The usage of the *MA*-algorithm with personal privacy settings introduces a variation of the run-time, both an increase and a decrease. The usage of a data-set with the size of 1,000,000 showed only increased run-times. In contrast, the usage of the *GMA*-algorithm with personal privacy settings introduced a significant run-time decrease for all tested configurations of up to 88.16% compared to the baseline. This significant improvement is based on the reduction of distinct values and anonymization hierarchies due to the *GMA*-algorithm, which reduces the run-time of privacy models. But, this improvement comes with a possible reduction of the final utility of the data-set, because the all values of an attribute are anonymized to the same level instead of individual levels. Therefore, outliers in the personal privacy settings can decrease the utility of the whole data-set. Thus, a trade-off between run-time efficiency and utility has to be made considering the use case, i.e. the properties of the data-set and corresponding personal privacy settings.

Therefore, the second research question *RQ2* is answered by the introduction of the *Policy-based De-identification* process, which is able to derive uniform privacy and utility requirements from varying LPL privacy policies and apply them to preserve the privacy of individuals. Furthermore, the quantitative evaluation showed that the PD process only add a run-time overhead of 5.58% compared to the sole application of privacy models with a tendency of a lower overhead for bigger data-sets, thus especially viable for data-warehouse scenarios. It has

been shown that an decrease of distinct values and anonymization hierarchies is related to an decreased run-time of privacy models. To reduce the possible negative impact of personal privacy settings the *GMA*-algorithm is introduced, which enabled a significant run-time reduction if personal privacy settings are used, but potentially reduced the utility of the resulting de-identified data-set.

The research questions *RQ1* and *RQ2* are answered in this thesis by the *Layered Privacy Language (LPL)* and the *Policy-based De-identification (PD)* process using LPL. Thus, the goal of this thesis to formalize privacy policies in machine-readable format integrating legal requirements of the *GDPR* and privacy-preserving technologies and to enable the enforcement of efficient policy-based and user-guided processing of personal data is achieved. In the following, an outlook for future work in the privacy domain is given in the context of LPL and the PD process.

8.2 FUTURE WORK

For future work, three main topics are considered. *Extended Scenarii* details possible extensions and optimizations for LPL and the PD process. *Company Compliance* details the challenges that have to be faced for companies to demonstrate their compliance after the processing of personal data. *User Experience* details user-friendly presentation and negotiation of privacy policies using privacy languages as the basis.

8.2.1 *Extended Scenarii*

The *Layered Privacy Language (LPL)* and the *Policy-based De-identification (PD)* process are discussed according to a data-warehouse scenario using multidimensional data. The extension of this scenario is subject to future work.

In fact, other data-set types could be considered for the de-identification process, e.g., transaction data, longitudinal data, graph data, and time series data. Each of the data types has properties that have to be preserved during the de-identification process. For example, longitudinal data contains several entries for the same user. Therefore, the affiliation of the records to the same entity has to be preserved during de-identification. Considering data-sets that contain the location data of users, specialized privacy models have to be applied to preserve the privacy [109] [178].

Furthermore, the focus on other domains is subject to future work, e.g., IoT [103] or e-Health [56] [138]. For example, devices in the IoT domain have restricted resources for computation, for which LPL may be further developed to fulfil specific storage capacity requirements. Because the PD process is resource intensive, the IoT device on which the PD process is run must be carefully selected. Additionally, it

has to be considered that IoT devices are heterogenous and highly interconnected, thus various privacy policies have to be managed to enable efficient access control and de-identification [155] [103].

Therefore, the extension of LPL and the PD process is subject to future work. The evaluation showed that the individual algorithms of the PD process do not add significant run-time overhead compared to the sole application of privacy models, but the integration of caching mechanisms and parallelization may further decrease the overhead. For example, a cache may be introduced that temporarily stores requests and corresponding de-identified data-sets. If a request is repeated and identified within the cache, the PD process can be skipped and the cached de-identified data-set can be returned. Because the caching of de-identified data-sets requires extra storage capacity for big data-sets, a trade-off between the storage capacity and run-time savings has to be found.

The parallelization of independent algorithms, e.g., *Set Maximum Anonymization* and *Set Privacy Group*, may decrease the run-time of the PD process. Considering the possible worse utility of the PD process with the *GMA*-algorithm compared to the PD process with the *MA*-algorithm, the data-set may be analysed regarding its personal privacy properties, while simultaneously the default PD process with *MA* is run to predict if the *GMA* is advantageous regarding the run-time and utility trade-off. If it is advantageous, the PD process with *GMA* may be started instead.

Therefore, it is essential to quantify the utility for which a suitable metric has to be defined, considering the *Accuracy*, *Completeness* and *Consistency* properties of the data-set [38]. Commonly, only *Accuracy* is considered to quantify the utility, e.g., information loss, but considering the use of personal privacy anonymization and pseudonymization also *Completeness* and *Consistency* of the data-set should be taken into account. For example, the usage of pseudonyms for EI attributes instead of deleting them during the application of the privacy model may be an improvement for the utility considering *Completeness*. Furthermore, in longitudinal data-sets, it is essential that the affiliation of the records to the same entity has to be preserved, which is possible using pseudonymization but may be error prone using anonymization. Therefore, the *Consistency* property of utility is affected. With such an utility metric, the properties of the PD process can be further optimized.

In summary, LPL and the PD process are subject to future works that extend them for use in other domains and with other data-set types. Furthermore, the optimization of the PD process considering performance improvements and detailed analysis of its utility properties is to be considered.

8.2.2 *Company Compliance*

Within this work, the life-cycle of the *Layered Privacy Language (LPL)* incorporated the creation and negotiation of privacy policies, as well as their pre-processing, storage, transfer and usage. Therefore, the life-cycle focused on the *ex ante processing* of personal data. But, there are also challenges that can be faced after the personal data is processed, i.e. *ex post processing* [89]. The processing entity, i.e. company, has to show its compliance to the legal framework – the *GDPR* – if audited by supervisory authorities. Hereby, the compliant privacy-preserving processing of personal data has to be accounted for.

These challenges can be tackled by logging the processing requests and their outcome, i.e. request and outcome of PAC and PD. Therefore, the logs may contain the requesting entity, purpose of the processing, attributes, and requested entities from the request in addition to common attributes, e.g. the time-stamp. These logs can be used as the basis to show compliance to supervisory authorities and to determine the origin of privacy incidents. Approaches for compliance checking and inference detection can be found in the literature [275] [52] [3].

To ensure the validity of the log entries it has to be ensured that they are not manipulated, i.e. log entries must be immutable. Therefore, log entries may be stored in a blockchain preventing unnoticed manipulation [172] [160] [285] [2] [61]. These technologies have to be combined to a holistic approach that considers all aspects of privacy from consent management, privacy-preserving processing, and ex post processing.

Considering that personal data is transferred alongside sticky policies, additional challenges have to be faced considering the compliant processing of personal data. First, it has to be considered that the original policies, i.e. underlying LPL privacy policies, may be altered, updated, or deleted. Therefore, also the related transferred policies have to be altered. Additionally, the validation of a refined policy against its updated original policy has to be repeated considering the distributed definitions of purposes, entities, and data.

Second, it has to be ensured that the transferred policies can be processed by third parties, e.g., the rules defined the *Purpose-Hierarchy* have to be accessible. Otherwise, authentication and authorization, i.e. PAC, may fail. Considering the generic definition of identifiers for elements, it has to be ensured that identifiers are globally unique instead of only within a company. Therefore, it should be strived for a standardization for a privacy vocabulary, e.g., for purposes, data groups and attributes [55].

8.2.3 *User Experience*

Transparency of privacy policies is important for enabling free and voluntary consent of the user. It is legally required to inform the user about the processing of his personal data. But in practice, privacy policies are rarely read by users because they require a high effort to be understood by the users [182]. Therefore, many users accept privacy policies without reading them (carefully) [253]. Additionally, it has been shown that privacy policies are too complex, long and demanding in their formulation [46]. Considering LPL privacy policies, privacy guarantees are detailed using privacy models, pseudonymization and anonymization, which are complex methods by themselves. Therefore, it is safe to assume that users would not understand if it is stated that their personal data is protected with *5-Anonymity* if it is processed for research purposes.

It is possible with LPL privacy policies to give the user various options for personalization. Hereby, the user is given the possibility to decide which data, i.e. attributes, and in what quality the data is used, i.e. the anonymization level. Furthermore, the data recipients for a purpose can be selected by the user. Thus, the user may be offered various personalization options, which enable him to decide about the processing of his personal data. But this variety of options may also overwhelm the user, because he is unable or unwilling to assess the impact of his decisions.

LPL is capable to express the legally required information that has to be provided by privacy policies, detail de-identification methods that protect the users personal data, and enable the user to personalize his privacy settings. But the challenge remains that the information provided by LPL privacy policies has to be refined and displayed such that users are informed about the processing of their personal data and enabled to personalize their privacy settings. This challenge can be tackled by providing users with suitable user interfaces. Within this thesis, proof-of-concept user interfaces for creation, presentation and negotiation of LPL privacy policies are presented, but they are neither finalized nor extensively evaluated. Therefore, extensive research has to be conducted to determine suitable user-friendly interfaces that inform the users comprehensibly about the contents of privacy policies, but should also provide detailed information on the processing of personal data if requested. The usage of standardized *Privacy Icons* is an interesting approach, but the creation, evaluation and standardization of *Privacy Icons* is a challenge by itself.

To enable user-friendly personalization of privacy policies is another challenge. First, the default settings should be privacy-friendly by default (*Privacy by Default* principle). Furthermore, the user has to be informed about the future impact of his decision, both considering the possible restriction of features of the service and possible risks where

the personal data are processed. For example, the user may decide about providing his current location to an online map service. If the user decided against the provision of his current location to the service, then he is not able to access features that provide him nearby points-of-interest (POIs). If the user decides that his current location is provided to the service, then he is able to use the features, but his location will be used to provide him personalized advertising. Therefore, the user is provided with all information to give an informed decision about the processing of his current location.

Although the user may be provided with all required information, he may be overwhelmed by the number of decisions that have to be taken, especially if this process has to be repeated for each service. Therefore, tools for supporting the decision-making for the personalization of privacy policy settings, based on the users preferences are subject to future research. Hereby, the user should be able to express preferences on the usage of his personal data, e.g., personal data should never be published, which are then matched against the services' privacy policy. The usage of privacy languages for expressing privacy policies, i.e. LPL, and privacy preferences seems advantageous, because the machine-readable formats can be automatically matched to determine if the privacy policy complies with the users' preferences. Furthermore, personal privacy settings may be altered on-demand to comply to the privacy preferences of the user. Therefore, a suitable privacy language for expressing users' privacy preferences is subject to future work.

In summary, the user has to be informed about the processing of his personal data and enabled to personalize his privacy settings for which LPL can be used as a basis. But the transparent presentation and negotiation of the privacy policy requires the research of suitable user interfaces. Furthermore, the creation of a privacy language modelling users' privacy preferences, which are matched against LPL privacy policies is subject to future work.

Part I

APPENDIX

A

POLICY-BASED DE-IDENTIFICATION – ARX ALGORITHMS

For the implementation of the *Policy-based De-identification (PD)* the ARX framework is used. ARX allows the anonymization of sensitive personal data supporting privacy models and algorithms for risk and utility measurements [223] [159].

For the implementation of the PD process, additional algorithms – *Data-set Transformation* and *Anonymization Hierarchy Transformation* – are required, which are highlighted in the following for completeness.

The algorithms *Data-set Transformation* and *Set Maximum Anonymization* are run before *Anonymization Hierarchy Transformation*, *Set Privacy Group*, *Privacy Model Substitution*, and *Apply Privacy Models* (see Figure A.1).

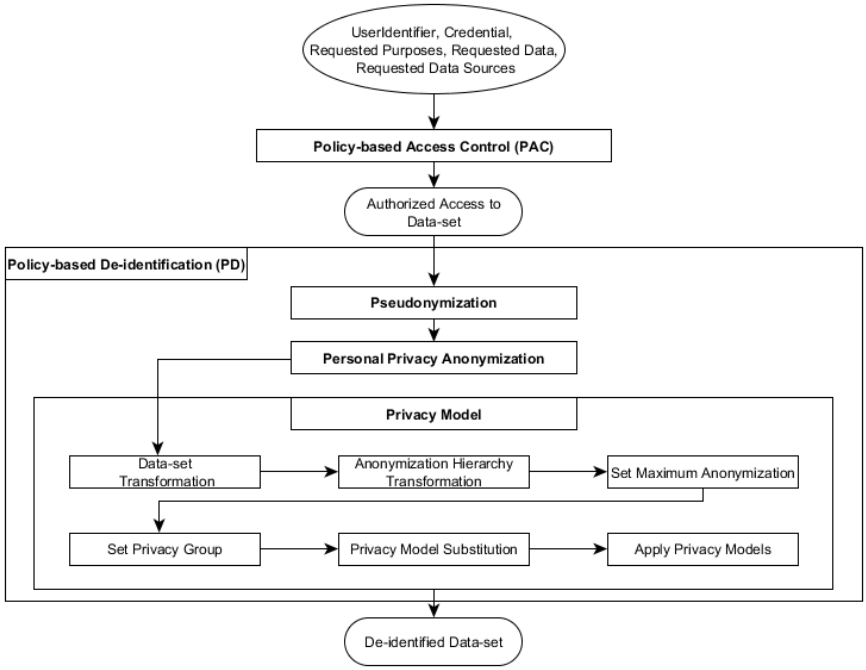


Figure A.1: *Policy-based De-identification (PD)* process step sequence with the ARX specific processes after the *Policy-based Access Control (PAC)* has been completed for a request.

A.1 DATA-SET TRANSFORMATION

Personal data and corresponding authorized purpose are encapsulated within the *DataWrapper*-object d*w*, whereas a data-set is represented by a set of d*w* objects. To enable the compatibility with ARX, each *DataWrapper*-object has to be transformed into a two-dimensional array (see Listing A.1). The first dimension represents the record and the second dimension represents the attributes of the record. This ARX specific process introduces some processing overhead to the PD process, but is excluded from the final measurements for the evaluation in Chapter 7. Next, another ARX specific process is detailed which aims at preparing the anonymization hierarchies to be processed.

```
  method: dataTransformation
  in: DW //DataWrapper list containing all records with corresponding purpose
4 out: dataArray //transformed data-set that can be processed by ARX

  dataArray[][] //two-dimensional array
  k = 0; //record counter

9 for (dw : DW) //iterate records
    for (l = 0; l < dataArray[0].length; l++) //iterate attributes

      //store value of attribute in dataArray
      dataArray[k + 1][l] = dw.dataList(dataArray[0][l]);
14
    k++;

  return dataArray;
```

Listing A.1: Pseudocode for the *Data-set Transformation*.

A.2 ANONYMIZATION HIERARCHY TRANSFORMATION

The *Anonymization Hierarchy Transformation* is also an ARX specific process like the *Data-set Transformation*. Although, the anonymization hierarchies $\widehat{\mathbf{HE}}$ already contain complete anonymization hierarchies for each attribute, specific requirements have to be met to be processed by ARX. First, it is verified if the first value (anonymization level '0') of the anonymization hierarchy matches the value that has to be anonymized. Next, the sizes of all anonymization hierarchies for an attribute have to be equal. These properties are verified again to guarantee a correct format of the anonymization hierarchies, although they may have already been fulfilled by the *Personal Privacy Anonymization* iff personal privacy requirements have been given by the users. Lastly, ARX requires the removal of duplicate hierarchies.

These properties are enforced by the *Anonymization Hierarchy Transformation* (see Listing A.2).

For each record, the anonymization hierarchy for each attribute is extracted and stored as an array. If the first entry of the array does not match the *hierarchyBase* (the first element of the *hierarchy*), it is added and the whole *hierarchy* is transformed into an array suitable for ARX. This array is then adjusted in length, if necessary, so that all hierarchies for the same attribute have the same length. If a hierarchy is too short, the last entry is copied and appended. The last entry is chosen because it is the most anonymous value of the hierarchy. This guarantees a fixed hierarchy length for the same attribute and causes no loss of privacy or utility. Thus, the anonymization hierarchies are transformed into a suitable format for ARX and are added to its configuration of the privacy model.

```
  method: anonymizationHierarchyTransformation
3 in: DW //DataWrapper list containing all records with corresponding purpose
  out: HEList //well-formed set of HE

  HEList; //arrayList with all anonymization hierarchies
  heBases; //arrayList with all first elements of the hierarchies
8
  // get all hierarchies out of the wrappers
  for (dw : DW)
    for (i = 0; i < sizeOf(dw.p.D̂; i++)
      //transform ĤE set to array
13    hierarchyArray = dw.p.D̂(i).am.ĤE;

      //prevent duplicate hierarchies
      if (not heBases(i).contains(hierarchyArray[0]))
        //add hierachyArray to HEList
18      HEList(i).add(hierarchyArray);
        //add first element of hierarchyArray to heBases
        heBases(i).add(hierarchyArray[0]);

  // for each HE in HEList unify hierarchy size
23 for (ĤE : HEList)
   adjustHierarchySize(ĤE, HEList);

  return HEList;
```

Listing A.2: Pseudocode for the *Anonymization Hiearchy Transformation*.

BIBLIOGRAPHY

[1] Harald Aamot, Christian Dominik Kohl, Daniela Richter, and Petra Knaup-Gregori. "Pseudonymization of patient identifiers for translational research." In: *BMC medical informatics and decision making* 13.1 (2013), p. 75.

[2] Abbas Acar, Hidayet Aksu, A. Selcuk Uluagac, and Mauro Conti. "A Survey on Homomorphic Encryption Schemes: Theory and Implementation." In: *ACM Comput. Surv.* 51.4 (July 2018), 79:1–79:35. ISSN: 0360-0300. DOI: 10.1145/3214303.

[3] Sushant Agarwal, Simon Steyskal, Franjo Antunovic, and Sabrina Kirrane. "Legislative Compliance Assessment: Framework, Model and GDPR Instantiation." In: *Privacy Technologies and Policy*. Ed. by Manel Medina, Andreas Mitrakas, Kai Rannenberg, Erich Schweighofer, and Nikolaos Tsouroulas. Cham: Springer International Publishing, 2018, pp. 131–149. ISBN: 978-3-030-02547-2.

[4] Charu C. Aggarwal. "On K-anonymity and the Curse of Dimensionality." In: *Proceedings of the 31st International Conference on Very Large Data Bases*. VLDB '05. Trondheim, Norway: VLDB Endowment, 2005, pp. 901–909. ISBN: 1-59593-154-6. URL: http://dl.acm.org/citation.cfm?id=1083592.1083696.

[5] Charu C. Aggarwal and Philip S. Yu. "A General Survey of Privacy-Preserving Data Mining Models and Algorithms." In: *Privacy-Preserving Data Mining: Models and Algorithms*. Ed. by Charu C. Aggarwal and Philip S. Yu. Boston, MA: Springer US, 2008, pp. 11–52. ISBN: 978-0-387-70992-5. DOI: 10.1007/978-0-387-70992-5_2.

[6] Dakshi Agrawal and Charu C. Aggarwal. "On the Design and Quantification of Privacy Preserving Data Mining Algorithms." In: *Proceedings of the Twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. PODS '01. New York, NY, USA: ACM, 2001, pp. 247–255. ISBN: 1-58113-361-8. DOI: 10.1145/375551.375602.

[7] R. Agrawal, P. Bird, T. Grandison, J. Kiernan, S. Logan, and W. Rjaibi. "Extending relational database systems to automatically enforce privacy policies." In: *21st International Conference on Data Engineering (ICDE'05)*. Apr. 2005, pp. 1013–1022. DOI: 10.1109/ICDE.2005.64.

[8] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. "Mining Association Rules Between Sets of Items in Large Databases." In: *SIGMOD Rec.* 22.2 (June 1993), pp. 207–216. ISSN: 0163-5808. DOI: 10.1145/170036.170072.

[9] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. "Hippocratic databases." In: *VLDB'02: Proceedings of the 28th International Conference on Very Large Databases*. Elsevier. 2002, pp. 143–154.

[10] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. "XPref: a preference language for P3P." In: *Computer Networks* 48.5 (2005), pp. 809 –827. ISSN: 1389-1286. DOI: 10.1016/j.comnet.2005.01.004.

[11] Irem Aktug and Katsiaryna Naliuka. "ConSpec – A Formal Language for Policy Specification." In: *Electron. Notes Theor. Comput. Sci.* 197.1 (Feb. 2008), pp. 45–58. ISSN: 1571-0661. DOI: 10.1016/j.entcs.2007.10.013.

[12] Inayat Ali, Sonia Sabir, and Zahid Ullah. "Internet of Things Security, Device Authentication and Access Control: A Review." In: *CoRR* abs/1901.07309 (2019). arXiv: 1901.07309. URL: http://arxiv.org/abs/1901.07309.

[13] Irwin Altman. *The Environment and Social Behavior: Privacy, Personal Space, Territory, and Crowding.* Monterey, Calif. : Brooks/-Cole Pub. Co., 1975.

[14] Anne H. Anderson. "A Comparison of Two Privacy Policy Languages: EPAL and XACML." In: *Proceedings of the 3rd ACM Workshop on Secure Web Services*. SWS '06. New York, NY, USA: ACM, 2006, pp. 53–60. ISBN: 1-59593-546-0. DOI: 10.1145/1180367.1180378.

[15] Julio Angulo, Simone Fischer-Hübner, Tobias Pulls, and Erik Wästlund. "Towards Usable Privacy Policy Display & Management - The PrimeLife Approach." In: *Proceedings of the Fifth International Symposium on Human Aspects of Information Security & Assurance (HAISA 2011)*. July 2011, pp. 108–118.

[16] C Ardagna, Laurent Bussard, Sabrina De Capitani Di Vimercati, Gregory Neven, E Pedrini, S Paraboschi, F Preiss, P Samarati, S Trabelsi, M Verdicchio, et al. "Primelife policy language." In: *W3C Workshop on Access Control Application Scenarios*. W3C. 2009.

[17] Article 29 Working Party. *Opinion 05/2014 on Anonymisation Techniques*. Apr. 2014. URL: https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2014/wp216_en.pdf.

[18] Article 29 Working Party. *Guidelines on transparency under Regulation 2016/679 (wp260rev.01)*. Apr. 2018. URL: https://ec.europa.eu/newsroom/article29/item-detail.cfm?item_id=622227.

[19] P. Ashley, S. Hada, G. Karjoth, and M. Schunter. "E-P3P Privacy Policies and Privacy Authorization." In: *Proceedings of the 2002 ACM Workshop on Privacy in the Electronic Society*. WPES '02. New York, NY, USA: ACM, 2002, pp. 103–109. ISBN: 1-58113-633-1. DOI: 10.1145/644527.644538.

[20] Paul Ashley, Satoshi Hada, Günter Karjoth, Calvin Powers, and Matthias Schunter. *Enterprise Privacy Authorization Language (EPAL 1.2)*. Tech. rep. IBM, 2003. URL: https://www.zurich.ibm.com/security/enterprise-privacy/epal/Specification/.

[21] K. Ashoka and B. Poornima. "Stipulation-Based Anonymization with Sensitivity Flags for Privacy Preserving Data Publishing." In: *Recent Findings in Intelligent Computing Techniques*. Ed. by Pankaj Kumar Sa, Sambit Bakshi, Ioannis K. Hatzilygeroudis, and Manmath Narayan Sahoo. Singapore: Springer Singapore, 2019, pp. 445–454. ISBN: 978-981-10-8639-7.

[22] Baik Hoh, M. Gruteser, Hui Xiong, and A. Alrabady. "Enhancing Security and Privacy in Traffic-Monitoring Systems." In: *IEEE Pervasive Computing* 5.4 (Oct. 2006), pp. 38–46. DOI: 10.1109/MPRV.2006.69.

[23] Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, and Jaehoon Amir Safavi. "Mitigating Poisoning Attacks on Machine Learning Models: A Data Provenance Based Approach." In: *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*. AISec '17. New York, NY, USA: ACM, 2017, pp. 103–110. ISBN: 978-1-4503-5202-4. DOI: 10.1145/3128572.3140450.

[24] Susan B. Barnes. "A privacy paradox: Social networking in the United States." In: *First Monday* 11.9 (2006). ISSN: 13960466. DOI: 10.5210/fm.v11i9.1394. URL: https://firstmonday.org/ojs/index.php/fm/article/view/1394.

[25] S. Barth, M.D.T. de Jong, M. Junger, P.H. Hartel, and J.C. Roppelt. "Putting the privacy paradox to the test: Online privacy and security behaviors among users with technical knowledge, privacy awareness, and financial resources." In: *Telematics and Informatics* (2019). ISSN: 0736-5853. DOI: 10.1016/j.tele.2019.03.003. URL: http://www.sciencedirect.com/science/article/pii/S0736585317307724.

[26] Susanne Barth and Menno DT De Jong. "The privacy paradox–Investigating discrepancies between expressed privacy concerns and actual online behavior–A systematic literature review." In: *Telematics and Informatics* 34.7 (2017), pp. 1038–1058.

[27] C. Bartolini and L. Siry. "The right to be forgotten in the light of the consent of the data subject." In: *Computer Law & Security Review* 32.2 (2016), pp. 218 –237. ISSN: 0267-3649. DOI: 10.1016/j.clsr.2016.01.005. URL: http://www.sciencedirect.com/science/article/pii/S026736491630019X.

[28] Lujo Bauer, Jay Ligatti, and David Walker. "Composing Security Policies with Polymer." In: *SIGPLAN Not.* 40.6 (June 2005), pp. 305–314. ISSN: 0362-1340. DOI: 10.1145/1064978.1065047.

[29] Roberto J Bayardo and Rakesh Agrawal. "Data privacy through optimal k-anonymization." In: *21st International conference on data engineering (ICDE'05)*. IEEE. 2005, pp. 217–228.

[30] Moritz Y. Becker, Cedric Fournet, and Andrew D. Gordon. "SecPAL: Design and semantics of a decentralized authorization language." In: *Journal of Computer Security* 18.4 (Jan. 2010), pp. 619–665. ISSN: 0926-227X. DOI: 10.3233/JCS-2009-0364.

[31] Moritz Y Becker, Alexander Malkis, and Laurent Bussard. "A framework for privacy preferences and data-handling policies." In: *Microsoft Research Cambridge Technical Report, MSR-TR-2009-128* (2009).

[32] Moritz Y. Becker, Alexander Malkis, and Laurent Bussard. "A Practical Generic Privacy Language." In: *Information Systems Security*. Ed. by Somesh Jha and Anish Mathuria. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 125–139. ISBN: 978-3-642-17714-9.

[33] Moritz Y Becker, Alexander Malkis, Laurent Bussard, et al. "S4P: A generic language for specifying privacy preferences and policies." In: *Microsoft Research* 167 (2010).

[34] K. Bekara, Y. Ben Mustapha, and M. Laurent. "XPACML eXtensible Privacy Access Control Markup Langua." In: *The Second International Conference on Communications and Networking*. Nov. 2010, pp. 1–5. DOI: 10.1109/COMNET.2010.5699807.

[35] Mihir Bellare, Ran Canetti, and Hugo Krawczyk. "Keying hash functions for message authentication." In: *Annual international cryptology conference*. Springer. 1996, pp. 1–15.

[36] E. Bertino, S. Merrill, A. Nesen, and C. Utz. "Redefining Data Transparency: A Multidimensional Approach." In: *Computer* 52.1 (Jan. 2019), pp. 16–26. ISSN: 0018-9162. DOI: 10.1109/MC.2018.2890190.

[37] Elisa Bertino and Igor Nai Fovino. "Information driven evaluation of data hiding algorithms." In: *International Conference on Data Warehousing and Knowledge Discovery*. Springer. 2005, pp. 418–427.

[38] Elisa Bertino, Dan Lin, and Wei Jiang. "A Survey of Quantification of Privacy Preserving Data Mining Algorithms." In: *Privacy-Preserving Data Mining: Models and Algorithms*. Ed. by Charu C. Aggarwal and Philip S. Yu. Boston, MA: Springer US, 2008, pp. 183–205. ISBN: 978-0-387-70992-5. DOI: 10.1007/978-0-387-70992-5_8.

[39] R. Bhatia and M. Singh. "Preserving Privacy in Healthcare Web Services Paradigm Through Hippocratic Databases." In: *Intelligent Computing, Communication and Devices*. Ed. by Lakhmi C. Jain, Srikanta Patnaik, and Nikhil Ichalkaranje. New Delhi: Springer India, 2015, pp. 177–188. ISBN: 978-81-322-2012-1.

[40] Raffael Bild, Klaus A Kuhn, and Fabian Prasser. "Safepub: A truthful data anonymization algorithm with strong privacy guarantees." In: *Proceedings on Privacy Enhancing Technologies* 2018.1 (2018), pp. 67–87.

[41] Joachim Biskup and Piero A. Bonatti. "Lying versus refusal for known potential secrets." In: *Data & Knowledge Engineering* 38.2 (2001), pp. 199 –222. ISSN: 0169-023X. DOI: https://doi.org/10.1016/S0169-023X(01)00024-6. URL: http://www.sciencedirect.com/science/article/pii/S0169023X01000246.

[42] Joachim Biskup and Piero A. Bonatti. "Controlled Query Evaluation for Known Policies by Combining Lying and Refusal." In: *Annals of Mathematics and Artificial Intelligence* 40.1 (Jan. 2004), pp. 37–62. ISSN: 1573-7470. DOI: 10.1023/A:1026106029043. URL: https://doi.org/10.1023/A:1026106029043.

[43] Joachim Biskup and Piero Bonatti. "Controlled query evaluation for enforcing confidentiality in complete information systems." In: *International Journal of Information Security* 3.1 (Oct. 2004), pp. 14–27. ISSN: 1615-5270. DOI: 10.1007/s10207-004-0032-1. URL: https://doi.org/10.1007/s10207-004-0032-1.

[44] Joachim Biskup and Piero Bonatti. "Controlled query evaluation with open queries for a decidable relational submodel." In: *Annals of Mathematics and Artificial Intelligence* 50.1 (June 2007), pp. 39–77. ISSN: 1573-7470. DOI: 10.1007/s10472-007-9070-5. URL: https://doi.org/10.1007/s10472-007-9070-5.

[45] Joachim Biskup and Hans Hermann Brüggeman. "The personal model of data: Towards a privacy-oriented information system." In: *Computers & Security* 7.6 (1988), pp. 575–597.

[46] Bitkom. *Stimmen Sie den Aussagen voll / eher zu? Datenschutzerklärungen...* Last accessed: 11.09.2019. 2015. URL: https://de.statista.com/statistik/daten/studie/467075/umfrage/beurteilung-der-datenschutzerklaerungen-von-online-diensten-in-deutschland/.

[47] Lynn A. Blewett, Julia A. Rivera Drew, Risa Griffin, Miram L. King, and Kari C. W. Williams. *IPUMS Health Surveys: National Health Interview Survey, Version 6.2 [dataset]. Minneapolis, MN: University of Minnesota*. Last accessed: 11.09.2019. 2017. DOI: 10.18128/D070.V6.2.

[48] Avrim Blum, Katrina Ligett, and Aaron Roth. "A Learning Theory Approach to Noninteractive Database Privacy." In: *J. ACM* 60.2 (May 2013), 12:1–12:25. ISSN: 0004-5411. DOI: 10.1145/2450142.2450148.

[49] Kathy Bohrer and Bobby Holland. *Customer Profile Exchange (CPExchange) Specification*. Version 1.0. Oct. 2000. URL: http://xml.coverpages.org/cpexchangev1_0F.pdf.

[50] B. A. Bonatti, S. Kirrane, I.M. ePetrova, L. Sauro, and E. Schlehahn. *The SPECIAL Usage Policy Language Vocabulary Vo.1*. Tech. rep. SPECIAL, 2018. URL: https://www.specialprivacy.eu/vocabs.

[51] P. A. Bonatti and S. Kirrane. "Big Data and Analytics in the Age of the GDPR." In: *2019 IEEE International Congress on Big Data (BigDataCongress)*. July 2019, pp. 7–16. DOI: 10.1109/BigDataCongress.2019.00015.

[52] Piero A. Bonatti. "Fast Compliance Checking in an OWL2 Fragment." In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*. International Joint Conferences on Artificial Intelligence Organization, July 2018, pp. 1746–1752. DOI: 10.24963/ijcai.2018/241.

[53] Piero Bonatti, Wouter Dullaert, Javier D. Fernàndez, Sabrina Kirrane, Uros Milosevic, and Axel Polleres. *The SPECIAL Policy Log Vocabulary Vo.5*. Tech. rep. SPECIAL, 2018. URL: https://aic.ai.wu.ac.at/qadlod/policyLog/.

[54] Piero Bonatti, Sabrina Kirrane, Iliana Mineva Petrova, Luigi Sauro, and Eva Schlehahn. *The SPECIAL Usage Policy Language Vo.1*. Tech. rep. SPECIAL, 2019. URL: https://aic.ai.wu.ac.at/qadlod/policyLanguage.

[55] Bert Bos et al. *Data Privacy Vocabulary vo.1*. Tech. rep. W3C Data Privacy Vocabularies and Controls Community Group, July 2019. URL: https://www.w3.org/ns/dpv.

[56] Stefano Braghin, Joao H Bettencourt-Silva, Killian Levacher, and Spiros Antonatos. "An Extensible De-Identification Framework for Privacy Protection of Unstructured Health Information: Creating Sustainable Privacy Infrastructures." In: *Studies in health technology and informatics* 264 (Aug. 2019), pp. 1140–1144. ISSN: 0926-9630. DOI: 10.3233/shti190404. URL: https://doi.org/10.3233/SHTI190404.

[57] T. Brekne, A. Årnes, and A. Øslebø. "Anonymization of ip traffic monitoring data: Attacks on two prefix-preserving anonymization schemes and some proposed remedies." In: *International Workshop on Privacy Enhancing Technologies*. Springer. 2005.

[58] Justin Brickell and Vitaly Shmatikov. "The Cost of Privacy: Destruction of Data-mining Utility in Anonymized Data Publishing." In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '08. New York, NY, USA: ACM, 2008, pp. 70–78. ISBN: 978-1-60558-193-4. DOI: 10.1145/1401890.1401904.

[59] S. Bugiel, S. Heuser, and A. Sadeghi. "Flexible and Fine-grained Mandatory Access Control on Android for Diverse Security and Privacy Policies." In: *Presented as part of the 22nd USENIX Security Symposium (USENIX Security 13)*. Washington, D.C.: USENIX, 2013, pp. 131–146. ISBN: 978-1-931971-03-4.

[60] Peter Buneman, Adriane Chapman, and James Cheney. "Provenance Management in Curated Databases." In: *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data*. SIGMOD '06. Chicago, IL, USA: ACM, 2006, pp. 539–550. ISBN: 1-59593-434-0. DOI: 10.1145/1142473.1142534. URL: http://doi.acm.org/10.1145/1142473.1142534.

[61] Benedikt Bünz, Shashank Agrawal, Mahdi Zamani, and Dan Boneh. *Zether: Towards Privacy in a Smart Contract World*. Cryptology ePrint Archive, Report 2019/191. https://eprint.iacr.org/2019/191. 2019.

[62] Ji-Won Byun, Elisa Bertino, and Ninghui Li. "Purpose Based Access Control of Complex Data for Privacy Protection." In: *Proceedings of the Tenth ACM Symposium on Access Control Models and Technologies*. SACMAT '05. New York, NY, USA: ACM, 2005, pp. 102–110. ISBN: 1-59593-045-0. DOI: 10.1145/1063979.1063998.

[63] *California Consumer Privacy Act (CCPA)*. June 2018. URL: https://leginfo.legislature.ca.gov/faces/billTextClient.xhtml?bill_id=201720180AB375.

[64] Jianneng Cao and Panagiotis Karras. "Publishing Microdata with a Robust Privacy Guarantee." In: *Proc. VLDB Endow.* 5.11 (July 2012), pp. 1388–1399. ISSN: 2150-8097. DOI: 10.14778/2350229.2350255.

[65] Center for Information Policy Leadership. *Ten steps to develop a multilayered privacy notice*. 2007.

[66] Shuchi Chawla, Cynthia Dwork, Frank McSherry, Adam Smith, and Hoeteck Wee. "Toward Privacy in Public Databases." In: *Theory of Cryptography*. Ed. by Joe Kilian. Berlin, Heidelberg:

Springer Berlin Heidelberg, 2005, pp. 363–385. ISBN: 978-3-540-30576-7.

[67] Jeffrey T Child and Sandra Petronio. "Unpacking the paradoxes of privacy in CMC relationships: The challenges of blogging and relational communication on the internet." In: *Computer-mediated communication in personal relationships* (2011), pp. 21–40.

[68] Jeffrey T Child and David A Westermann. "Let's be Facebook friends: Exploring parental Facebook friend requests from a communication privacy management (CPM) perspective." In: *Journal of Family Communication* 13.1 (2013), pp. 46–59.

[69] *Commission Implementing Decision (EU) 2016/1250 of 12 July 2016 pursuant to Directive 95/46/EC of the European Parliament and of the Council on the adequacy of the protection provided by the EU-U.S. Privacy Shield (notified under document C(2016) 4176)*. 2016. URL: http://data.europa.eu/eli/dec_impl/2016/1250/oj.

[70] Ilenia Confente, Giorgia Giusi Siciliano, Barbara Gaudenzi, and Matthias Eickhoff. "Effects of data breaches from user-generated content: A corporate reputation analysis." In: *European Management Journal* 37.4 (2019), pp. 492 –504. ISSN: 0263-2373. DOI: https://doi.org/10.1016/j.emj.2019.01.007. URL: http://www.sciencedirect.com/science/article/pii/S0263237319300234.

[71] 104th Congress. *Health Insurance Portability and Accountability Act (HIPAA)*. Public Law 104-191. 1996. URL: https://www.govinfo.gov/content/pkg/PLAW-104publ191/html/PLAW-104publ191.htm.

[72] Lorrie Cranor. *Web privacy with P3P*. " O'Reilly Media, Inc.", 2002.

[73] Frédéric Cuppens and Nora Cuppens-Bouhlahia. "Modeling contextual security policies." In: *International Journal of Information Security* 7.4 (Aug. 2008), pp. 285 –305. DOI: 10.1007/s10207-007-0051-9. URL: https://hal.archives-ouvertes.fr/hal-01207773.

[74] DIN 5008. *Schreib- und Gestaltungsregeln für die Text- und Informationsverarbeitung*. 2011.

[75] Joan Daemen and Vincent Rijmen. *AES Proposal: Rijndael*. 1999.

[76] Chenyun Dai, Dan Lin, Elisa Bertino, and Murat Kantarcioglu. "An Approach to Evaluate Data Trustworthiness Based on Data Provenance." In: *Secure Data Management*. Ed. by Willem Jonker and Milan Petković. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 82–98. ISBN: 978-3-540-85259-9.

[77] Chenyun Dai, Gabriel Ghinita, Elisa Bertino, Ji-Won Byun, and Ninghui Li. *TIAMAT: a Tool for Interactive Analysis of Microdata Anonymization Techniques*. Tech. rep. Dept. of Computer Science - Purdue University, Database Security - Oracle Corp., 2009. URL: http://www.vldb.org/pvldb/2/vldb09-114.pdf.

[78] Nicodemos Damianou, Naranker Dulay, Emil Lupu, and Morris Sloman. "The Ponder Policy Specification Language." In: *Proceedings of the International Workshop on Policies for Distributed Systems and Networks*. POLICY '01. London, UK, UK: Springer-Verlag, 2001, pp. 18–38. ISBN: 3-540-41610-2. URL: http://dl.acm.org/citation.cfm?id=646962.712108.

[79] Quynh H Dang. *Secure hash standard*. Tech. rep. Federal Inf. Process. Stds. (NIST FIPS) - 180-4, 2015. DOI: 10.6028/NIST.FIPS.180-4.

[80] Data Transfer Project. *Data Transfer Project Overview and Fundamentals*. July 2018.

[81] S. B. Davidson, S. Khanna, S. Roy, J. Stoyanovich, V. Tannen, and Y. Chen. "On Provenance and Privacy." In: *Proceedings of the 14th International Conference on Database Theory*. ICDT '11. New York, NY, USA: ACM, 2011, pp. 3–10. ISBN: 978-1-4503-0529-7. DOI: 10.1145/1938551.1938554.

[82] Susan B. Davidson, Sanjeev Khanna, Sudeepa Roy, and Sarah Cohen Boulakia. "Privacy Issues in Scientific Workflow Provenance." In: *Proceedings of the 1st International Workshop on Workflow Approaches to New Data-centric Science*. Wands '10. New York, NY, USA: ACM, 2010, 3:1–3:6. ISBN: 978-1-4503-0188-6. DOI: 10.1145/1833398.1833401.

[83] Susan B. Davidson, Sanjeev Khanna, Tova Milo, Debmalya Panigrahi, and Sudeepa Roy. "Provenance Views for Module Privacy." In: *Proceedings of the Thirtieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. PODS '11. New York, NY, USA: ACM, 2011, pp. 175–186. ISBN: 978-1-4503-0660-7. DOI: 10.1145/1989284.1989305.

[84] John Dempsey, Gavin Sim, and Brendan Cassidy. "Designing for GDPR-Investigating Children's Understanding of Privacy: A Survey Approach." In: *Proceedings of the 32nd International BCS Human Computer Interaction Conference (HCI 2018)* (2018). DOI: 10.14236/ewic/HCI2018.26.

[85] W. Diffie and M. E. Hellman. "Special Feature Exhaustive Cryptanalysis of the NBS Data Encryption Standard." In: *Computer* 10.6 (June 1977), pp. 74–84. ISSN: 0018-9162. DOI: 10.1109/C-M.1977.217750.

[86] W. Diffie and M. Hellman. "New directions in cryptography." In: *IEEE Transactions on Information Theory* 22.6 (Nov. 1976), pp. 644–654. DOI: 10.1109/TIT.1976.1055638.

[87] Johannes Drepper. *Vo86-01 Anon-Tool*. Last accessed: 11.09.2019. 2014. URL: http://www.tmf-ev.de/Themen/Projekte/V08601{_}AnonTool.aspx.

[88] Dheeru Dua and Efi Karra Taniskidou. *UCI Machine Learning Repository*. Last accessed: 11.09.2019. University of California, Irvine, School of Information and Computer Sciences. 2017. URL: https://archive.ics.uci.edu/ml/datasets/adult.

[89] Wouter Dullaert, Uros Milosevic, Jonathan Langens, Arnaud S'Jongers, Nora Szepes, Vincent Goossens, Nathaniel Rudavsky-Brody, Ward Delabastita, Sabrina Kirrane, and Javier Fernandez. *Deliverable No 3.4 – Transparency & Compliance Release*. Tech. rep. SPECIAL, Jan. 2019. URL: https://www.specialprivacy.eu/images/documents/SPECIAL_D34_M25_V10.pdf.

[90] Cynthia Dwork. "Differential Privacy." In: *Automata, Languages and Programming*. Springer Berlin Heidelberg, 2006, pp. 1–12. DOI: 10.1007/11787006_1.

[91] Cynthia Dwork. "Differential Privacy." In: *Encyclopedia of Cryptography and Security*. Ed. by Henk C. A. van Tilborg and Sushil Jajodia. Boston, MA: Springer US, 2011, pp. 338–340. ISBN: 978-1-4419-5906-5. DOI: 10.1007/978-1-4419-5906-5_752.

[92] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. "Our Data, Ourselves: Privacy Via Distributed Noise Generation." In: *Advances in Cryptology - EUROCRYPT 2006*. Ed. by Serge Vaudenay. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 486–503. ISBN: 978-3-540-34547-3.

[93] Khaled El Emam and Luk Arbuckle. *Anonymizing health data: case studies and methods to get you started*. " O'Reilly Media, Inc.", 2013.

[94] Khaled El Emam and Fida Kamal Dankar. "Protecting Privacy Using k-Anonymity." In: *Journal of the American Medical Informatics Association* 15.5 (Sept. 2008), pp. 627–637. ISSN: 1527-974X. DOI: 10.1197/jamia.M2716. eprint: http://oup.prod.sis.lan/jamia/article-pdf/15/5/627/21382979/15-5-627.pdf.

[95] Anas Abou El Kalam, Yves Deswarte, Gilles Trouessin, and Emmanuel Cordonnier. "A Generic Approach for Healthcare Data Anonymization." In: *Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society*. WPES '04. New York, NY, USA: ACM, 2004, pp. 31–32. ISBN: 1-58113-968-3. DOI: 10.1145/1029179.1029188.

[96] Hal Lockhart Erik Rissanen Bill Parducci. *eXtensible Access Control Markup Language (XACML) Version 3.0*. Tech. rep. OASIS, 2013. URL: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html.

[97] European Commission. *EU-US data transfers - How personal data transferred between the EU and US is protected*. Last accessed: 11.09.2019. URL: https://ec.europa.eu/info/law/law-topic/data-protection/data-transfers-outside-eu/eu-us-data-transfers_en.

[98] European Convention. *Charter of Fundamental Rights of the European Union*. Dec. 2000. URL: http://www.europarl.europa.eu/charter/pdf/text_en.pdf.

[99] Stèphanie Faber. *Understanding the Layered Approach to International Data Transfers Under GDPR*. Feb. 2019. URL: https://www.securityprivacybytes.com/2019/02/understanding-the-layered-approach-to-international-data-transfers-under-gdpr/.

[100] Benjamin Fabian and Tom Göthling. "Privacy-preserving Data Warehousing." In: *Int. J. Bus. Intell. Data Min.* 10.4 (Oct. 2015), pp. 297–336. ISSN: 1743-8195. DOI: 10.1504/IJBIDM.2015.072210.

[101] Andreas Faldum and Klaus Pommerening. "An optimal code for patient identifiers." In: *Computer Methods and Programs in Biomedicine* 79.1 (2005), pp. 81 –88. ISSN: 0169-2607. DOI: 10.1016/j.cmpb.2005.03.004. URL: http://www.sciencedirect.com/science/article/pii/S0169260705000672.

[102] J. Fan, J. Xu, M.H. Ammar, and S.B. Moon. "Prefix-preserving IP address anonymization: measurement-based security evaluation and a new cryptography-based scheme." In: *Computer Networks* 46.2 (2004), pp. 253 –272. ISSN: 1389-1286. DOI: 10.1016/j.comnet.2004.03.033.

[103] Kai Fan, Huiyue Xu, Longxiang Gao, Hui Li, and Yintang Yang. "Efficient and privacy preserving access control scheme for fog-enabled IoT." In: *Future Generation Computer Systems* 99 (2019), pp. 134 –142. ISSN: 0167-739X. DOI: 10.1016/j.future.2019.04.003.

[104] Liyue Fan and Li Xiong. "Real-time Aggregate Monitoring with Differential Privacy." In: *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*. CIKM '12. New York, NY, USA: ACM, 2012, pp. 2169–2173. ISBN: 978-1-4503-1156-4. DOI: 10.1145/2396761.2398595.

[105] Ferdinando Fioretto, Terrence W. K. Mak, and Pascal Van Hentenryck. "Differential Privacy for Power Grid Obfuscation." In: *CoRR* abs/1901.06949 (2019). arXiv: 1901.06949. URL: http://arxiv.org/abs/1901.06949.

[106] Ronald Aylmer Fisher and Frank Yates. *Statistical tables for biological, agricultural and medical research*. Oliver and Boyd, 1948.

[107] Garrett M Fitzmaurice, Nan M Laird, and James H Ware. *Applied longitudinal analysis*. Vol. 998. John Wiley & Sons, 2012.

[108] Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu. "Privacy-preserving Data Publishing: A Survey of Recent Developments." In: *ACM Comput. Surv.* 42.4 (June 2010), 14:1–14:53. ISSN: 0360-0300. DOI: 10.1145/1749603.1749605.

[109] B. Gedik and Ling Liu. "Location Privacy in Mobile Systems: A Personalized Anonymization Model." In: *25th IEEE International Conference on Distributed Computing Systems (ICDCS'05)*. June 2005, pp. 620–629. DOI: 10.1109/ICDCS.2005.48.

[110] *General Data Protection Regulation*. Regulation (EU) 2016 of the European Parliament and of the Council of on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC. Apr. 2016. URL: http://data.europa.eu/eli/reg/2016/679/oj.

[111] A. Gerl and S. Becher. "Policy-Based De-Identification Test Framework." In: *2019 IEEE World Congress on Services (SERVICES)*. Vol. 2642-939X. July 2019, pp. 356–357. DOI: 10.1109/SERVICES.2019.00101.

[112] Armin Gerl. "Extending Layered Privacy Language to Support Privacy Icons for a Personal Privacy Policy User Interface." In: *Proceedings of British HCI 2018*. BCS Learning and Development Ltd., Belfast, UK, 2018, p. 5.

[113] Armin Gerl and Felix Bölz. "Layered Privacy Language (LPL) Pseudonymization Extension for Health Care." In: *MEDINFO 2019: Health and Wellbeing e-Networks for All*. Ed. by Lucila Ohno-Machado and Brigitte Séroussi. Vol. 264. Studies in Health Technology and Informatics. 2019, pp. 1189 –1193. DOI: 10.3233/SHTI190414.

[114] Armin Gerl and Bianca Christina Meier. "Privacy in the Future of Integrated Health Care Services - Are Privacy Languages the Key?" In: *Seventh International Workshop on e-Health Pervasive Wireless Applications and Services 2019 (eHPWAS'19)*. Barcelona, Spain, Oct. 2019.

[115] Armin Gerl and Bianca Meier. "The Layered Privacy Language Art. 12 – 14 GDPR Extension – Privacy Enhancing User Interfaces." In: *Datenschutz und Datensicherheit - DuD* 43.12 (2019), pp. 747–752. ISSN: 1862-2607. DOI: 10.1007/s11623-019-1200-9. URL: https://doi.org/10.1007/s11623-019-1200-9.

[116] Armin Gerl and Bianca Meier. "The Layered Privacy Language Art. 12 - 14 GDPR Extension - Privacy Enhancing User Interfaces." In: *INFORMATIK 2019: 50 Jahre Gesellschaft für Informatik – Informatik für Gesellschaft*. Ed. by Klaus David, Kurt Geihs, Martin Lange, and Gerd Stumme. Bonn: Gesellschaft für Informatik e.V., 2019, pp. 311–324. DOI: 10.18420/inf2019_51.

[117] Armin Gerl, Bianca Meier, and Stefan Becher. "Let Users Control Their Data – Privacy Policy-Based User Interface Design." In: *Human Interaction and Emerging Technologies*. Ed. by Tareq Ahram, Redha Taiar, Serge Colson, and Arnaud Choplin. Vol. 1018. Advances in Intelligent Systems and Computing 1. Springer International Publishing, 2019, pp. 790 –795. DOI: 10.1007/978-3-030-25629-6.

[118] Armin Gerl and Dirk Pohl. "The Right to data portability between legal possibilities and technical boundaries." In: *Practical Implementation of the Right to Data Portability- Legal, Technical and Consumer-Related Implications*. Stiftung Datenschutz. 2017, pp. 208–224.

[119] Armin Gerl and Dirk Pohl. "Critical Analysis of LPL according to Articles 12 - 14 of the GDPR." In: *Proceedings of International Conference on Availability, Reliability and Security*. ARES 2018. Hamburg, Germany, Aug. 2018, p. 9. DOI: 10.1145/3230833.3233267.

[120] Armin Gerl and Florian Prey. "LPL Personal Privacy Policy User Interface: Design and Evaluation." In: *Mensch und Computer 2018 - Tagungsband*. Bonn: Gesellschaft für Informatik e.V., 2018.

[121] Armin Gerl, Nadia Bennani, Harald Kosch, and Lionel Brunie. "LPL, Towards a GDPR-Compliant Privacy Language: Formal Definition and Usage." In: ed. by A. Hameurlain and R. Wagner. Vol. Transactions on Large-Scale Databases and Knowledge-Centered Systems (TLDKS). Lecture Notes in Computer Science (LNCS) 10940 XXXVII. Springer-Verlag GmbH Germany, part of Springer Nature 2018, 2018. Chap. 2, pp. 41–80. DOI: 10.1007/978-3-662-57932-9_2.

[122] Tyrone Grandison and Kristen LeFevre. "Hippocratic Database." In: *Encyclopedia of Cryptography and Security*. Ed. by Henk C. A. van Tilborg and Sushil Jajodia. Boston, MA: Springer US, 2011, pp. 556–559. ISBN: 978-1-4419-5906-5. DOI: 10.1007/978-1-4419-5906-5_679.

[123] Todd J. Green, Grigoris Karvounarakis, and Val Tannen. "Provenance Semirings." In: *Proceedings of the Twenty-sixth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. PODS '07. Beijing, China: ACM, 2007, pp. 31–40. ISBN: 978-1-59593-685-1. DOI: 10.1145/1265530.1265535. URL: http://doi.acm.org/10.1145/1265530.1265535.

[124] Sebastian Greger. *User-centred transparency design for privacy – Part I: The layered approach*. Last accessed: 11.09.2019. Aug. 2018. URL: https://sebastiangreger.net/2018/08/user-centred-transparency-design-the-layered-approach.

[125] Guozhang Wang. *Cornell Anonymization Toolkit*. Last accessed: 11.09.2019. 2014. URL: https://sourceforge.net/projects/anony-toolkit/.

[126] S. Gutwirth, Y. Poullet, P.J.A. de Hert, J. Nouwt, and C. de Terwange. *Reinventing data protection?* English. Pagination: 330. Springer, 2009. ISBN: 9781402094972.

[127] H.Shin, J. Vaidya, and V. Atluri. "Anonymization models for directional location based service environments." In: *Computers & Security* 29.1 (2010), pp. 59 –73. ISSN: 0167-4048. DOI: 10.1016/j.cose.2009.07.006.

[128] Satoshi Hada and Michiharu Kudo. *XML Access Control Language: Provisional Authorization for XML Documents*. Oct. 2000. URL: http://xml.coverpages.org/xacl-spec200102.html.

[129] Ragib Hasan, Radu Sion, and Marianne Winslett. "Introducing Secure Provenance: Problems and Challenges." In: *Proceedings of the 2007 ACM Workshop on Storage Security and Survivability*. StorageSS '07. Alexandria, Virginia, USA: ACM, 2007, pp. 13–18. ISBN: 978-1-59593-891-6. DOI: 10.1145/1314313.1314318. URL: http://doi.acm.org/10.1145/1314313.1314318.

[130] Ragib Hasan, Radu Sion, and Marianne Winslett. "Preventing History Forgery with Secure Provenance." In: *Trans. Storage* 5.4 (Dec. 2009), 12:1–12:43. ISSN: 1553-3077. DOI: 10.1145/1629080.1629082. URL: http://doi.acm.org/10.1145/1629080.1629082.

[131] Thomas Heinis and Gustavo Alonso. "Efficient Lineage Tracking for Scientific Workflows." In: *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*. SIGMOD '08. Vancouver, Canada: ACM, 2008, pp. 1007–1018. ISBN: 978-1-60558-102-6. DOI: 10.1145/1376616.1376716. URL: http://doi.acm.org/10.1145/1376616.1376716.

[132] Johannes Heurix and Thomas Neubauer. "Privacy-preserving storage and access of medical data through pseudonymization and encryption." In: *International Conference on Trust, Privacy and Security in Digital Business*. Springer. 2011, pp. 186–197.

[133] Mike Hintze and Khaled El Emam. "Comparing the benefits of pseudonymisation and anonymisation under the GDPR." In: *Journal of Data Protection & Privacy* 2.2 (2018), pp. 145–158. URL: https://www.ingentaconnect.com/content/hsp/jdpp/2018/00000002/00000002/art00005.

[134] ISO 639-1. *Codes for the representation of names of languages - Part 1: Alpha-2 code*. Vienna, Austria, 2002.

[135] ITU E.123. *Notation for national and international telephone numbers, e-mail addresses and web addresses*. Tech. rep. International Telecommunication Union, 2001. URL: https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-E.123-200102-I!!PDF-E&type=items.

[136] ITU E.164. *The international public telecommunication numbering plan*. Tech. rep. International Telecommunication Union, 2010. URL: https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-E.164-201011-I!!PDF-E&type=items.

[137] Parsa Ismail and Howes Ken. *KDD Cup 1998 Data*. UCI KDD Archive - Information and Computer Science, University of California, Irvine. Last accessed: 11.09.2019. 1999. URL: https://kdd.ics.uci.edu/databases/kddcup98/kddcup98.html.

[138] Leonardo Horn Iwaya, Simone Fischer-Hübner, Rose-Mharie Åhlfeldt, and Leonardo A Martucci. "Mobile Health Systems for Community-Based Primary Care: Identifying Controls and Mitigating Privacy Threats." In: *JMIR Mhealth Uhealth* 7.3 (Mar. 2019), e11642. ISSN: 2291-5222. DOI: 10.2196/11642. URL: http://mhealth.jmir.org/2019/3/e11642/.

[139] Vijay S. Iyengar. "Transforming Data to Satisfy Privacy Constraints." In: *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '02. New York, NY, USA: ACM, 2002, pp. 279–288. ISBN: 1-58113-567-X. DOI: 10.1145/775047.775089.

[140] J. Iyilade and J. Vassileva. "P2U: A Privacy Policy Specification Language for Secondary Data Sharing and Usage." In: *Proc. IEEE Security and Privacy Workshops*. May 2014, pp. 18–22. DOI: 10.1109/SPW.2014.12.

[141] Johnson Iyilade and Julita Vassileva. "A Framework for Privacy-Aware User Data Trading." In: *User Modeling, Adaptation, and Personalization*. Springer, Jan. 2013. ISBN: 978-3-642-38843-9. DOI: 10.1007/978-3-642-38844-6_28.

[142] S. Ji, P. Mittal, and R. Beyah. "Graph Data Anonymization, De-Anonymization Attacks, and De-Anonymizability Quantification: A Survey." In: *IEEE Communications Surveys Tutorials* 19.2 (2017), pp. 1305–1326. ISSN: 1553-877X. DOI: 10.1109/COMST.2016.2633620.

[143] Adam N Joinson, Ulf-Dietrich Reips, Tom Buchanan, and Carina B Paine Schofield. "Privacy, trust, and self-disclosure online." In: *Human–Computer Interaction* 25.1 (2010), pp. 1–24.

[144] Marc Joye and Benoît Libert. "A Scalable Scheme for Privacy-Preserving Aggregation of Time-Series Data." In: *Financial Cryptography and Data Security*. Ed. by Ahmad-Reza Sadeghi. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 111–125. ISBN: 978-3-642-39884-1.

[145] Lalana Kagal. *Rei: APolicy Language for the Me-Centric Project*. Tech. rep. HP Labs, 2002.

[146] Lalana Kagal, Ian Jacobi, and Ankesh Khandelwal. "Gasping for air why we need linked rules and justifications on the semantic web." In: *CSAIL Technical Reports* (2011). URL: http://hdl.handle.net/1721.1/62294.

[147] B. Kaliski. *RFC 1319 - The MD2 Message-Digest Algorithm*. Tech. rep. Network Working Group, Apr. 1992. DOI: 10.17487/RFC1319. URL: https://tools.ietf.org/html/rfc1319.

[148] B. Kaliski. *PKCS #5: Password-Based Cryptography Specification Version 2.0*. Tech. rep. Network Working Group, Sept. 2000. DOI: 10.17487/RFC2898. URL: https://tools.ietf.org/html/rfc2898.

[149] Murat Kantarcioglu, Ali Inan, and Mehmet Kuzu. *Anonymization ToolBox*. Last accessed: 11.09.2019. 2012. URL: http://cs.utdallas.edu/dspl/cgi-bin/toolbox/index.php?go=home.

[150] Günter Karjoth, Matthias Schunter, and Michael Waidner. "Platform for Enterprise Privacy Practices: Privacy-Enabled Management of Customer Data." In: *Privacy Enhancing Technologies*. Ed. by Roger Dingledine and Paul Syverson. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 69–84. ISBN: 978-3-540-36467-2.

[151] Saffija Kasem-Madani and Michael Meier. "Security and Privacy Policy Languages: A Survey, Categorization and Gap Identification." In: *CoRR* abs/1512.00201 (2015). URL: http://arxiv.org/abs/1512.00201.

[152] Florian Kerschbaum. "Distance-preserving Pseudonymization for Timestamps and Spatial Data." In: *Proceedings of the 2007 ACM Workshop on Privacy in Electronic Society*. WPES '07. New York, NY, USA: ACM, 2007, pp. 68–71. ISBN: 978-1-59593-883-1. DOI: 10.1145/1314333.1314346.

[153] A. Khan, K. Choromanski, A. Pothen, S. M. Ferdous, M. Halappanavar, and A. Tumeo. "Adaptive Anonymization of Data using b-Edge Cover." In: *2018 SC18: The International Conference for High Performance Computing, Networking, Storage, and Analysis (SC)*. Vol. 00. 2018, pp. 743–753.

[154] S. M. Khan and K. W. Hamlen. "AnonymousCloud: A Data Ownership Privacy Provider Framework in Cloud Computing." In: *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*. June 2012, pp. 170–176. DOI: 10.1109/TrustCom.2012.94.

[155] Wazir Zada Khan, Mohammed Y. Aalsalem, Muhammad Khurram Khan, and Quratulain Arshad. "Enabling Consumer Trust Upon Acceptance of IoT Technologies Through Security and Privacy Model." In: *Advanced Multimedia and Ubiquitous Engineering*. Ed. by James J. (Jong Hyuk) Park, Hai Jin, Young-Sik Jeong, and Muhammad Khurram Khan. Singapore: Springer Singapore, 2016, pp. 111–117. ISBN: 978-981-10-1536-6.

[156] Ankesh Khandelwal, Jie Bao, Lalana Kagal, Ian Jacobi, Li Ding, and James Hendler. "Analyzing the AIR Language: A Semantic Web (Production) Rule Language." In: *Web Reasoning and Rule Systems: Fourth International Conference, RR 2010, Bressanone/Brixen, Italy, September 22-24, 2010. Proceedings*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 58–72. ISBN: 978-3-642-15918-3. DOI: 10.1007/978-3-642-15918-3_6.

[157] J. Klensin. *RFC 3696 - Application Techniques for Checking and Transformation of Names*. Tech. rep. Network Working Group, Feb. 2004. DOI: 10.17487/RFC3696. URL: https://tools.ietf.org/html/rfc3696.

[158] J. Klensin. *RFC 5321 - Simple Mail Transfer Protocol*. Tech. rep. Network Working Group, Oct. 2008. DOI: 10.17487/RFC5321. URL: https://tools.ietf.org/html/rfc5321.

[159] Florian Kohlmayer, Fabian Prasser, and Klaus A. Kuhn. "The cost of quality: Implementing generalization and suppression for anonymizing biomedical data with minimal information loss." In: *Journal of Biomedical Informatics* 58 (2015), pp. 37 – 48. ISSN: 1532-0464. DOI: 10.1016/j.jbi.2015.09.007. URL: http://www.sciencedirect.com/science/article/pii/S1532046415002002.

[160] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou. "Hawk: The Blockchain Model of Cryptography and Privacy-Preserving Smart Contracts." In: *2016 IEEE Symposium on Security and Privacy (SP)*. May 2016, pp. 839–858. DOI: 10.1109/SP.2016.55.

[161] John Krumm. "Inference Attacks on Location Tracks." In: *Pervasive Computing*. Ed. by Anthony LaMarca, Marc Langheinrich, and Khai N. Truong. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 127–143. ISBN: 978-3-540-72037-9.

[162] Jürgen Kühling, Mario Martini, Johanna Heberlein, Benjamin Kühl, David Nink, Quirin Weinzierl, and Michael Wenzel. *Die Datenschutz-Grundverordnung und das nationale Recht*. Verlagshaus Monsenstein und Vannderdat OHG Münster, 2016.

[163] H. Kumar, S. Kumar, R. Joseph, D. Kumar, S. K. Shrinarayan Singh, P. Kumar, and H. Kumar. "Rainbow table to crack password using MD5 hashing algorithm." In: *2013 IEEE Conference on Information Communication Technologies*. Apr. 2013, pp. 433–439. DOI: 10.1109/CICT.2013.6558135.

[164] Ponnurangam Kumaraguru, Lorrie Cranor, Jorge Lobo, and Seraphin Calo. "A Survey of Privacy Policy Languages." In: *Workshop on Usable IT Security Management (USM '07) at Symposium On Usable Privacy and Security '07* (2007). URL: http://precog.iiitd.edu.in/Publications_files/Privacy_Policy_Languages.pdf.

[165] Martin Lablans, Andreas Borg, and Frank Ückert. "A RESTful interface to pseudonymization services in modern web applications." In: *BMC medical informatics and decision making* 15.1 (2015), p. 2.

[166] D. D. Lamanna, J. Skene, and W. Emmerich. "SLAng: a language for defining service level agreements." In: *The Ninth IEEE Workshop on Future Trends of Distributed Computing Systems, 2003. FTDCS 2003. Proceedings.* May 2003, pp. 100–106. DOI: 10.1109/FTDCS.2003.1204317.

[167] N. Li, T. Li, and S. Venkatasubramanian. "t-Closeness: Privacy Beyond k-Anonymity and l-Diversity." In: *Proc. IEEE 23rd Int. Conf. Data Engineering*. Apr. 2007, pp. 106–115. DOI: 10.1109/ICDE.2007.367856.

[168] Ninghui Li, Wahbeh Qardaji, and Dong Su. "On sampling, anonymization, and differential privacy or,k-anonymization meets differential privacy." In: *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security - ASIACCS 12*. ACM Press, 2012. DOI: 10.1145/2414456.2414474.

[169] Tiancheng Li and Ninghui Li. "On the Tradeoff Between Privacy and Utility in Data Publishing." In: *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '09. New York, NY, USA: ACM, 2009, pp. 517–526. ISBN: 978-1-60558-495-9. DOI: 10.1145/1557019.1557079.

[170] Xueping Liang, Sachin Shetty, Deepak Tosh, Charles Kamhoua, Kevin Kwiat, and Laurent Njilla. "ProvChain: A Blockchain-based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability." In: *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud*

and Grid Computing. CCGrid '17. Piscataway, NJ, USA: IEEE Press, 2017, pp. 468–477. ISBN: 978-1-5090-6610-0. DOI: 10.1109/CCGRID.2017.8.

[171] Grigorios Loukides and Jianhua Shao. "Data Utility and Privacy Protection Trade-off in K-anonymisation." In: *Proceedings of the 2008 International Workshop on Privacy and Anonymity in Information Society*. PAIS '08. New York, NY, USA: ACM, 2008, pp. 36–45. ISBN: 978-1-59593-965-4. DOI: 10.1145/1379287.1379296.

[172] Faiza Loukil, Chirine Ghedira-Guegan, Khouloud Boukadi, and Aïcha Nabila Benharkat. "Towards an End-to-End IoT Data Privacy-Preserving Framework Using Blockchain Technology." In: *Web Information Systems Engineering – WISE 2018*. Ed. by Hakim Hacid, Wojciech Cellary, Hua Wang, Hye-Young Paik, and Rui Zhou. Cham: Springer International Publishing, 2018, pp. 68–78. ISBN: 978-3-030-02922-7.

[173] Gavin Lowe. "An attack on the Needham-Schroeder public-key authentication protocol." In: *Information Processing Letters* 56.3 (1995), pp. 131 –133. ISSN: 0020-0190. DOI: https://doi.org/10.1016/0020-0190(95)00144-2. URL: http://www.sciencedirect.com/science/article/pii/0020019095001442.

[174] Rongxing Lu, Xiaodong Lin, Xiaohui Liang, and Xuemin (Sherman) Shen. "Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing." In: *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*. ASIACCS '10. Beijing, China: ACM, 2010, pp. 282–292. ISBN: 978-1-60558-936-7. DOI: 10.1145/1755688.1755723. URL: http://doi.acm.org/10.1145/1755688.1755723.

[175] T. Ma, J. Cao, and A. Otgonbayar. "Entropy Based Personal Privacy Measurement in Ubiquitous Computing." In: *2014 7th International Conference on Ubi-Media Computing and Workshops*. July 2014, pp. 33–36. DOI: 10.1109/U-MEDIA.2014.36.

[176] M. L. Maag, L. Denoyer, and P. Gallinari. "Graph Anonymization Using Machine Learning." In: *2014 IEEE 28th International Conference on Advanced Information Networking and Applications*. May 2014, pp. 1111–1118. DOI: 10.1109/AINA.2014.20.

[177] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. "L-diversity: Privacy Beyond K-anonymity." In: *ACM Trans. Knowl. Discov. Data* 1.1 (Mar. 2007). ISSN: 1556-4681. DOI: 10.1145/1217299.1217302.

[178] Mohamed Maouche, Sonia Ben Mokhtar, and Sara Bouchenak. "HMC: Robust Privacy Protection of Mobility Data against Multiple Re-Identification Attacks." In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2.3 (Sept. 2018), pp. 1–25. URL: https://hal.archives-ouvertes.fr/hal-01954041.

[179] Lorrie Cranor & Marc Langheinrich & Massimo Marchiori. *A P3P Preference Exchange Language 1.0 (APPEL1.0)*. Tech. rep. W3C, 2002. URL: https://www.w3.org/TR/P3P-preferences/.

[180] Yutaka Matsuo, Naoaki Okazaki, Kiyoshi Izumi, Yoshiyuki Nakamura, Takuichi Nishimura, Kôiti Hasida, and Hideyuki Nakashima. "Inferring Long-term User Properties Based on Users' Location History." In: *IJCAI*. 2007, pp. 2159–2165.

[181] Francesca Mauro and Debora Stella. "Brief Overview of the Legal Instruments and Restrictions for Sharing Data While Complying with the EU Data Protection Law." In: *International Conference on Web Engineering*. Springer. 2016, 57 bibrangedash 68.

[182] Aleecia M. McDonald and Lorrie Faith Cranor. "The cost of reading privacy policies." In: *I/S: A Journal of Law and Policy for the Information Society* 4 (2008).

[183] Matthias Mehldau. *Iconset for Data-privacy Declarations v0.1 - Let's simple declare what data is how used, stored, given away or deleted.* June 2018. URL: https://cdn.netzpolitik.org/wp-upload/data-privacy-icons-v01.pdf.

[184] Per Håkon Meland, Karin Bernsmed, Martin Gilje Jaatun, Humberto Nicolás Castej´n, and Astrid Undheim. "Expressing cloud security requirements for SLAs in deontic contract languages for cloud brokers." In: *International Journal of Cloud Computing* 3.1 (2014). PMID: 58831, pp. 69–93. DOI: 10.1504/IJCC.2014.058831. eprint: http://www.inderscienceonline.com/doi/pdf/10.1504/IJCC.2014.058831.

[185] Tim Menzies, Ekrem Kocagüneli, Leandro Minku, Fayola Peters, and Burak Turhan. "Chapter 16 - How To Keep Your Data Private." In: *Sharing Data and Models in Software Engineering*. Ed. by Tim Menzies, Ekrem Kocagüneli, Leandro Minku, Fayola Peters, and Burak Turhan. Boston: Morgan Kaufmann, 2015, pp. 165 –196. ISBN: 978-0-12-417295-1. DOI: 10.1016/B978-0-12-417295-1.00016-3. URL: http://www.sciencedirect.com/science/article/pii/B9780124172951000163.

[186] Taneli Mielikäinen. "Privacy Problems with Anonymized Transaction Databases." In: *Discovery Science*. Ed. by Einoshin Suzuki and Setsuo Arikawa. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 219–229. ISBN: 978-3-540-30214-8.

[187] D. Mohapatra and M.R. Patra. "Graph Anonymization Using Hierarchical Clustering." In: *Computational Intelligence in Data Mining*. Ed. by Himansu Sekhar Behera, Janmenjoy Nayak, Bighnaraj Naik, and Ajith Abraham. Singapore: Springer Singapore, 2019, pp. 145–154. ISBN: 978-981-10-8055-5.

[188] Victor Morel and Raúl Pardo. "Three Dimensions of Privacy Policies." In: *arXiv e-prints*, arXiv:1908.06814 (Aug. 2019), arXiv:1908.06814. arXiv: 1908.06814 [cs.CY].

[189] Ben Moskowitz and Aza Raskin. *Privacy Icons Project (beta release)*. June 2011. URL: https://wiki.mozilla.org/Privacy_Icons.

[190] Mehmet Munur, Sarah Branam, and Matt Mrkobrad. *Best Practices in Drafting Plain-Language and Layered Privacy Policies*. Sept. 2012. URL: https://iapp.org/news/a/2012-09-13-best-practices-in-drafting-plain-language-and-layered-privacy/.

[191] National Highway Traffic Safety Administration, 1200 New Jersey Avenue, SE Washington, DC 20590. *Fatality Analysis Reporting System (FARS)*. Last accessed: 11.09.2019. URL: https://www.nhtsa.gov/research-data/fatality-analysis-reporting-system-fars.

[192] Roger M. Needham and Michael D. Schroeder. "Using Encryption for Authentication in Large Networks of Computers." In: *Commun. ACM* 21.12 (Dec. 1978), pp. 993–999. ISSN: 0001-0782. DOI: 10.1145/359657.359659. URL: http://doi.acm.org/10.1145/359657.359659.

[193] M. E. Nergiz, C. Clifton, and A. E. Nergiz. "MultiRelational k-Anonymity." In: *2007 IEEE 23rd International Conference on Data Engineering*. Apr. 2007, pp. 1417–1421. DOI: 10.1109/ICDE.2007.369025.

[194] Mehmet Ercan Nergiz, Maurizio Atzori, and Chris Clifton. "Hiding the Presence of Individuals from Shared Databases." In: *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*. SIGMOD '07. New York, NY, USA: ACM, 2007, pp. 665–676. ISBN: 978-1-59593-686-8. DOI: 10.1145/1247480.1247554.

[195] Mehmet Ercan Nergiz, Maurizio Atzori, and Yucel Saygin. "Towards Trajectory Anonymization: A Generalization-based Approach." In: *Proceedings of the SIGSPATIAL ACM GIS 2008 International Workshop on Security and Privacy in GIS and LBS*. SPRINGL '08. New York, NY, USA: ACM, 2008, pp. 52–61. ISBN: 978-1-60558-324-2. DOI: 10.1145/1503402.1503413.

[196] Thomas Neubauer and Johannes Heurix. "A methodology for the pseudonymization of medical data." In: *International Journal of Medical Informatics* 80.3 (2011), pp. 190 –204. ISSN: 1386-5056. DOI: 10.1016/j.ijmedinf.2010.10.016. URL: http://www.sciencedirect.com/science/article/pii/S1386505610002042.

[197] B. C. Neuman and T. Ts'o. "Kerberos: An Authentication Service for Computer Networks." In: *Comm. Mag.* 32.9 (Sept. 1994), pp. 33–38. ISSN: 0163-6804. DOI: 10.1109/35.312841. URL: https://doi.org/10.1109/35.312841.

[198] Geoffrey K. Neumann, Paul Grace, Daniel Burns, and Mike Surridge. "Pseudonymization risk analysis in distributed systems." In: *Journal of Internet Services and Applications* 10.1 (Jan. 2019), p. 1. ISSN: 1869-0238. DOI: 10.1186/s13174-018-0098-z. URL: https://doi.org/10.1186/s13174-018-0098-z.

[199] Qun Ni, Shouhuai Xu, Elisa Bertino, Ravi Sandhu, and Weili Han. "An Access Control Language for a General Provenance Model." In: *Secure Data Management*. Ed. by Willem Jonker and Milan Petković. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 68–88. ISBN: 978-3-642-04219-5.

[200] Qun Ni, Elisa Bertino, Jorge Lobo, Carolyn Brodie, Clare-Marie Karat, John Karat, and Alberto Trombeta. "Privacy-aware Role-based Access Control." In: *ACM Trans. Inf. Syst. Secur.* 13.3 (July 2010), 24:1–24:31. ISSN: 1094-9224. DOI: 10.1145/1805974.1805980.

[201] Rita Noumeir, Alain Lemay, and Jean-Marc Lina. "Pseudonymization of radiology data for research purposes." In: *Journal of digital imaging* 20.3 (2007), pp. 284–295.

[202] Daniel Oberle, Alistair Barros, Uwe Kylau, and Steffen Heinzl. "A Unified Description Language for Human to Automated Services." In: *Inf. Syst.* 38.1 (Mar. 2013), pp. 155–181. ISSN: 0306-4379. DOI: 10.1016/j.is.2012.06.004.

[203] Aleksander Øhrn and Lucila Ohno-Machado. "Using Boolean reasoning to anonymize databases." In: *Artificial Intelligence in Medicine* 15.3 (1999), pp. 235 –254. ISSN: 0933-3657. DOI: 10.1016/S0933-3657(98)00056-6.

[204] Aafaf Ouaddah, Hajar Mousannif, Anas Abou Elkalam, and Abdellah Ait Ouahman. "Access control in the Internet of Things: Big challenges and new opportunities." In: *Computer Networks* 112 (2017), pp. 237 –262. ISSN: 1389-1286. DOI: https://doi.org/10.1016/j.comnet.2016.11.007. URL: http://www.sciencedirect.com/science/article/pii/S1389128616303735.

[205] Ed. P. Resnick. *RFC 5322 - Internet Message Format*. Tech. rep. Network Working Group, 2008. DOI: 10.17487/RFC5322. URL: https://tools.ietf.org/html/rfc5322.

[206] M. Palmirani, A. Rossi, M. Martoni, and M. Hagan. "A methodological framework to design a machine-readable privacy icon set." In: *Data Protection / LegalTech Proceedings of the 21st International Legal Informatics Symposium IRIS* (Feb. 2018). ISSN: 0302-9743. DOI: 10.1007/978-3-319-98349-3_11. URL: https://ssrn.com/abstract=3195937.

[207] S. Pearson and M. Casassa-Mont. "Sticky Policies: An Approach for Managing Privacy across Multiple Parties." In: *Computer* 44.9 (Sept. 2011), pp. 60–68. ISSN: 0018-9162. DOI: 10.1109/MC.2011.225.

[208] Peter-Paul de Wolf. *μ-ARGUS home page*. Last accessed: 11.09.2019. 2015. URL: http://neon.vb.cbs.nl/casc/mu.htm.

[209] A. Petit, T. Cerqueus, S. B. Mokhtar, L. Brunie, and H. Kosch. "PEAS: Private, Efficient and Accurate Web Search." In: *2015 IEEE Trustcom/BigDataSE/ISPA*. Vol. 1. Aug. 2015, pp. 571–580. DOI: 10.1109/Trustcom.2015.421.

[210] Albin Petit, Sonia Ben Mokhtar, Lionel Brunie, and Harald Kosch. "Towards Efficient and Accurate Privacy Preserving Web Search." In: *Proceedings of the 9th Workshop on Middleware for Next Generation Internet Computing*. MW4NG '14. New York, NY, USA: ACM, 2014, 1:1–1:6. ISBN: 978-1-4503-3222-4. DOI: 10.1145/2676733.2676734. URL: http://doi.acm.org/10.1145/2676733.2676734.

[211] S Petronio. *Boundaries of Privacy: Dialectics of Disclosure*. SUNY series in communication studies. Albany, NY: State University of New York Press, 2002. ISBN: 0-7914-5515-7.

[212] Sandra Petronio and Jennifer Reierson. "Regulating the privacy of confidentiality: Grasping the complexities through communication privacy management theory." In: *Uncertainty, information management, and disclosure decisions: Theories and applications* (2009), pp. 365–383.

[213] John Sören Pettersson. "A brief evaluation of icons in the first reading of the European parliament on COM (2012) 0011." In: *IFIP International Summer School on Privacy and Identity Management*. Springer. 2014, pp. 125–135.

[214] Eugenia Politou, Efthimios Alepis, and Constantinos Patsakis. "Forgetting personal data and revoking consent under the GDPR: Challenges and proposed solutions." In: *Journal of Cybersecurity* 4.1 (Mar. 2018). ISSN: 2057-2085. DOI: 10.1093/cybsec/tyy001.

[215] Eugenia Politou, Alexandra Michota, Efthimios Alepis, Matthias Pocs, and Constantinos Patsakis. "Backups and the right to be forgotten in the GDPR: An uneasy relationship." In: *Computer Law & Security Review* 34.6 (2018), pp. 1247 –1257. ISSN: 0267-3649. DOI: 10.1016/j.clsr.2018.08.006.

[216] K Pommerening and M Reng. *Secondary use of the electronic health record via pseudonymisation Medical and Care Compunetics, vol. 1*. 2004.

[217] Klaus Pommerening. "Medical Requirements for Data Protection." In: *IFIP Congress (2)*. 1994, pp. 533–540.

[218] Giorgos Poulis, Aris Gkoulalas-Divanis, Grigorios Loukides, Spiros Skiadopoulos, and Christos Tryfonopoulos. *The SECRETA system*. Last accessed: 11.09.2019. 2014. URL: http://users.uop.gr/~poulis/SECRETA/index.html.

[219] Calvin S Powers, Paul Ashley, and Matthias Schunter. "Privacy promises, access control, and privacy management. Enforcing privacy throughout an enterprise by extending access control." In: *Electronic Commerce, 2002. Proceedings. Third International Symposium on*. IEEE. 2002, pp. 13–21.

[220] F. Prasser, F. Kohlmayer, and K. A. Kuhn. "A Benchmark of Globally-Optimal Anonymization Methods for Biomedical Data." In: *2014 IEEE 27th International Symposium on Computer-Based Medical Systems*. May 2014, pp. 66–71. DOI: 10.1109/CBMS.2014.85.

[221] Fabian Prasser. *ARXConfiguration*. Last accessed: 11.09.2019. Dec. 2018. URL: https://github.com/arx-deidentifier/.

[222] Fabian Prasser and Florian Kohlmayer. *ARX-Developement-API*. Last accessed: 11.09.2019. 2018. URL: https://arx.deidentifier.org/development/api/.

[223] Fabian Prasser, Florian Kohlmayer, Ronald Lautenschlaeger, and Klaus A Kuhn. "Arx-a comprehensive tool for anonymizing biomedical data." In: *AMIA Annual Symposium Proceedings*. Vol. 2014. American Medical Informatics Association. 2014, p. 984.

[224] V. Primault, S. B. Mokhtar, and L. Brunie. "Privacy-Preserving Publication of Mobility Data with High Utility." In: *2015 IEEE 35th International Conference on Distributed Computing Systems*. June 2015, pp. 802–803. DOI: 10.1109/ICDCS.2015.117.

[225] Konstantinos Psaraftis, Theodoros Anagnostopoulos, and Klimis Ntalianis. "Customized Recommendation System for Optimum Privacy Model Adoption." In: *International Journal of Economics and Management Systems*. Vol. 3. Aug. 2018, pp. 155 –163. URL: https://www.iaras.org/iaras/filedownloads/ijems/2018/007-0023(2018).pdf.

[226] Balaji Raghunathan. *The complete book of data anonymization: from planning to implementation*. Auerbach Publications, 2013.

[227] Philip Raschke, Axel Küpper, Olha Drozd, and Sabrina Kirrane. "Designing a GDPR-Compliant and Usable Privacy Dashboard." In: *Privacy and Identity Management. The Smart Revolution: 12th IFIP WG 9.2, 9.5, 9.6/11.7, 11.6/SIG 9.2.2 International Summer School, Ispra, Italy, September 4-8, 2017, Revised Selected Papers*. Ed. by Marit Hansen, Eleni Kosta, Igor Nai-Fovino, and Simone Fischer-Hübner. Cham: Springer International Publishing, 2018, pp. 221–236. ISBN: 978-3-319-92925-5. DOI: 10.1007/978-3-319-92925-5_14.

[228] Vibhor Rastogi and Suman Nath. "Differentially Private Aggregation of Distributed Time-series with Transformation and Encryption." In: *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*. SIGMOD '10. New York, NY, USA: ACM, 2010, pp. 735–746. ISBN: 978-1-4503-0032-2. DOI: 10.1145/1807167.1807247.

[229] Vibhor Rastogi, Dan Suciu, and Sungho Hong. "The Boundary Between Privacy and Utility in Data Publishing." In: *Proceedings of the 33rd International Conference on Very Large Data Bases*. VLDB '07. Vienna, Austria: VLDB Endowment, 2007, pp. 531–542. ISBN: 978-1-59593-649-3. URL: http://dl.acm.org/citation.cfm?id=1325851.1325913.

[230] S Ram Prasad Reddy, K VSVN Raju, and V Valli Kumari. "A Novel Approach for Personalized Privacy Preserving Data Publishing with Multiple Sensitive Attributes." In: *International Journal of Engineering & Technology* 7.2.20 (2018), pp. 197–206. ISSN: 2227-524X. DOI: 10.14419/ijet.v7i2.20.13296. URL: https://www.sciencepubco.com/index.php/ijet/article/view/13296.

[231] R. L. Rivest, A. Shamir, and L. Adleman. "A Method for Obtaining Digital Signatures and Public-key Cryptosystems." In: *Commun. ACM* 26.1 (Jan. 1983), pp. 96–99. ISSN: 0001-0782. DOI: 10.1145/357980.358017. URL: http://doi.acm.org/10.1145/357980.358017.

[232] R. Rivest. *RFC 1321 - The MD5 Message-Digest Algorithm*. Tech. rep. Network Working Group, Apr. 1992. DOI: 10.17487/RFC1321. URL: https://tools.ietf.org/html/rfc1321.

[233] Arianna Rossi and Monica Palmirani. "From Words to Images Through Legal Visualization." In: *AI Approaches to the Complexity of Legal Systems*. Springer, 2015, pp. 72–85.

[234] Arianna Rossi and Monica Palmirani. "DaPIS: a Data Protection Icon Set to Improve Information Transparency under the GDPR." In: *Knowledge of the Law in the Big Data Age. Frontiers*

in Artificial Intelligence and Applications 317 (2019), pp. 181 –195. DOI: 10.3233/FAIA190020.

[235] Cristina Rottondi, Giulia Mauri, and Giacomo Verticale. "A data pseudonymization protocol for smart grids." In: *2012 IEEE Online Conference on Green Communications (GreenCom)*. IEEE. 2012, pp. 68–73.

[236] S. Ruj, M. Stojmenovic, and A. Nayak. "Privacy Preserving Access Control with Authentication for Securing Data in Clouds." In: *2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)*. May 2012, pp. 556–563. DOI: 10.1109/CCGrid.2012.92.

[237] Jeena Mariam Saji, Kalyani Bhongle, Sharayu Mahajan, Soumya Shrivastava, and Ashwini Jarali. "Advancement in Personalized Web Search Engine with Customized Privacy Protection." In: *Progress in Intelligent Computing Techniques: Theory, Practice, and Applications*. Ed. by Pankaj Kumar Sa, Manmath Narayan Sahoo, M. Murugappan, Yulei Wu, and Banshidhar Majhi. Singapore: Springer Singapore, 2018, pp. 405–413. ISBN: 978-981-10-3376-6.

[238] P. Samarati. "Protecting respondents identities in microdata release." In: *IEEE Transactions on Knowledge and Data Engineering* 13.6 (Nov. 2001), pp. 1010–1027. ISSN: 1041-4347. DOI: 10.1109/69.971193.

[239] Pierangela Samarati and Latanya Sweeney. "Generalizing Data to Provide Anonymity when Disclosing Information (Abstract)." In: *Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*. PODS '98. New York, NY, USA: ACM, 1998, pp. 188–. ISBN: 0-89791-996-3. DOI: 10.1145/275487.275508.

[240] Pierangela Samarati and Latanya Sweeney. *Protecting Privacy when Disclosing Information: k-Anonymity and Its Enforcement through Generalization and Suppression*. Tech. rep. Computer Science Laboratory, SRI International, 1998. URL: http://www.csl.sri.com/papers/sritr-98-04/.

[241] Bruce Schneier. "Description of a new variable-length key, 64-bit block cipher (Blowfish)." In: *Fast Software Encryption*. Ed. by Ross Anderson. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 191–204. ISBN: 978-3-540-48456-1.

[242] M. Sehatkar and S. Matwin. "HALT: Hybrid anonymization of longitudinal transactions." In: *2013 Eleventh Annual Conference on Privacy, Security and Trust*. July 2013, pp. 127–134. DOI: 10.1109/PST.2013.6596046.

[243] S. Sen, S. Guha, A. Datta, S. K. Rajamani, J. Tsai, and J. M. Wing. "Bootstrapping Privacy Compliance in Big Data Systems." In: *2014 IEEE Symposium on Security and Privacy*. May 2014, pp. 327–342. DOI: 10.1109/SP.2014.28.

[244] Kumar Sharad and George Danezis. "An Automated Social Graph De-anonymization Technique." In: *Proceedings of the 13th Workshop on Privacy in the Electronic Society*. WPES '14. New York, NY, USA: ACM, 2014, pp. 47–58. ISBN: 978-1-4503-3148-7. DOI: 10.1145/2665943.2665960.

[245] Elaine Shi, HTH Chan, Eleanor Rieffel, Richard Chow, and Dawn Song. "Privacy-preserving aggregation of time-series data." In: *Annual Network & Distributed System Security Symposium (NDSS)*. Internet Society. 2011.

[246] Irina Shklovski, Scott D Mainwaring, Halla Hrund Skúladóttir, and Höskuldur Borgthorsson. "Leakiness and creepiness in app space: Perceptions of privacy and mobile app use." In: *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*. ACM. 2014, pp. 2347–2356.

[247] Ben Shneiderman. "The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations." In: *The Craft of Information Visualization*. Ed. by Benjamin B. Bederson and Ben Shneiderman. Interactive Technologies. San Francisco: Morgan Kaufmann, 2003, pp. 364 –371. ISBN: 978-1-55860-915-0. DOI: 10.1016/B978-155860915-0/50046-9. URL: http://www.sciencedirect.com/science/article/pii/B9781558609150500469.

[248] George L. Sicherman, Wiebren De Jonge, and Reind P. Van de Riet. "Answering Queries Without Revealing Secrets." In: *ACM Trans. Database Syst.* 8.1 (Mar. 1983), pp. 41–59. ISSN: 0362-5915. DOI: 10.1145/319830.319833. URL: http://doi.acm.org/10.1145/319830.319833.

[249] Yogesh L Simmhan, Beth Plale, and Dennis Gannon. "Karma2: Provenance management for data-driven workflows." In: *International Journal of Web Services Research (IJWSR)* 5.2 (2008), pp. 1–22.

[250] NIST-FIPS Standard. "Announcing the advanced encryption standard (AES)." In: *Federal Information Processing Standards Publication* 197.1-51 (2001), pp. 3–3.

[251] Jeff Stapleton and Ralph Spencer Poore. "Tokenization and Other Methods of Security for Cardholder Data." In: *Information Security Journal: A Global Perspective* 20.2 (2011), pp. 91–99. DOI: 10.1080/19393555.2011.560923. eprint: https://doi.org/10.1080/19393555.2011.560923.

[252] Konrad Stark. *Open Anonymizer*. Last accessed: 11.09.2019. 2009. URL: https://sourceforge.net/projects/openanonymizer/.

[253] Nili Steinfeld. "'I agree to the terms and conditions': (How) do users read privacy policies online? An eye-tracking experiment." In: *Computers in Human Behavior* 55 (2016), pp. 992–1000. ISSN: 07475632.

[254] Marc Martinus Jacobus Stevens et al. *Attacks on hash functions and applications*. Mathematical Institute, Faculty of Science, Leiden University, 2012.

[255] William H. Stufflebeam, Annie I. Antón, Qingfeng He, and Neha Jain. "Specifying Privacy Policies with P3P and EPAL: Lessons Learned." In: *Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society*. WPES '04. New York, NY, USA: ACM, 2004, pp. 35–35. ISBN: 1-58113-968-3. DOI: 10.1145/1029179.1029190.

[256] Dan Svirsky. "Why Are Privacy Preferences Inconsistent?" In: *The Harvard John M. Olin Fellow's Discussion Paper Series* 81 (Jan. 2019). ISSN: 1936-5357.

[257] L. Sweeney. "k-Anonymity: A model for protecting privacy." In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10.05 (2002), pp. 557–570. DOI: 10.1142/S0218488502001648.

[258] Latanya Sweeney. "Achieving k-Anonymity Privacy Protection using Generalization and Suppression." In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10.05 (2002), pp. 571–588. DOI: 10.1142/S021848850200165X. eprint: https://doi.org/10.1142/S021848850200165X.

[259] Matthias Templ, Alexander Kowarik, and Bernhard Meindl. *sdcMicro: Statistical Disclosure Control Methods for Anonymization of Microdata and Risk Estimation*. Last accessed: 11.09.2019. 2017. URL: https://cran.r-project.org/web/packages/sdcMicro/index.html.

[260] Robert Thibadeau. "A Critique of P3P: Privacy on the Web." In: *The eCommerce Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh* 24 (2000).

[261] S. Trabelsi, J. Sendor, and S. Reinicke. "PPL: PrimeLife Privacy Policy Engine." In: *2011 IEEE International Symposium on Policies for Distributed Systems and Networks*. June 2011, pp. 184–185. DOI: 10.1109/POLICY.2011.24.

[262] T. M. Truta and B. Vinay. "Privacy Protection: p-Sensitive k-Anonymity Property." In: *22nd International Conference on Data Engineering Workshops (ICDEW'06)*. Apr. 2006, pp. 94–94. DOI: 10.1109/ICDEW.2006.116.

[263] U.S. Bureau of Labor Statistics, American Time Use Survey 2 Massachusetts Avenue, NE Suite 4675 Washington, DC 20212-0001. *American Time Use Survey (ATUS)*. Last accessed: 11.09.2019. URL: https://www.bls.gov/tus/.

[264] Max-R. Ulbricht and Frank Pallas. "YaPPL - A Lightweight Privacy Preference Language for Legally Sufficient and Automated Consent Provision in IoT Scenarios." In: *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Ed. by Joaquin Garcia-Alfaro, Jordi Herrera-Joancomartí, Giovanni Livraga, and Ruben Rios. Cham: Springer International Publishing, 2018, pp. 329–344. ISBN: 978-3-030-00305-0.

[265] Henk CA Van Tilborg and Sushil Jajodia. *Encyclopedia of cryptography and security*. Springer Science & Business Media, 2014.

[266] Nataraj Venkataramanan and Ashwin Shriram. *Data privacy: principles and practice*. Chapman and Hall/CRC, 2016.

[267] J. Wang, Y. Luo, Y. Zhao, and J. Le. "A Survey on Privacy Preserving Data Mining." In: *2009 First International Workshop on Database Technology and Applications*. Apr. 2009, pp. 111–114. DOI: 10.1109/DBTA.2009.147.

[268] Ke Wang and Benjamin C. M. Fung. "Anonymizing Sequential Releases." In: *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '06. New York, NY, USA: ACM, 2006, pp. 414–423. ISBN: 1-59593-339-5. DOI: 10.1145/1150402.1150449.

[269] Ke Wang, Benjamin C. M. Fung, and Philip S. Yu. "Handicapping attacker's confidence: an alternative to k-anonymization." In: *Knowledge and Information Systems* 11.3 (Apr. 2007), pp. 345–368. ISSN: 0219-3116. DOI: 10.1007/s10115-006-0035-5.

[270] Lun Wang, Joseph P. Near, Neel Somani, Peng Gao, Andrew Low, David Dao, and Dawn Song. "Data Capsule: A New Paradigm for Automatic Compliance with Data Privacy Regulations." In: *arXiv e-prints*, arXiv:1909.00077 (Aug. 2019), arXiv:1909.00077. arXiv: 1909.00077 [cs.CY].

[271] Pete Warden. *Why you can't really anonymize your data-It's time to accept and work within the limits of data anonymization*. May 2011. URL: https://www.oreilly.com/ideas/anonymize-data-limits.

[272] E. Welbourne, L. Battle, G. Cole, K. Gould, K. Rector, S. Raymer, M. Balazinska, and G. Borriello. "Building the Internet of Things Using RFID: The RFID Ecosystem Experience." In: *IEEE Internet Computing* 13.3 (May 2009), pp. 48–55. ISSN: 1089-7801. DOI: 10.1109/MIC.2009.52.

[273] Lorrie Cranor & Brooks Dobbs & Serge Egelman & Giles Hogben & Jack Humphrey & Marc Langheinrich & Massimo Marchior & Martin Presler-Marshall & Joseph Reagle & Matthias Schunter & David A. Stampley & Rigo Wenning. *The Platform for Privacy Preferences 1.1 (P3P1.1) Specification*. Tech. rep. W3C, 2006. URL: https://www.w3.org/TR/P3P11/.

[274] Alan F. Westin. "Privacy and Freedom." In: *Atheneum Press* (1967).

[275] Patrick Westphal, Javier David Fernandez Garcia, Sabrina Kirrane, and Jens Lehmann. "SPIRIT: A Semantic Transparency and Compliance Stack." EN. In: *Proceedings of the 14th International Conference on Semantic Systems*. CEUR Workshop Proceedings. Aachen, 2018, pp. 1–1. URL: http://epub.wu.ac.at/id/eprint/6491.

[276] Sebastian Wilhelm and Armin Gerl. "Policy-based Authentication and Authorization based on the Layered Privacy Language." In: *Datenbanksysteme für Business, Technologie und Web (BTW 2019)*. Gesellschaft für Informatik, Bonn, 2019.

[277] Xiaokui Xiao and Yufei Tao. "Personalized Privacy Preservation." In: *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data*. SIGMOD '06. New York, NY, USA: ACM, 2006, pp. 229–240. ISBN: 1-59593-434-0. DOI: 10.1145/1142473.1142500.

[278] Yang Yang, Xianghan Zheng, Wenzhong Guo, Ximeng Liu, and Victor Chang. "Privacy-preserving smart IoT-based healthcare big data storage and self-adaptive access control system." In: *Information Sciences* 479 (2019), pp. 567 –592. ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2018.02.005. URL: http://www.sciencedirect.com/science/article/pii/S0020025518300860.

[279] T. You, W. Peng, and W. Lee. "Protecting Moving Trajectories with Dummies." In: *2007 International Conference on Mobile Data Management*. May 2007, pp. 278–282. DOI: 10.1109/MDM.2007.58.

[280] Ting Yu, Ninghui Li, and Annie I. Antón. "A Formal Semantics for P3P." In: *Proceedings of the 2004 Workshop on Secure Web Service*. SWS '04. New York, NY, USA: ACM, 2004, pp. 1–8. ISBN: 1-58113-973-X. DOI: 10.1145/1111348.1111349.

[281] R. Yue, Y. Li, T. Wang, and Y. Jin. "An Efficient Adaptive Graph Anonymization Framework for Incremental Data Publication." In: *2018 5th International Conference on Behavioral, Economic, and Socio-Cultural Computing (BESC)*. Nov. 2018, pp. 103–108. DOI: 10.1109/BESC.2018.8697301.

[282] K. Zhao and L. Ge. "A Survey on the Internet of Things Security." In: *2013 Ninth International Conference on Computational Intelligence and Security*. Dec. 2013, pp. 663–667. DOI: 10.1109/CIS.2013.145.

[283] Bin Zhou, Jian Pei, and WoShun Luk. "A Brief Survey on Anonymization Techniques for Privacy Preserving Publishing of Social Network Data." In: *SIGKDD Explor. Newsl.* 10.2 (Dec. 2008), pp. 12–22. ISSN: 1931-0145. DOI: 10.1145/1540276.1540279.

[284] Ye Zhu, Yongjian Fu, and Huirong Fu. "On Privacy in Time Series Data Mining." In: *Advances in Knowledge Discovery and Data Mining*. Ed. by Takashi Washio, Einoshin Suzuki, Kai Ming Ting, and Akihiro Inokuchi. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 479–493. ISBN: 978-3-540-68125-0.

[285] G. Zyskind, O. Nathan, and A. '. Pentland. "Decentralizing Privacy: Using Blockchain to Protect Personal Data." In: *2015 IEEE Security and Privacy Workshops*. May 2015, pp. 180–184. DOI: 10.1109/SPW.2015.27.

FOLIO ADMINISTRATIF

THESE DE L'UNIVERSITE DE LYON OPEREE AU SEIN DE L'INSA LYON

NOM : Gerl

DATE de SOUTENANCE : 05.12.2019

Prénoms : Armin

TITRE : Modelling of a Privacy Language and Efficient Policy-based De-identification

NATURE : Doctorat

Numéro d'ordre : 2019LYSEI105

Ecole doctorale : Informatique et Mathématiques

Spécialité : Informatique

RESUME :
The processing of personal information is omnipresent in our data-driven society enabling personalized services, which are regulated by privacy policies. Although privacy policies are strictly defined by the General Data Protection Regulation (GDPR), no systematic mechanism is in place to enforce them. Especially if data is merged from several sources into a data set with different privacy policies associated, the management and compliance to all privacy requirements is challenging during the processing of the data set. Privacy policies can vary hereby due to different policies for each source or personalization of privacy policies by individual users. Thus, the risk for negligent or malicious processing of personal data due to defiance of privacy policies exists.

To tackle this challenge, we propose a privacy-preserving framework. Within this framework privacy policies are expressed in the proposed \emph{Layered Privacy Language (LPL)} which allows to specify legal privacy policies and privacy-preserving de-identification methods. The policies are enforced by a \emph{Policy-based De-identification (PD)} process. The PD process enables efficient compliance to various privacy policies simultaneously while applying pseudonymization, personal privacy anonymization and privacy models for de-identification of the data-set. Thus, the privacy requirements of each individual privacy policy are enforced filling the gap between legal privacy policies and their technical enforcement.

MOTS-CLÉS :
De-identification, GDPR, Privacy Language

Laboratoire (s) de recherche : Laboratoire d'InfoRmatique en Image et Systèmes d'information

Directeur de thèse:

Brunie, Lionel Professeur des Universités	INSA Lyon	Co-directeur de thèse
Kosch, Harald Professeur des Universités	Universität Passau	Co-directeur de thèse
Bennani, Nadia Maître de Conférences	NSA Lyon	Co-directeurice de thèse

Président de jury :

Composition du jury :

Bertino, Elisa Directeur de Recherche	Purdue University	Rapporteure
Benzekri, Abdelmalek Professeur des Universités	Université Paul Sabatier-Toulouse	Rapporteur
Granitzer, Michael Professeur des Universités	Universität Passau	Examinateur
Lenz, Richard Professeur des Universités	Friedrich-Alexander Universität	Examinateur
Cuppens, Frédéric Professeur des Universités	Télécom Bretagne	Examinateur

INSA | INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

DEPARTEMENT FEDORA
FORMATION PAR LA RECHERCHE
ET ÉTUDES DOCTORALES
Bâtiment INSA direction

Année Universitaire 2017-2018

ECOLES DOCTORALES – DISCIPLINES
A REMPLIR LORS DE VOTRE INSCRIPTION

Doctorant :
Nom : Gerl Prénom : Armin **Signature :**

Directeur de thèse :
Nom : Brunet Prénom : Lionel **Signature :**

UNE SEULE DISCIPLINE POSSIBLE, à choisir parmi la liste suivante :

ECOLES DOCTORALES n° code national	DISCIPLINES	Cocher la case correspondante
ED CHIMIE DE LYON (Chimie, Procédés, Environnement) EDA206	Chimie	☐
	Procédés	☐
	Environnement	☐
HISTOIRE, GEOGRAPHIE, AMENAGEMENT, URBANISME, ARCHEOLOGIE, SCIENCE POLITIQUE, SOCIOLOGIE, ANTHROPOLOGIE (ScSo) EDA483	Géographie – Aménagement - Urbanisme	☐
ELECTRONIQUE, ELECTROTECHNIQUE, AUTOMATIQUE (E.E.A.) EDA160	Automatique	☐
	Génie Electrique	☐
	Electronique, micro et nanoélectronique, optique et laser	☐
	Ingénierie pour le vivant	☐
	Traitement du Signal et de l'Image	☐
EVOLUTION, ECOSYSTEMES, MICROBIOLOGIE, MODELISATION (E2M2) EDA 341	Paléoenvironnements et évolution	☐
	Micro-organismes, interactions, infections	☐
	Biologie Evolutive, Biologie des Populations, écophysiologie	☐
	Biomath-Bioinfo-Génomique évolutive	☐
	Ecologie des communautés, fonctionnement des écosystèmes, écotoxicologie	☐
INFORMATIQUE ET MATHEMATIQUES DE LYON (InfoMaths) EDA 512	Informatique	☒
	Informatique et applications	☐
	Mathématiques et applications	☐
	Génie Industriel	☐
INTERDISCIPLINAIRE SCIENCES-SANTE (EDISS) EDA205	Biochimie	☐
	Physiologie	☐
	Ingénierie biomédicale	☐
ED MATERIAUX DE LYON EDA 034	Matériaux	☐
MEGA DE LYON (MECANIQUE, ENERGETIQUE, GENIE CIVIL, ACOUSTIQUE) (MEGA) EDA162	Mécanique des Fluides	☐
	Génie Mécanique	☐
	Biomécanique	☐
	Thermique Energétique	☐
	Génie Civil	☐
	Acoustique	☐

Pour le Directeur de l'INSA Lyon
Et par délégation

Florence POPOWYCZ
Directrice du Département FEDORA
Formation par la Recherche Et des Etudes Doctorales

Cette liste est mise à jour annuellement par le Département FEDORA en collaboration avec les Ecoles Doctorales.