Génération de systèmes de gestion de données personnalisés fondée sur les ontologies : une application à la physique expérimentale des particules

Blerina Gkotse

PSL UNIVERSITÉ PARIS

THÈSE DE DOCTORAT

DE L'UNIVERSITÉ PSL

Préparée à MINES ParisTech

Gèneration de systèmes de gestion de donées personnalisés fondée sur les ontologies: une application à la physique expérimentale des particules

Ontology-based Generation of Personalised Data Management Systems: an Application to Experimental Particle Physics

Soutenue par

Blerina GKOTSE

Le 25 Septembre 2020

École doctorale n°621

Ingénierie des systèmes, matériaux, mécanique, energétique

Spécialité

Informatique temps-réel, robotique et automatique

Composition du jury :

Laurent DUSSEAU Professeur, Université de Montpellier	*Président de jury*
Laura GONELLA Maître de conférences, University of Birmingham	*Rapporteur*
Jean-Baptiste LAMY Maître de conférences, Université Paris 13	*Rapporteur*
Theodora VARVARIGOU Professeur, National Technical University of Athens	*Examinateur*
Federico RAVOTTI Docteur, CERN	*Examinateur*
Pierre JOUVELOT Directeur de recherche, MINES ParisTech	*Directeur de thèse*

MINES ParisTech | PSL

Acknowledgements

This Ph.D. thesis has received funding from the European Union's Horizon 2020 Research and Innovation programme under Grant Agreement no. 654168.

In my acknowledgements, I would like, first of all, to thank Pierre Jouvelot my thesis director in MINES ParisTech, PSL University and Federico Ravotti, my CERN supervisor, for their support and guidance throughout these years. Their help and suggestions were precious for this thesis work and made me understand the essence of research and how it is performed.

I would like to thank Laura Gonella and Jean-Baptiste Lamy for reviewing this manuscript and providing me with valuable input. Moreover, I thank Theodora Varvarigou and Laurent Dusseau for participating in my thesis jury.

In addition, I would like to thank the CRI team of MINES ParisTech and more specifically Francois Irigoin, Jérôme Adnot, Olivier Hermant and Bruno Sguerra. I'd also like to express my gratitude to the CERN EP-DT group and in particular Petra Riedler, Michael Moll, Burkhard Schmidt and Mar Capeans.

I am also very grateful of being a member of the CERN IRRAD team and working with great colleagues. More specifically, I'd like to thank Maurice, who guided me through my CERN Technical Student projects and taught me a lot of useful things, and Giuseppe, for being always very helpful and resolving all kinds of problems. I would also like to thank Georgi, Isidre, Rania, Viktoria, Jes, Emanuele and Martin for providing feedback to my work. Moreover, I'd like to thank the students Katarina, Konul, Alexander, Alfredo, Sabina, Magnus and Jose who are working or worked with me. Special thanks also to Dorota, for the support she has given to us all these years.

Except from my IRRAD family, I would also like to thank my real family, my parents, that provided me with a proper education, pushed me forward and always believed in me.

Special thanks to my best friends in Greece, Eleni, Giorgos, Angela, Foteini, Katerina and Efthymis. Also, to my good friends in Geneva, Marcos, Johanie, Moritz, Lucia, Clara and Nicola.

Last but not least, I would like to express my gratitude to Grigoris for encouraging me and believing in me throughout the years.

Contents

Acknowledgements i

1 Introduction 1

1.1 Objectifs . . . 2
1.2 Contributions . . . 3
1.3 Plan . . . 3
1.4 Liste de publications . . . 4
1.5 Goals . . . 6
1.6 Thesis Contributions . . . 7
1.7 Outline . . . 8
1.8 Published Work . . . 9

2 Background 10

2.1 Web Semantics . . . 11
2.1.1 Definitions . . . 11
2.1.2 Ontology Languages . . . 15
2.1.3 Types of Ontologies . . . 16
2.1.4 Ontology Development and Visualisation Tools . . . 17
2.2 Irradiation Experiments . . . 17
2.2.1 Irradiation Facilities and Online Database . . . 19
2.2.2 Experimental Particle Physics (EPP) . . . 21
2.2.3 CERN and Proton Irradiation Facility (IRRAD) . . . 23

3 Related Work 26

3.1 Data Management Systems in EPP . . . 27
3.1.1 Digital Users Office (DUO) . . . 28
3.1.2 Monitoring and Visualisation Tools . . . 29
3.1.3 Reporting Tools . . . 31

3.2 Domain Ontologies in EPP . . . 33
3.3 Ontology-based Data Management Systems . . . 34
3.4 UI Customisation and Personalisation . . . 35
3.4.1 User-Experience Research Findings . . . 37
3.4.2 UI Customisation . . . 41
3.4.3 Recommendation Models . . . 42

4 Data Management System Generation with GenAppi 44
4.1 Ontology-based Web Application Generation Ontology (OWAO) 45
4.2 Data Management Web Application Generation (GenAppi) . . . 47
4.2.1 Algorithm and Workflow . . . 47
4.2.2 Discussion . . . 50
4.3 UI Customisation and Personalisation . . . 51
4.3.1 Recommendation Generation . . . 51
4.3.2 Popularity Model . . . 53
4.3.3 ontowalk2vec, a New Ontology Embedding Model . . . 54
4.4 Summary . . . 59

5 Application of GenAppi to EPP Data Management 60
5.1 IRRAD Data Manager (IDM) . . . 61
5.1.1 Design Life-Cycle . . . 61
5.1.2 Development . . . 62
5.1.3 Architecture . . . 64
5.1.4 Deployment . . . 65
5.1.5 Functionalities . . . 66
5.2 Irradiation Experiment Data Management Ontology (IEDM) . . 69
5.2.1 Imported Ontologies . . . 70
5.2.2 Design Methodology . . . 71
5.2.3 Core Structure . . . 73
5.2.4 FCC-Radmon, a Use Case of Irradiation Experiment . . . 77
5.3 Comparison of GenAppi-Generated IEDM-Based Data Management System and IDM . . . 78
5.3.1 IEDM-dedicated Data Management System . . . 78

5.3.2 Comparison with IDM 80
5.4 Validation with Other Ontologies 83
5.5 Summary 84

6 UI Personalisation with ontowalk2vec 85
6.1 Classification Techniques and Evaluation Metrics 86
6.2 Algorithm for ontowalk2vec and Classification 90
6.3 Experiment with the MUTAG ontology 91
6.3.1 Experimental Setup 92
6.3.2 Results 92
6.4 Using ontowalk2vec with OWAO 98
6.4.1 Experimental Setup 98
6.4.2 Experiment 1: Classification of Preferences 100
6.4.3 Experiment 2: Classification of Popularity 102
6.4.4 Non-Zero Dataset 104
6.4.5 Cosine Similarity of OWAO Embeddings 106
6.5 Optimisation 107
6.5.1 Experimental Setup 108
6.5.2 Evaluation Metrics and Results 110
6.6 Summary 113

7 Conclusion 114
7.1 Contributions 114
7.2 Perspectives 116
7.3 Épilogue 117
7.4 Thesis Contributions 118
7.5 Perspectives 119
7.6 Epilogue 120

Bibliography 122

Glossary 137

Acronyms 138

A Irradiation Facilities Database and Website **A-1**

A.1 Functionalities A-1

A.2 Database Content A-2

B Test Beam Facilities Database and Website **B-1**

B.1 Functionalities B-1

B.2 Database Content B-1

C IRRAD Data Manager (IDM) Installation **C-1**

List of Figures

2.1 W3C Semantic Web Stack or Layer Cake for the years 2000, from Fabien Gandon "A Survey of the First 20 Years of Research on Semantic Web and Linked Data" [Gan18] (Names explained in the Glossary and Acronyms sections). 12

2.2 A new version of the Semantic Web Stack in 2018, from Fabien Gandon "A Survey of the First 20 Years of Research on Semantic Web and Linked Data" [Gan18] (Names explained in the Glossary and Acronyms sections). 13

2.3 Excerpt from Google Knowledge Graph showing information related to some artists (e.g., Michelangelo) and the connections among them. 14

2.4 Excerpt from SOSA (Sensors, Observations, Samples and Actuators) ontology containing some instances. 15

2.5 The wine ontology as illustrated through the Protégé platform. On the left section the ontology classes are shown (yellow points), while individual instances and their properties and annotations are shown in the rest of the interface. 18

2.6 Bottle sterilisation by electron-beam irradiation. 19

2.7 Online Irradiation Facilities Database platform containing data entries for irradiation facilities worldwide. 20

2.8 The CERN accelerator complex [Mob16]. 23

2.9 IRRAD Facility irradiation zone. 24

3.1 SOLARIS Digital Users Office (DUO) [SLR]. 28

3.2 Web Extensible Display Manager used at Jefferson Lab for control systems display (from R.J. Slominski and T. Larrieu., Web Extensible Display Manager 2 [SL19]). 30

3.3 Default Tornado DS index page (from M. Broseta et al., A Web-Based Report Tool for Tango Control Systems via WebSockets [BRB+17]). 32

3.4 A schematic view of the DetectorFinalState ODP (from David Carral et al, "An Ontology Design Pattern for Particle Physics Analysis" [CCDT+15]). 33

3.5 A screenshot of the Epic charting software used by many hospitals in the US. The black arrow shows where the alerts are shown (from C. Savard S. Jonathan Shariat, "Tragic Design: The Impact of Bad Product Design and How to Fix It" [SS17]). 36

3.6 Contrast table (from Jeremy Girard, "How to Contrast Background and Foreground Colors in Web Design, 2019" [Gir19]). . . 39

3.7 Gutenberg Diagram (from Steven Bradley, Design Fundamentals [Bra18]). 40

3.8 F Pattern (from Steven Bradley, Design Fundamentals [Bra18]). . 40

3.9 Z Pattern (from Steven Bradley, Design Fundamentals [Bra18]). . 40

3.10 F reading pattern eye tracking by Nielsen Norman Group [PWNG14]. 41

4.1 OWAO excerpt (light-blue colour: Domain Ontology-related entities, yellow colour: Model related-entities, green colour: Web Application-related entities, dark-blue colour: Operation-related entities, orange colour: UI widget-related entities). 45

4.2 GenAppi generator workflow. 48

4.3 OWAO part for describing UI preferences. 52

4.4 Recommender model. The user is shown different UI Styles for a specific UI and chooses his/her preference, which is provided as a feedback to OWAO. 54

4.5 word2vec architectures. The CBOW architecture predicts the current word based on the context; the Skip-gram mode predicts surrounding words, given a current word [MCCD13]. 56

4.6 Starting from node u, a Breadth-first search (BFS) traversal is depicted in red arrows, while a Depth-first search (DFS) path is depicted in blue allows (from A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks" [GL16]). 56

4.7 node2vec random walk generation based on the hyperparameters p and q. The walk transitions from t to v and is evaluating its next possible steps out of node v (from A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks" [GL16]). . . . 57

4.8 Embedding workflow with ontowalk2vec. The OWAO ontology is used as an input for node2vec and RDF2Vec where node2vec random walks and RDF triples are extracted and used as input sentences to word2vec Skip-gram. The generated word2vec feature vectors can then be used for a recommender system. 58

5.1 Balsamiq prototypes. 63
5.2 IDM architecture. 65
5.3 IDM functionalities. 66
5.4 IEDM online documentation. 72
5.5 Graph representation in Protégé of the IEDM ontology, with the focus on the `iedm:IrradiationExperiment` class. The left column illustrates the classes of IEDM while the right column, the object properties. 76
5.6 Login page of the IEDM-dedicated application. 79
5.7 Registering a new sample in the IEDM-dedicated application. . . 79
5.8 WebVOWL visualisation page of IEDM from the generated application. 80
5.9 "Create user instance" form of IDM, at the top; generated web application version, at the bottom. 81
5.10 List user interface of IDM at the top, generated web application at the bottom. 82
5.11 The List view of a class *BT33_Marine_Animal_Type* of the generated web application using MarineTLO as an input ontology. . 83

6.1 MNIST classification with t-SNE (from L. van der Maaten and G. Hinton, "Visualizing data using t-SNE" [MH08]). 88
6.2 Cosine similarity. 89
6.3 Excerpt from MUTAG depicting am instance of Compound (d30). 93
6.4 Accuracy plot for the different methodologies. 94
6.5 node2vec t-SNE visualisation of mutagenic and non-mutagenic elements. 96
6.6 RDF2Vec t-SNE visualisation of mutagenic and non-mutagenic elements. 96
6.7 ontowalk2vec t-SNE classification of mutagenic and non-mutagenic elements. 97

6.8 Excerpt from OWAO depicting a User instance (`User1`) and his/her preferences for a given UI component. 99
6.9 t-SNE visualisation for OWAO preferences. 102
6.10 t-SNE visualisation for OWAO popularity. 104
6.11 t-SNE visualisation for OWAO popularity without 0 values. . . . 106
6.12 BFS F1 score histogram. 112
6.13 WL F1 score histogram. 112

A.1 Screenshot of the irradiation facility details provided to users. . . A-3

B.1 A screenshot from the test beam facilities and beamlines view. . B-2
B.2 A screenshot from the Super Proton Synchrotron (SPS) test beam facility data. B-3
B.3 A screenshot from the entry data of the T9 beamline of SPS. . . B-4

List of Tables

5.1 Comparison of GenAppi-Generated IEDM Application against IDM. 82

6.1 Confusion matrix for binary classification. 87
6.2 Models' accuracy scores using the MUTAG dataset. 94
6.3 Models' confusion matrices for the MUTAG ontology. 95
6.4 Background and font colour preferences for ontowalk2vec. 100
6.5 Font size preferences. 100
6.6 Text alignment preferences. 100
6.7 Accuracy for OWAO preferences. 101
6.8 Confusion matrix for OWAO preferences. 101
6.9 Popularity values thresholds. 102
6.10 Accuracy for OWAO popularity. 103
6.11 Confusion matrix of OWAO popularity. 103
6.12 Accuracy for OWAO popularity with non-zero popularity values. 105
6.13 Confusion matrix of OWAO popularity without 0 values. 105
6.14 Top 5 highest cosine similarities for `owao:User251` using BFS. . . 107
6.15 Top 5 highest cosine similarities for owao:User251 using WL. . . 107
6.16 Instances for similarity comparison with `owao:white_black_medium_left`. 109
6.17 Instances for similarity comparison with `owao:User251`. 110
6.18 Hyperparameter and their range, used to form combinations. . . 111
6.19 Top F1 score (> 0.9) for BFS. 112
6.20 Top F1 score (> 0.9) for WL. 112
6.21 Top cosine similarities for `owao:ui_style_white_black_medium_left` using BFS. 113

Chapter 1

Introduction

Version française

De nos jours, le progrès est croissant dans les domaines de la science et de la technologie. Par exemple, dans le domaine de la physique, de grandes infrastructures pour héberger des expériences de physique ont été construites pour explorer la matière et l'univers en profondeur, en fournissant des réponses à certaines questions ouvertes [E+19] tout en laissant d'autres à aborder. Ces efforts se traduisent par la production de volumes importants de données hétérogènes qui doivent être suivies, traitées et analysées. Pendant ce temps, dans le domaine de l'informatique, de nombreux progrès technologiques ont eu un large impact sur tous les aspects de la vie. Les ordinateurs, les smartphones et les applications logicielles permettent, facilitent ou accélèrent des tâches importantes de la vie courante.

Malgré les avancées significatives dans tous les domaines scientifiques, il reste difficile de combler le fossé entre les différents domaines et sous-domaines cités ci-dessus, car les connaissances acquises dans un domaine particulier ne sont pas toujours transférées à d'autres, ce qui entraîne souvent un manque de communication et de fertilisation croisée des idées entre eux. Cela entraîne une perte d'informations et de savoir-faire importante, alors qu'ils aurait pu être utilisés pour développer davantage un domaine scientifique donné. Cet état de fait montre la nécessite de davantage de recherches interdisciplinaires, ce qui est la clé de ce travail de thèse, qui s'appuie sur des idées issues du développement de logiciels en informatique, de la conception d'interfaces orientées utilisateur et de la physique expérimentale.

Les applications logicielles avancées sont généralement des systèmes assez complexes et nécessitent un temps de développement et de test important pour les développeurs, mais également une formation spéciale pour les utilisateurs, qui ont besoin d'apprendre à utiliser de tels systèmes, induisant des coûts en temps et en main-d'œuvre importants. Par conséquent, afin d'obtenir une intégration logicielle meilleure et plus rapide dans les domaines scientifiques, et en particulier en physique expérimentale, il est nécessaire à la fois de minimiser le temps de

développement et de rendre les systèmes logiciels plus intuitifs et conviviaux afin qu'ils puissent être plus facilement adoptés par la population cible visée.

1.1 Objectifs

Dans le domaine de la physique expérimentale des particules (EPP, pour Èxperimental Particle Physics"), une multitude d'expériences sont effectuées chaque jour pour découvrir de nouveaux phénomènes physiques tout en fournissant des explications toujours plus précises de ce qui existe dans le monde physique. A cet effet, la gestion et l'analyse des données hétérogènes issues de telles expériences sont cruciales pour la précision et la reproductibilité des expériences réalisées. Cependant, les questions de la préservation, de l'intégration et de l'intercommunication des données n'ont pas reçu l'attention nécessaire qu'elles méritent, principalement pour les raisons mentionnées dans les paragraphes précédents.

Cette thèse vise à s'appuyer sur les technologies du web sémantique pour la standardisation de la gestion des données dans le domaine des EPP. Les technologies du web sémantique sont connues pour faciliter l'interopérabilité et l'intégration des données, tout en permettant la communication entre les systèmes informatiques. L'un de ces outils est la notion d'ontologie, qui est une structure de connaissances déclarative spécifique à un domaine d'application donné (et expliquée plus en détail dans les chapitres suivants). Les principaux objectifs de cette thèse sont donc les suivants.

- *Le premier objectif de ce travail est, en s'appuyant sur les concepts liés à la notion d'ontologie, de fournir une axiomatisation de certaines notions de gestion des données spécifiques à un certain type d'expériences EPP, à savoir les "expériences d'irradiation". Cet effort aboutit à la formalisation d'une base de connaissances commune sur laquelle nous espérons que pourra s'appuyer la communauté de la physique des irradiations.*

- *Le second objectif de ce travail de thèse est d'automatiser, à partir de cette ontologie, le processus de développement de systèmes dédiés de gestion de données (via une application web). L'application résultante, fondée sur une ontologie pour EPP, peut être considérée comme un cas d'utilisation de l'idée plus générale consistant à utiliser des ontologies pour la génération automatique de systèmes de gestion de données.*

- *Enfin, le troisième objectif de cette thèse est d'améliorer la convivialité de ces systèmes de gestion de données générés automatiquement en introduisant des méthodes de personnalisation de l'interface utilisateur (UI).*

1.2 Contributions

Afin d'atteindre les objectifs mentionnés précédemment, ce travail de thèse comprend les trois principales contributions suivantes.

IEDM *(Irradiation Experiment Data Management ontology - Ontologie de gestion des données d'expériences d'irradiation) L'ontologie IEDM définit les principaux concepts de gestion des données des expériences d'irradiation.*

GenAppi *Il s'agit d'une nouvelle méthodologie de génération d'applications web pour la gestion des données dans le domaine des EPP, fondée sur des ontologies. Il utilise IEDM comme cas d'utilisation pour la génération d'applications Web.*

ontowalk2vec *Cette nouvelle méthode, développée par l'intégration de plusieurs techniques de pointe de traitement automatique du langage naturel (NLP, pour "Natural Language Processing"), peut être utilisée pour améliorer le processus de recommandation personnalisé de fonctionnalités d'interface utilisateur (UI) selon le profil de ce dernier.*

En plus de ces trois contributions fondamentales, des développements plus pratiques sont également à attribuer à ce travail de thèse.

IDM *(IRRAD Data Manager) Cette application web dédiée est actuellement utilisée quotidiennement dans le Centre d'irradiation des protons du CERN (IRRAD) [CER] [GGMR15]. Elle a inspiré le développement d'IEDM et de GenAppi.*

Bases de données et sites Web *de facilités d'irradiation et faisceaux pour la qualification des détecteurs. A la suite des recherches menées pour analyser les expériences d'irradiation effectuées dans les installations d'irradiation du monde entier, les infrastructures dédiées EPP pour la réalisation d' expériences d'irradiation et les infrastructures offrant des faisceaux pour la qualification des détecteurs également utilisées pour les expériences EPP, deux sites web et bases de données distincts ont été développés. Ils contiennent les données accumulées lors de ces recherches.*

1.3 Plan

Ce chapitre d'introduction, le chapitre 1, présente un aperçu général de ce travail de thèse en décrivant les motivations, les objectifs et les principales contributions. Le chapitre 2 fournit les bases nécessaires en sémantique web, EPP et expériences d'irradiation pour profiter pleinement de la lecture de ce manuscrit de thèse. Le lecteur averti est invité à simplement parcourir ces pages.

Le chapitre 3 détaille les travaux connexes dans les domaines des systèmes logiciels pour EPP, les ontologies liées à EPP, la génération d'interfaces utilisateur à partir d'ontologies, la notion d'expérience utilisateur et les systèmes de recommandation utilisés pour la personnalisation des interfaces utilisateur.

Le chapitre 4 présente la méthodologie GenAppi développée pour la génération d'applications web de gestion de données. De plus, une nouvelle ontologie, pour les applications web fondées sur des ontologie (OWAO), utilisée comme couche de support de la méthodologie GenAppi, est introduite. Un autre nouveau concept décrit dans ce chapitre est ontowalk2vec, un algorithme utilisé pour la génération de vecteurs de représentation d'ontologie ("embeddings") pour les systèmes de recommandation (un chapitre complet est ensuite dédié à cette nouvelle technique).

Le chapitre 5 fournit une description détaillée de certaines applications des méthodologies introduites précédemment. Premièrement, l'application web IDM est présentée. Ensuite, l'ontologie IEDM, qui décrit des concepts généraux pour la gestion des données d'expériences d'irradiation, est présentée. Dans la dernière partie de ce chapitre, l'application web générée par l'utilisation de GenAppi sur IEDM est présentée et comparée à l'IDM développé manuellement.

Le chapitre 6 se concentre sur la comparaison d'ontowalk2vec avec les méthodes NLP actuelles de pointe utilisées pour fournir des incorporations ("embeddings") d'ontologies. Notre méthodologie est ensuite testée avec les données de l'ontologie OWAO et évaluée sur la possibilité d'être utilisée pour recommander des fonctionnalités d'interface utilisateur personnalisées en fonction des préférences de l'utilisateur et des similitudes d'interface utilisateur. Enfin, le chapitre 7 présente les conclusions de ce travail de thèse et examine les perspectives futures possibles.

1.4 Liste de publications

Des parties de ce travail de thèse sont liées aux publications suivantes:

- B. Gkotse, P. Jouvelot, and F. Ravotti, "IEDM: An Ontology for Irradiation Experiments Data Management," in The Semantic Web: ESWC2019 Satellite Events. Cham: Springer International Publishing, Portoroz, Slovenia, 2019, pp. 80–83 [GJR19b];
- B. Gkotse, P. Jouvelot, and F. Ravotti, "Automatic Web Application Generation from an Irradiation Experiment Data Management Ontology (IEDM)," in 17th International Conference on Accelerator and Large Experimental Physics Control Systems, ICALEPCS 2019, 2019. [Online]. Available: http://icalepcs2019.vrws.de/papers/tubpl01.pdf [GJR19a];

- B. Gkotse, P. Jouvelot, G. Pezzullo, and F. Ravotti, “The IRRAD Data Manager (IDM),” in Proceedings of ICALEPCS 2019, New York, NY, USA, 2019 [GJPR19];
- B. Gkotse, M. Glaser, P. Jouvelot, E. Matli, G. Pezzullo, and F. Ravotti, “Towards a Unified Environmental Monitoring, Control and Data Management System for Irradiation Facilities: the CERN IRRAD Use Case,” in 17th European Conference on Radiation and Its Effects on Components and Systems (RADECS), 2017, pp. 1–8 [GGJ+17];
- B. Gkotse, M. Brugger, P. Carbonez, S. Danzeca, A. Fabich, R. G. Alia, M. Glaser, G. Gorine, M. R. Jaekel, I. M. Suau, G. Pezzullo, F. Pozzi, F. Ravotti, M. Silari, and M. Tali, “Irradiation Facilities at CERN,” in 17th European Conference on Radiation and Its Effects on Componentsand Systems (RADECS), 2017, pp. 1–7 [GBC+17];
- B. Gkotse and G. Gorine, "Specifications for IRRAD sample & user management system and online database fixed", AIDA-2020-MS16, CERN, 2016. [Online]. Available: https://cds.cern.ch/record/2159521 [GG16];

En plus de ces publications, un article relatif au chapitre 6, “Ontology Embeddings for UI Personalization with ontowalk2vec”, est en cours de préparation pour soumission à un journal.

English Version

Nowadays, there has been an increasing progress in the fields of science and technology. For example, in the physics domain, big infrastructures for hosting physics experiments have been built to explore matter and universe in depth, providing answers to some open questions [E+19] while leaving others to address. These efforts result in the production of large volumes of heterogeneous data that need to be monitored, processed and analysed. Meanwhile, in the field of computer science, numerous technological advancements have had a wide impact in all aspects of life. Computers, smartphones and software applications enable, facilitate or accelerate important tasks of life.

Despite the significant advancements in every scientific field, it still remains difficult to bridge the gap among the different domains and sub-domains cited above, due to the fact that acquired knowledge in a particular domain is not always transferred to others, resulting often in a lack of communication and cross-fertilisation of ideas among them. This leads to a loss of important information and know-how that could have been utilised for further development of a given scientific area. This calls for more interdisciplinary research, something that is key to this present work, which builds upon ideas originating from software development in computer science, user-oriented interface design and experimental physics.

Advanced software applications are usually quite complex systems and require significant time of development and testing for developers but also special training for the users who need to learn to operate such systems, inducing significant time and manpower costs. Therefore, in order to achieve a better and rapid software integration in the scientific domains, and in particular in experimental physics, it is necessary to both minimise the development time and make software systems more intuitive and user-friendly so that they can be more easily adopted by their intended population target.

1.5 Goals

In the field of Experimental Particle Physics (EPP), various experiments are performed every day for discovering new physical phenomena while also providing ever more precise explanations for what exists in the physical world. For this purpose, the management and analysis of the heterogeneous data emanating from such experiments are crucial for the accuracy and reproducibility of the performed experiments. However, data preservation, integration and intercommunication have not been given the necessary attention they deserve, mostly for the reasons mentioned in the previous paragraphs.

This thesis aims at building upon web semantic technologies for the standardisation of data management in the field of EPP. Web semantic technologies

are known to facilitate data interoperability and integration, while allowing for communication among computer systems. One of these tools is the notion of an ontology, which is a declarative knowledge structure specific to a given application domain (and further explained in the following chapters). The overall objective of this thesis is thus as follows.

- The first goal of this work is, building upon ontology-related concepts, to provide an axiomatisation of some of the data management notions specific to a certain type of EPP experiments, namely the "irradiation experiments". This effort ends up formulating a common basis of knowledge to build upon in the physics community.
- The second goal of this thesis work is to automatise, from this ontology, the development process of dedicated data management systems (web application). These applications based on ontologies for EPP can be seen as a use case for the more general idea of using ontologies for the automatic generation of data management systems.
- Last but not least, the third goal of this thesis is to enhance the usability of such generated data management systems by introducing user interface (UI) personalisation methods.

1.6 Thesis Contributions

In order to achieve the previously mentioned goals, this thesis work includes the following three main contributions.

IEDM (Irradiation Experiment Data Management ontology) The IEDM ontology defines the main concepts for the data management of irradiation experiments.

GenAppi This is a new methodology for generating web applications for data management in the field of EPP, based on ontologies. It uses IEDM as a use case for the web application generation.

ontowalk2vec This new method developed by the integration of multiple state-of-the-art Natural Language Processing (NLP) techniques can be used for recommending User Interface (UI) features for personalisation.

In addition to these three fundamental contributions, some more practical developments also came out of this thesis work.

IDM (IRRAD Data Manager) This custom-built web application is currently used daily in the CERN Proton Irradiation Facility (IRRAD) [CER] [GGMR15]. It inspired the development of IEDM and GenAppi.

Irradiation Facilities and Test Beam databases and websites As by-products of the research conducted for analysing the irradiation experiments performed in irradiation facilities worldwide, the EPP-dedicated infrastructures for performing irradiation experiments, and the test beam infrastructures also used for EPP experiments, two separate websites and databases were developed thanks to the data accumulated from this research.

1.7 Outline

This current introduction chapter, Chapter 1, presents a general overview of this thesis work by describing the motivations, goals and main contributions.

Chapter 2 provides the necessary background in web semantics, EPP and irradiation experiments in order to facilitate the reading of this thesis manuscript. The knowledgeable reader is welcome to just skim through these pages.

Chapter 3 details the related work for the fields of software systems in EPP, ontologies related to EPP, generation of UIs from ontologies, user experience and recommender systems used for the personalisation of UIs.

Chapter 4 introduces the GenAppi methodology developed for the generation of data management web applications. Moreover, the Ontology-based Web Application Ontology (OWAO) used as a support layer of the GenAppi methodology is detailed. Another new concept described in this chapter is ontowalk2vec, an algorithm used for the generation of ontology representation vectors (embeddings) for recommender systems (a full chaper is later dedicated to this new technique).

Chapter 5 provides a detailed description of some applications of the previously introduced methodologies. First. the IDM web application is presented. Then, the IEDM ontology, which includes general concepts for the data management of irradiation experiment, is presented. In the last part of this chapter, the web application generated by the use of GenAppi is presented and compared against the manually developed IDM.

Chapter 6 focuses on the comparison of ontowalk2vec with the current state-of-the art NLP methods used for providing ontology embeddings. Our methodology is then tested with the OWAO ontology data and evaluated the possibility of being used for recommending personalised UI features according to the user preferences and UI similarities.

Finally, Chapter 7 presents the conclusions of this thesis work and discusses possible future perspectives.

1.8 Published Work

Parts of this thesis work are related to the following publications:

- B. Gkotse, P. Jouvelot, and F. Ravotti, “IEDM: An Ontology for Irradiation Experiments Data Management,” in The Semantic Web: ESWC2019 Satellite Events. Cham: Springer International Publishing, Portoroz, Slovenia, 2019, pp. 80–83 [GJR19b];
- B. Gkotse, P. Jouvelot, and F. Ravotti, “Automatic Web Application Generation from an Irradiation Experiment Data Management Ontology (IEDM),” in 17th International Conference on Accelerator and Large Experimental Physics Control Systems, ICALEPCS 2019, 2019. [Online]. Available: http://icalepcs2019.vrws.de/papers/tubpl01.pdf [GJR19a];
- B. Gkotse, P. Jouvelot, G. Pezzullo, and F. Ravotti, “The IRRAD Data Manager (IDM),” in Proceedings of ICALEPCS 2019, New York, NY, USA, 2019 [GJPR19];
- B. Gkotse, M. Glaser, P. Jouvelot, E. Matli, G. Pezzullo, and F. Ravotti, “Towards a Unified Environmental Monitoring, Control and Data Management System for Irradiation Facilities: the CERN IRRAD Use Case,” in 17th European Conference on Radiation and Its Effects on Components and Systems (RADECS), 2017, pp. 1–8 [GGJ+17];
- B. Gkotse, M. Brugger, P. Carbonez, S. Danzeca, A. Fabich, R. G. Alia, M. Glaser, G. Gorine, M. R. Jaekel, I. M. Suau, G. Pezzullo, F. Pozzi, F. Ravotti, M. Silari, and M. Tali, “Irradiation Facilities at CERN,” in 17th European Conference on Radiation and Its Effects on Componentsand Systems (RADECS), 2017, pp. 1–7 [GBC+17];
- B. Gkotse and G. Gorine, "Specifications for IRRAD sample & user management system and online database fixed", AIDA-2020-MS16, CERN, 2016. [Online]. Available: https://cds.cern.ch/record/2159521 [GG16];

In addition to these publications, a paper related to Chapter 6, "Ontology Embeddings for UI Personalisation with ontowalk2vec", is being prepared for journal submission.

Chapter 2

Background

Version française

Ce travail de thèse met en jeu deux domaines scientifiques différents, celui du web sémantique et celui des expériences d'irradiation, faisant partie de la Physique expérimentale des particules (EPP). Afin de permettre une meilleure compréhension et faciliter la lecture de ce manuscrit de thèse, ce chapitre présente les notions de base nécessaires sur ces deux domaines. Dans la première partie du chapitre, une définition de l'approche sémantique du web est donnée tandis que plusieurs exemples illustrent les grands principes de ce domaine et les standards utilisés en pratique. La deuxième partie de ce chapitre fournit une brève description du domaine des EPP, des expériences d'irradiation et des installations d'irradiation. Le plus grand laboratoire de l'EPP, le Laboratoire européen de physique des particules (CERN)[CER], est présenté, en particulier parce que le CERN Proton Irradiation Facility (IRRAD) [GGMR15] est utilisé comme infrastructure d'étude de cas dans cette thèse.

English version

This thesis work involves two different scientific fields, the one of web semantics and the one of irradiation experiments, part of Experimental Particle Physics (EPP). To allow for a better understanding and facilitate the reading of this thesis manuscript, this chapter is dedicated to the presentation of the necessary basic concepts about these two fields. In the first part of the chapter, some definitions about the web semantics are given while several examples demonstrate the main principles of this field and the standards used in practice. The second part of this chapter provides a brief description of the field of EPP, irradiation experiments and irradiation facilities. The largest laboratory in EPP, the European Laboratory for Particle Physics (CERN) [CER], is presented, in particular since the CERN Proton Irradiation Facility (IRRAD) [GGMR15] is taken as a case-study infrastructure in this thesis.

2.1 Web Semantics

The field of Web semantics was part of the World Wide Web Consortium's (W3C) vision about what is called the Semantic Web or Web of Linked Data [W3C]. Semantic Web denotes the technologies needed to define machine-readable data structures [TR06] where a well-defined meaning of their information content is provided, enabling computers' intercommunication and people's cooperation [BLHL01]. Web semantic technologies allow for a more efficient knowledge sharing and searching by providing context and structure to any specific knowledge field. This is made possible by the definition of certain standards which are known as *Ontologies*. In 2000, the Semantic Web was based on the stack of standards displayed in Figure 2.1. This stack has been extended with further semantic web standards, and it is nowadays better represented by Figure 2.2 [Gan18]. The name of the standards presented in these two figures are included in the Glossary and Acronyms sections. The main building blocks of these stacks, used also in this thesis work, will be further discussed in this chapter.

Ontologies have been used in Artificial Intelligence (AI) for years for various purposes such as knowledge formalisation, interoperability, complex querying and inference [Noy04]. Nowadays, academia and industry have chosen to base their information systems on both ontologies [Smi04] and knowledge graphs (KGs) [Pau17], the descendants of ontologies, in order to allow for better data integration and communication. Ontologies and KGs are suitable for domain-specific knowledge formalisation, and are broadly used in many domains such as biomedicine [S+07], bioinformatics [AC18] and law [PFOL18]. In addition to formalisation, ontologies may have many practical applications, although these have not been really explored deeply so far on a wide scale. The following section describes what ontologies and KGs are and how they can be practically used.

2.1.1 Definitions

The term Ontology derives from the ancient Greek words ὄν (on), the present participle of verb "to be", and λόγος (logos), (one) who speaks (in a certain way), or (one) who treats of (a certain subject) [OXF]. According to historical evidence, questions about the very existence of human nature have puzzled philosophers and humanity as a whole since the Ancient Greek times, where students of Aristotle used the term "metaphysics" (meaning literally "what comes after *Physics*") to refer to the work that Aristotle called "first philosophy". Aristotle spoke about the role of substance, essence and axioms [Coh16]. This work fed later the studies of other philosophers, and the term Ontology was coined by Rudolf Göckel (Goclenius) in his *Lexicon philosophicum* and Jacob Lorhard (Lorhardus) in his *Theatrum philosophicum*, referring to "metaphysics" or "first philosophy" as "ontology", to denote "what exists" or might exist [Smi04].

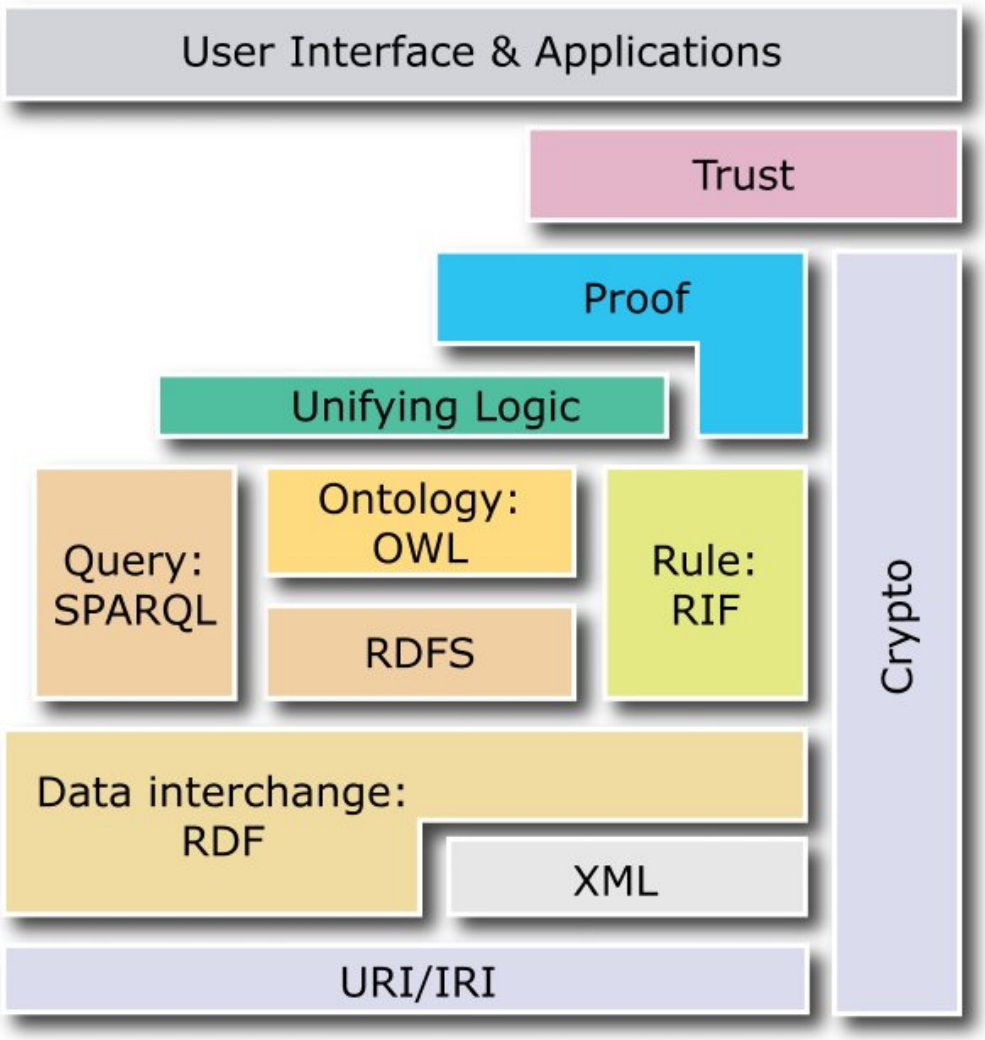


Figure 2.1: W3C Semantic Web Stack or Layer Cake for the years 2000, from Fabien Gandon "A Survey of the First 20 Years of Research on Semantic Web and Linked Data" [Gan18] (Names explained in the Glossary and Acronyms sections).

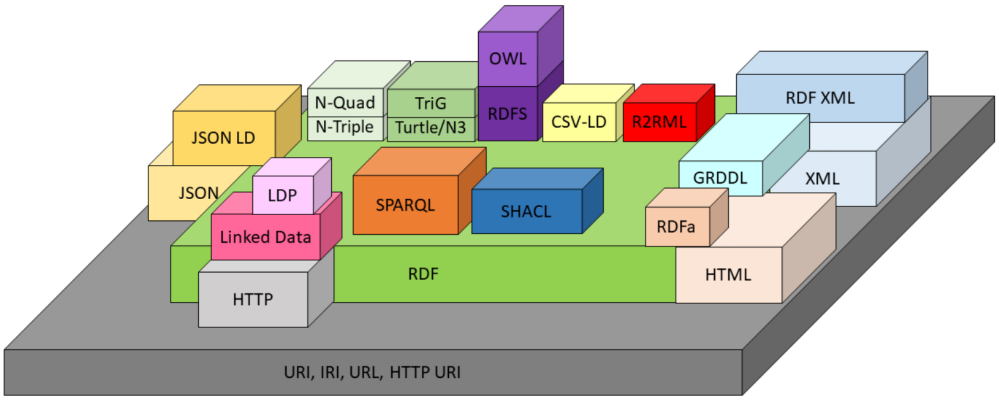


Figure 2.2: A new version of the Semantic Web Stack in 2018, from Fabien Gandon "A Survey of the First 20 Years of Research on Semantic Web and Linked Data" [Gan18] (Names explained in the Glossary and Acronyms sections).

As in philosophy, in computer science too ontologies are used for the exhaustive description and classification of "what exists". However, in computer science, "what exists" is what can be represented as knowledge. An ontology is defined as an explicit specification of a conceptualisation [Gru93], a model that defines concepts and relations among them for representing an area of knowledge, usually called a "domain". One of the main motivations of using an ontology is to solve the problem of semantic meaning and reach a common understanding of the structure of some specific information among people or software agents (e.g., computers, robots). Ontologies are used for sharing domain information and enable seamless interoperability, thus enabling the analysis and reuse of domain knowledge [NM01]. Moreover, the structure of an ontology follows a specific logic that allows for inference on data, link prediction, reasoning, query answering and information extraction. Another use of ontologies is data integration; heterogeneous data coming from different sources and databases can be described in a common manner via an appropriate ontology. Last but not least, the semantics provided by an ontology make it suitable for Natural Language Processing (NLP) algorithms and recommender systems.

More in detail, an ontology is a set of domain-specific definitions and descriptions containing the following concepts:

- **class**, i.e., an entity of the domain;
- **property**, i.e., an attribute, of specific type (e.g., string, integer, etc.), that helps to describe a class – such an attribute is also called a **data property**;

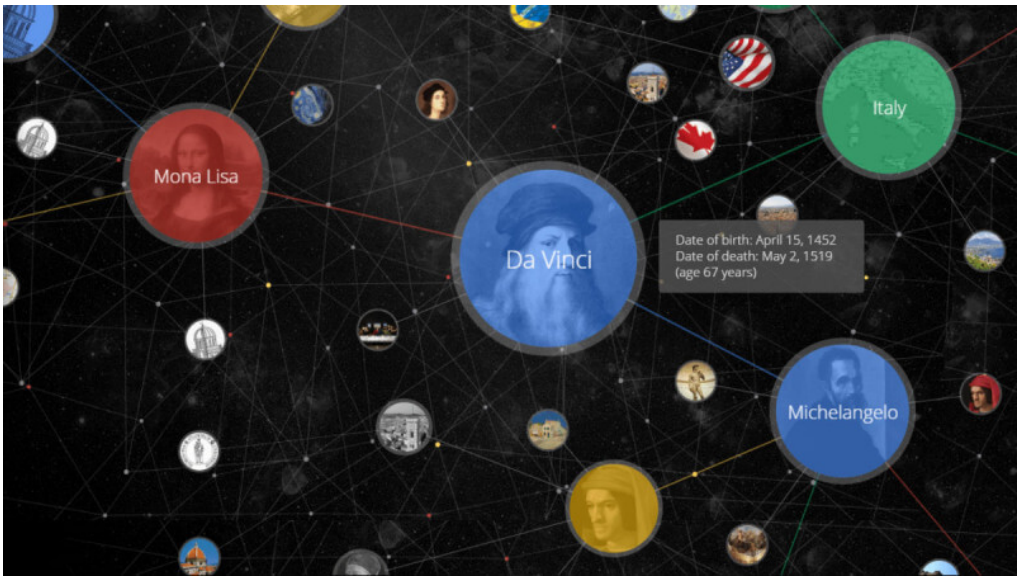


Figure 2.3: Excerpt from Google Knowledge Graph showing information related to some artists (e.g., Michelangelo) and the connections among them.

- **relation**, i.e., a link between two or more classes in order to describe a semantic relation among them – a relation is often called an **object property**.

An ontology can be also represented by a graph-like or only tree-like structure where the nodes are the classes and the edges are the relations. In the case of a tree, the ontology is actually a taxonomy, since classes are only organised in a hierarchical manner, representing a specific classification and inheritance relationship.

Once an ontology is defined, instances of its classes can be created and linked together with the purpose of representing or annotating datasets and resources. Nowadays, an ontology is considered to be the sole schema describing the interrelations and restrictions of resources, whereas the whole set of schema and instances is referred to as a knowledge base (KB) or knowledge graph (KG) [Pau17], a term coined by Google [GKG]. An excerpt of the Google Knowledge Graph is illustrated in Figure 2.3, showing information related to some artists (e.g., Michelangelo) and the connections among them.

To illustrate these concepts, in Figure 2.4, we present an excerpt from the well-established and broadly used ontology of Sensors, Observations, Samples and Actuators (SOSA), providing examples of classes, objects and data properties [JHC+19]. In this example, `Sensor`, `Observation` and `ObservableProperty` are ontology classes. Concepts such as `madeBySensor`, `observes` or `madeObser`

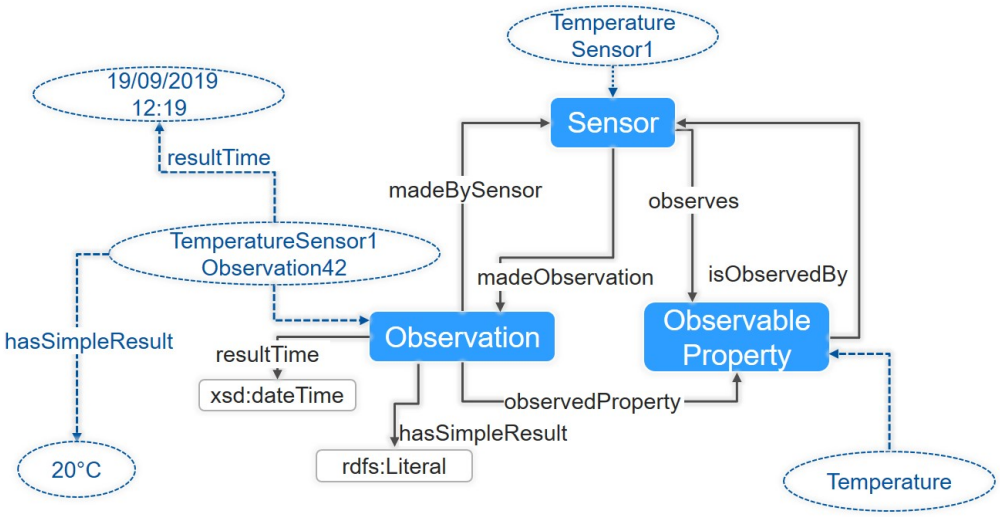


Figure 2.4: Excerpt from SOSA (Sensors, Observations, Samples and Actuators) ontology containing some instances.

`vation` act as semantic relations among classes, and therefore they are object properties of the ontology. `resultTime` and `hasSimpleResult` are data properties of the class `Observation`, and are of type `xsd:dateTime` and `rdfs:Literal`.

An ontology is considered a knowledge base or KG when it is enriched with instances (individuals) of classes and properties, as are the instances `Temperature Sensor1Observation42` or `19/09/2019 12:19` shown in the example, in Figure 2.4.

Each element of an ontology or a KG is defined by a unique identifier, which is called a URI, for Uniform Resource Identifier (URI), and that can be used to help finding such elements on the World Wide Web. For example, the URI of the SOSA class `Sensor` is `http://www.w3.org/ns/sosa/Sensor`.

2.1.2 Ontology Languages

Ontologies and KGs can be specified using languages dedicated to ontology description; each has its own structure and syntax. The languages that prevail nowadays because of their simplicity and expressiveness are the Resource Description Framework (RDF) [RDF], the Resource Description Framework Schema (RDFS) [RDS] and the Web Ontology Language (OWL) [OWL]. These standards are also used in this thesis work.

- **RDF** is a standard model for describing resources over the web. An RDF description is a set of triples in the form of subject-predicate-object. An

example of such a triple is the expression `TemperatureSensor1 observes Temperature`. Triples are saved in the ontologies. However, when there is a considerable amount of data to be stored in an ontology, dedicated storage units called triple stores are used.

- **RDFS** (RDF Schema) is an extension of RDF and supports definitions of basic ontology elements such as classes with their hierarchy and properties with their domain, range and hierarchy. Thus, RDFS can be used to describe taxonomies of classes and properties and is well suited for expressing simple ontologies.

- **OWL** is a W3C recommendation for a web ontology language. It aims at addressing the expressive limitations of RDFS. OWL enhances the expressiveness of RDFS by providing the means to represent relations between classes such as disjointedness, union, and intersection restrictions on property values such as cardinality.

  One major reason for the choice of OWL as the representation language for semantic web was that it is compatible with XML and RDF and that it provides additional expressiveness features allowing users to formally describe more types of classes, properties, individuals, and relationships than either XML or RDF can. OWL also provides semantics that give to terms defined in OWL a precise meaning so that they can be used effectively in applications that require interoperability.

2.1.3 Types of Ontologies

In the literature, different classifications of ontologies exist, depending on their level of abstraction, their scope and the specific domain they represent. The most often used classification divides the ontologies into four categories.

- **Upper/Top-level/Foundational Ontologies** describe abstract and general concepts in the domain of discourse. This type of ontologies is used as a foundation for further development of more low-level ontologies describing specific domains, applications or tasks [Jin18]. An example of a well-established top-level ontology is the Suggested Upper Merged Ontology (SUMO) being used for research and applications in linguistics and reasoning [PNL02]. This type of ontology is also referred to as a meta-ontology.

- **Domain Ontologies** represent a specific field of knowledge such as computer science or biology. One example of a domain ontology is the Computer Science Ontology (CSO) [STM+18], containing concepts of the computer science field.

- **Task Ontologies** are used to conceptualise and achieve a specific task. An example is the Robot Task Ontology (RTO), developed with the purpose of describing robot task structures and reasoning across various robotic domains [BSRF+17].

- **Application Ontologies** are local ontologies that describe a specific viewpoint. Such an ontology can be also considered as a combination of a domain ontology and a task ontology in order to achieve a particular purpose within an application [RPKC11, FEDB00]. An example of application ontology is the Experimental Factor Ontology (EFO) that combines parts of several biological ontologies, such as anatomy, disease and chemical compounds, with the purpose of annotating, analysing and visualising data of the European Bioinformatics Institute (EBI) databases [MRZP08].

2.1.4 Ontology Development and Visualisation Tools

Throughout the years, several software tools have been developed to facilitate the ontology creation, editing and visualisation processes. They provide the means for analysing, modifying, and maintaining an ontology as it evolves over time. One of the most popular tools and the one that was also chosen for the development of ontologies in this thesis work is Protégé [MPT15].

Protégé was developed by the Medical Informatics at Stanford University. It is an open-source and standalone integrated software tool used by ontology engineers and domain experts for developing knowledge-based systems. Protégé integrates several functionalities to support the major stages of an ontology's life cycle (creation, visualisation, and editing). It allows for the integration of new modules, offering underlying environments that are independent of the description language. Figure 2.5 shows a screenshot of Protégé where classes and individuals of a wine and food ontology (used as an example ontology in prior related work [NM01]) are represented.

2.2 Irradiation Experiments

As mentioned before, this interdisciplinary thesis lies at the frontier of two different domains. In the previous paragraphs, an overview of web semantic technologies was provided. In this section, further details concerning irradiation experiments (IEs) in Experimental Particle Physics (EPP) are presented.

In an irradiation experiment (IE), a piece of material (e.g., electronic chip, silicon detector, etc.) is purposefully exposed to radiation (of electromagnetic nature or particles). IEs find applications in several scientific and technical fields. For example, in the field of space and avionics, the materials composing aircraft

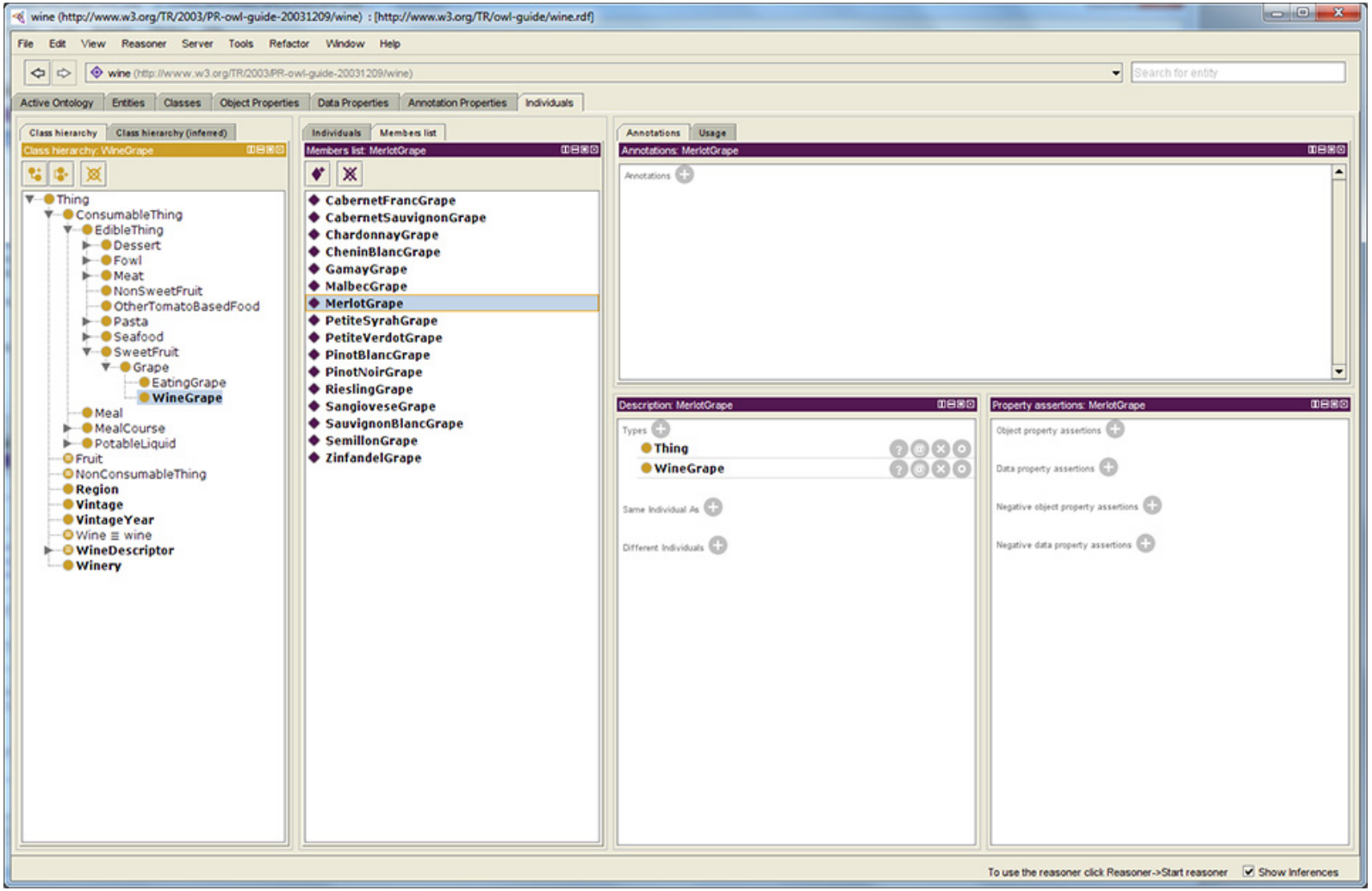


Figure 2.5: The wine ontology as illustrated through the Protégé platform. On the left section the ontology classes are shown (yellow points), while individual instances and their properties and annotations are shown in the rest of the interface.

Figure 2.6: Bottle sterilisation by electron-beam irradiation.

or spaceships suffer from radiation damage during their flights. Therefore, engineers need to test space components and materials during their development projects [KMBS12]. Other examples can be found in industry, where irradiation experiments are used for various purposes such as food and other items sterilisation (see Figure 2.6) [SNL62] or seed treatment [HEA]. IEs are also nowadays performed on patients as part of radiotherapy treatments [EBCTCG05] or for medical imaging in many hospital environments [AAB+98].

In EPP, the purpose of performing an IE is typically the qualification of materials, detectors and electronic components in a radiation environment equivalent to the one these devices will encounter in the actual High-Energy Physics (HEP) experiments, thus simulating, in a short time, long-term radiation-induced degradation effects. Even though all these IEs are performed for different purposes and in various locations (e.g. scientific institutes, industries and sometimes hospitals), these infrastructures can all be assimilated, at a fundamental level, to what is called an "irradiation facility".

2.2.1 Irradiation Facilities and Online Database

In order to gather detailed information about existing irradiation facilities but also get a better understanding of the needs of the irradiation-facility community, a research about the existent irradiation facilities and their common characteristics and practices for performing experiments was conducted. Aiming at finding relevant information and build a dataset, several publications were analysed, websites were explored and interviews with facility coordinators were conducted.

This research resulted in a practical contribution to the EPP community, namely the online irradiation facilities database and website developed during

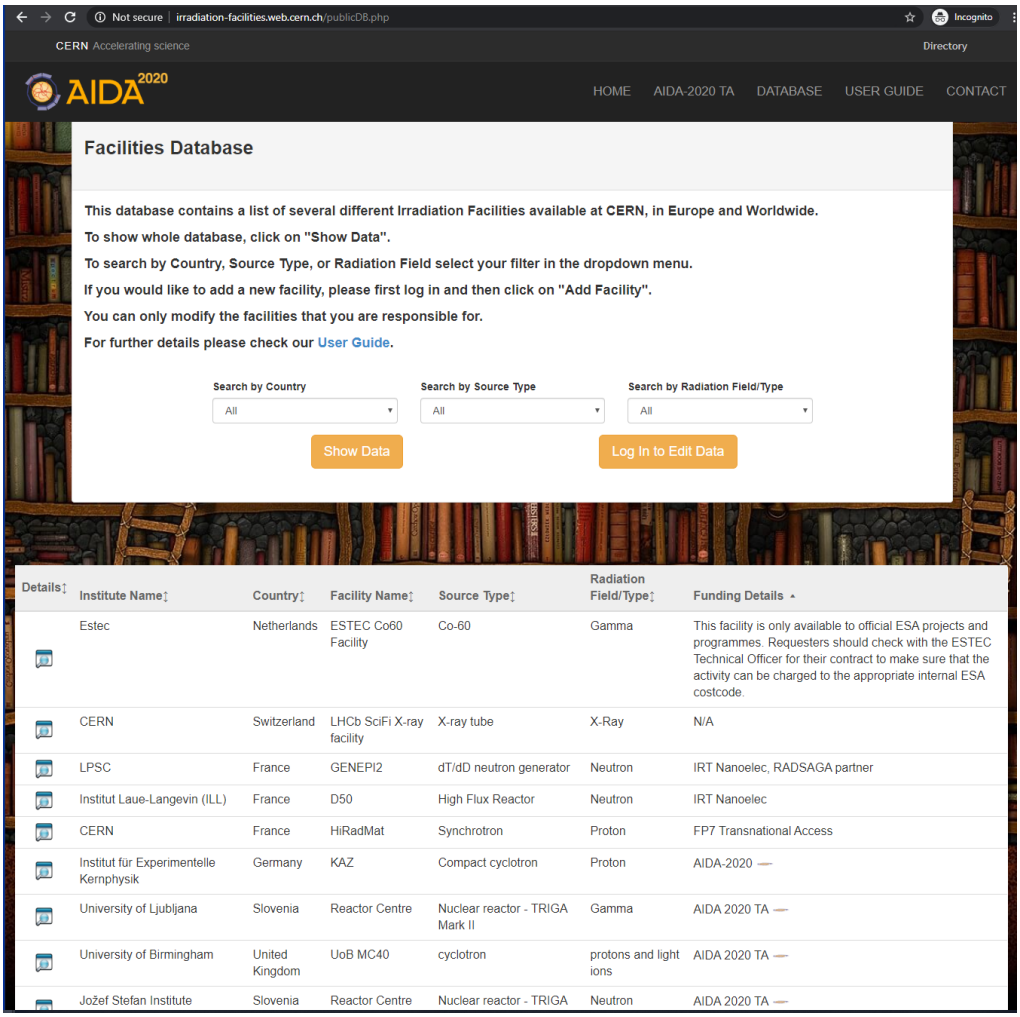


Figure 2.7: Online Irradiation Facilities Database platform containing data entries for irradiation facilities worldwide.

the framework of this thesis work [CIF] and populated with the data found from the conducted research[1]. This database includes more than 200 entries containing irradiation facilities data and allows scientists and engineers to search for the most suitable facility for their testing needs [GG16]. Requirements such as particle type, radiation field, etc. may vary for an irradiation experiment. Therefore, scientists and engineers need to find the best suitable facility for performing their experiments. Moreover, the availability of the facility, location and technical sup-

[1]Even though this project is performed within the framework of this thesis work, it is considered as a basis for the main contributions of this and therefore included in this background chapter.

port provided to the users are important information that they would need to know. In addition, the facility coordinators can update the facility's information on her own, allowing thus the data to be kept up-to-date. Furthermore, by the use of Linux "cron" jobs, annual reminders are sent to the facility coordinators, reminding them to update their data, if need be [GG17]. A screenshot of the online website with the database data is provided in Figure 2.7.

Following the success of this work, another project based on similar functionalities was later developed for test-beam facilities, following upon the request emanating from the HEP test beam community [TBD, GARW19]. Test-beam facilities are facilities serving a complementary purpose to irradiation facilities. They are used before and after IEs to ensure that the testing equipment is functioning as expected.

More details about the functionalities of the irradiation facilities and test beam databases and websites can be found in Appendixes A and B.

2.2.2 Experimental Particle Physics (EPP)

In the domain of EPP, IEs are a prerequisite to ensure that the experimental equipment can withstand radiation, and is, as it is usually called, "radiation hard". There is a need of several irradiation facilities worldwide to perform this type of experiments.

The irradiation facilities may have different characteristics and ways of operation, but they all have similar characteristics and follow the same EPP principles. Each irradiation facility provides a radiation field composed of one or more types of particles (e.g., protons, electrons, etc.) delivered by a specific source such as an accelerator or a radioactive isotope. Scientists and engineers from different communities of experimental particle physics can have their equipment, often called Device Under Test (DUT), tested in this type of facilities.

When referring to irradiation experiments, there are certain basic quantities that need to be defined and will be used in this thesis work. These quantities are defined below.

- **Activity** The activity A of a radioactive source in a particular energy state at a given time t is defined as the quotient of dN by dt, where dN is the expectation value of the number of spontaneous nuclear transitions from that energy state in the time interval dt: [Sta]

$$A = dN/dt \quad [\text{Bq}]. \tag{2.1}$$

- **Fluence** The particle fluence Φ is the quotient dN by dA, where dN is the number of particles incident on an area dA [Rav18]:

$$\Phi = dN/dA \quad [\text{cm}^{-2}]. \tag{2.2}$$

- **Dose** An absorbed dose D is related to the stochastic quantity called "energy imparted" ε, and is defined as the mean energy imparted $d\bar{\varepsilon}$ by radiation to a matter of mass dm of a specified material [Rav18]:

$$D = d\bar{\varepsilon}/dm \quad [\text{Gy}]. \tag{2.3}$$

- **Momentum** Given the kinetic energy E [MeV] of a particle of rest mass M [MeV] and momentum P [MeV/c], the energy-momentum relation is given by the followin equation [Rav06]:

$$E = \sqrt{P^2 + M^2} - M \quad [\text{MeV}], \tag{2.4}$$

  where the particle is supposed to move at the speed of light c, assumed here equal to 1.

- **Interaction Lengths**:
  - **Radiation Length** The radiation length in a material is the mean distance (in cm) needed to reduce the energy on the incoming beam by a factor of $1/e$, depending on its energy level.
  - **Nuclear Interaction Length** The nuclear interaction length is the mean distance travelled by a hadronic particle such as a proton before undergoing an inelastic nuclear interaction with the material atoms or molecules.
  - **Nuclear Collision Length** The nuclear collision length is the mean free path of a particle before undergoing a nuclear reaction, for a given particle in a given medium. The collision length is smaller than the nuclear interaction length because the latter excludes the elastic and quasi-elastic (diffractive) reactions from its definition.

  Usually these values are pre-calculated for each basic element and provided in the literature, for example via the portal of the Particle Data Group (PDG) [T+18]. In order to compute one of these lengths for a composite material made of n basic elements, the general mass-corrected formula is:

$$X_0/W_0 = \sum_{i=1}^{n} X_i/W_i, \tag{2.5}$$

  where:
  - W_0 is the total mass of the sample (in g);
  - X_0 is the combined nuclear length of the sample (in g.cm^{-2});
  - W_i is the mass of the individual component i (in g);
  - X_i is the nuclear length of the individual component i (in g.cm^{-2}).

The goal of an irradiation experiment is to ensure that the DUT reaches a certain cumulated fluence, or dose, equivalent to the one estimated to be received in the longer term, when this type of DUT is installed in a given HEP experiment. In general, the nuclear lengths have to be minimised as much as possible, so that the secondary particles produced as a byproduct of the interaction of the beam particles with the material do not affect much the experiments' results.

2.2.3 CERN and Proton Irradiation Facility (IRRAD)

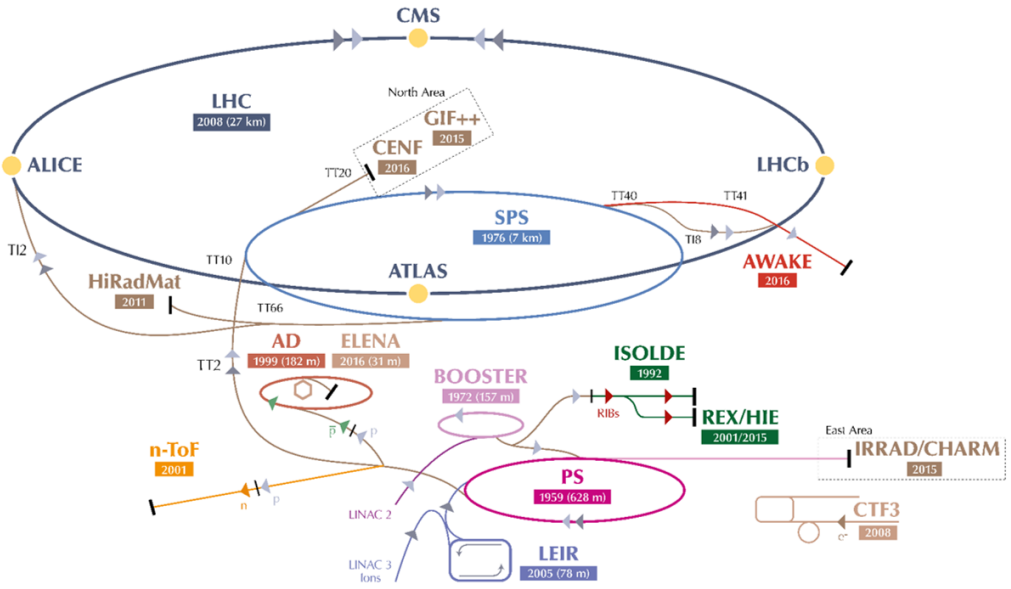


Figure 2.8: The CERN accelerator complex [Mob16].

CERN is the European Laboratory of Particle Physics [CER], providing a complex set of accelerators, beam lines and detectors including the largest worldwide accelerator, the Large Hadron Collider (LHC) [EB08]. Two particle (proton or ion) beams are accelerated in two opposite directions in a collider and, as shown in Figure 2.8, at four specific points where particle detectors are located, these beams collide and produce different kinds of particles that need to be detected. These detectors at the LHC are: ATLAS (A Toroidal LHC Apparatus) [C+08b], CMS (Compact Muon Solenoid) [C+08a], ALICE (A Large Ion Collider Experiment) [A+08a] and LHCb (Large Hadron Collider beauty) [A+08b].

To produce the physics that need to be studied, the particle beams need to reach certain energy levels (14 TeV). For this reason, the beams are accelerated in steps by several ancillary accelerators before reaching the LHC. These acceler-

ators, except from acting as injectors for the LHC, also deliver, through different beam lines, beams of particles to several irradiation facilities and test-beam facilities, which are dedicated infrastructures where detectors can be tested to ensure that, after irradiation, they are still properly functional.

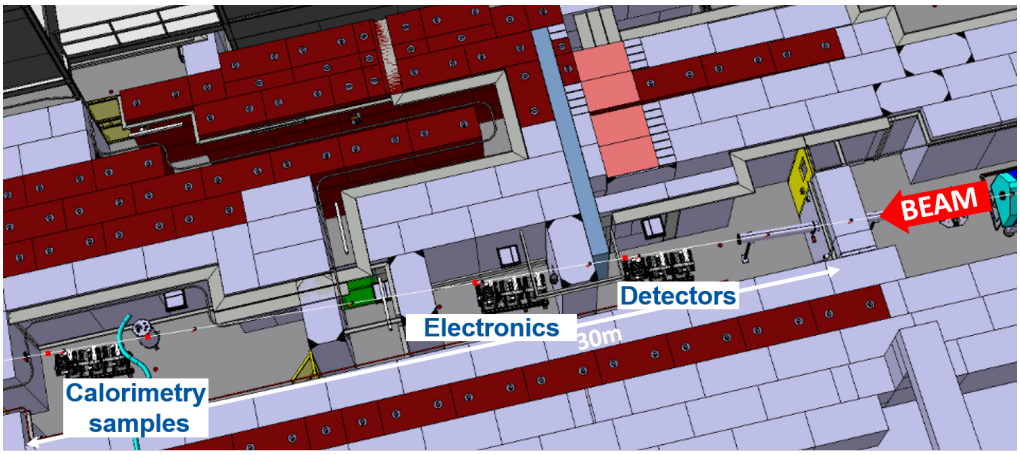


Figure 2.9: IRRAD Facility irradiation zone.

One of these accelerators is the Proton Synchrotron accelerator, which is delivering a proton beam to the Proton Irradiation Facility (IRRAD) located in the East Area of PS (see bottom right corner of Figure 2.8). The IRRAD facility is used for the qualification of materials, electronic systems and detector components for HEP experiments. A beam of protons with a momentum of 24 GeV/c and a typical size of 12×12 mm^2 accelerated by the PS is delivered to IRRAD, in pulses, and used for irradiation experiments.

Every year in IRRAD, hundreds of electronic and detector components, called "samples", are tested and qualified against radiation. Figure 2.9 shows the experimental area of IRRAD where the samples get installed on a couple of remotely controlled movable stages, called IRRAD tables [GMR13], and the proton beam is impinging on them. Dosimetry of the proton beam is an important part of the IRRAD operation; thus, usually during the irradiation experiments, pure aluminium foils (which function as dosimeters) are placed together with the samples. They are used for assessing the actual cumulated proton fluence through Aluminium activation and subsequently gamma spectrometry [CFG+17]. As a result, the registration, planning and follow-up of these operations require the tracking and management of several hundreds of samples and dosimeters each year. For example, for the year of 2018, 792 samples and 401 dosimeters were irradiated belonging to 81 irradiation experiments coming from 92 users from different CERN and external teams. Even though, the data derived from these experiments is not in the scale of what we call nowadays "Big Data" (denoting

Gigabytes of data), they still correspond to a considerable amount of data to be managed through spreadsheets or paper notebooks, especially for small (or understaffed) experimental teams.

Chapter 3

Related Work

Version française

Dans le chapitre précédent, les avantages de l'utilisation d'ontologies et de KG pour la formalisation et la normalisation des connaissances ont été présentés, tandis que des connaissances de base sur les expériences d'irradiation en EPP ont été introduites. Dans les sections suivantes, l'état actuel des connaissances sur les systèmes logiciels liés à la gestion des données dans l'EPP est présenté, et certains travaux connexes sur les ontologies construites à des fins liées à l'EPP sont discutés. En outre, les principaux travaux antérieurs sur les ontologies dédiées aux interfaces utilisateur (UI) et la génération d'interfaces fondées sur des ontologies sont discutés. L'importance d'une expérience utilisateur (UX) de qualité et d'une personnalisation de l'interface utilisateur efficace est soulignée par plusieurs résultats de recherche liés aux préférences des utilisateurs pour différents types d'interface ; ceci est présenté dans les paragraphes suivants. Ainsi donc, un autre aspect étudié dans cette thèse est l'utilisabilité des interfaces de gestion des données et comment celles-ci peuvent être améliorées en utilisant des systèmes de recommandation fondés sur des ontologies.

English version

In the previous chapter, the advantages of ontologies and KGs for the formalisation and standardisation of knowledge were presented, while some background knowledge about irradiation experiments in EPP was introduced. In the following sections, the current state of the art on software systems related to data management in EPP is presented, and some related work on ontologies built for EPP-related purposes are discussed. Furthermore, previous work on User Interface (UI) ontologies and ontology-based UI generation is discussed. The importance of having efficient User Experience (UX) and UI personalisation is emphasised by several research findings on UI preferences, presented in the following sections. Therefore, another issue addressed in this thesis is the usability of data management UIs and how it can be optimized via ontology-based recommender systems.

3.1 Data Management Systems in EPP

Most of the software tools present in EPP infrastructures are platforms for handling requests for experiments, control systems for operating the infrastructures, and libraries for the data monitoring, visualisation and analysis of the subsequent results. For the specific case of irradiation facilities, in the past, little work has been done on data management systems, since a large part of the community relies on customised proprietary software, spreadsheets or paper notebooks for handling and saving the produced data. Furthermore, little attention has been given to the sharing of data management knowledge and availability of previous experimental results. This less-than-optimal situation is something that can, we believe, be addressed by the use of ontologies. Furthermore, on a more practical side, there seems to exist no related work in EPP on generating software systems based on ontologies, something we want to pursue. However, some effort has been put from the web semantics community towards this direction.

Since not much work on data management systems was found in irradiation facilities, the work related to this issue from the broader EPP community is investigated. These systems provide solutions on different aspects of EPP experiments, such as the following.

- **Information registration** This specification includes the experiment (e.g., type of experiment, requested dose/fluence, requested particle type, etc.), the contact information of the users participating to the experiments as well as the definition of their roles (e.g., operator, experiment's responsible person) and information about the material or DUT (e.g., dimensions, type, composition, etc...).
- **Infrastructure availability** The planning of the experiments but also the availability of materials and tools to be used in and for the experiments are described there.
- **Monitoring and visualisation** Software tools for monitoring an experiment and visualising the results are of major importance in order to ensure the proper operation of an experiment.
- **Reporting** Providing data related to a performed experiment is necessary for further data analysis and long-term knowledge sharing.

Of course, control systems are key software tools for controlling any experiment in EPP. However, these control systems are used mainly during its operational phase, and although they manage a significant amount of data, they are not used for further data handling and processing. Therefore, additional tools are integrated on top for monitoring the devices, visualising the results and report generation. The following related work is thus focused mainly on the software tools contributing to the data management of experiments' life cycle.

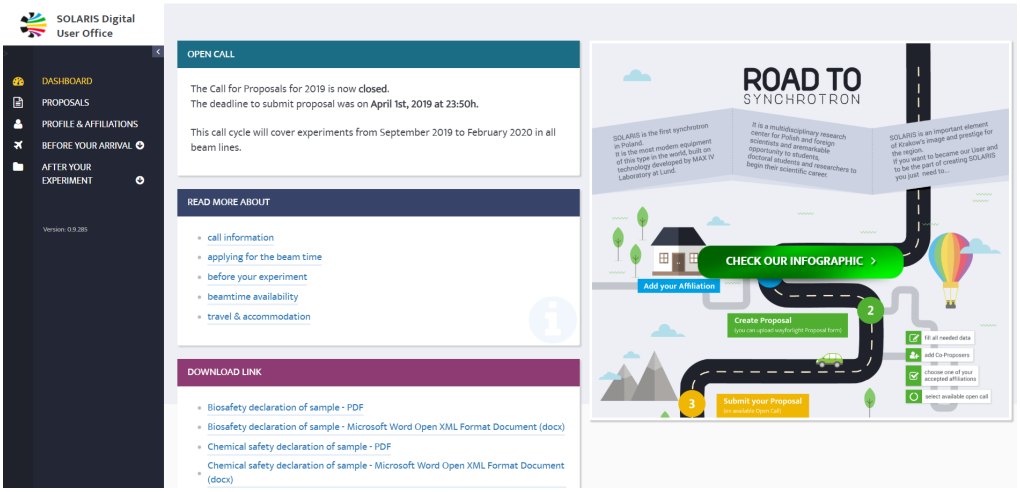


Figure 3.1: SOLARIS Digital Users Office (DUO) [SLR].

3.1.1 Digital Users Office (DUO)

A class of tools used for data handling in accelerator facilities is called Digital Users Offices (DUO). DUOs consist of central tools used for the organisation of the users' experimental operations. They are web applications where users can submit their proposals for beam time in several beam lines and test beam facilities (see Section 2.2.1). They also provide information to users about the regulations for dosimetry and personal safety. Moreover, DUOs handle reports and documentation prepared and provided by the facility managers.

For instance, one such platform is used in the Paul Scherrer Institute (PSI) in Switzerland [PSI]. The dedicated platform is built in PHP, a server-side web programming language, and uses Oracle[1], a specific type of relational database. In the PSI DUO, the user can create and edit his/her beam proposals. They can apply for access cards or dosimeters. Furthermore, PSI DUO is also used for the management of newsletters, mailing lists and the publication database [AW05].

Driven by this development, other Digital Users Offices were developed, such as the DOOR Online Office for Research with Photons [**?**] for DESY in Germany [DES], the Digital User Office portal GATE [GAT] for HZB in Germany [HZB] and the Users Net set [SUN] for SOLEIL in France [SOL]. More recently, another web application for the national synchrotron radiation center in Poland (SOLARIS) was built [SLR]. SOLARIS DUO uses the latest web technologies, such as Spring Boot[2], which is a framework of the Java[3] programming language

[1]Oracle relational database https://www.oracle.com/database/
[2]Spring Boot Java framework https://spring.io/projects/spring-boot
[3]Java programming language java.com

for back-ends, and Angular [4], which is a framework of JavaScript [5] for front-ends, and MySQL [6], another type of relational database. As illustrated in Figure 3.1, in addition to the functionalities of PSI DUO, SOLARIS DUO provides a more user-friendly interface, displaying different dedicated views according to the users roles (Radiation Manager, Safety Manager, Beamline Manager and Reviewer). Moreover, it provides a guide on the steps that a user should follow before and after an experiment [Szy17].

The necessity of DUOs is obvious in any accelerator infrastructure. They facilitate the management of users data, the organisation and planning of beam proposals and experiments, while they are also used by the facility team to provide reports to users. However, DUOs are utilised only as an initial step of registering experiments or for the final reports. They do not include detailed information about the Devices Under Test (DUT) and the dosimetry tools. Therefore, a constant follow-up of the overall experiment is not possible.

3.1.2 Monitoring and Visualisation Tools

Monitoring and visualisation tools are widely utilised in EPP. They are used for various purposes such as time series analytics or displaying the status of experiments and control systems; they allow for a somewhat intuitive understanding of the ongoing processes and visual alarming of possible malfunctions during experiments.

An important contribution in the field of monitoring and visualisation tools is the WebJive web application [WJV], developed in the MAX-IV institute [MAX], in Sweden. It is a web-based application for monitoring and controlling Tango [7] devices [G+03]. WebJive uses React[8], a JavaScript library, for the client side that communicates with a GraphQL server [9] developed in MAX-IV, named TangoGQL. With WebJive, users can view and modify the devices' information, properties and attributes. Moreover, the system can be configured by the users in order to create their own UI dashboards.

Another web application, used at Jefferson Lab, in the USA [JLA], is the Web Extensible Display Manager (WEDM) [SL19]. As illustrated in Figure 3.2, WEDM provides the users a read-only access to their EPICS[10] control system screens from a web browser in remote offices and from mobile devices.

[4] Angular framework angularjs.org
[5] JavaScript https://www.javascript.com/
[6] MySQL database www.mysql.com
[7] Tango is a control system offering a set of communication and logging services.
[8] React JavaScript library https://reactjs.org/
[9] GraphQL query language for APIs https://graphql.org/
[10] A set of open-source software tools, libraries and applications developed collaboratively and used worldwide to create distributed real-time control systems for scientific instruments such as particle accelerators, telescopes and other large scientific experiments https://epics.anl.gov/

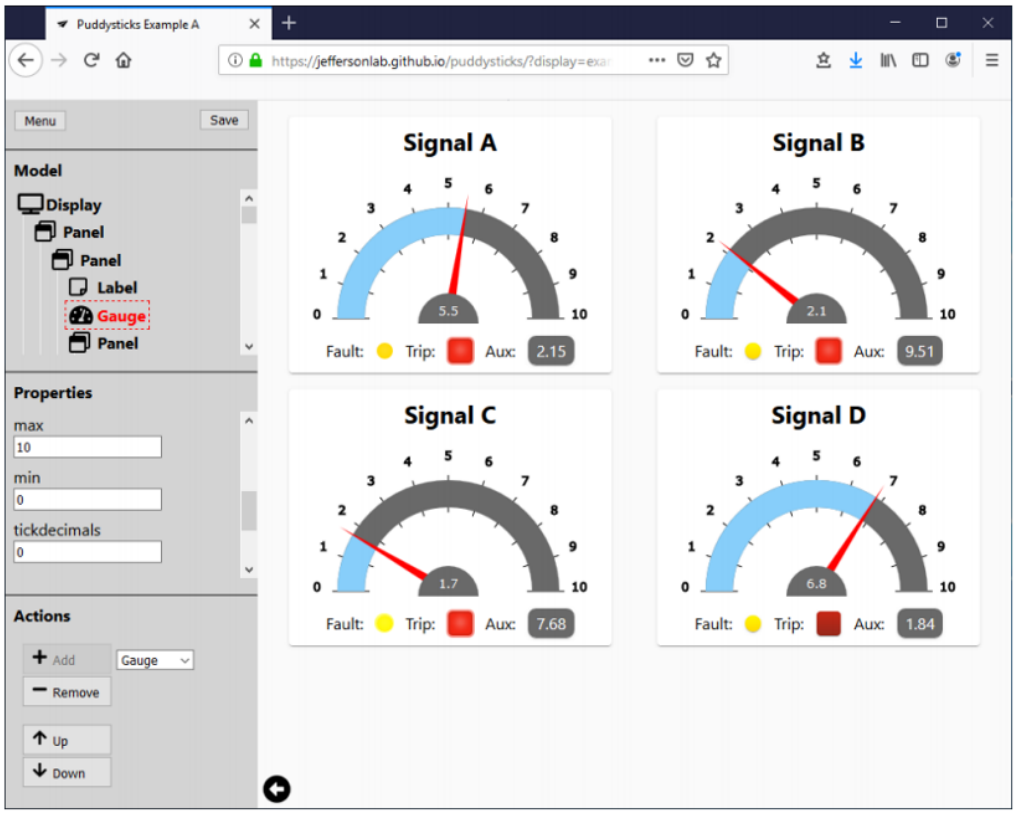


Figure 3.2: Web Extensible Display Manager used at Jefferson Lab for control systems display (from R.J. Slominski and T. Larrieu., Web Extensible Display Manager 2 [SL19]).

In addition to web-based applications, Graphical User Interfaces (GUIs) and frameworks are also employed for data visualisation in EPP. A good example is the Mantid [11] data analysis framework [MAN, C+17], used for the processing and visualisation in near real time of live experimental data at the European Spallation Source (ESS), in Sweden [L+11, ESS], and ISIS Neutron and Muon Source, in the UK [Fin07, ISI]. Other developments include Graphical User Interfaces (GUIs) built with the PyQt framework[12] for the support of new user interfaces for the Sirius beam lines in SOLEIL, France [SIR].

While the EPP visualisation and monitoring systems allow for a smoother operation of experiments, they are usually dependent on the underlying control systems. As described in the previous paragraphs, most of them are used as a

[11] Mantid provides a framework that supports high-performance computing and visualisation of materials science data www.mantidproject.org

[12] PyQt is one of the Python bindings for the Qt cross-platform C++ framework https://riverbankcomputing.com/software/pyqt/intro

front-end interface for displaying and analysing data from control systems such as EPICS or Tango. This has a consequence on the scalability and portability of these tools, preventing them from being adopted by the other infrastructures, which may not use the appropriate control systems. Moreover, their multi-layer architecture may not be suitable for smaller-scale infrastructures such as irradiation facilities and may induce high maintenance costs in order to keep the system up-to-date and secure.

3.1.3 Reporting Tools

Reporting tools display important information for the life cycle of an experiment and its final results. These reports need to be detailed enough and tailored to the data requirements of the users or other scientists involved in the experiment. In order to achieve that, reporting tools should provide features that allow users to customise the report format.

One such example is the Tornado DS tool shown in Figure 3.3. The Tornado DS tool is a web-based application that was developed for the Tango control systems used in the beam lines at Synchrotron Light sources at ALBA [Ein11, ALB], in Spain. By the use of the Tornado DS tool, scientists can configure, create and display their own reports. The tool is integrated in a device server called Tornado DS, which initially starts a Tornado [TOR] web server, uses WebSocket[13] and behaves as a gateway between all the devices in the database and the report tool [BRB+17].

Another reporting tool was developed for data analytics of SCADA systems at CERN [CER] [Boy09] [SGBST17]. The tool reports data analytics directly into SCADA systems[14], allowing for better operation. The software can also display more complex analyses involving process interconnections. Moreover, the reporting tool can obtain the metadata and analysis results of any database system that supports the SQL standards.

As for the visualisation systems, also most of the EPP reporting software tools are strongly connected to the control and data acquisition systems of their infrastructures. This does not allow for an easy integration with software in other EPP infrastructures, such as irradiation facilities, that may act in smaller-scale software infrastructures and with lightweight control systems.

[13] WebSocket is a computer communications protocol, providing full-duplex communication channels over a single TCP connection.

[14] A control system architecture that uses computers, networked data communications and graphical user interfaces to monitor and control large processes.

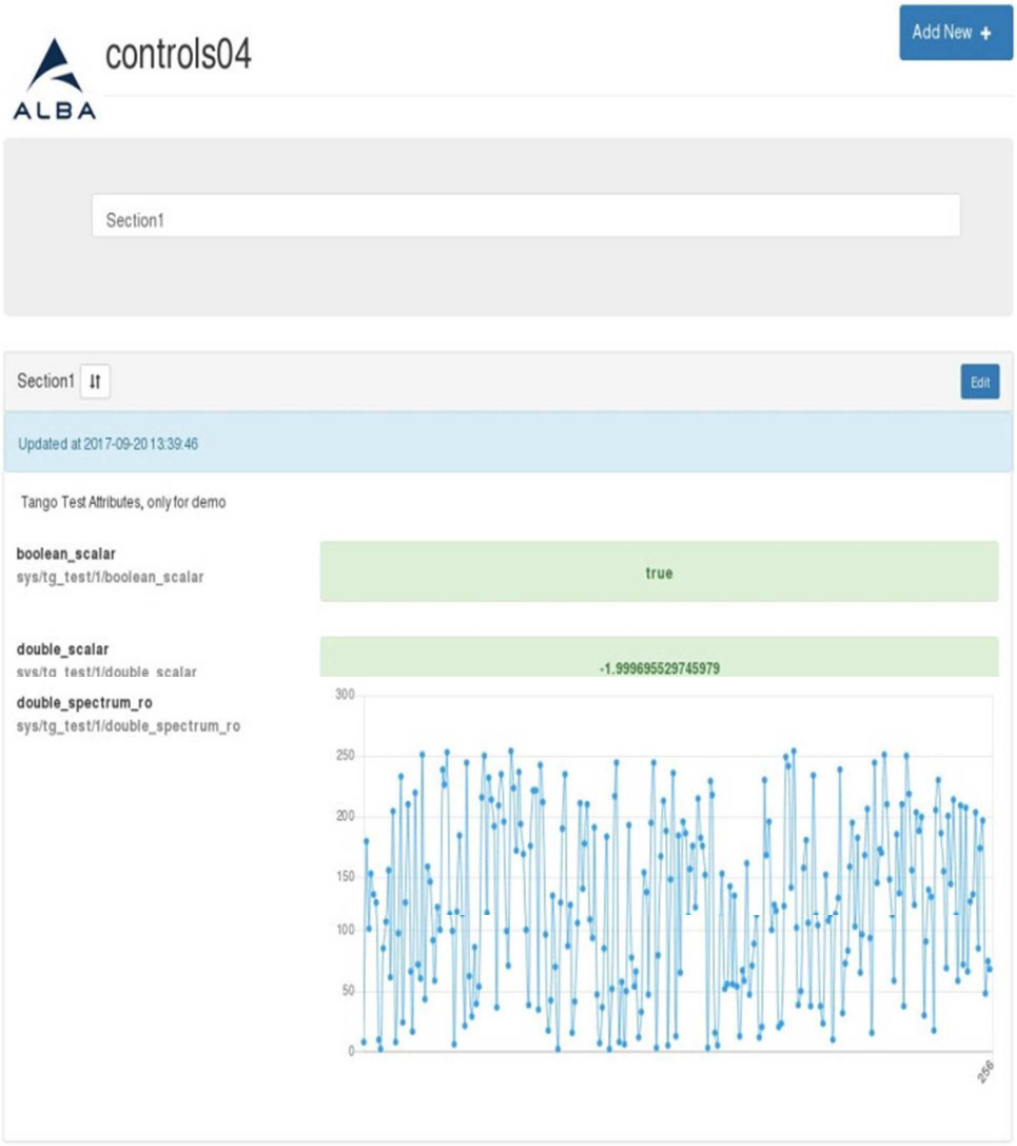


Figure 3.3: Default Tornado DS index page (from M. Broseta et al., A Web-Based Report Tool for Tango Control Systems via WebSockets [BRB+17]).

3.2 Domain Ontologies in EPP

In the literature, several ontologies attempting to formalise scientific knowledge exist. One example is the Knowledge Graph for Science, where the authors work on axiomatising the knowledge present in the scientific literature [AKP+18]. Another related work is the Web Physics Ontology [Vje17], which describes physics equations and relations among physical quantities. However, this ontology is limited to the domain of electromagnetism and mechanics, and there is no reference to EPP. Another approach is the ontology design pattern proposed for the particle physics detectors final state [CCDT+15] as illustrated in Figure 3.4. Although this work includes concepts that are typical in EPP experiments, it focuses only on the analysis of particle physics data and not data management concepts. Other related ontologies include concepts for specific domains such as the synchrotron ontology [SP17] or the ontology for radiation protection procedures in the medical field [BLH17].

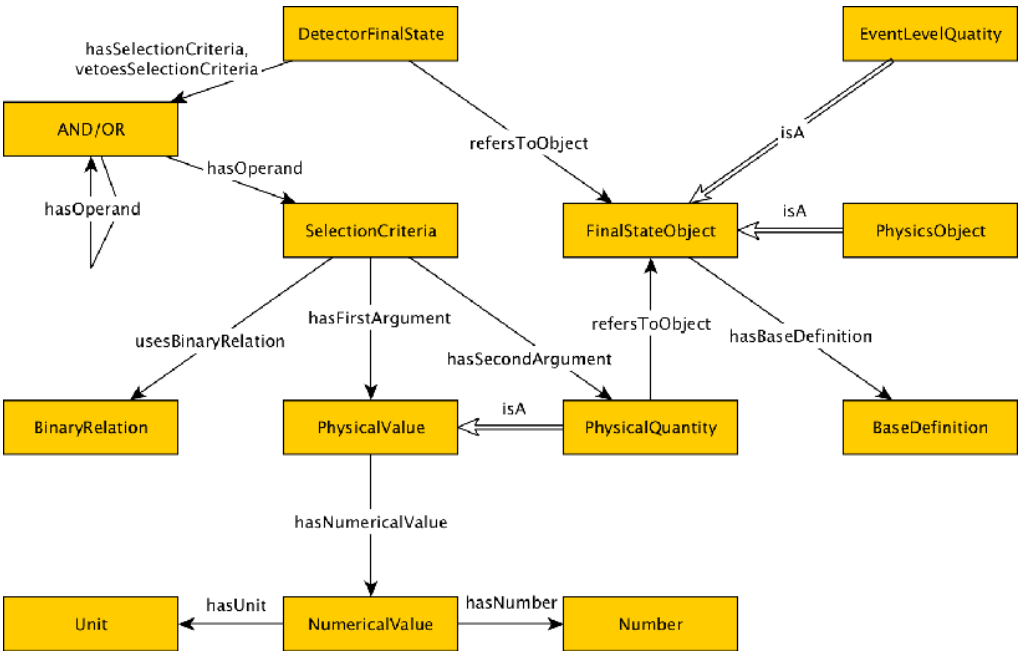


Figure 3.4: A schematic view of the DetectorFinalState ODP (from David Carral et al, "An Ontology Design Pattern for Particle Physics Analysis" [CCDT+15]).

Besides these, there have been two ontologies that could instead partially represent concepts of data management in EPP. The first ontology is the Ontology of Scientific Experiments (EXPO), which introduces concepts specifically linked to the notions of experimental design, scientific methods, and other core principles of experiments [SK07]. The second ontology is the Units of Measure (OM)

ontology, which describes entities from many physics-related domains, including concepts from particle physics [RvAT13]. This ontology is thus necessary for the formalisation of the most commonly used physical quantities.

3.3 Ontology-based Data Management Systems

The formalisation and standardisation of knowledge that ontologies provide can be used for the description of the UI components that form the fabrics of the vast majority of computer software. In the literature, there have been several attempts to describe, at least, part of this type of knowledge [SUO, LUI, HUH10]; UI ontologies have been developed to fulfil specific requirements and describe certain parts of a UI or web application. For example, the Semantic UI ontology includes concepts related to the interface elements of its own UI framework [SUO]. Even though this ontology seems suitable for our work, it does not include concepts related to the visualisation of the elements (e.g., font size).

Concepts such as font size are, however, included in the UI ontology [LUI] supported by the community behind Linked Open Vocabularies (LOV) [LOV]. This ontology describes concepts of UI elements, relates these entities to properties of style (e.g., colour or background colour) and is suitable for describing forms and sequences in widgets[15].

Yet, both ontologies describe only the graphical front-end part of a user interface, and they do not address concepts related to actual data operations. Such an issue is treated by RDFa[16] User Interface Language (RaUL) [HUH10], a user interface ontology used for the description and structure of web forms as RDF statements that introduces concepts such as *CRUDOperation* about specific CRUD operations (Create, Read, Update and Delete). These are the four fundamental operations that any data management system is expected to, at least, provide.

The idea of automatically generating UIs from a given ontology, on which part of this thesis work is based, has been envisioned before. One example is the work about ontology-based UI development where a User Interface Ontology (UIO) is used to describe UI properties and their semantic relationships [Sha11]. By the use of UIO and a VCards[17] domain ontology, mappings between these two ontologies are created and used to instantiate a user interface for creating and editing VCards. Even though the idea is indeed similar, it does not get up to the point of automatically generating user interfaces, as we describe in this work. Moreover, UIO is neither properly documented nor publicly available, and therefore this work is not readily reusable.

[15] Fragments or a set of fragments of a user interface (e.g., form, table, list, etc.).

[16] Resource Description Framework in attributes provides ways to add metadata annotations to Web documents.

[17] VCards is a file format for electronic business cards.

Another example where an ontology of user interfaces is used is Hierarchical User Interface Component Architecture (HUICA) [Eng18]. This ontology describes the whole process of user-interface development from the wireframes to a Model-View-Controller (MVC) architecture for UI interactive components. This ontology is divided in three parts: the design part, where concepts such as wireframes are described; the UI part, where UI elements such as Widgets or Composites [18] are included; and the last part, used for an MVC architecture. Since the main focus of this work is the HUICA architecture, the author provides only a visual schema of this ontology without providing the ontology in a machine-readable or even formal format (e.g., RDF). Moreover, the MVC architecture is used only in the front end for UI interactive components, which is a domain where several software tools exist and is not the focus of our work.

In other related work [MLA18], ontologies are transformed into relational databases based on specific rules that can then be used for the development of a web application. However, a final operational web application is not presented. Another study found about ontology-based UI generation [HKP17] requires that users provide inputs to an Application Ontology that is transformed into a Target Ontology used for UI generation. However, in this work, the user is asked to customise the Application Ontology (AO) used for generating the UI, and some proprietary annotations are required, adding some limitations to the universality and user-friendliness of this approach.

3.4 UI Customisation and Personalisation

Good User Experience (UX), the fact of having well designed and intuitive UI, is an important aspect that has been well studied and used, in particular in digital marketing and industry. Here the goal is to drive people's attention in order to buy targeted products and achieve best profit. Unfortunately, UX is an aspect often neglected or given secondary priority in scientific domains such as in medicine or EPP. Forgetting to properly address this issue, though, may be quite dangerous and even deadly. As described in the book of *Tragic Design* [SS17], an example of bad UX design on the visibility of a control system status was the cause of the partial meltdown of the nuclear power plant in Pennsylvania, in 1979, which, luckily, had no consequences on the plant workers or the public. Unfortunately, that was not the case for a young hospitalised cancer patient. The nurses treating the patient were too distracted by the medical software used for keeping the patient's medical information due to an overload of information and bad colour choices (see Figure 3.5). A consequence is that the most important information, namely that the treatment required three-days hydration, was not visible, resulting in the death of the patient due to toxicity and dehydration.

[18] High-level fragment of a user interface (e.g., panel, interacting dialog, etc.)

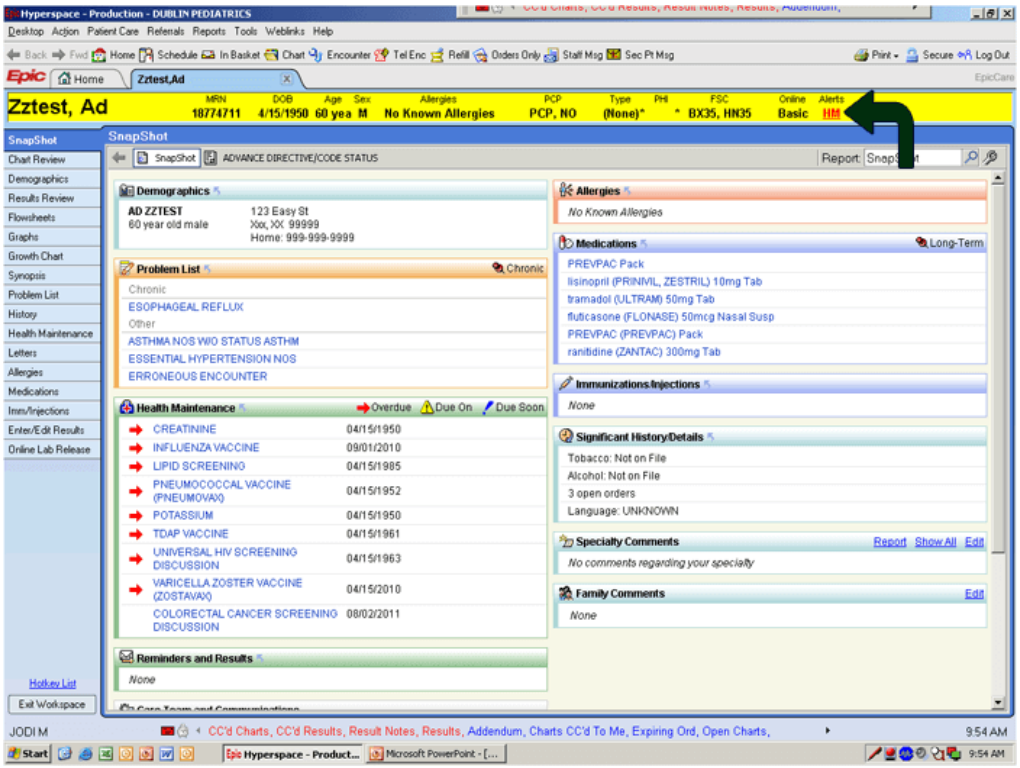


Figure 3.5: A screenshot of the Epic charting software used by many hospitals in the US. The black arrow shows where the alerts are shown (from C. Savard S. Jonathan Shariat, "Tragic Design: The Impact of Bad Product Design and How to Fix It" [SS17]).

These examples make us understand the importance of efficient UX. In EPP, in addition to the risk in the control systems of missing an alert and having some equipment failure, a bad interface leading to improper action may also cause data loss, unsuccessful experiments or false research conclusions. Therefore, this thesis work also focuses on maximising the user experience of the generated web applications by taking as main usability metrics the legibility, readability and aestheticism of the user interfaces. More specifically, legibility is a measure of how fast and well users can distinguish the letters; readability relates to how easy it is for people to read words, phrases, and blocks; and aestheticism refers to how pleasant the specific UI is for the user. In this research, only the most basic UI style elements that can make a difference in these metrics are taken into consideration, but this choice of elements could naturally be extended in future work:

- background colour;

- font colour;
- font size;
- font family;
- text alignment.

3.4.1 User-Experience Research Findings

Many studies have been conducted over the years on researching the best design principles, and several heuristics have been defined. Most of these research efforts conclude on the fact that there is not a unique choice for the best UI design. Yet, there are certain factors that can influence the UI choice preferences. Some of the most relevant factors are listed below.

- **UI Content** The conducted studies show different results depending on the fact that the page is depicting text or pictures.
- **User's physiology** This factor includes, for example, the user's age or her visual deficits. Studies have shown that almost 50% of the population has astigmatism, making reading on a computer screen somewhat difficult.
- **User's cognitive capability** Not all the users have the same cognitive capabilities. For example people suffering from dementia clearly need more enhanced and adapted UIs [MSJ19].
- **User's cultural background** According to the users' cultural background, what may look appealing for one nationality may have a negative impact on another one. Studies show that the perception of aestheticism clearly varies with the users' cultural background, which thus needs to be taken into consideration [RB13].

Based on these previous factors, some empirical observations presented in the literature and related to the main UI style elements studied in this thesis are detailed in the following paragraphs.

Background and Font Colour

Colour and the way it affects humans have been studied extensively in various different scientific fields such as psychology, human computer interaction or web design. For instance, studies show that in the case that a page contains mainly text, participants were 26% more accurate in reading dark-coloured text on a light background [BC83]. This observation is also supported via experts in the Sensory Perception and Interaction Research Group (University of British

Columbia) by the fact that the large majority of population (about 50%) has astigmatism, which means that with a bright background the iris slightly closes, decreasing the effect of the "deformed" lens, whereas with a dark background, the iris opens to receive more light, and the deformation of the lens creates a much fuzzier focus of the eye [Har]. While it is recommended to use white background on black text, other research studies argue that dark background with the correct contrasting font colour may be equally good (e.g., white text on black background) [Nie99, HH04]. Another survey about the background colour preferences on blogs state that 47% of the interviewees responded that they prefer a light-colour background; 36% of them responded that it depends on the blog; 10% stated that they always preferred a black background; while 6% of them claimed that the colour was irrelevant [Row09]. In the case of web pages depicting images, it is usually suggested to use a black or dark-coloured background, since the dark background emphasises better the content and increases aestheticism.

As previously mentioned, age also plays an important role on user preferences. For example, middle-aged or older people tend to prefer lighter background colour, whereas younger people prefer darker colours. Children and teenagers tend to prefer more colourful and playful background colours.

Another important aspect is colour psychology. For example, an important research observation is that people in general tend to consider short-wavelength colours (blue, green, etc.) as more pleasant than long-wavelength colours (red, yellow, etc). During the research conducted in the work of [GS59], participants rated colours based on preference, which resulted in the following rank ordering, from most to least preferred:

1. blue;
2. green;
3. purple;
4. violet;
5. red;
6. orange;
7. yellow.

A typical example that suggests that these results are coherent is the fact that in most UIs used in the western countries, for example, the *Save* or *Confirm* buttons, associated to an action of approval, are depicted as green, whereas *Delete* buttons, associated to a dangerous action, are depicted as red. This is explained by the fact that short-wave-length colours elicit higher negative or

positive arousal, depending on context, in contrast to short-wave-length colours [Wil66, JH74].

Taking into consideration the above research studies, the main conclusion derived concerning colours is that there should be a high contrast between the background and font colour so that the content of the page is readable. The colour contrast table in Figure 3.6 shows an evaluation of contrast of the main colour w.r.t. each other [Gir19].

		Background								
		Red	Orange	Yellow	Green	Blue	Violet	Black	White	Gray
Foreground	Red		Poor	Good	Poor	Poor	Poor	Good	Good	Poor
	Orange	Poor		Poor	Poor	Poor	Poor	Good	Poor	Poor
	Yellow	Good	Good		Poor	Good	Poor	Good	Poor	Good
	Green	Poor	Poor	Poor		Good	Poor	Good	Poor	Good
	Blue	Poor	Poor	Good	Good		Poor	Poor	Good	Poor
	Violet	Poor	Poor	Good	Poor	Poor		Good	Good	Poor
	Black	Poor	Good	Good	Good	Poor	Good		Good	Poor
	White	Good	Good	Poor	Poor	Good	Good	Good		Good
	Gray	Poor	Poor	Good	Good	Poor	Poor	Poor	Good	

Figure 3.6: Contrast table (from Jeremy Girard, "How to Contrast Background and Foreground Colors in Web Design, 2019" [Gir19]).

Moreover, in order to have more measurable metrics for colour contrast but also for luminescence, the World Wide Web consortium has published an algorithm that determines the contrast (Equation 3.1) and luminescence (Equation 3.2) respectively:

$$\begin{aligned} contrast = max(red_1, red_2) - min(red_1, red_2) + max(green_1, green_2) \\ -min(green_1, green_2) + max(blue_1, blue_2) - min(blue_1, blue_2), \end{aligned} \tag{3.1}$$

where red_1, $green_1$ and $blue_1$ are the RGB values of the first colour, and red_2, $green_2$ and $blue_2$, the RGB values of the second colour to compare.

$$luminescence = (299red + 587green + 114blue)/1000, \tag{3.2}$$

where red, $green$ and $blue$ are the RGB values of the specific colour.

Font Size and Family

Font size is related to the visual capabilities of each individual. Elder users (above 65 years old) would prefer large font sizes in order to make the text more visible (e.g., 18-20 point size) [SB10] while according to research, children prefer 12-14 point size [Nie]. Concerning the font family, sans-serif seems to be usually more legible, while serif fonts look more elegant and appealing to users.

Text Alignment

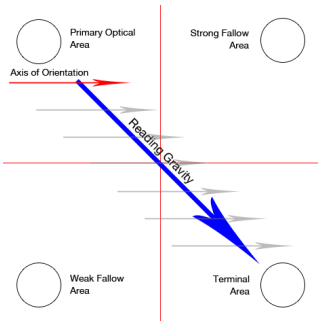


Figure 3.7: Gutenberg Diagram (from Steven Bradley, Design Fundamentals [Bra18]).

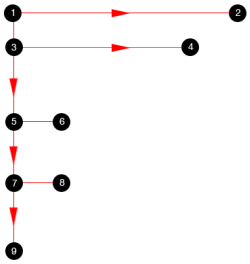


Figure 3.8: F Pattern (from Steven Bradley, Design Fundamentals [Bra18]).

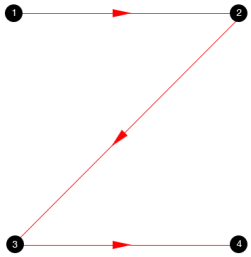


Figure 3.9: Z Pattern (from Steven Bradley, Design Fundamentals [Bra18]).

According to previous research, users, and particularly from the Western countries where text is read from left to right, seem to follow specific reading patterns. The first one is the Gutenberg Diagram, attributed to the typographer Edmund Arnold, who is said to have developed the concept in the 1950s. Since people are trained to read from the top left to the bottom right naturally, they instinctively scan the text starting from the top left down to the bottom right, as shown in Figure 3.7. This applies mainly for text-heavy pages with not any particular hierarchy.

Another study suggests that users follow an F-pattern reading pattern for text-heavy content [PWNG14]. For simpler elements, another common pattern is the Z-pattern. As is obvious by its name, the reading follows a Z-shaped path, starting from the top left to the top right, moving diagonally to the bottom left and then bottom right [Bra18]. From the above studies, one can deduce that it is more common and popular among the people of left-to-right-reading cultures to place more information on the left and center and thus to have a left or center alignment.

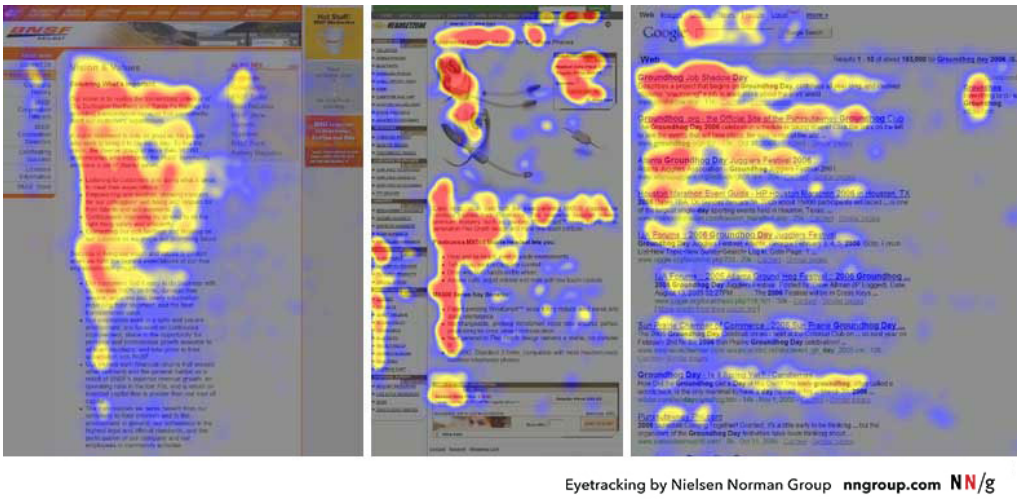


Figure 3.10: F reading pattern eye tracking by Nielsen Norman Group [PWNG14].

3.4.2 UI Customisation

Throughout the years, UX principles have been developed in order to create intuitive and user-friendly interfaces [LHB10]. Nevertheless, it is now broadly accepted that the approach of "one size fits all" is not realistic in a user community composed by different cognitive capabilities, culture and physiology. Therefore, one important good practice suggested in the Interaction Design field is to conduct several user research and usability tests before releasing a software product. These studies, however, try to approximately assess the future satisfaction of users by taking as an example the one of a small sample of users, supposed to be representative. In several research studies, it is however stressed that the best solution would be to allow users to customise their UI [DS13].

Some UIs provide certain preference-configuration capabilities, so that the users can change some basic display parameters. Moreover, several browser extensions that enable the modification of certain styling characteristics of the displayed web pages, such as the font size or font colour, have been developed [AFO, MIN]. This is implemented by the installation of additional JavaScript code that it is somehow covering the issue, but this could also cause some security concerns.

A more pertinent related work is found in a thesis for web page enhancement on desktop and mobile browsers [Yu13]. The purpose of this work is to improve web page readability, skimmability and continuous reading support on mobile devices. A specific format transformation called Jenga is presented, which allows

for a content alignment of text in a format that is more easily readable. An extension that specifies the most significant sentences by applying the LexRank algorithm [ER04] is also presented. Finally a mobile browser considered more readable is proposed. Other related work base their UI customisation on the cognitive style of the users. In the work of [Li08], the content of a specific information palette is adapted according to the users' cognitive style based on the websites he/she has visited. These cognitive style dimensions are defined as "analytic-holistic" or "deliberative-impulsive". In addition to that, each website receives a specific rating related to its specific cognitive dimension, and in this way a clustering of the users is performed. In the work of [Per11], a website is adapted to different morphs [19] based on the click stream of each user; the algorithm can thus learn how to pick the optimal morph based on aggregate user results.

A lot of effort has been put on UI customisation for accessibility. For example, in another related work [ABBR19], a tool for blind people called SuggestOmatic is proposed; it uses a personalised and unsupervised approach for predicting the most likely next browsing action of a user, and proactively suggesting it. In their methodology, they use Logical Segments (LS), which are collections of related HTML elements that share common spatial and functional properties with a discernible visual boundary such as menu, sidebar, search or login forms, main-content, search results, filter options, footer, user comments section, widgets, etc. They construct this model by the use of custom-defined Web-Entity Ontologies and techniques for extracting the visual blocks from a web page.

Nowadays, most of the research related to UI customisation focuses on the automatic suggestion of products to users by the use of different types of recommender systems, a route we are also taking in this thesis. Yet, the purpose of our work is optimising user experience for operational efficiency and not for maximising financial profit. Therefore, the UI customisation in this work is, contrarily to a significant part of the related work, more context-oriented than content-oriented.

3.4.3 Recommendation Models

In order to reach a better understanding on the kind of recommender systems that would be suitable for this thesis work, a detailed research has been conducted on the current state of the art of this field. In the literature, there have been different approaches on recommender systems development depending on the use of the recommender and the data availability related to the recommended objects or users. The three major types of recommenders are based on the popularity, content-based, and collaborative filtering models.

[19] A morph denotes a specific appearance ("look and feel") of a website.

Popularity Model

Popularity models recommend to the users only the items that are most popular at the current time, based on the users previously selecting the specific item. If it does not suggest personalised recommendations, it is however considered a baseline model that could be used when there is not enough information about the users or the items.

Collaborative Filtering Model

Collaborative filtering models are based on the fact that there is a substantial number of users and user-item interactions. These models can also be divided in two subcategories.

- **Model-based** A user-item interaction model is built where representations of users and items are learned from an interaction matrix.
- **Memory-based** This method relied on the similarities among the users and the items. There are two approaches in this case. In the first approach, users with similar preferences are identified, forming a "neighbourhood", and then the systems recommend items according to the preferred items of their neighbour users. In the second approach, items similar to what a user has already selected are proposed.

Collaborative filtering techniques seem to provide accurate results when there is enough data. However, the main problem of this method is called Cold Start, a common problem in recommender systems, when there are not yet enough user ratings or data to meaningfully classify the specific item.

Content-based Model

Content-based models are usually preferred when there is enough information about a specific item (e.g., its description) and may be not enough user ratings. This approach solves the Cold Start problem, since, based on the item's description, it will be recommended to users who have been shown to prefer similar items. Therefore, it is user-independent, and there is transparency on the reason why this item was recommended. The main drawback of this model is its over-specialisation, since it is going to be suggesting to the user only closely similar items.

However, this approach seems more suitable for building a recommender system in the framework of this thesis work since there is enough information on the characteristics of the possible UI styles that will be provided by an ontology, while there will not be much user data at deployment time.

Chapter 4

Data Management System Generation with GenAppi

Version française

*Dans ce chapitre, certaines des principales contributions de ce travail de thèse sont décrites. Dans un premier temps, une ontologie des applications Web fondées sur des ontologies (OWAO) est introduite. Cette ontologie décrit les concepts, opérations et axiomes liés à la génération d'applications Web via l'extraction de données à partir d'une ontologie de domaine. Les concepts OWAO sont intégrés dans une méthodologie générale de génération d'applications Web (GenAppi), qui peut être utilisée pour créer une application Web Django dédiée à n'importe quelle ontologie de domaine fournie par l'utilisateur [DJA]. De plus, OWAO permet également d'enregister les préférences d'affichage de l'interface utilisateur des utilisateurs de l'application Web. Pour tirer parti de ces données, une deuxième méthodologie est présentée et utilisée pour générer des vecteurs de caractéristiques d'ontologie OWAO, ou des plongements (*embeddings*), qui peuvent être utilisés pour recommander automatiquement des interfaces utilisateur personnalisées.*

English version

In this chapter, some of the main contributions of this thesis work are described. At first, a new ontology, Ontology-based Web Application Ontology (OWAO), is introduced. This ontology describes concepts, operations and axioms related to the generation of web applications via data extraction from a domain ontology. The OWAO concepts are integrated in a methodology built on top for the generation of web applications (GenAppi), which can be used to create Django web applications dedicated to any user-provided domain ontology [DJA]. In addition, OWAO also enables the saving of the UI display preferences of each web application' user. Building upon these data, a second methodology is presented and used for generating new OWAO ontology feature vectors, or embeddings,

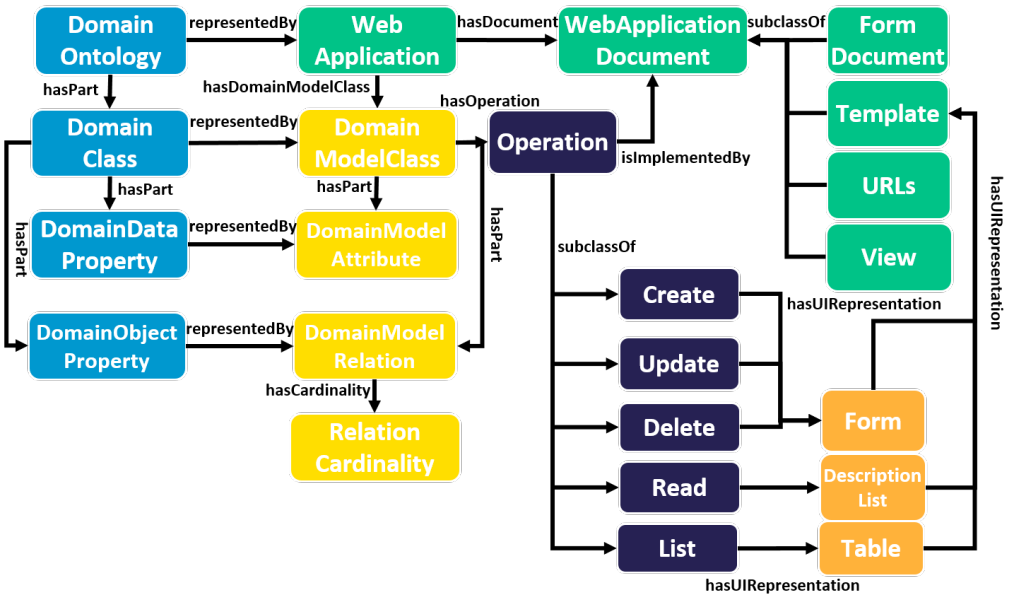


Figure 4.1: OWAO excerpt (light-blue colour: Domain Ontology-related entities, yellow colour: Model related-entities, green colour: Web Application-related entities, dark-blue colour: Operation-related entities, orange colour: UI widget-related entities).

that can be used to recommend personalised UIs.

4.1 Ontology-based Web Application Generation Ontology (OWAO)

We introduce a new generic ontology for web application generation, namely Ontology-based Web Application Generation Ontology (OWAO). It can be seen as a meta-ontology of (meta-)concepts used to describe all the entities present in any preexisting domain ontology, together with those needed to automatically generate and manage a web application of the target ontology instances. The OWAO ontology contains concepts that describe the domain ontology, its model, the (future) web application and its operations (including UI widgets and the UI user preferences).

OWAO serves various purposes in the process of generating a dedicated web application for a given domain:

- it provides the necessary rules, guidelines and restrictions for a consistent web application generation;

- it stores instances of the created files and thus aids in better documenting the application;
- it enables the saving of the UI display preferences for each user;
- it is used as the UI description model for generating UI preferences embeddings for a recommender system.

As already mentioned, the generated web application is based on Django. This is a design choice motivated by the fact that Django is an open-source high-level framework [DJA] based on Python, which is a well established and supported programming language. Django is a modern web technology that ensures better security, for example via Cross-Site Request Forgery (CSRF)[1] protection, quick development and less manual coding, avoiding redundancy.

Figure 4.1 provides an excerpt of OWAO related to the process of generating a web application from a domain ontology. For modularity purposes in the presentation, the part of the OWAO ontology related to UI preferences is further detailed in Section 4.3. As displayed in Figure 4.1, a domain ontology (an instance of `DomainOntology`) is considered to be composed of elements of the classes `DomainClass`, `DomainDataProperty` and `DomainObjectProperty` concepts. These domain concepts are then mapped to the OWAO concepts that describe any Django web application, thus opening the door to the ultimate automatic management of all the data belonging to the domain ontology.

Django web applications follow a Model-View-Template (MVT) architecture. The Model layer defines the domain data structure describing the tables and fields of the underlying database storing the application data. The View layer defines the processes used for retrieving, formatting or saving the data into the database defined in the Model layer. The Templates are used to render the information to the client. In order to generate a domain-specific web application, the concepts of the domain ontology have to be mapped to the corresponding concepts of the MVT architecture, which will enable the automatic generation of the MVT-compliant documents (instances of `WebApplicationDocument`) for the web application.

The domain concepts must be mapped to the web-application-related concepts (`DomainModelClass`, `DomainModelAttribute` and `DomainModelRelation`) of the Django web application model. These mappings are used for the generation of the Model of the web application (see Subsection 4.2 for more details). Then, for each `DomainModelClass`, the UI `Operation`s such as `Create`, `Read`, `Update`, `Delete` or `List` that are to be supported, must be specified. In the generated web application, this means that the appropriate `WebApplicationDocument`s will be

[1] Cross-Site Request Forgery (CSRF) is an IT-security attack that makes users perform malicious requests through the web application when they are authenticated

available to the user, for example `View` documents implementing the functionalities of each `Operation`, and this for each `DomainModelClass`. At the front-end level, every `Operation` is represented by a specific UI fragment `UIFragment`, e.g., `Form` or `Table`.

The next chapter will provide a detailed application of OWAO in the field of EPP, and thus put all these new concepts more clearly in action.

4.2 Data Management Web Application Generation (GenAppi)

The GenAppi generator, developed in the framework of this thesis work and applied in the domain of EPP (see next chapter), is also built in Python. Exploiting state-of-the-art technologies for ontologies and web applications, GenAppi is based on three main technologies. The first one is Owlready2, a Python module built with the purpose of enabling ontology-oriented development in Python, which in GenAppi acts as a bridge between the ontology and the web application actual code [Lam17]. The second technology are the Jinja2 templates, used for the software representation of the actual UI fragments [JIN]. The third one is Django [DJA], the Python framework mentioned in the previous section and used for building and running the whole web application.

4.2.1 Algorithm and Workflow

The GenAppi software tool follows a specific workflow for the generation of OWAO-based web applications. The algorithm for generating web applications is presented in pseudocode in the Algorithm 1 block below; its main steps are also displayed in Fig. 4.2 and described in this section.

Loading Ontologies

Owlready2 [Lam17] is the module that GenAppi uses for handling ontologies within Python; it enables existing ontologies to be represented as Python data structures, opening the way to their dynamic management via Python at run time. It allows for a quick access and search of ontologies as well as creating, editing and deleting ontology instances. Since the Django framework is based on Python, the Owlready2 module is simply imported in GenAppi. As shown in Line 1 of Algorithm 1, when GenAppi is started, the domain and OWAO ontologies are loaded into GenAppi via Owlready2 in order to extract the necessary information.

Figure 4.2: GenAppi generator workflow.

Domain-Ontology-to-Model Mapping

Following the user-provided OWAO mappings, the domain-ontology classes (`DomainClass`) are transformed to Python classes (`DomainModelClass`) used to create the Model of the future Django web application. Data properties (`DomainDataProperty`) are transformed into attributes (`DomainModelAttribute`), and their datatypes are transformed to the equivalent datatypes in the model, e.g., `xsd:string` to `TextField`. Object properties (`DomainObjectProperty`) are mapped to relations (`DomainModelRelation`) of the Model. Their cardinality (`RelationCardinality`) constrains the Django relationship (e.g., ForeignKey or ManyToMany).

Using an example taken from the Irradiation Experiment Data Manage-

Algorithm 1 GenAppi
Input: *domain_ontology*
Output: client code and server

```
load domain_ontology, owao
load Jinja_templates
map_domain_ontology_to_model(domain_ontology, owao)
create URL_file
for op in owao.Operations() do
    create view_file(op)
    get template(op) from Jinja_templates
    for cl in domain_ontology.classes() do
        adjust template(op) to cl
        view_file(op).append(cl.template(op))
        create cl.URL(op)
        URL_file.append(cl.URL(op))
    end for
end for
WebVOWL_JSON_file ← owl2vowl(domain_ontology)
assemble files in directory
migrate Model
start server
```

ment (IEDM) ontology [GJR19b], detailed in the next chapter, the OWL triple `IrradiationExperiment createdBy exactly 1 User` is transformed to the field `createdBy = ForeignKey("User")` in the Python class corresponding to the `IrradiationExperiment` domain class.

Operation Handling

As displayed in Figure 4.1, CRUD operations such as `Create` or `Update` can be associated to Model classes (`DomainModelClass`). For each of these operations, GenAppi provides a corresponding Jinja2 [JIN] template that includes the code implementation of the operation, independently of the Model. GenAppi uses these templates to generate *View* Python functions, used for the execution of the operations for each class. These functions are responsible for handling the communication (read/write) among the ontology or the defined triple store, the database and the front end.

Moreover, the Django framework provides an integrated authentication and authorisation system. In order to use these functionalities, specific Jinja2 templates were created to generate the user interfaces that allow users to register and sign into the web application.

Generating User Interfaces

As shown in Line 9 of the Algorithm, for each operation and ontology class, GenAppi adapts a preexisting Django template. These templates are needed to provide the UI presentation logic of the generated web application, rendering and presenting data to the users. A unique URL is associated to each template instance, and acts as a link to the corresponding `View` document.

Domain-Ontology-to-JSON-File Creation

WebVOWL is a software tool that allows for the easy visualisation and editing of ontologies [WLA18]. This tool builds upon the conversion of the ontology to JSON. WebVOWL is integrated in our framework and customised to fit the configurations of a Django application. This allows for an easy graph-like visualisation of the domain ontology that is at the origin of the generated application.

Web Application Packaging

After the generation of all the previously mentioned files, GenAppi assembles them in specific directories. Moreover, the OWAO and domain ontologies are copied in the web application, allowing for easier accessibility and portability, while making it possible to create future instances of the domain ontology classes through the web application, whenever the user inputs new data.

Model Migration

The last step of GenAppi is to handle the Model migration to a dedicated database. The default database managed by Django is SQLite3, but this can be changed easily in the settings. In addition, the instances are saved also in a local copy of the ontology. After the migration, the administrator can start the web server, and the web application is finally ready for use. The web application runs on localhost and the name is taken by the domain ontology but it can also be changed through the Django settings file.

From that point on, instances of the user-specific domain classes can be generated and managed via the UI client automatically generated by GenAppi, in compliance with OWAO-specified requirements. Moreover, the instances are also stored in the domain ontology or configured triple store.

4.2.2 Discussion

In the framework of this thesis, GenAppi is used for the generation of data management systems dedicated to irradiation experiments via the use of the IEDM

domain ontology. The ontology's main concepts and the whole application of this workflow will be further described and analysed in the dedicated Chapter 5.

The algorithm described in the previous paragraphs shows that the time for the generation of the web application strongly depends on the number of classes of the ontology and the operations to be generated. Given that an ontology has m classes and n OWAO operations, the time complexity would be

$$O(mn). \tag{4.1}$$

Moreover, some preliminary tests on generating web applications from preexisting domain ontologies such as SOSA [JHC+19], EXPO [SK07] and Marinetlo, an ontology for the marine domain [TAB+13] have shown positive results that GenAppi is not bound to IEDM and is indeed domain-independent.

The code of GenAppi as well as the OWAO ontology can be found in the online resources[2].

4.3 UI Customisation and Personalisation

Section 3.4.1 highlights the importance of good user experience and the fact that user preferences may vary depending on different factors such as the UI content, users' cultural background, physiology or cognitive capability. Moreover, several research studies have pointed out that the best solution for addressing this diversity would be to allow users to customise their own UI [DS13]. Therefore, an important issue when thinking of providing automatically generated web applications is the question of UI customisation.

4.3.1 Recommendation Generation

In order to enhance the quality of GenAppi-generated applications, one can think of having such systems offer recommendations regarding good UI styles to users. For example, users could get a recommendation for a given UI style that seems more suitable to their needs, or they could also receive the suggestion to customise some specific parameters (e.g., font size or background colour) according to best practices. In order to provide such a service, the features of the UI preferences that can be personalised, as well as their instances created when a user chooses his/her own settings, are saved in the OWAO ontology specific to each application. This information can then be used as a prior knowledge for recommending specific UI styles to users.

In order to help provide such a service, concepts and relations representing some key UI preferences have already been introduced in the OWAO ontology.

[2] https://gitlab.cern.ch/irrad/genappi

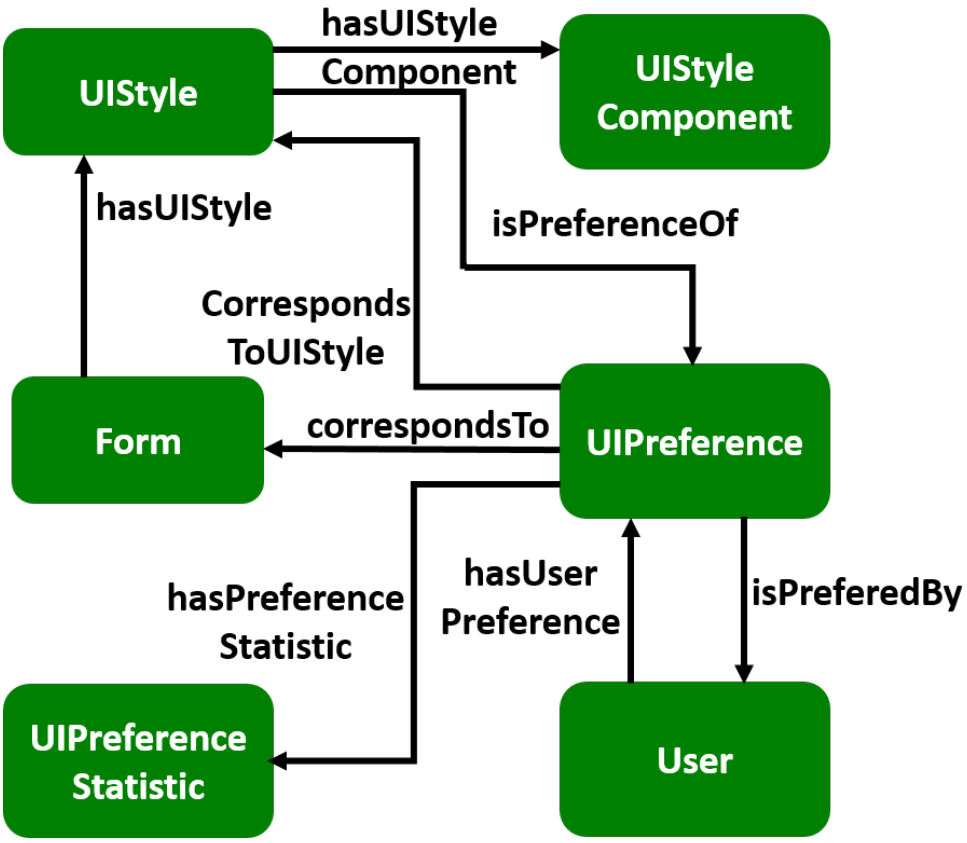


Figure 4.3: OWAO part for describing UI preferences.

Figure 4.3 demonstrates the example of the `owao:Form` UI fragment. A specific instance of `owao:Form`, generated by GenAppi, will have a specific `UIStyle` composed of instances of the `owao:UIStyleComponent` class. Some subclasses of `owao:UIStyleComponent` are `owao:Color` and `owao:FontSize`. A certain `UIStyle` is a preference of the class `owao:UIPreference` that corresponds to a specific UI fragment, such as the `owao:Form` in this case. Each `owao:User` has a set of elements of `owao:UIPreference`.

Given all the choices made by the users, different statistical values can be gathered, to help the recommendation process; they are instances of the `owao:UIPreferenceStatistic` class. Our approach is then to consider the specific ontology structure as well as its data instances as a useful input dataset for building and testing an OWAO-provided recommendation model, further described in the sequel.

However, before proceeding along those lines, some issues need to be addressed. The first one is the question of access to data, needed to populate such

recommender systems. Research was thus conducted for finding publicly available data related to UI preferences in well-known platforms that include open datasets such as the Google Dataset Search[3], Kaggle[4] or Microsoft Research Open Data[5]. However, it appears that there is no open-access data available describing user preferences for UIs. This is because most recommender systems are used for commercial and advertising purposes, in order to recommend products to users and maximise the profit of the vendor. In our case, the aim is to maximise the user experience of the generated web application.

The second limitation when thinking of building a recommender system for a generated application is that the user community is rather limited, at least at first. This is a serious issue, since an important step when building recommender systems is its training phase, which leads to the problem of Cold Start, when a new user first registers onto a web application and prior data is lacking or scarce.

Considering these two limitations, our methodology for UI recommendation follows a hybrid approach, mixing the popularity (see Section 3.4.3) and content-based models (see Section 3.4.3); this latest model uses ontologies as its source dataset. More specifically, the popularity model is chosen in order to confront the problem of Cold Start by suggesting to the user the UI style that is currently more popular among the users. However, since the most popular profile is not always the best choice for each user's taste, a hybrid approach based on ontology embeddings is studied in order to provide more personalised recommendations to the user.

4.3.2 Popularity Model

The popularity model is a simple approach that can give to the recommender a head start when a new user arrives. As illustrated in Figure 4.4, the recommender system can propose to the user different UI styles, based on the previous users' choices. According to the UI style that this new user selects, feedback is obtained and used to update the OWAO ontology, via the creation or updating of new instances of the classes related to UI preferences. The recommender system will process the new feedback, based on the current state of the OWAO UI preferences, and periodically recommend the most popular UI styles to the users. Since the users are able to change their preferences from the interface, the OWAO-defined statistical reporting values related to UI preferences will change with time, leading to more pertinent recommendations to old and new users alike.

[3]Google Dataset Search https://datasetsearch.research.google.com/

[4]Kaggle is a platform for entering in machine learning competitions by using open datasets https://www.kaggle.com/

[5]https://msropendata.com/

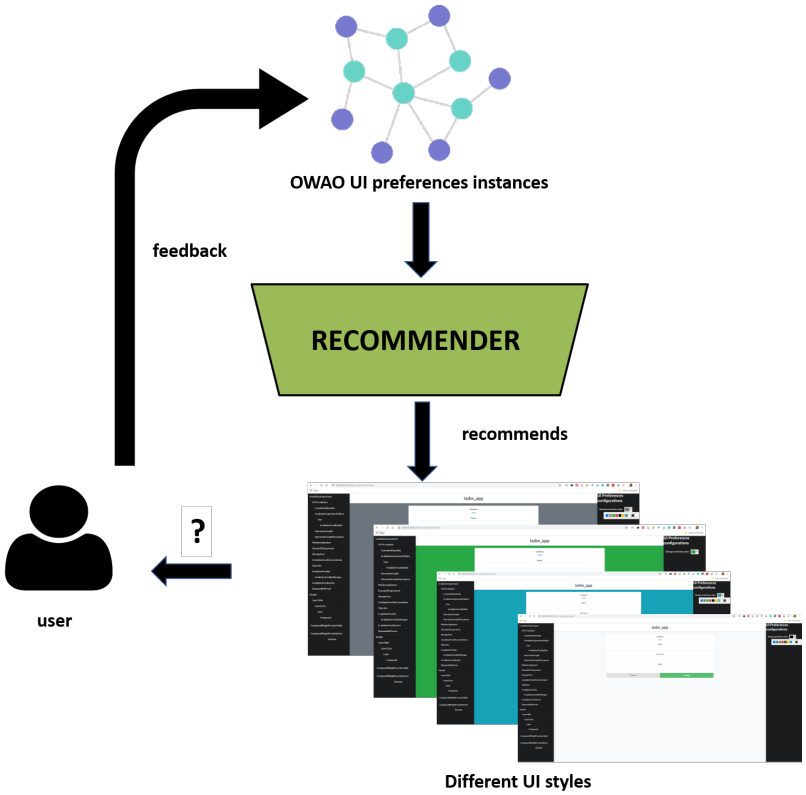


Figure 4.4: Recommender model. The user is shown different UI Styles for a specific UI and chooses his/her preference, which is provided as a feedback to OWAO.

4.3.3 ontowalk2vec, a New Ontology Embedding Model

The popularity model, described in the previous subsection, is recommending the UI Style that is most preferred by the current users, but it is not providing personalised recommendations. For this reason, more advanced recommender system technology is needed for fulfilling our goal of providing user-specific UI recommendations as part of GenAppi-derived applications.

Most of the current content-based recommenders use the Bag Of Words (BOW) technique for recommendations. This methodology is based on the principle of having a vocabulary of chosen words and counting the frequency of the words in the specific documents. This method does not take into consideration the proximity and order of the words. For example, they cannot notice the difference of a colour that is a background colour or a font colour, because they count only the frequency of the specific word, here "colour", in the document but not its different semantics. For example, according to these approaches, a UI with white background colour would have a high similarity score with a white font

colour, which is not correct. Having an ontology, as we do here, can contribute in classifying, for example, if this colour will be a background or font one, and this specific information should not be dismissed by the recommender.

In order to tackle this issue, a new model-based hybrid approach is implemented in this thesis. This methodology relies on word representations as vectors or, as they are called, "embeddings" that can be used to mathematically define similarity values for the instances of the ontology. The model does not only learn which instances are similar, but also the structure of the ontology, providing further information to improve the computation of the relatedness relationship for the instances.

The developed methodology, called "ontowalk2vec" is inspired by the current reference state-of-the-art models for word embeddings and random walks of graphs generation. These models are further analysed below, after which we define ontowalk2vec. A whole chapter (Chapter 6) is dedicated to analysing the performance of this new embeddings.

word2vec

One of the most influential works concerning word embeddings in the recent years is the NLP model called word2vec [MCCD13]. This model takes as input sentences, seen as sequences of words, and compute a vector embedding for each word. In word2vec, two different architectures are proposed: the Continuous Bag of Words (CBOW) and the Continuous Skip-gram (see Figure 4.5). CBOW predicts, provided a set of words as context, a word that could fit next in that context. On the other hand, in the Skip-gram approach, when a word is provided, its fitting in a specific classification is used to predict the context.

node2vec

The word2vec model is efficient for extracting word embeddings in text, for example in an article. However, it is not sufficient when the input is a graph. As an ontology can be assimilated to a graph, we need to extract embeddings from graphs. For graph embeddings, a model called node2vec was introduced [GL16]. This model uses the two classic search strategies in graphs: the Breadth-first Search (BFS) and Depth-first Search (DFS). These two algorithms provide two different ways of visiting the nodes of a graph. In BFS, starting from a specific node u, all the closest neighbouring nodes are visited first, as depicted in red arrows in Figure 4.6. In contrast to BFS, DFS visits the nodes in depth, as depicted in blue arrows in Figure 4.6.

These search algorithms can be used to extract random walks from a graph in order to build node sequences that can then be considered as sentences and input in word2vec. The node2vec algorithm relies on two hyperparameters: p,

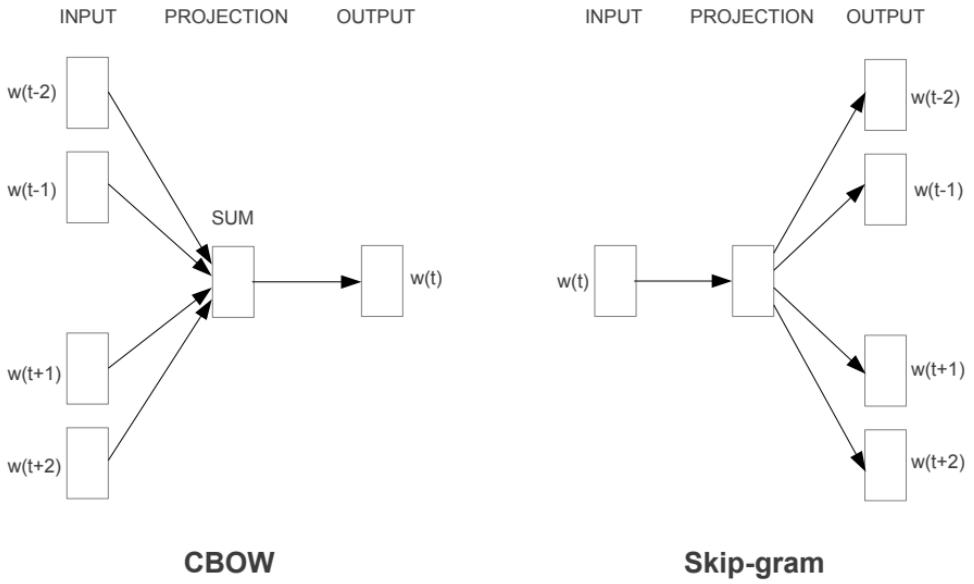


Figure 4.5: word2vec architectures. The CBOW architecture predicts the current word based on the context; the Skip-gram mode predicts surrounding words, given a current word [MCCD13].

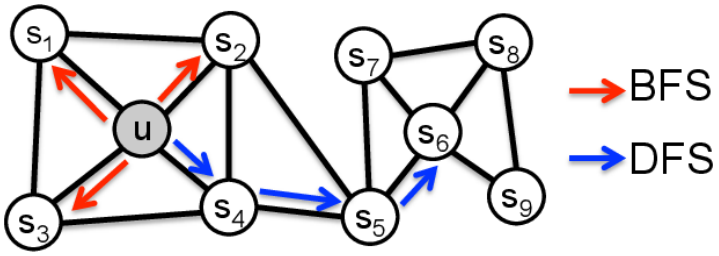


Figure 4.6: Starting from node u, a Breadth-first search (BFS) traversal is depicted in red arrows, while a Depth-first search (DFS) path is depicted in blue allows (from A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks" [GL16]).

which controls the probability that a node in the walk is revisited, and q, which controls whether a BFS or DFS approach should be employed depending on its value. Figure 4.7 shows the possible random walks passing through t to v; the next node to be traversed in this particular random walk is stochastically selected according to the various branching probabilities, which depend on a.

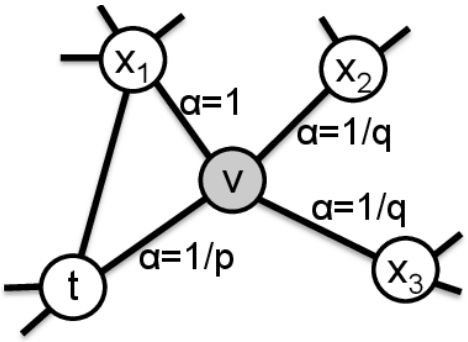


Figure 4.7: node2vec random walk generation based on the hyperparameters p and q. The walk transitions from t to v and is evaluating its next possible steps out of node v (from A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks" [GL16]).

RDF2Vec

RDF2Vec is used for extracting embeddings from RDF Graphs [RRN+19]. In order to generate adequate random walks, RDF2Vec uses a BFS of a certain depth, as in node2vec, but does not offer any hyperparameter for adjusting the paths of the generated randow walks. In addition, RDF2Vec also utilises the Weisfeiler-Lehman algorithm, which computes subtree kernels for graph comparison [dV13]. RDF2Vec plays an important role in the generation of random walks for ontologies and is thus integrated in our ontowalk2vec model, described in the following paragraph. A python implementation of RDF2vec is the pyRDF2Vec [VSA+20], used also as a base for our code.

ontowalk2vec

To take full advantage of our ontology-based approach to data management system generation, we developed, in the framework of this thesis work, a new technique for computing ontology embeddings; named ontowalk2vec, it is inspired by the two main models detailed above, i.e., RDF2Vec [RRN+19] and node2vec [GL16]. Both models focus on creating different random walks on an ontology or a graph, and utilising these random walks as input sentences to word2vec [MCCD13]. These two methodologies are considered complementary within our method. The node2vec algorithm focuses on the structural part of the ontology by treating it as a graph and extracting random walks - sentences of different lengths - from it, while neglecting the object and data properties. On the other hand, RDF2Vec focuses on the RDF triples including instances and their relations (object and data properties); this method also relies on word2vec, used for generating the final embeddings.

As previously described, word2vec follows two specific training architectures: CBOW and Skip-gram. In the CBOW architecture, the order of the word instances does not influence the result. However, in ontologies and KGs, the structure and order of the instances is significant for their semantic meaning, and therefore CBOW is not suitable for our purpose. For example, in a RDF triple, the subject and object are strictly non interchangeable, otherwise the whole semantic meaning changes. In contrast to CBOW, the Skip-gram architecture focuses on the context of the instances and provides better semantic accuracy in comparison to CBOW [MCCD13]. For these reasons, Skip-gram has been selected as our training architecture of choice.

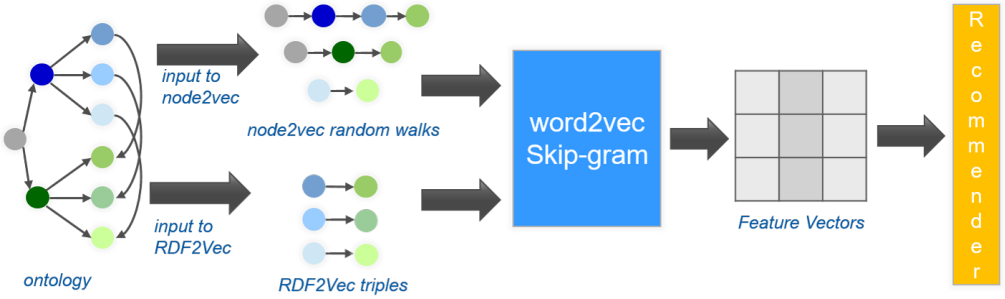


Figure 4.8: Embedding workflow with ontowalk2vec. The OWAO ontology is used as an input for node2vec and RDF2Vec where node2vec random walks and RDF triples are extracted and used as input sentences to word2vec Skip-gram. The generated word2vec feature vectors can then be used for a recommender system.

As displayed in Figure 4.8, in ontowalk2vec, random walks are extracted from OWAO using both the node2vec and RDF2Vec methodologies and fed to word2vec. After training word2vec Skip-gram with the specific random walks, the entity embeddings are generated and can then be used for the recommender system.

All these models rely on several hyperparameters (e.g., iteration number, learning rate, window size, etc.) that need to be initialised before starting training the model and generating embeddings. Even though in many works these hyperparameters are initialised with their default values, studies have shown that the best values for these strongly depend on the type of data the models are trained on, and thus these parameters should be chosen with caution; otherwise they may affect the final accuracy of the model [CDLRL18, Dil19]. For these reasons and in order to maximise the accuracy of our proposed methodology, a thorough investigation of this issue was carried out by performing several experiments and evaluating the final results. Further information about these experiments and the informed choice of hyperparameters is provided in Chap-

ter 6, where details are provided using two domain ontologies, ontowalk2vec is compared with the current state-of-the-art methods, the model is optimised and results by using the UI preferences stored in OWAO are presented.

4.4 Summary

In this chapter, the methodology GenAppi for generating a web application from a domain ontology is presented. This methodology is based on the OWAO ontology providing the rules for the generation and documentation of the generated application. Furthermore, OWAO is used for the storage of the users' UI preferences. This allows for building, on top of OWAO, a methodology for the development of a recommender system that can customise and suggest to the users UI styles according to their UI preferences, thus providing a better UX when using GenAppi-generated management systems.

In the following chapter, an application of the GenAppi methodology for the EPP community, more precisely the data management of irradiation experiments, is demonstrated. Then, a final chapter (Chapter 6) presents extensively the results obtained when fine-tuning the generation process of OWAO ontology embeddings and how these can be used for building a dedicated recommender system for each generated web application.

Chapter 5

Application of GenAppi to EPP Data Management

Version française

Ce chapitre présente une application de GenAppi en détaillant un cas d'utilisation de la gestion des données dans le domaine de la physique expérimentale des particules. Plus précisément, ce travail de thèse a été initié par la compilation des exigences pour la gestion des données d'une installation d'irradiation spécifique, le Proton Irradiation Facility (IRRAD) [GGMR15] au CERN [CER]. Une application Web sur mesure (appelée IRRAD Data Manager, ou IDM) [GJPR19] pour cette installation a été développée, et des concepts clés décrivant la gestion des données des expériences d'irradiation ont été identifiés et formalisés par une ontologie pour la gestion des données des expériences d'irradiation (IEDM) [GJR19b]. Dans cet exemple, IEDM est utilisé comme entrée d'ontologie de domaine au logiciel GenAppi afin de générer un équivalent de l'application web IDM. Enfin, l'application Web générée est comparée à IDM.

English version

This chapter presents an application of GenAppi by detailing a data management use case in the domain of Experimental Particle Physics. More specifically, this thesis work was initiated by the compilation of the requirements for the data management of a specific irradiation facility, the Proton Irradiation Facility (IRRAD) [GGMR15] at CERN [CER]. A custom-made web application (called IRRAD Data Manager, or IDM) [GJPR19] for this facility was developed and key concepts describing the data management of irradiation experiments were identified and formalised by a new ontology, the Irradiation Experiment Data Management (IEDM) [GJR19b] ontology. In this example, IEDM is used as a domain ontology input to GenAppi in order to generate an equivalent to the IDM web application. Finally, the generated web application is compared to IDM.

5.1 IRRAD Data Manager (IDM)

As mentioned in Section 2.2.3, IRRAD is a reference facility for performing irradiation experiments and is considered as the main use case for this thesis work. In the early years of IRRAD, a software application called Sample Manager was developed and used on a local computer for the registration of the samples to be irradiated. Since 2014, the IRRAD facility was upgraded in order to cope with the increasing demand for irradiation experiments at CERN, and therefore this system was outdated, running only on local computers with specific drivers, and could not be upgraded. Thus, these facts called for the development of a new web-based system with specific software requirements. These requirements were that the new system should be in line with the new IRRAD facility software needs, integrate into the modern software infrastructure of CERN, and provide better User Experience (UX). Since this application would be online, security was also a crucial issue to be considered. Moreover, after the irradiation experiments are completed, the samples become radioactive, and therefore this system had to be also compliant with the CERN traceability procedures and able to send/receive data from the official system used by CERN for the traceability of potential radioactive equipment (TREC) [KBMA+13]. Last but not least, the scalability and portability of the system were also important aspects taken into consideration for long-term developments and associated to facility upgrade plans.

5.1.1 Design Life-Cycle

As stated in the previous section, User Experience was an important aspect for the new IDM web application. In particular, IDM was supposed to provide an intuitive interface targeting different user groups such as physicists, engineers and technicians. Therefore, the IDM design was based on UX universal principles in order to be intuitive and user-friendly [LHB10]. In particular, a User-Centered Design (UCD) approach was applied, which allows for a better understanding and identification of the users' requirements and goals [PRS15].

Research and Exploration

The first phase of a UCD (process) focuses on the users' needs and tasks. For this reason, the CERN Irradiation Facilities online database was used in order to find information and details about existing irradiation facilities at CERN and worldwide [CIF]. Then, we interviewed coordinators, operators and users of IRRAD and of some of the other irradiation facilities listed in the irradiation facilities database. This allowed for a better understanding of the current state of the art regarding the used software tools. We thus gained a better insight of

the users' needs, reaching the conclusion that there was no off-the-shelf existing tool that could already fulfil these requirements.

Insight Synthesis

The second phase of a UCD requires taking into consideration the previously acquired user insights and defining specific user requirements and use cases. In this step, roles, permissions and tasks for each role were described in detail [GG16]. For instance, one specification is that an administrator is authorised to view all the experimental data of the facility, whereas standard users can only access the data of the experiments they are participating in.

UI Prototyping

After several iterations of this requirement and use case definition process, the next step is the design of (possibly many) User Interface (UI) prototypes. For producing such a UI prototype but without implementing any actual functionality, the use of a wireframe software such as Balsamiq was deemed necessary [BAL]. Balsamiq is a graphical-user-interface application that allows developers to build and test easily UI prototypes. Figure 5.1 provides some examples of such early UI design prototypes.

Evaluation

The final step of a UCD, before the actual software implementation, is the evaluation of the projected designs. The already built prototypes were tested and evaluated by the facility operation team, and the most suitable prototype was chosen for implementation. This final UCD phase may, in practice, require to return to the UI prototyping phase if none of the proposed UI designs end up fitting the evaluators' expectations.

5.1.2 Development

The four main concerns guiding the software design and development of IDM were: compliance with the CERN software infrastructure, security, scalability and portability specifications. These issues are discussed in this subsection, while detailing our software choices, the IDM architecture and its deployment facets.

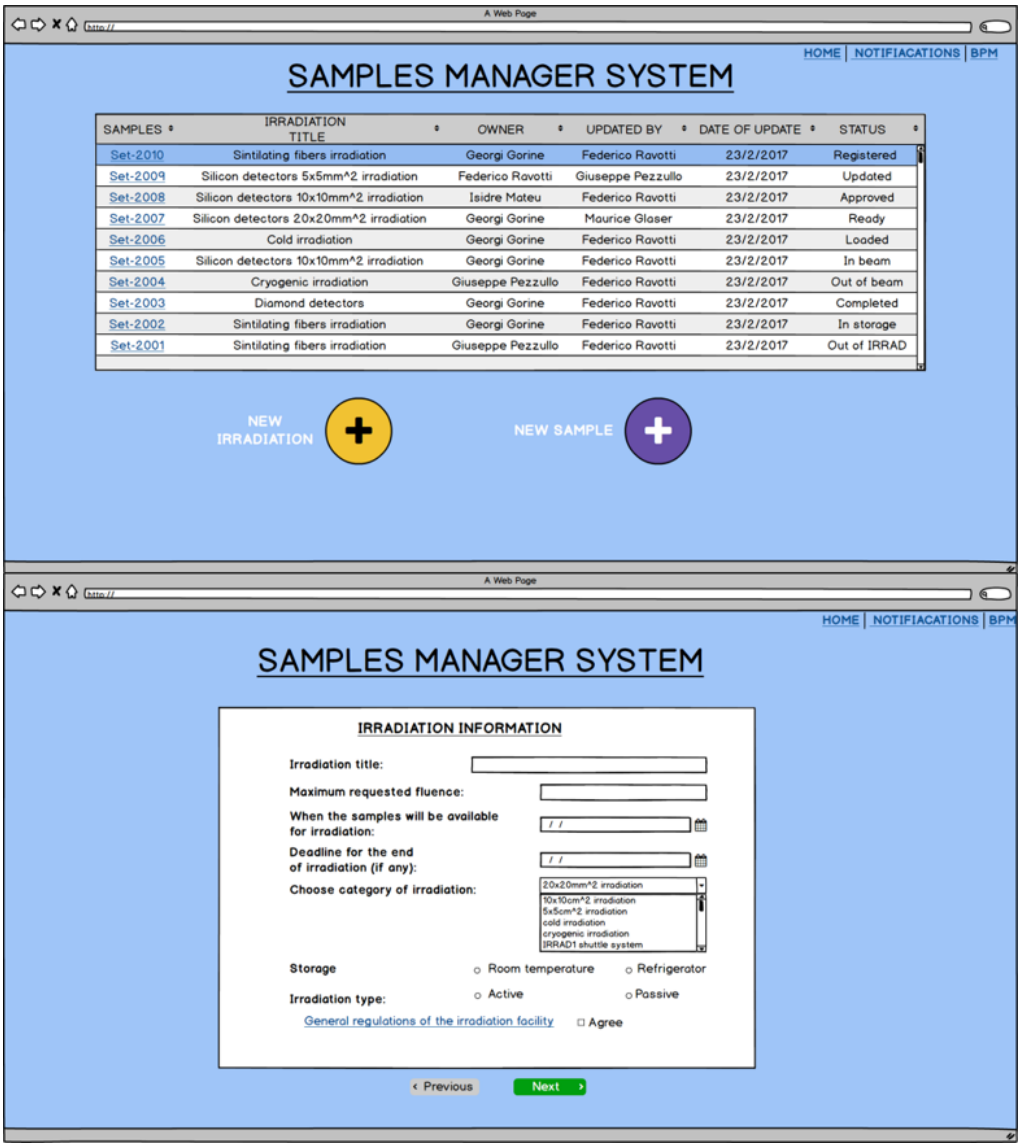


Figure 5.1: Balsamiq prototypes.

Software Choices

To tackle these main concerns, the following state-of-the-art web technologies were chosen.

- **Django** was chosen for the development of IDM. As already mentioned in Chapter 4, it is an open-source Python framework [DJA] that ensures better security and robustness over other frameworks.

- **Oracle Database** was selected as the default database for storing the IDM data. The choice of the Oracle database is based on the fact that Oracle is broadly used in other CERN software infrastructures and is also well supported by CERN, thus allowing for better database maintenance and backup opportunities [ORA].

- **JavaScript and JQuery** were chosen for the purpose of making IDM more user-friendly. JavaScript and JQuery, a JavaScript library that helps with the high-level manipulation of HTML and other web-related data structures, were chosen for the implementation of the IDM front end in order to simplify the development of its interactive features [JQU].

- **Semantic UI**: The IDM front end aims for a minimalistic design, focusing only on the components necessary to be compliant with the UX principles [LHB10]. For this reason, the layout of IDM was implemented via the Semantic UI development framework [SUI], which provides enough aestheticism and responsiveness to user interfaces to fulfil the users' requirements we had gathered.

5.1.3 Architecture

The IDM software architecture is mainly based on the Django architecture (see Figure 5.2), i.e., a Model-View-Template (MVT) architecture.

The Model layer provides the structure and primitives for the manipulation of user data. As shown in Figure 5.2, this layer is, in this case, directly mapped to an Oracle database, which means that each Model class is represented by a database table. In addition to the IDM data, the Oracle database stores also operational information related to the irradiation facility coming from the dedicated IRRAD control system [GGJ+17] such as the particle flux, read out by IDM and used for the real-time computation of the estimated cumulated proton fluence.

The View layer provides the logic for user request processing and response. In this layer, the Infor[1]-provided suite of web services for Enterprise Asset Management (EAM) [Mar13] is integrated in order to allow for a transparent communication between IDM and TREC [KBMA+13]. These web services implement the requests for CRUD operations in the Infor EAM system used in the back end of TREC [Mar13].

Finally, the Template layer provides a specific syntax used for rendering the information and presenting it to users. As shown in Figure 5.2, the Semantic UI framework, JavaScript and the JQuery library were integrated in these templates in order to provide an intuitive and interactive user interface for IDM.

[1] www.infor.com

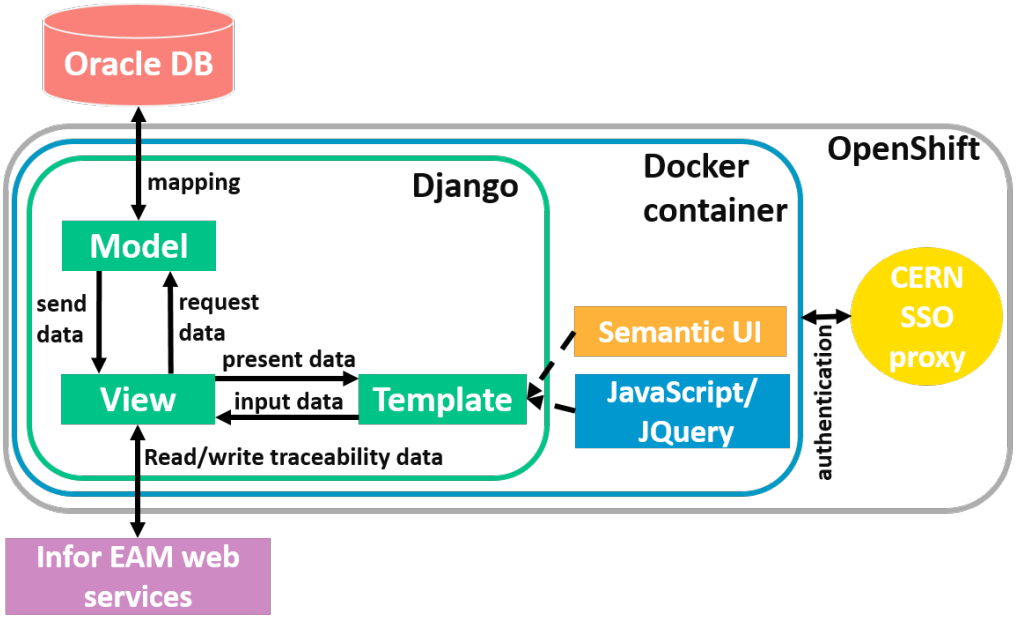


Figure 5.2: IDM architecture.

5.1.4 Deployment

The so-called "DevOps" technologies are used in modern software development projects to seamlessly provide Continuous Integration and Continuous Delivery (CI/CD) of software. Accordingly, a Docker container [DOC] is used for the containerisation of the IDM software, in order to deal with potential dependency issues when running this software in various computing environments. Leveraging the software services provided by CERN, this Docker image is built on the CERN Platform-as-a-Service (PaaS) infrastructure, based on RedHat OpenShift, which makes the automatic process of building Docker images [LPW17] somewhat easier. OpenShift is based on Kubernetes, an open-source system for automating deployment for containerised applications [KUB].

In line with our focus on security issues, IDM uses the CERN Single Sign-On (SSO) system for user authentication, ensuring in this way a level of security for IDM comparable to other CERN services. For this purpose, a basic Apache server was deployed; it acts as a SSO authenticating proxy for the application.

Appendix C shows some guidelines on the way to install and deploy IDM in localhost.

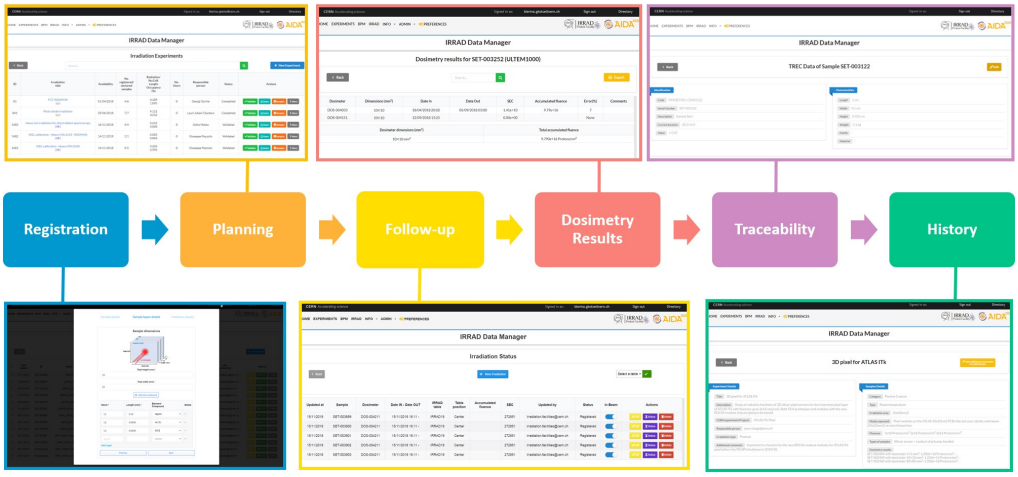


Figure 5.3: IDM functionalities.

5.1.5 Functionalities

IDM provides the key functionalities needed for the data management of the IRRAD facility, the follow-up of irradiation experiments, and the traceability of the IRRAD samples and other potentially radioactive components (see Figure 5.3).

Registration

All the data related to an irradiation experiment are registered in IDM. Users first register into the dedicated mailing list (e-group) of IRRAD users; this provides the permission to the users to log into IDM. Once logged in, they can request to perform an irradiation experiment by providing details such as information about the samples to be irradiated, the type of irradiation experiment requested and the availability of the components to be tested. Moreover, for reasons of data privacy, users can only see and modify the experiments to which they are associated and take part in. Thus, when a user registers an experiment, he/she also needs to assign access permissions to the specific users that this experiment will be visible to.

On the IRRAD side, the facility coordinators who have access to the IDM "administrator" view can approve the experiment requests, if the experiment complies with the planning of the facility and the irradiation procedures. Once the irradiation experiment is approved, the users are then allowed to register the samples corresponding to their experiments; they can also provide additional

details such as the quantity, dimensions and physical composition of the samples; this data is necessary for computing the parameters related to the operation.

Planning

In IRRAD, there are nine remotely controlled, movable stages (IRRAD tables) along the path of the proton beam; they are used to place the samples for irradiation [GMR13] (see Section 2.2.3). This setting introduces some complexity when planning the irradiation experiments. IDM enables a better planning of the experiments' sequence by providing an overview of the registered experiments and samples, and of their availability, to the facility coordinators and operators. Through IDM, the coordinators can assign the samples to a suitable support and keep track of their positions. Moreover, they can associate dosimeters to the relevant samples; those will be used for assessing after irradiation, through spectrometry, the exact cumulated fluence.

Furthermore, the fact that the samples are positioned in series, one after the other, along the beam line causes another challenge in the planning of the irradiation experiments in IRRAD. Even though a high-energy beam does not degrade completely after interaction with a thin layer of matter, however, this interaction causes a degradation of the proton-beam quality along the beam line, if a significant amount of matter with considerable density (i.e., interaction length) is put simultaneously into the beam. For mitigating this phenomenon, users are asked to register the precise material composition of their samples. This information is then used for the computation of experiment-specific proton interaction lengths (see Section 2.2.2), which provide the necessary information to the coordinators about the amount of samples that can be irradiated at the same time and in a given order in beam.

Dosimetry

As previously mentioned, dosimeters are used in order to assess the cumulated fluence actually received by the samples; therefore they are placed together with the samples during the irradiation experiment. To get an accurate value, a gamma-spectrometry analysis of these devices is performed once an irradiation experiment is completed [CFG+17]. These data are registered and provided to the users as the final dosimetry results, necessary for further physics data analysis.

Follow-up

Once an irradiation experiment is started, IDM provides a status panel for the follow-up of the experiment; it is refreshed every five seconds. The start of an

experiment triggers the corresponding functions for reading the beam flux values and providing an estimate of the cumulated fluence. When the IDM-estimated cumulated fluence reaches the user-requested fluence, the irradiation experiment is considered as completed, and a notification is sent to the coordinators.

Traceability

CERN uses a dedicated system for ensuring the proper traceablity of potentially radioactive equipment [KBMA+13]. In order for IDM to be compliant with these traceability procedures, naming conventions compatible with TREC are used in IDM when labelling the samples and dosimeters used in the facility. Moreover, through IDM, the relevant labels can be printed and then attached to the corresponding samples and dosimeters.

When samples have to be transported outside the IRRAD facility to a CERN area or shipped to another location outside CERN, the users need to issue a transport request and transfer the relevant data to the CERN radiation protection service [RP], through TREC. Similarly to IDM, TREC requires the registration of certain characteristics of the samples such as the dimensions or the material type. Since these data are already registered in IDM, the advantage of integrating these data into the TREC system is obvious, so that the users do not have to perform the same registration twice, in both systems. Moreover, the facility operators need to collect the samples after irradiation into containers; those must be also registered in TREC. This requires creating associations between samples and containers and registering them directly in TREC through IDM. To implement these functionalities, IDM uses the Infor EAM web services for data exchange [Mar13].

History

For various key reasons such as minimising the amount of radioactive waste, which has an important environmental impact, reducing the cost of an irradiation experiment, reducing the use of the beam, saving beam time and improving efficiency, facility operators and users should be attentive not to repeat similar irradiation experiments and to keep the amount of material set in the beam to a minimum. To help in these matters, IDM implements functionalities that allow for knowledge sharing among IRRAD users by keeping the history of the previously performed experiments. Users can choose to make their experiment details visible to other IRRAD users. This way, the latter can learn from past experiments, in particular the procedures that need to be followed and the expected results.

UI Customisation

One important principle of good User Experience design is the idea that there is no "one size fits all" rule. Therefore, IDM needs to be easily customisable and appealing to the various users' tastes. Thus, IDM provides functionalities that allow users to customise their interfaces. For example, they can change certain display configurations such as background colour or font size. In the following chapter (Chapter 6), a new way to automatise this UI customisation by the use of recommender systems and ontology embeddings is discussed in detail.

5.2 Irradiation Experiment Data Management Ontology (IEDM)

IDM is a key tool for the data management of irradiation experiments in the IRRAD facility. However, it is highly customised to the IRRAD requirements and strongly linked to the CERN software infrastructure. On the other hand, there are hundreds of irradiation facilities that could profit from such a development. Although, these facilities use and produce knowledge of high scientific value, after some research, it was found that most of the teams in charge of the operation of IEs do not have a dedicated software such as IDM. They follow informal procedures for their overall data handling; they often store their data in ad-hoc spreadsheets in local computers or online, and even sometimes only on paper. This practice makes irradiation facilities more prone to the risk of data errors, corruption or even loss, hence the clear need for the introduction of a standardised approach in the data management of IEs.

Ontologies are known to facilitate knowledge sharing, reusability, and formalisation of a specific domain. Therefore, the general knowledge linked to the management of data associated with IEs, gained by the research and development of IDM, was encoded in a new ontology, called the Irradiation Experiment Data Management (IEDM) ontology. IEDM has been built by investigating and analysing the elements and practices commonly used in irradiation facilities around the world, building thus upon the knowledge of domain experts to provide a first step towards the structured and hopefully commonly agreeable-upon management of IE data.

Developing an ontology is a long and incremental process that calls for significant testing along the way. In order to provide the first evidence of the validity of our model and demonstrate how the ontology can be used, an actual IE is described in this thesis work using IEDM. FCC-Radmon [GPM+18], an experiment that has been performed in the IRRAD facility and which is linked to the current development of a future particle monitoring device for the proposed Future Circular Collider (FCC) [FCC] under consideration at CERN, was chosen.

The development of IEDM is part of a larger project undertaken at CERN and linked to the management of IE data [AID]. In addition to the previous motivation related to domain formalisation, IEDM is also used as a domain ontology from which a whole web application dedicated to IE data management can be automatically generated. Even further down the line, IEDM could be the foundational stone of an ontology for all irradiation facilities. These long-term goals have had a significant impact on the way IEDM is structured, thus requiring specific methodological stances, which are described in the sequel. In particular, the development of IEDM relies heavily on existing open-source standards such as OWL or physics-related ontologies (see Section 3.2).

5.2.1 Imported Ontologies

One of the best practices while developing an ontology is to reuse as much as possible the earlier ontology developments that we believe can partially describe irradiation experiments. The ones integrated in the IEDM ontology are described below.

- **EXPO** The Ontology of Scientific Experiments (EXPO) is a general ontology for the formalisation of scientific experiments [SK07]. It introduces concepts specifically linked to the notions of experimental design, scientific methods, and other core principles of experiments. Inheriting some of its structure from the Suggested Upper Merged Ontology (SUMO) [PNL02], EXPO similarly classifies all its concepts into an abstract level and a physical one, providing in this way an elegant and quite flexible design. Taking advantage of both the SUMO-inspired abstract/physical structure and of the fundamental science-related entities of EXPO, IEDM derives many of its key classes from these two upper-level ontologies.

- **OM** However, since EXPO does not elaborate on physical quantities and units, restricting its key constructs to the logical structure of scientific experiments, it is not enough to cover all of IEDM needs. In particular, IEDM has to be able to describe actual experimental parameters such as the total fluence (number of particles received by unit of area) impinging on a piece of material during an IE. We rely on OM, the Units of Measure ontology, for the representation of experimental quantities and of their physical units [RvAT13]. OM contains entities from many physics-related domains, including concepts from particle physics. This ontology is thus necessary for the formalisation of the physical quantities related to IEs and, as such, is the second most important ontology used for the development of IEDM.

- **FOAF** The third ontology used to develop some concepts of IEDM is the Friend-of-a-Friend ontology (FOAF) [BM14]. This widely used ontology

aims to describe networks of people, their activities, and their relations. For example, FOAF can be used to represent a social network on the web [Gol08]. IEDM uses FOAF mostly to describe the characteristics of the various individuals involved in IEs.

5.2.2 Design Methodology

As mentioned above, in this work, we aim to maximise the reusability of the upper-level ontologies that were described in Section 5.2.1. Therefore, we anchored the OWL-based definition of IEDM on the three foundational ontologies EXPO, OM and FOAF. These ontologies supplied the proper concepts, relations, and definitions for the representation of irradiation experiments, allowing for a more explicit taxonomy and axiomatisation (see Section 2.1).

In IEDM, EXPO was reused for describing abstract and physical concepts linked to the various features of an irradiation experiment, from the definition of its requirements to the representation of its experimental results. For example, the class `expo:AdminInfoExperiment` is used for representing the administrative information of an irradiation experiment. Yet, to ensure compatibility with the potential future enhancements of EXPO and other ontologies, one important IEDM design decision was to never copy, override or modify them. Instead, when more specific information was needed, we introduced IEDM-specific variants of existing classes, using OWL namespaces to avoid ambiguities. Thus, as an example, the class `iedm:AdminInfoIrradiationExperiment` was created as a subclass of `expo:AdminInfoExperiment`, instead of directly updating the later.

The OM ontology was chosen to represent the experimental quantities specifically used when handling irradiation experiments. Entities such as `om:Energy`, `om:AbsorbedDose` (a dose is the amount of radiation energy delivered to matter per unit of mass, see Definition 2.3) or `om:Activity` (an activity is a number of atom decays per unit of time, see Definition 2.1) are fundamental concepts that appear in an irradiation experiment, and they exist in OM.

Another essential notion for the IEDM ontology is the concept of a user. The term "user" has a broad definition and can be a group, an organisation, or a person. Therefore, we employed the definition of `foaf:Agent`, which contains, as subclasses, these three types of entities.

Since the domain of irradiation experiments is larger than those covered by the three foundational ontologies used to describe some of the IEDM concepts, the second phase of the IEDM development required introducing concepts more specific to irradiation experiments per se. In this step, a top-down approach was followed: IE-specific concepts were added as subclasses of the upper ontologies' classes. For instance, `iedm:Element`, denoting the notion of atoms, is a subclass of `expo:Object`, and the notions of `iedm:RelativisticMomentum` or `iedm:Fluence` were added.

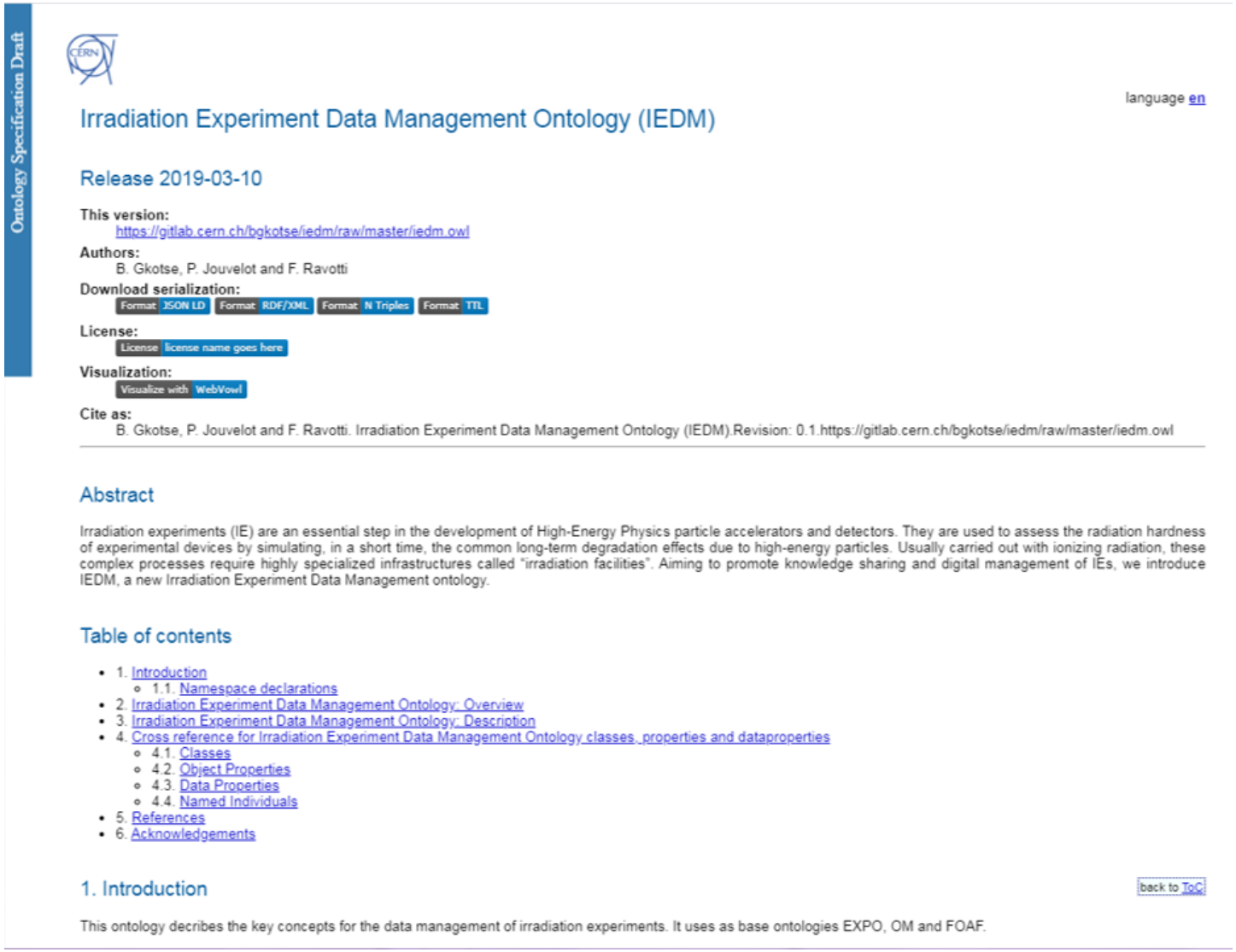


Figure 5.4: IEDM online documentation.

Moreover, new relations were introduced, i.e., superclasses were created via IE-specific OWL object properties[2]. For example, the constraint `iedm:hasResult some iedm:CumulatedQuantity` is a superclass of the class `iedm:Irradiation-Experiment`, which is, in fine, the only top-level new class introduced by IEDM.

During the integration of IEDM with the upper ontologies, several issues were detected. One of them was that a concept could appear in more than one of the three upper ontologies (e.g., `expo:Quantity` and `om:Quantity`). In that case, the design choice was to use the concept whose definition better fit the model. There were also cases where two concepts were in fact complementary to each other (e.g., `expo:Agent` and `foaf:Agent`), and hence another IEDM class (`iedm:User`) was created that would be the subclass of both, taking advantage of OWL multiple inheritance for classes.

Finally, given the importance of documentation in ontology development and usage, dedicated annotations are used as internal elements providing information for each new IEDM entity. As a consequence, as shown in Figure 5.4, by the use of the WIDOCO wizard for documenting ontologies [Gar17], external documentation is provided online[3].

5.2.3 Core Structure

This section is dedicated to highlighting, analysing, and explaining the core entities and relations that characterise IEDM. In the following paragraphs, the core classes of IEDM are presented.

Irradiation Experiments

Most of the entities that are described revolve around the Irradiation Experiment class `iedm:IrradiationExperiment`. However, for easier comprehension, the description starts from the Irradiation Experiment Object class (`iedm:IrradiationExperimentObject`) and radiation field (`iedm:RadiationField`), which represent the key elements of an IE.

- **Radiation Field** A radiation field (`iedm:RadiationField`) is necessary in order to perform an irradiation experiment. An `iedm:RadiationField` can be composed of particles (`iedm:Particle`) of the same type (`iedm:SingularField`) or more (`iedm:MixedField`).
- **Irradiation Experiment Object** The class `iedm:IrradiationExperimentObject` is a subclass of `expo:Object`; it represents an object brought

[2]Note that OWL sees any logical constraint as defining a full-fledged class in itself, an unusual design feature even to software specialists used to Object-Oriented principles.

[3]http://cern.ch/iedm

by a user to the irradiation facility infrastructure that is exposed to an `iedm:RadiationField` or an object that is being irradiated. A Device Under Test (DUT) during an irradiation experiment is an instance of `iedm:DUT`. The purpose of performing an irradiation experiment is that the DUTs reach some target cumulated quantity (`iedm:CumulatedQuantity`), which is a subclass of `iedm:DosimetricQuantity`. Two of the most common cumulated quantities in irradiation experiments are the absorbed dose (`iedm:AbsorbedDose`) or fluence (`iedm:Fluence`), which are both included in the model (see definitions in Section 2.2.2).

- **Irradiation Experiment**: In Experimental Particle Physics, an Irradiation Experiment (`iedm:IrradiationExperiment`) denotes a whole experiment where DUTs are exposed under a specific `iedm:RadiationField`. An instance of `iedm:IrradiationExperiment` has some explicit detailed specifications, which describe experimental methods and include a plan of actions that is defined by EXPO as `expo:ProcedureExecuteExperiment`. In IEDM, this class is extended with three subclasses:
  - the first one is the `iedm:PassiveStandardIrradiation`, where an `iedm:DUT` is simply put into a `iedm:RadiationField`;
  - the second class is `iedm:PassiveCustomIrradiation`, for which very specific `iedm:TechnicalRequirements` have to be specified and implemented by the irradiation experiment operator;
  - finally, `iedm:ActiveIrradiation` represents the third category and includes experiments requiring an active data acquisition (DAQ) device and, typically, the usage of a readout system during the experiment execution.

  Each `iedm:IrradiationExperiment` instance belongs to one category of `expo:ProcedureExecuteExperiment`, which is defined by the OWL expression `iedm:hasIrradiationCategory exactly 1 expo:ProcedureExecuteExperiment`.

- **DUT Irradiation**: The DUT Irradiation class (`iedm:DUTIrradiation`) refers to a specific irradiation experiment related to a unique DUT. In contrast to `iedm:IrradiationExperiment`, this class depends on one and only one `iedm:DUT`, and it can be considered as part of an `iedm:IrradiationExperiment`. For this reason, the relation `iedm:IrradiationExperiment iedm:hasPart some iedm:DUTIrradiation` is defined. The `iedm:DUTIrradiation` has two specific points in time, from `iedm:TimePosition`, that denote the start and the stop times of the radiation exposure.

DUT Material Composition

As already previously highlighted (see Sections 2.2.2, 5.1.5), the interaction length values are important quantities for performing irradiation experiments. In order to compute these values, the detailed description of the material composition of a DUT should be provided.

- **Interaction Length and Occupancy** The parameters that define the degree of interaction of a particle with matter (interaction lengths) are important quantities for an irradiation experiment (see also Section 2.2.2). In an actual irradiation experiment as, for example, the one performed at CERN IRRAD (see Section 5.2.4), these values need to be kept as low as possible since a high degree of interaction (caused by too many DUTs placed in the `iedm:RadiationField` at the same time) can produce an excess of secondary particles that, in turn, can interfere with the result of an `iedm:IrradiationExperiment`. Occupancy is defined as the sum of the ratios of the actual length for each device layer (`iedm:Layer`) of the DUTs against the interaction length of each layer. These quantities, which are dependent upon the DUT materials and radiation fields and are computed by a dedicated physics-simulation program, are defined as `iedm:InteractionLength` and `iedm:InteractionLengthOccupancy` [LR11].
- **Element, Compound, and Device Layer** IEDM includes also the concepts of elements (`iedm:Element`), corresponding to the basic entries of the Mendeleev periodic table, Compounds (`iedm:Compound`), which are mixtures of elements, and Layers (`iedm:Layer`), which are used to describe the 1D structure of a typical DUT along the `iedm:RadiationField` main propagation axis. These sets of information are necessary for properly computing quantities related to the `iedm:InteractionLength` and `iedm:InteractionLengthOccupancy` instances.

Users

An `iedm:User` is defined as a subclass of both `expo:SentientAgent` and `foaf:Agent`. According to the definition of EXPO [SK07], the `expo:SentientAgent` describes a user that has rights but may or may not have responsibilities and the ability to reason. If the latter is present, it can also have cognitive abilities. A `foaf:Agent` can be a person, a group, or an organisation. By specifying that the `iedm:User` class has both of them as superclasses, both concepts can be combined.

In an irradiation facility, a user can have one or multiple roles during an irradiation experiment. These IEDM-specific roles are represented by subclasses

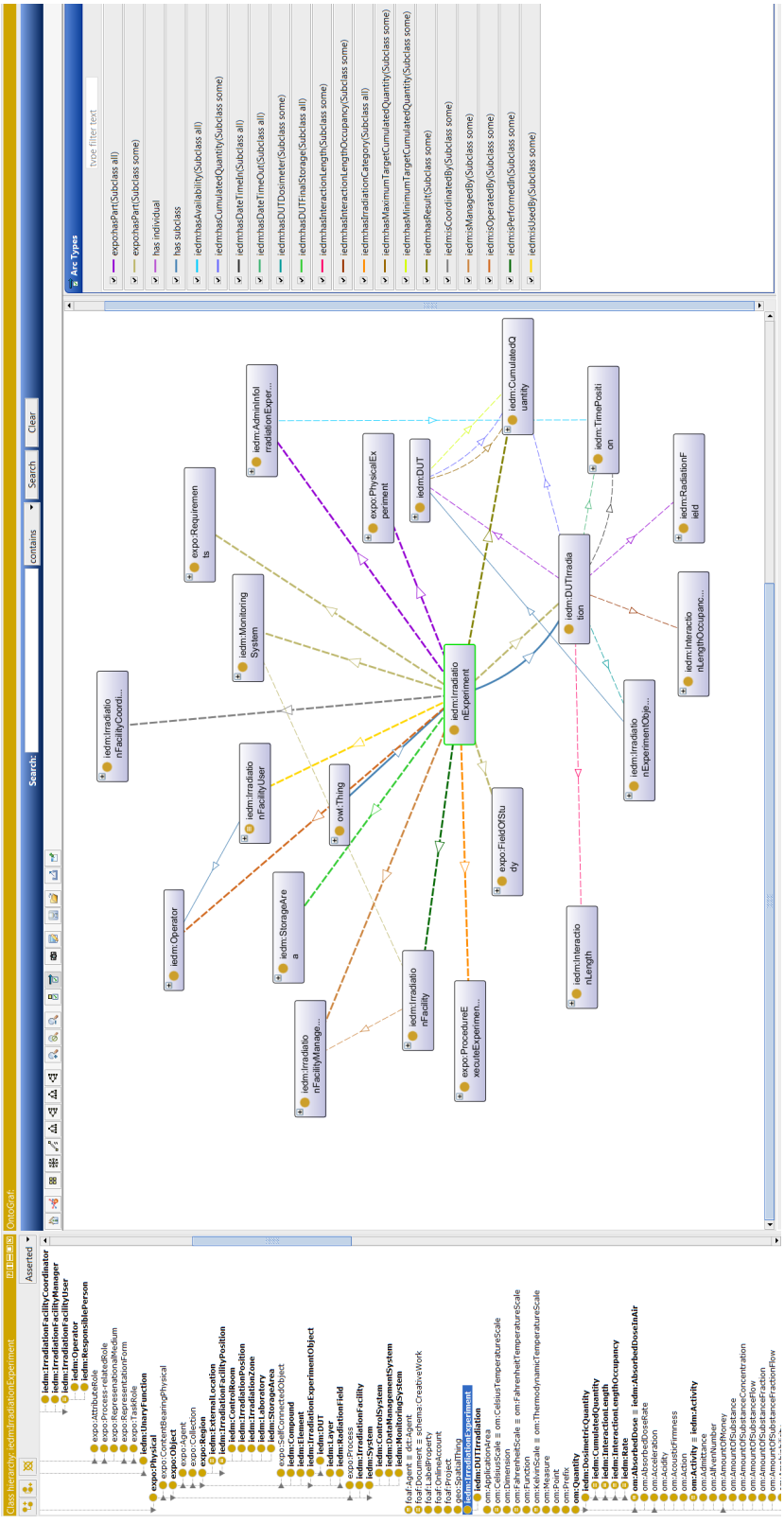


Figure 5.5: Graph representation in Protégé of the IEDM ontology, with the focus on the `iedm:Irradiation Experiment` class. The left column illustrates the classes of IEDM while the right column, the object properties.

of `expo:User`, which is an `expo:SubjectRole`, viewed in EXPO as a predicate, following SUMO's practice. IEDM defines the roles of irradiation facility coordinator (`iedm:IrradiationFacilityCoordinator`), irradiation facility manager (`iedm:IrradiationFacilityManager`) and irradiation facility user (`iedm:IrradiationFacilityUser`). The class `iedm:IrradiationFacilityUser` has two subclasses: the operator (`iedm:Operator`), who is a person that performs the irradiation experiment, and the responsible person (`iedm:ResponsiblePerson`), i.e., the person in charge of that specific irradiation experiment.

Discussion

In the above subsections, the main entities linked to an `iedm:IrradiationExperiment` are presented. However, many more concepts exist in the IEDM ontology, in particular regarding the OWL object properties that relate these classes together. In total, IEDM consists of 115 classes and 941 annotations. Relations among the classes are represented with 24 object properties and 16 data properties. Online, publicly available documentation[3] is provided for further validation and review by domain experts. The IEDM ontology and related resources can be found in its Gitlab repository[4]. All IEDM classes and relations are explained with proper annotations, definitions, and comments for easier understanding.

5.2.4 FCC-Radmon, a Use Case of Irradiation Experiment

To assess the validity, coherence, and completeness of the IEDM ontology, we applied it to the representation of an actual irradiation-experiment use case. In IRRAD, one of the IEs performed during the 2018 irradiation run was FCC-Radmon [GPM+18]. With the development of future accelerators such as the 100-Tev 80-kilometer-long FCC [FCC] being proposed by CERN, and its related experiments, targeting always higher performance comes the need to monitor more intense radiation levels. This calls for the development of new generations of radiation monitors that can cope with these increased radiation levels. Thus, in the FCC-Radmon experiment, a new technology of particle-fluence monitor was tested against radiation-damage levels that could be found in a future accelerator such as the FCC [GPM+18].

Focusing mainly on the instances of the core classes (see Section 5.2.3), the FCC-Radmon IE is described via the concepts provided by IEDM. The name for this specific IE instance is `iedm:FCC-Radmon`. For this experiment, the irradiation of a specific `iedm:DUT`, which in this case is the `iedm:PCB5-run2017` (a DUT already irradiated in the run of 2017), we have created the `iedm:FCC-Radmon-Irradiation` instance. For measuring the actual `iedm:Fluence`, using a dedicated dosimeter, during the `iedm:FCC-Radmon` experiment in the irradiation facil-

[4] https://gitlab.cern.ch/irrad/iedm

ity `iedm:CERN_IRRAD`, the `iedm:Operator1` has installed an `iedm:Irradiation-ExperimentObject` that, in this case, is the instance `iedm:Dosimeter004139`. The `iedm:FCC-RadmonIrradiation` instance has a specific start time (an `iedm:-TimePosition`, which in this example is `iedm:_2018_03_30_12h_00`), time at which the DUT is exposed to the proton radiation field (`iedm:Protons_24GeV`), and a specific end time, at which the radiation exposure is completed `iedm:_201-8_11_12_18h_00`. The `iedm:CumulatedQuantity` of interest, an `iedm:Fluence` in this experiment, is represented by the instance `iedm:_3e17_protons_per_square_cm` and may suffer, from `expo:MeasurementError`, a variation of `iedm:_7-_per_cent`.

For this specific experiment example, 43 instances of IEDM classes were created. Moreover, as a more general check of ontology consistency, IEDM was also verified with the Pellet reasoner [SPG+07]. The complete use case can be found in the online resources

5.3 Comparison of GenAppi-Generated IEDM-Based Data Management System and IDM

For the purpose of testing the GenAppi methodology, IEDM has been used as a domain ontology input. We compare the resulting IE-dedicated data management system to IDM.

5.3.1 IEDM-dedicated Data Management System

As explained in the previous chapter, its classes and object and data properties are thus transformed into the Django model of an IEDM-specific web application. This model is mapped by default to a SQLite3 database. In addition, and to enable the use of more advanced web semantic technologies, Owlready2 has been integrated in the generated web application, allowing then for communication between the web application and the ontology. In particular, this means that it is also possible, in addition to what is done with the relational database, to save and update data directly in the ontology, or in a triple store if one is interested in making the storage and querying of large amount of data more efficient. Views, templates and URLs of the MVT architecture are generated; they implement the operations for each class of the ontology. In Figures 5.6 and 5.7, some screenshots of the generated user interfaces are presented.

As described in the previous paragraphs, the key UI operations are defined in OWAO, and the relevant functions for the model classes are generated by GenAppi. Figure 5.6 shows the "Sign Up" page, which actually uses the authentication system provided by Django. In Figure 5.7, one specific example of

[4]https://gitlab.cern.ch/irrad/iedm

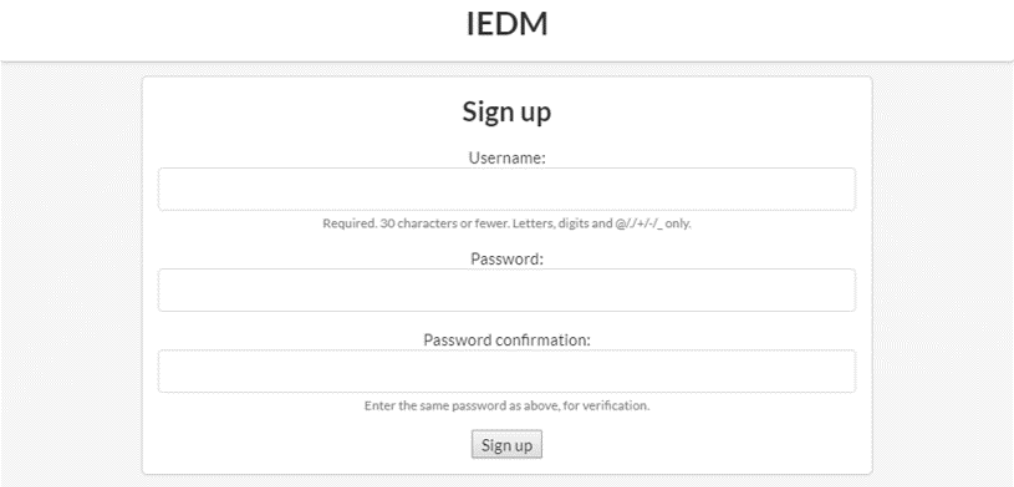


Figure 5.6: Login page of the IEDM-dedicated application.

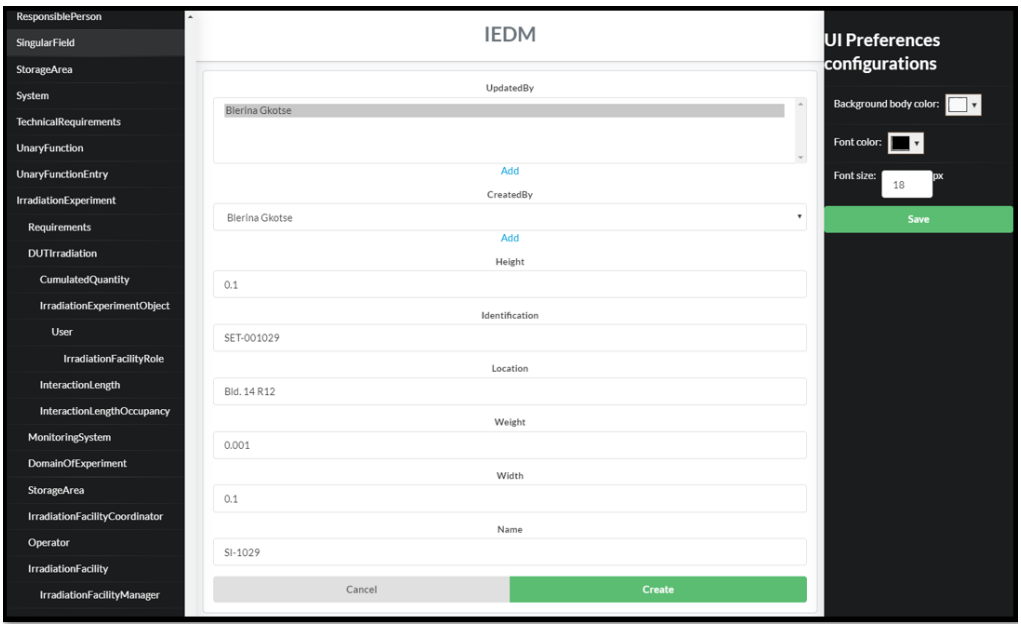


Figure 5.7: Registering a new sample in the IEDM-dedicated application.

a `Create`-sample operation form is demonstrated, where the OWL superclasses defined by triples corresponding to object or data properties are transformed into the input fields of a form adequate for creating the corresponding class instances. If this data property were instead an object property, the GenAppi-generated application would add a foreign key to the instance. In addition, if there are no instances of the class referred to yet, the user is prompted by the application to first add an instance of the class to which the foreign key is associated.

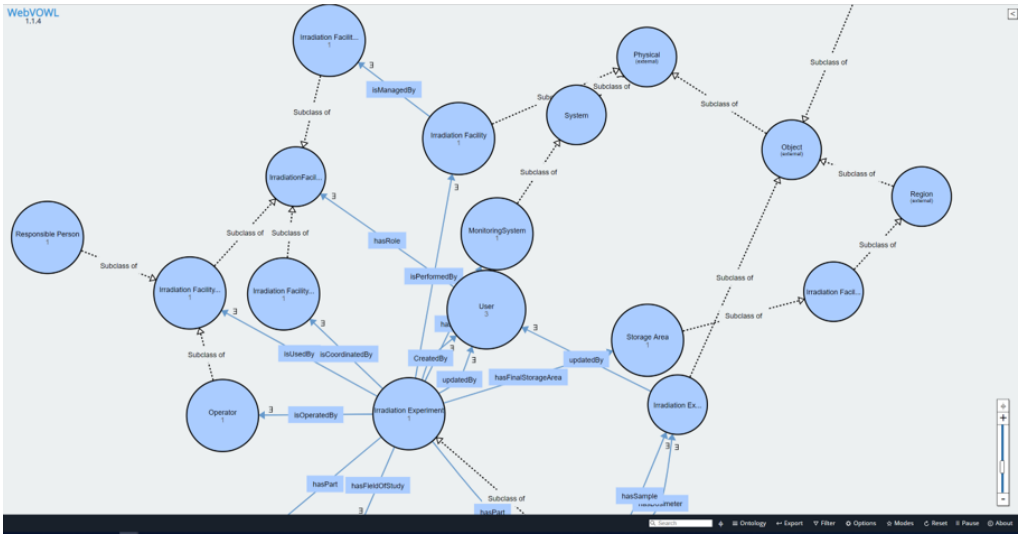


Figure 5.8: WebVOWL visualisation page of IEDM from the generated application.

As an additional useful tool, the generated web application also includes a WebVOWL visualisation page of the domain ontology, as presented in Figure 5.8.

5.3.2 Comparison with IDM

In Section 5.1, the IRRAD Data Manager (IDM) used for handling the data of the IRRAD facility was presented. IDM follows loosely the concepts present in the IEDM ontology, but it has been manually developed and fulfils the requirements specific to IRRAD experiments. IDM is used by the IRRAD coordinator, operators and users for the registration of data. It provides specific functionalities dedicated to the planning of the irradiation experiments and their follow-up. The results of the experiments are also available via this tool, which communicates with external systems. Finally, it keeps a history of the performed experiments and of the components that were irradiated [GJPR19].

When comparing the GenAppi-derived system and IDM, it is clear that IDM was developed for the same purpose as the IEDM-specific web application pre-

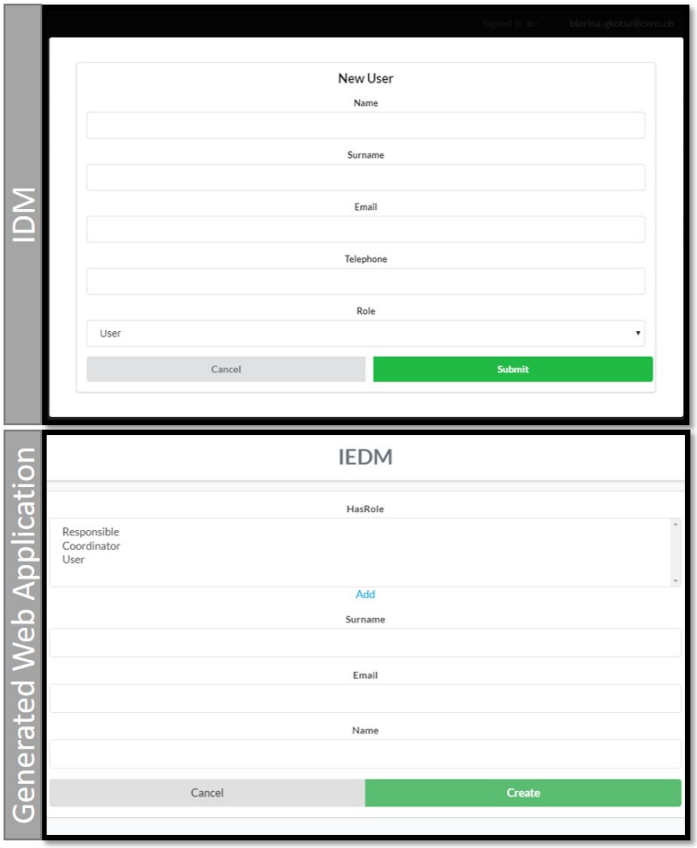


Figure 5.9: "Create user instance" form of IDM, at the top; generated web application version, at the bottom.

sented above, namely the management of irradiation experiments data. However, IDM may not be suitable for other facilities, because it strictly follows the operational requirements as specified by the IRRAD management. But, more importantly, the technologies IDM is based on are strongly linked to the CERN software infrastructure. First, the Django model of IDM is based on the Oracle relational database, which may not be available in other facilities. In GenAppi, the user can specify the database that the web application should use. Moreover, IDM is not built on top of semantic technologies as GenAppi is; these allow for easier knowledge sharing and interoperability. These technologies open the way to complex querying, reasoning and inference. This type of advanced analysis is further enabled by the possibility of saving instances directly in the ontology or in a triple store.

As far as UX is concerned, as shown in Figures 5.9 and 5.10, there are strong similarities between the UIs of IDM and the IEDM-specific GenAppi-generated web application.

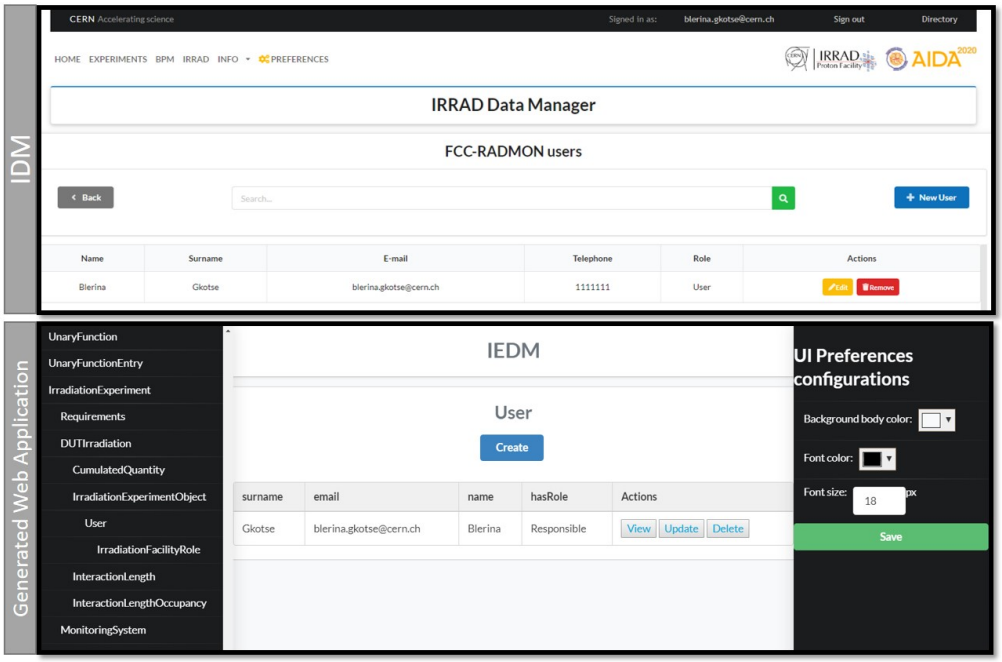


Figure 5.10: List user interface of IDM at the top, generated web application at the bottom.

	IDM	Generated Web Application
Purpose	IRRAD data management	Any domain
Software Infrastructure	CERN	Free and open-source
Storage	Oracle Database	Any relational database, ontology or triple store
Web Semantic Technologies	No	Yes, allowing for data integration
Functionalities	More advanced (e.g., interaction length calculation)	CRUD operations

Table 5.1: Comparison of GenAppi-Generated IEDM Application against IDM.

Regarding the additional services provided by both systems, note that the IDM authentication service is the CERN Single Sign On (SSO) system, while the generated application relies upon the default Django authenticating system. Table 5.1 summarises all above mentioned points of comparison of the two systems.

However, IDM implements additional and complex functionalities that need to be defined by the developer. For instance, it integrates formulas to compute physics-related values that cannot be easily defined in an ontology. This would require further work.

As a final comment, this IEDM-based use case suggests that GenAppi-generated web applications can be considered as valuable data management backbones from which further implementation and functionality customisation according to the needs of the domain at hand can be pursued, as is done with other commercial software tools (e.g., SIMATIC WinCC[5] [WCC]). The ease with which one can obtain such a powerful platform from the mere specification of a domain ontology should be a strong point for this approach.

5.4 Validation with Other Ontologies

Figure 5.11: The List view of a class *BT33_Marine_Animal_Type* of the generated web application using MarineTLO as an input ontology.

In order to understand if the GenAppi methodology can be used for the generation of web applications for other domain ontologies, three other ontologies were chosen for the tests:

- **SOSA**, the ontology that describes concepts of sensors, observations, samples and actuators [JHC+19];

[5]Siemens SIMATIC WinCC is a commercial tool providing customisable process-visualisation systems with functions for monitoring automated processes

- **EXPO**, the ontology that defines concepts of scientific experiments [SK07];
- **MarineTLO**, the ontology that includes concepts for the marine and terrestrial domains [TAB+13].

The tests with GenAppi suggest that the methodology is not bound to the IEDM model and UIs can be generated also for different ontologies. Image 5.11 illustrates an example of a list view of the class *BT33_Marine_Animal_Type* from the generated web application using the ontology MarineTLO.

5.5 Summary

This chapter presents the design, development and functionalities of the IRRAD Data Manager (IDM), the initial point for the introduction and creation of IEDM, a domain ontology built for the purpose of describing data management entities and relations in the field of irradiation experiments. IEDM is used as an input ontology for the GenAppi methodology presented in Chapter 4. The generated web application is presented and compared against the custom-made IDM web application.

It is clear that IDM has more advanced features than the IEDM-derived application. However, IDM is strongly linked to the IRRAD infrastructure, whereas the generated application can be adapted and customised according to the data management requirements of other interested facilities. This generic framework also introduces a set of best practices and a common reference on the way data are treated in the irradiation-experiments community.

Chapter 6

UI Personalisation with ontowalk2vec

Version française

Ce chapitre décrit les expériences, le processus d'évaluation et les résultats obtenus avec le modèle ontowalk2vec de génération de plongements ("embeddings") d' ontologie qui a été détaillé dans la section 4.3.3. Dans la première section de ce chapitre, les techniques de classification et les paramètres d'évaluation de ces expériences sont présentés. Dans la deuxième section, l'algorithme d'ontowalk2vec et les classifications associées sont précisés en pseudocode. Dans les sections suivantes, ontowalk2vec est comparé aux meilleurs modèles actuels : RDF2Vec [RRN+19] et node2vec [GL16]. À cette fin, MUTAG [DLL], une ontologie de référence utilisée également dans RDF2Vec et contenant des données de molécules complexes, est pris comme un ensemble de données de référence. Ensuite, ontowalk2vec est testé avec l'ontologie OWAO, car notre but ultime est de fournir des plongements cohérents pour OWAO, de façon à ce qu'ils puissent être utilisés pour la recommandation de préférences d'interface utilisateur. À cet effet, une deuxième expérience est mise en place en instanciant l'ontologie OWAO sur la base des données collectées à partir des résultats de la recherche sur les préférences d'interface utilisateur présentées dans la section 3.4.1. De plus, étant donné que les modèles intégrés dans ontowalk2vec tels que word2vec contiennent des hyperparamètres spécifiques qui peuvent être réglés pour fournir de meilleurs résultats pour des données spécifiques, une section dédiée présente certains résultats sur l'optimisation des hyperparamètres en utilisant des plongements générés par OWAO.

English version

This chapter describes the experiments, evaluation process and results obtained by the ontowalk2vec model for generating ontology embeddings that was detailed in Section 4.3.3. In the first section, the classification techniques and evaluation

metrics for these experiments are presented. In the second section, the algorithm of ontowalk2vec and the classifications are explained via pseudocode. In the following sections, ontowalk2vec is compared to the current state-of-the-art models: RDF2Vec [RRN+19] and node2vec [GL16]. For this purpose, MUTAG [DLL], a benchmark ontology used also in RDF2Vec and containing data from complex molecules, is taken as a benchmark dataset. Then ontowalk2vec is tested with the OWAO ontology, since our ultimate goal is to provide consistent embeddings for OWAO that can be used for the recommendation of UI preferences. For this purpose, a second experiment is set up by instantiating the OWAO ontology based on the data collected from the research findings about UI preferences presented in Section 3.4.1. Moreover, since the models integrated in ontowalk2vec such as word2vec contain specific hyperparameters that can be tuned to provide better results for specific data, a dedicated section presents some results about hyperparameters' optimisation by the use of OWAO generated embeddings.

6.1 Classification Techniques and Evaluation Metrics

In order to evaluate the embeddings generated via ontowalk2vec, it is necessary to define certain experiments, such as classification problems, and assess whether the results predicted by the model after being trained are accurate. In the framework of this thesis work, this will allow for a clear understanding of whether ontowalk2vec can perform better than the current state-of-the-art models and provide appropriate embeddings for the UI preferences stored in the OWAO ontology, as described in Section 4.3. Common evaluation metrics used for classifiers are the accuracy score, confusion matrix and t-distributed Stochastic Neighbor Embedding (t-SNE) plot, explained in the following paragraphs.

Classifiers

Two classical methodologies, namely Support Vector Machine (SVM) and Random Forest (RF), provided by Scikit-learn python module [PVG+11], were used for classification purposes. Provided some labelled training data are available, SVM outputs the optimal hyperplane[1] that best classify its training set and, hopefully, new samples [WLW04]. A random forest uses a number of decision trees on smaller groups of data contained in its input dataset and averages the results in order to improve accuracy [Bre01].

[1] A hyperplane, a term used in geometry, is any codimension-1 vector subspace of a vector space [Wei]. Embeddings are multidimensional vectors, and therefore a hyperplane can be used to try to separate these embeddings for classification purposes.

Accuracy

One of the key metrics used for the evaluation of classification models is the classification accuracy score, also included in Scikit-learn module [PVG+11]. Given some labelled test data, the accuracy measures the amount of correctly predicted labels over the overall number of labels. If $\hat{y}_i$ is the predicted value of the i-th sample and y_i is the corresponding actual value, then accuracy$(y, \hat{y})$ is the fraction of correct predictions over $n_{samples}$, defined as:

$$\text{accuracy}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} 1(\hat{y}_i = y_i), \tag{6.1}$$

where $1(\hat{y}_i = y_i)$ is equal to 1 if its argument is true, and 0 otherwise.

According to Equation 6.1, a high accuracy, i.e., a large number of correct predictions, is such that the value of the accuracy score should approach 1. On the contrary, low accuracy means that the value approaches 0.

Confusion Matrix

Models	Actual Negative	Actual Positive
Predicted Negative	$C_{0,1}$ (True Negative)	$C_{0,1}$ (False Negative)
Predicted Positive	$C_{1,0}$ (False Positive)	$C_{1,1}$ (True Positive)

Table 6.1: Confusion matrix for binary classification.

Another metric to evaluate classification performance is the confusion matrix. By definition, the element $C_{i,j}$ of a confusion matrix C is the number of observations known to be in group i and predicted to be in group j. In the case of binary classification, a confusion matrix includes the count of true negatives $C_{0,0}$, false negatives $C_{0,1}$, false positives $C_{1,0}$ and true positives $C_{1,1}$.

According to the definition, the accuracy of a model is considered here high when the numbers $C_{0,0}$ and $C_{1,1}$ of true negatives and positives are high (approaching the number of actual negative and positive data points, respectively, and depicted in Table 6.1 in green), while the numbers $C_{1,0}$ and $C_{0,1}$ (depicted in Table 6.1 in red) of false positives and negatives should be as low as possible, approaching a value of 0.

t-distributed Stochastic Neighbor Embedding (t-SNE)

In order to provide a visual evaluation for data classification, the notion of t-distributed Stochastic Neighbor Embedding (t-SNE) is often used [MH08]. t-

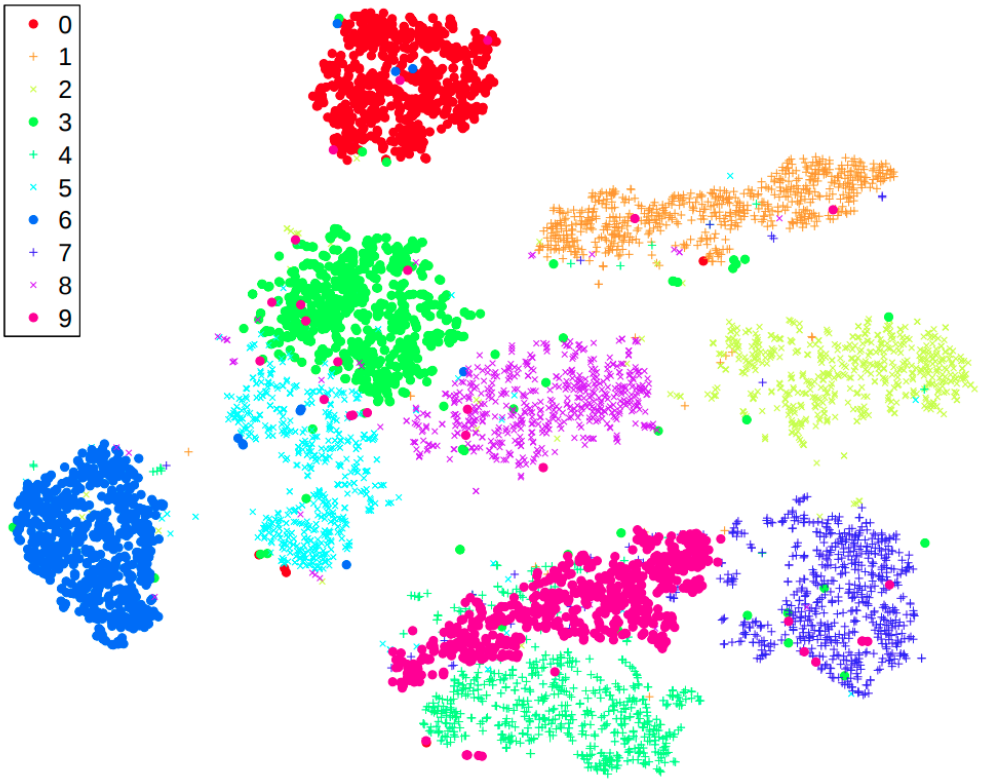


Figure 6.1: MNIST classification with t-SNE (from L. van der Maaten and G. Hinton, "Visualizing data using t-SNE" [MH08]).

SNE is a method used for the visualisation of high-dimensional data. It converts similarities between data points to joint probabilities and minimises the Kullback-Leibler divergence[2] between the joint probabilities of the low-dimensional embedding and the high-dimensional data [Kul87]. Figure 6.1 shows an example of t-SNE visualisation for the reference dataset MNIST[3], which contains handwritten digits that need to be classified as the number they correspond to. A classifier is considered as providing accurate results when the points corresponding to the same class (in this case, same number) are grouped together.

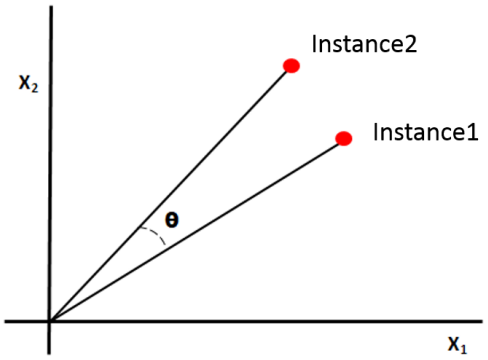


Figure 6.2: Cosine similarity.

Cosine Similarity

Having a high-accuracy classification model is important to ensure that such a model is robust; in our case, this would ensure that we managed to encode ontology instances with representative yet automatically generated embeddings. By comparing these vector embeddings, one can get an approximation of the proximity and relatedness of the corresponding ontology instances. To define such a comparison function, the similarity value of two vector embeddings is often specified via the use of the cosine similarity metric. Assuming that an ontology instance 1 is represented by the embeddings as vector A and an instance 2 is represented by the embeddings as vector B, the cosine similarity value $cos(\theta)$ of the two instances is defined as

$$cos(\theta) = \frac{A.B}{\|A\|\|B\|}, \tag{6.2}$$

where $A.B$ denotes the scalar product of the two vectors A and B, and $\|A\|$, $\|B\|$ the norms of the two vector.

According to Equation 6.2, the closer the vectors are, the more the similarity value approaches 1, the more similar the instances are. In the case where the cosine similarity value approaches 0, the two instances are considered less similar.

[2]Kullback-Leibler divergence, also known as relative entropy, denotes the "distance" of probability distributions that are being compared

[3]The MNIST dataset is publicly available from http://yann.lecun.com/exdb/mnist/index.html

6.2 Algorithm for ontowalk2vec and Classification

The pseudocode for the ontowalk2vec algorithm and its related classification methods is given as Algorithm 2, below. This algorithm is based on the methodology described in Section 4.3.3.

Once started, the algorithm reads the file of the ontology for which the embeddings will be generated and the training and test data sets, to be used for the classifications. The training and test data files contain the data points (ontology instances) that will be classified and the actual classification labels. Then, these data points and labels are split into different lists (*training_members* and *test_members* for the data points, and *training_labels* and *test_labels* for the labels), to be used for the training process and, later, for testing whether the predictions of the classifiers are accurate.

In Lines 4 and 5, the ontology is translated as a directed node2vec graph object, using the boolean parameter *directed*, which specifies that the graph is directed, and the hyperparameters p and q. The latter are used for the preprocessing of the transition probabilities that guide the computation of the random walks, as indicated in Line 6. Based on these probabilities, random walks on the graph are gathered in *node2vec_walks*. Subsequently, random walks using the Breadth First Search (BFS) method are generated by RDF2Vec (bfs_walks). Both *node2vec_walks* and bfs_walks are used as input sentences of word2vec, which produces one set of final embeddings ($bfs\_embeddings$). The same process is performed for the Weisfeiler-Lehman (WL) method, used as a method alternative to BFS in RDF2Vec for generating random walks; the generated embeddings are stored in *wl_embeddings*. As shown in Line 12, for both types of embeddings, the two Random Forest and SVM classifiers, detailed in Section 6.1, are used for training the generated embeddings using *training_members* with its labels *training_labels*. After the training, the classifiers are expected to predict accurately the labels of the *test_members*. The accuracy score and confusion matrix are computed, while a t-SNE plot is also generated.

Given a pair of instances ($instance_1$, $instance_2$), the similarity value can be computed in order to evaluate how similar these two instances are. For providing embedding-based recommendations, given an ontology instance $instance_1$, the instances exhibiting the highest similarity value with $instance_1$ are selected. This will allow our recommender system to suggest items similar to the one used as input, in this case, $instance_1$.

As already mentioned, the pseudocode for the ontopath2vec and classification algorithm is a simplified version of the actual code; the presentation above intends to provide a mostly high-level understanding of our new algorithm. The full code as well as the data used for the experiments can be found in the online

Algorithm 2 ontowalk2vec and classification
Input: $ontology$, $training\_data$, $test\_data$, $instance_1$, $instance_2$
Output: ontology embeddings and classification analysis data

Read $ontology$, $training\_data$, $test\_data$
Separate $training\_data$ to $training\_members$ and $training\_labels$
Separate $test\_data$ to $test\_members$ and $test\_labels$
$graph \leftarrow ontology\_to\_graph\_conversion(ontology)$
$node2vec\_graph \leftarrow node2vec.graph(graph, directed, p, q)$
$node2vec\_graph.preprocess\_transition\_probabilities()$
$node2vec\_walks \leftarrow node2vec\_graph.simulate\_walks()$
$bfs\_walks \leftarrow RDF2VecBFS(graph, training\_members \cup test\_members)$
$bfs\_embeddings \leftarrow word2vec(bfs\_walks \cup node2vec\_walks)$
$wl\_walks \leftarrow RDF2VecWL(graph, training\_members \cup test\_members)$
$wl\_embeddings \leftarrow word2vec(wl\_walks \cup node2vec\_walks)$
for $embeddings$ in $[bfs\_embeddings, wl\_embeddings]$ **do**
    Separate $embeddings$ to $training\_embeddings$ and $test\_embeddings$
    $rf \leftarrow RandomForestClassifier()$
    $svm \leftarrow SVM()$
    for $classifier$ in $[rf, svm]$ **do**
        $classifier.fit(training\_embeddings, training\_labels)$
        $classifier.predictions \leftarrow classifier.predict(test\_embeddings)$
        $classifier.accuracy \leftarrow$
            $classifier.accuracy\_score(test\_labelspredictions)$
        $classifier.confusion\_matrix \leftarrow$
            $confusion\_matrix(test\_labels, classifier.predictions)$
    end for
    $tsne \leftarrow TSNE().fit\_transform(embeddings)$
    Plot $tsne$
    $top\_similarities \leftarrow embeddings.most\_similar(instance_1)$
    $similarity\_value \leftarrow embeddings.similarity(instance_1, instance_2)$
end for

resources[4].

6.3 Experiment with the MUTAG ontology

The input ontology of the first experiment is a reference ontology called MUTAG. The MUTAG ontology is part of DL-Learner, a framework for supervised machine learning in OWL, RDF and Description Logics [DLL]. MUTAG contains information about 340 complex molecules that could be carcinogenic or

[4]https://gitlab.cern.ch/irrad/ontowalk2vec

as it is called in the ontology, "MUTAGenic".In order to compare the accuracy of ontowalk2vec to the other state-of-the-art models (node2vec and RDF2Vec), the setup of a classification experiment based on the MUTAG data is necessary. After the training, the classifiers should be able to predict the labels of the test data as accurately as possible. This prediction quality is measured by the accuracy score and confusion matrix, while the classification of the two categories is visualised through t-SNE plots, as detailed in Section 6.1.

6.3.1 Experimental Setup

The classification of the complex molecules is provided by a boolean data property in the ontology, which is called `mutag:isMutagenic`. If the molecule is mutagenic, the data property is true; otherwise, false. Figure 6.3 demonstrates an instance of a molecule (d30) and its structured composition. Classification is a learning task, therefore training and learning of specific labels provided from training data is required. In this case, the labels are 0 or 1 according to the value of the data property `mutag:isMutagenic`. The `mutag:isMutagenic` value is removed from the dataset so that the classification is only performed through the embeddings of the molecular instances; these are generated based on the ontology structure and the labels provided during the training.

In order to perform this training, a dedicated dataset was prepared, containing the 340 molecules with their labels. The dataset was then divided in 80% training data and 20% test data. Following the steps of Algorithm 2 in Section 6.2, the models are generating the embeddings for all data (both training and test data), and then the classifiers are trained by the generated embeddings of the training data (80% of the total data).

6.3.2 Results

Experiments were performed with the node2vec, RDF2Vec Breadth First Search (BFS), RDF2Vec Weisfeiler-Lehman (WL), ontowalk2vec BFS and ontowalk2vec WL. In order to validate the results and according to standard techniques in machine learning, the same experiments were run 10 times [HS19]. The accuracy scores provided by the two classification methods Random Forest (RF) and Support Vector Machine (SVM) are reported here via the average value and standard deviation of these 10 iterations.

Accuracy

The accuracy results are summarised graphically in Figure 6.4, and the exact values are also provided in Table 6.2. According to the presented results, both

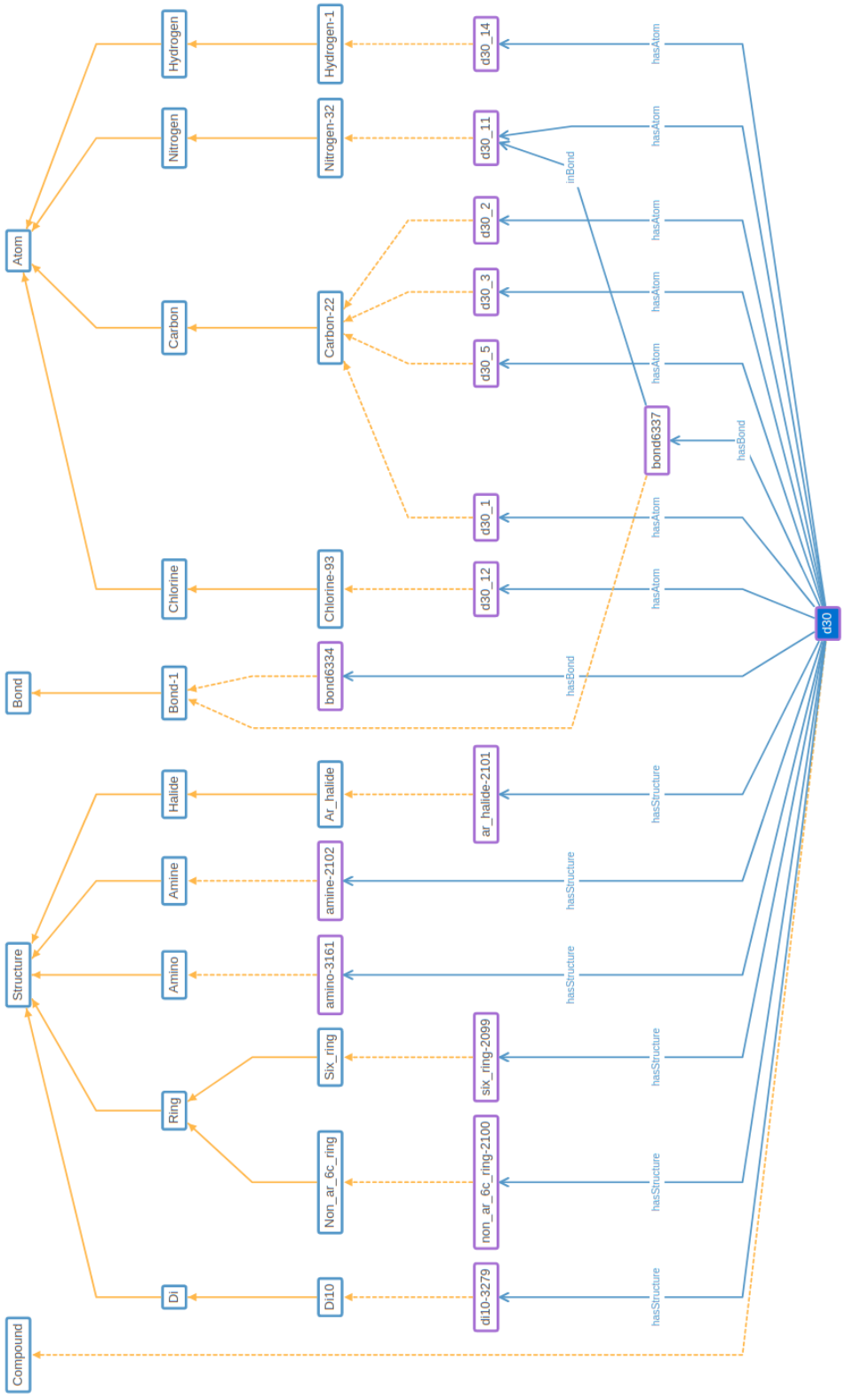


Figure 6.3: Excerpt from MUTAG depicting am instance of Compound (d30).

ontowalk2vec BFS and WL seem to provide, for the MUTAG ontology, on average, higher accuracies, in both classification problems, Random Forest and SVM. More specifically, one can observe in Figure 6.4 and Table 6.2 that the accuracies using Random Forest classification for both ontowalk2vec BFS and WL is exactly 0.74, and the same value is output by the ontowalk2vec WL with SVM classification. The best result in this experiment (0.76) is provided by ontowalk2vec BFS with SVM classification.

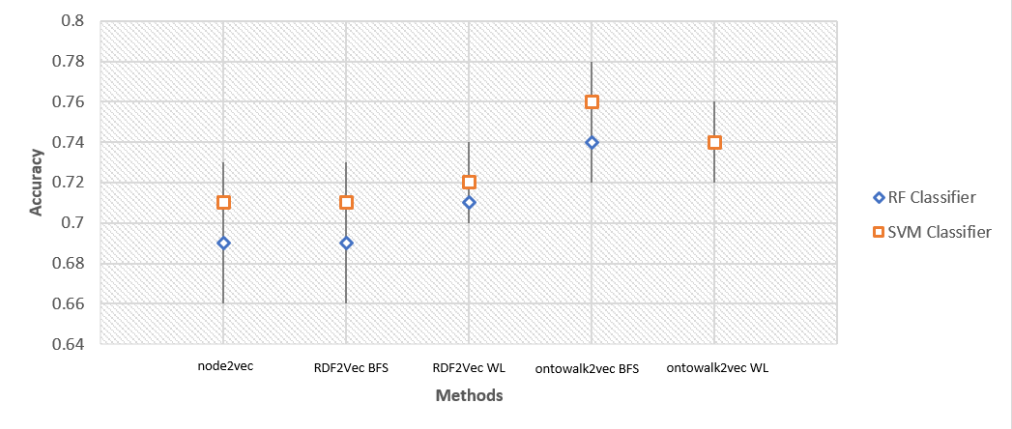


Figure 6.4: Accuracy plot for the different methodologies.

Model	Random Forest	SVM
node2vec	0.69 (±0.03)	0.71 (±0.02)
RDF2vec BFS	0.69 (±0.03)	0.71 (±0.02)
RDF2vec WL	0.71 (±0.01)	0.72 (±0.02)
ontowalk2vec BFS	**0.74** (±0.02)	**0.76** (±0.02)
ontowalk2vec WL	**0.74** (±0.02)	**0.74** (±0.01)

Table 6.2: Models' accuracy scores using the MUTAG dataset.

Confusion Matrix

In order to investigate further the accuracy results, we also use confusion matrices for understanding which predictions are affecting positively or negatively our results. According to Table 6.3, ontowalk2vec appears to have high prediction scores, on average, for predicting the negative data points, which are actually the non-mutagenic molecules. This is outlined in Table 6.3 for the Random Forest classification with BFS (41.1) and with WL (41.2), while using the SVM classification, one gets 41.2 and 43.1 for BFS and WL, respectively. Furthermore, the

Actual Value	Negative	Negative	Positive	Positive
Predicted Value	Negative	Positive	Negative	Positive
node2vec RF	35.4 (±2)	9.6 (±2)	11.4 (±0.5)	11.6 ((±2))
node2vec BFS - SVM	40.7 (±2.3)	4.3 (±2.3)	14.8 (±1.9)	8.2 (±1.9)
RDF2vec BFS - RF	35.9 (±2.2)	9.1 (±2.2)	10.4 (±1.9)	12.6 (±1.9)
RDF2vec BFS - SVM	40.7 (±1.8)	4.3 (±1.8)	14.5 (±1.3)	8.5 (±1.3)
RDF2vec WL - RF	36.8 (±2.1)	8.2 (±2.1)	13.2 (±1.5)	9.8 (±1.5)
RDF2vec WL - SVM	37.7 (±3.1)	7.3 (±3.1)	14.1 (±1.9)	8.9 (±1.9)
ontowalk2vec BFS - RF	**41.1** (±1.1)	**4** (±1.1)	14 (±1.1)	9 (±1.1)
ontowalk2vec BFS - SVM	**41.2** (±2.1)	**3.8** (±2.1)	12.8 (±2.3)	10.2 (±2.3)
ontowalk2vec WL - RF	**41.2** (±1.7)	**3.8** (±1.7)	14.7 (±1.1)	8.3 (±1.1)
ontowalk2vec WL - SVM	**43.1** (±1.4)	**1.9** (±1.4)	16.4 (±0.8)	6.6 (±0.8)

Table 6.3: Models' confusion matrices for the MUTAG ontology.

false positives are low (as depicted in Table 6.3 for Random Forest classification with BFS (4) and with WL (3.8), while using the SVM classification, one gets 3.8 and 1.9 for BFS and WL, respectively. However, the results are in line with the other tested models for predicting the molecules that are considered mutagenic. This is probably due to the fact the MUTAG ontology contains less data points representing mutagenic molecules; thus the model does not get well trained to learn with high accuracy the positive labels.

t-SNE

Our generated embeddings for all the given data points are also visualised using t-SNE plots. In the following figures 6.5, 6.6 and 6.7, the red points represent the molecules of the test data that were classified as non-mutagenic while the green points show the molecules that were classified as mutagenic. t-SNE should be able to separate the points of different colours in different colour groups, and there should be some significant distance among the colour groups.

Figure 6.5 is the t-SNE visualisation produced by the embeddings generated

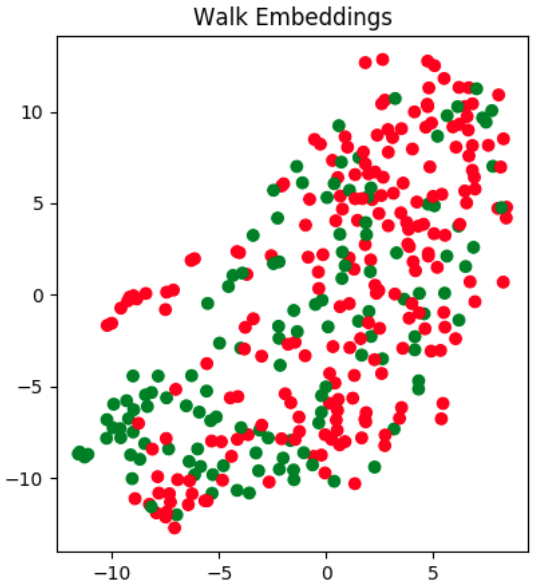


Figure 6.5: node2vec t-SNE visualisation of mutagenic and non-mutagenic elements.

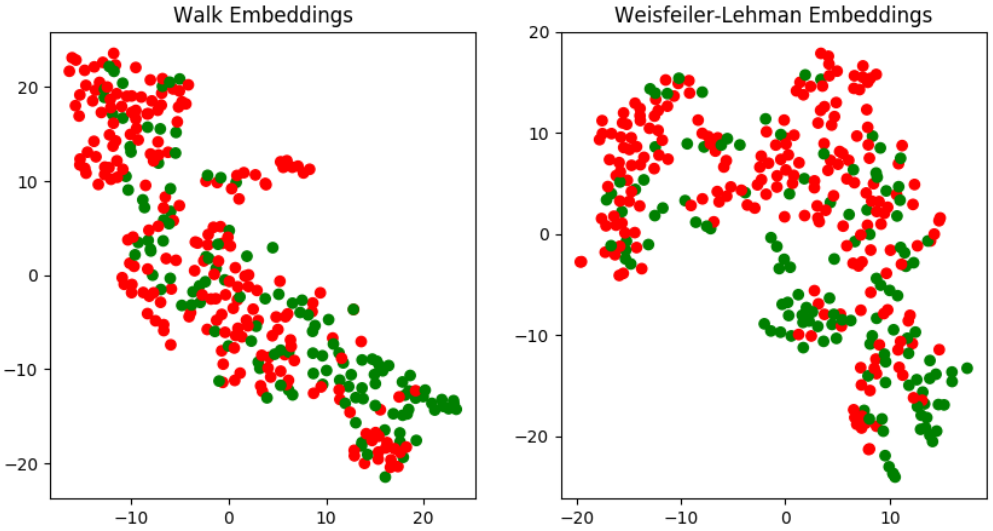


Figure 6.6: RDF2Vec t-SNE visualisation of mutagenic and non-mutagenic elements.

by the node2vec model. As it can be observed from the figure, the data points classified are quite mixed, and there is no clear distinction among the colour groups. This means that the vectors representing mutagenic and non-mutagenic

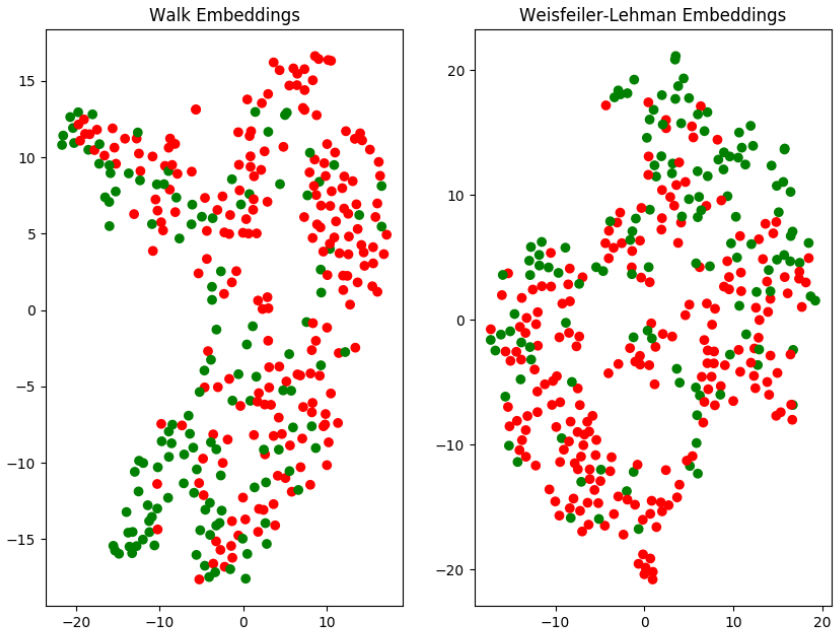


Figure 6.7: ontowalk2vec t-SNE classification of mutagenic and non-mutagenic elements.

molecules do not have high similarity distance. The same observation can be made for Figure 6.6, where the results from RDF2Vec are shown. Both BFS and Weisfeiler-Lehman embeddings do not seem to provide any clear classification of the two vector categories.

Figure 6.7 presents the t-SNE plots of ontowalk2vec. From the figure, it can be observed that there seems to exist more similarity among the vector embeddings. In the first plot, related to the BFS-generated embeddings, one can see a slim separation of the two classes. The mutagenic molecules (green points) seem to appear more numerous on the left side of the plot while the non-mutagenic ones (red points) seem to gather slightly in the right part of the plot. Similar observation can be made for the Weisfeiler-Lehman embeddings; it appears that the non-mutagenic molecules tend to be more numerous in the lower part of the plot, while the mutagenic molecules appear more often in the top part of the plot. This means that ontowalk2vec provides better embeddings for classifying the data than node2vec and RDF2Vec. However, it is admittedly not leading to a very clear distinction. This is studied further in Subsection 6.5, where the issue of hyperparameter optimisation is discussed.

6.4 Using ontowalk2vec with OWAO

In the previous section, it was suggested that based on different evaluation metrics (accuracy, confusion matrix and t-SNE), ontowalk2vec is providing, on average, better classification results and thus better ontology embeddings for the MUTAG ontology. In this section, the experiments are dedicated to the main purpose of building the ontowalk2vec model, which is to provide qualitative vector embeddings for the instances of the OWAO ontology and, more specifically, the part related to the UI preferences of users. Therefore, in the following section, detailed experiments with the OWAO ontology are prepared, in order to assess how the ontowalk2vec model is performing there.

6.4.1 Experimental Setup

In the case of OWAO, it was first needed to elaborate on the generation of ontology instances by compiling the current research findings concerning UI preferences, detailed in Section 3.4.1, and thus creating artificial data, given the lack of existing ones. More specifically, we focus on four UI components: the background colour, font colour, font size and text alignment. Since our interfaces are built using the Semantic UI framework [SUI], the discrete values of the previously mentioned UI Style components are those defined via the Semantic UI classes. For instance, as shown in Table 6.5, for the font size, the discrete defined values can be: tiny, small, medium, large, etc. In order to build instances of the class `owao:UIStyle`, all the combinations of font size, text alignment, background and font colour are generated. However, only the combinations that, according to the research findings, are generally more appealing to users would logically be preferred by them.

For example, Figure 6.8 shows a simplified excerpt of OWAO instances related to UI preferences. The figure shows the instance of a `owao:User1` associated to an instance of `owao:UIPreference` named `owao:white_black_center_Form1_pref_1` corresponding to an instance of `owao:Form` class named `owao:Form1`. This preference corresponds to the `owao:UIstyle` instance named `owao:ui_style_white_ black_mini_center`.

Tables 6.4, 6.5, and 6.6 show the percentages of users that would prefer a specific UI style based on our User Experience (UX) research findings (see Section 3.4.1). Since the background and font colours have a high interdependence, the background colour is taken as the independent parameter that varies the most, while the font colour is either black or white.

Table 6.4 depicts only the colour combinations that would provide a non-zero percentage value, but in the dataset generated for OWAO, all combinations are taken into account. For instance, if we assume that 1,000 users are registered in an OWAO-derived web application, the users that would prefer a UI style of

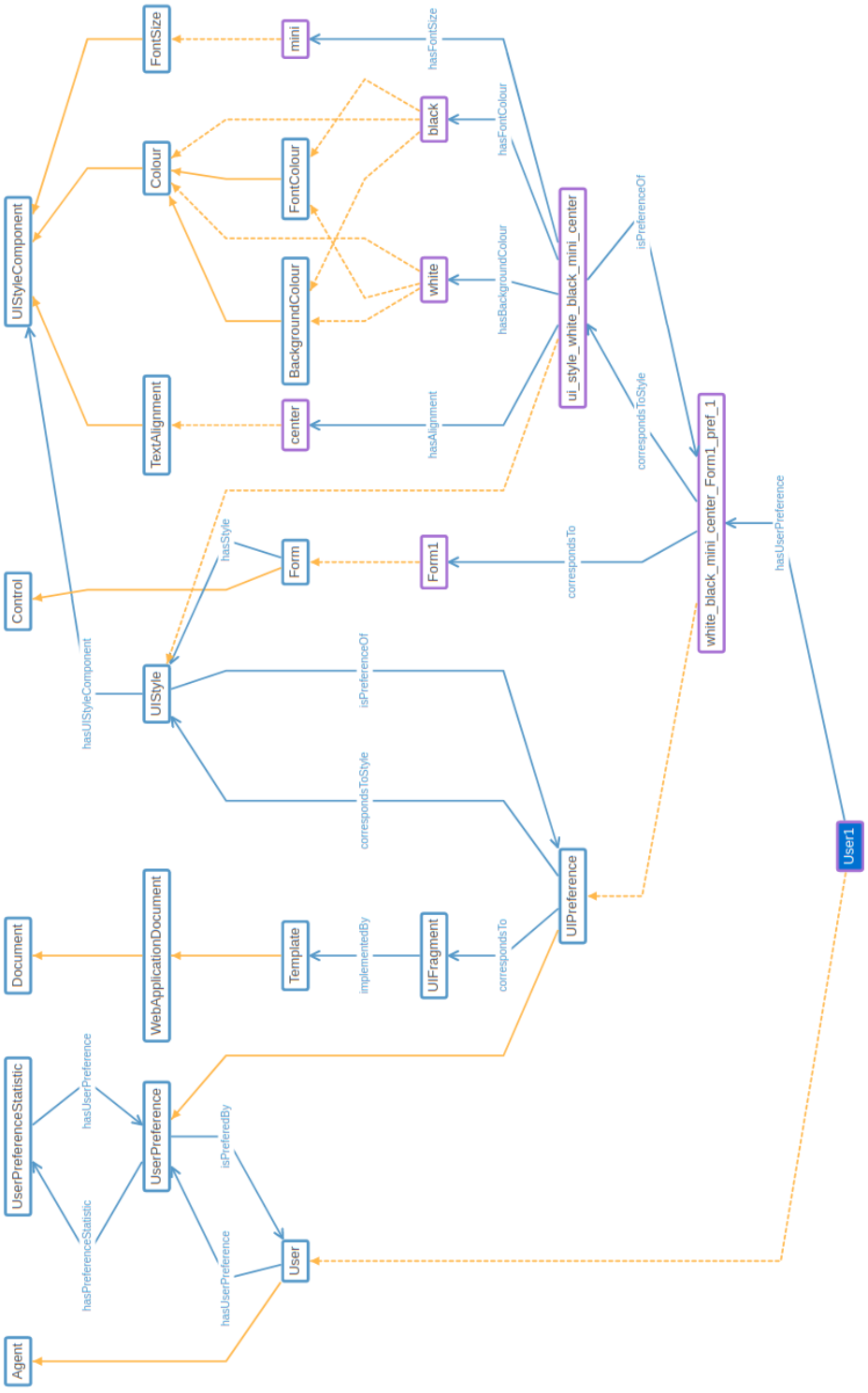


Figure 6.8: Excerpt from OWAO depicting a User instance (`User1`) and his/her preferences for a given UI component.

Background Colour	Font Colour	%
white	black	24
light grey	black	23
black	white	10
dark grey	white	10
blue	white	8
green	white	7
purple	white	6
red	white	5
orange	white	4
yellow	white	3

Table 6.4: Background and font colour preferences for ontowalk2vec.

Font Size	%
mini	1
tiny	4
small	10
medium	40
large	30
big	10
huge	4
massive	1

Table 6.5: Font size preferences.

Text Alignment	%
left	60
center	30
right	10

Table 6.6: Text alignment preferences.

white background with black font colour, medium font size and left alignment would be: $0.24 \times 0.4 \times 0.6 \times 1000 = 57$ users.

6.4.2 Experiment 1: Classification of Preferences

Following the same experimental setup as for MUTAG, a simple binary classification problem is introduced. Instead of the data property `mutag:isMutagenic`, in OWAO, a boolean data property `owao:preferred` is introduced and associ-

ated to a `owao:UIStyle` in order to denote if the UI style composed from specific UI components is preferred by users or not. This means that if at least 1 user prefers the specific `owao:UIStyle` instance, the `owao:preferred` data property is true, otherwise false. The ontowalk2vec model should predict, by exploiting the OWAO ontology data, whether a specific UI style is preferred by the users.

	Random Forest	SVM
BFS	1.00 (±0.00)	1.00 (±0.00)
WL	1.00 (±0.00)	0.99 (±0.01)

Table 6.7: Accuracy for OWAO preferences.

Actual Value	Negative	Negative	Positive	Positive
Predicted Value	Negative	Positive	Negative	Positive
ontowalk2vec BFS - RF	343 (±0)	0 (±0)	0 (±0)	57 (±0)
ontowalk2vec BFS - SVM	343 (±0)	0 (±0)	0 (±0)	57 (±0)
ontowalk2vec WL - RF	343 (±0)	0 (±0)	0 (±0)	57 (±0)
ontowalk2vec WL - SVM	343 (±0)	0 (±0)	2 (±2)	55 (±2)

Table 6.8: Confusion matrix for OWAO preferences.

Table 6.7 shows the accuracy score based on the binary classification using as label the data property `owao:preferred` in order to evaluate the ontowalk2vec model on whether a UI style is preferred by any user or not. As in the MUTAG experiment, also in this one, the algorithm run 10 times, and values presented are the mean values while the errors shown are the standard deviation. According to the table, for both methods BFS and Weisfeiler-Lehman (WL) and with both classifiers, Random Forest and SVM, the accuracy value is 1 or approaching 1. This means that the model is always accurate on the predictions, except for the case of the WL method for the SVM classification where the value is 0.99, being still quite high.

Observing in more detail the predictions presented by Table 6.8 depicting the confusion matrix of the models, the false positives and false negatives are as expected, in most of the cases equal to 0.

In addition to that, the t-SNE plots show a clear classification of the two categories, preferred and not preferred. The points of preferred values (in green colour) are well separated of the not-preferred ones, and no mixture of the two categories is observed.

According to previous metrics, it seems that the classification experiment for the OWAO data using the ontowalk2vec model is functional. However, in order to avoid evaluating the model only on a simplified classification task, more experiments are performed. The model is now evaluated not on a binary classification

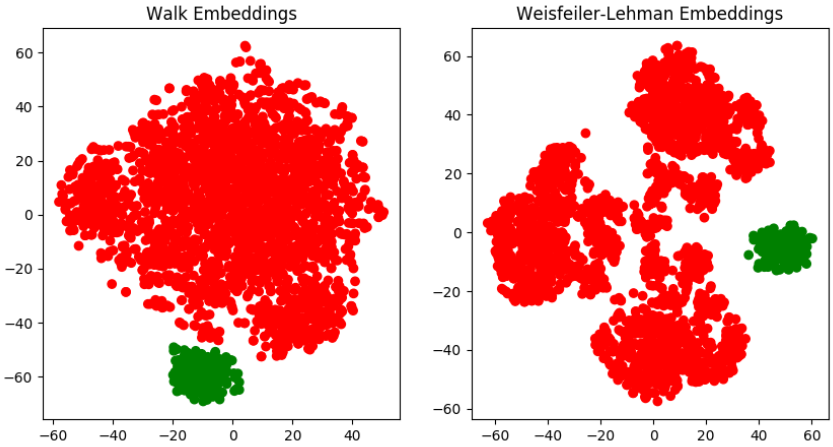


Figure 6.9: t-SNE visualisation for OWAO preferences.

problem but on a multidimensional classification.

6.4.3 Experiment 2: Classification of Popularity

For a better and more reliable assessment of classification performance, the data property `owao:popular` is introduced. The value of this data property of type `float` denoted the percentage of users having the specific UI preference. It is used to classify the UI style as: most, medium, less and not popular. Four instead of two values are used now for classifying the data points. These labels are (i) zero popularity, (ii) low popularity, (iii) medium popularity and (iv) high popularity, while the thresholds for providing these labels are defined in Table 6.9.

Since in our example, all the UI style combinations are taken into account, there are quite a lot of UI style instances with zero preference. The set thresholds for classifying the UI styles, consequently rather low, are provided by the table. This way, the OWAO dataset is tested with more labels, making the classification problem more challenging and thus providing deeper insights about our models.

Popularity	%
zero popularity	0%
low popularity	0% < value < 1%
medium popularity	1% < value < 3%
high popularity	3% < value

Table 6.9: Popularity values thresholds.

	Random Forest	SVM
BFS	0.990 (±0.001)	0.995 (±0.002)
WL	0.993 (±0.002)	0.995 (±0.001)

Table 6.10: Accuracy for OWAO popularity.

		Actual Values			
	Popularity	zero	low	medium	high
Predicted Values	BFS Random Forest				
	zero	**372 (±0)**	0 (±1.1)	0 (±1.1)	0 (±1.1)
	low	0 (±0)	**24 (±0)**	0 (±0)	0 (±0)
	medium	0 (±0)	1.8 (±0.4)	**0.2 (±0.4)**	0 (±0)
	high	0 (±0)	1.3 (±0.7)	0.7 (±0.7)	**0 (±0)**
	BFS SVM				
	zero	**372 (±0)**	0 (±0)	0 (±0)	0 (±0)
	low	0 (±2.1)	**23.9 (±0.3)**	0.1 (±0.3)	0 (±0)
	medium	0 (±0)	0.9 (±0.7)	**1.1 (±0.7)**	0 (±0)
	high	0 (±0)	0.5 (±0.8)	0.9 (±0.8)	**0.5 (±0.7)**
	WL Random Forest				
	zero	**372 (±0)**	0 (±0)	0 (±1.1)	0 (±1.1)
	low	0 (±0)	**24 (±0)**	0 (±0)	0 (±0)
	medium	0 (±1.7)	0.9 (±0.8)	**1.1 (±0.9)**	0 (±0)
	high	0.1 (±0.3)	0.2 (±0.6)	1.7 (±0.7)	**0 (±0)**
	WL SVM				
	zero	**372 (±0)**	0 (±0)	0 (±0)	0 (±0)
	low	0 (±0)	**24 (±0)**	0 (±2.3)	0 (±0)
	medium	0 (±0)	0.1 (±0.3)	**1.9 (±0.3)**	0 (±0)
	high	0 (±0)	0 (±0)	1.6 (±0.5)	**0.4 (±0.5)**

Table 6.11: Confusion matrix of OWAO popularity.

Whole Dataset

After running ontowalk2vec and the same classification methods as before, the accuracy scores were calculated and depicted in Table 6.10. According to this table, one can observe that the accuracy scores remain high (more than 0.990) in all the methods.

Table 6.11 presents the confusion matrix for the four labels. In this type of table, the accurate results, meaning the ones where the predicted label is the actual one, are found in the diagonal of each sub-table that shows the results for each method and classifier, while the rest of the values should be 0. For example, in the first sub-table presenting the results of the ontowalk2vec BFS using Ran-

dom Forest, the diagonal shows that 372 of the data points were predicted with as having "zero popularity", while the actual label was indeed "zero popularity". It can be observed that, for the labels where enough data points exist ("zero" and "low"), there are very few wrong predictions, while the classification works less well for the other labels.

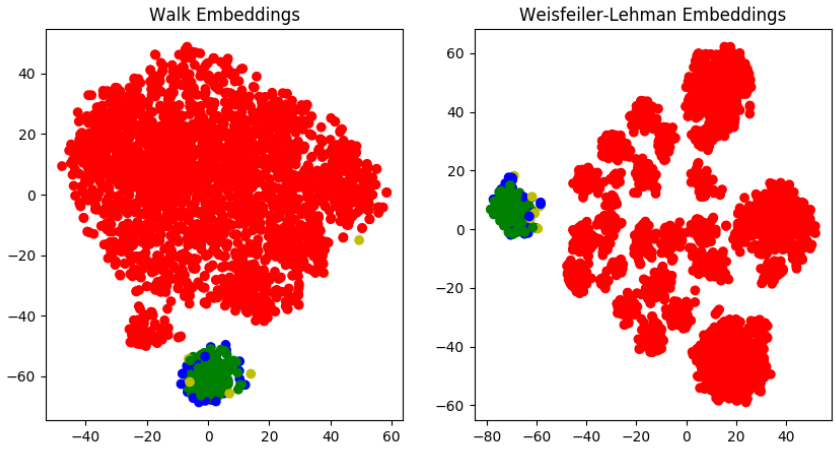


Figure 6.10: t-SNE visualisation for OWAO popularity.

The t-SNE plots in Figure 6.10 are also classifying the selected data points. The points corresponding to the "zero popularity" values (red points) seem to be well separated from the other points. However, the plots do not show a clear distinction of the other categories, namely low popularity (green colour points), medium popularity (blue colour points) and high popularity (yellow colour points).

6.4.4 Non-Zero Dataset

The observations above call for a final experiment for evaluating the model, i.e., to perform classification only on the points that have a non-zero popularity, while not taking into account the zero values that may provide some bias to our model.

After modifying the training and test data accordingly, the algorithm is ran again for 10 iterations. As shown in Table 6.12, the accuracy scores for the Random Forest classifier is slightly dropping for both the BFS (0.83) and WL (0.86) methods. However, the SVM classifier seems to provide still high accuracy and succeeds in predicting the labels (BFS: 0.94 and WL: 0.92).

	Random Forest	SVM
BFS	0.83 (±0.04)	0.94 (±0.04)
WL	0.86 (±0.06)	0.92 (±0.05)

Table 6.12: Accuracy for OWAO popularity with non-zero popularity values.

	Actual Values			
	Popularity	low	medium	high
Predicted Values	BFS Random Forest			
	low	**11.7 (±0.4)**	0.3 (±0.4)	0 (±0)
	medium	**1 (±0.8)**	**17 (±0.8)**	0 (±0)
	high	0 (±0)	5.4 (±1.4)	**2.6 (±1.4)**
	BFS SVM			
	low	**11.5 (±1.2)**	0.5 (±1.2)	0 (±0)
	medium	1.2 (±1.1)	**16.5 (±1.2)**	0.2 (±0.4)
	high	0 (±0)	0.9 (±0.3)	**7.1 (±0.3)**
	WL Random Forest			
	low	**11.2 (±0.8)**	0.8 (±0.8)	0 (±0)
	medium	1.3 (±1.1)	**16.2 (±1.6)**	0.5 (±0.7)
	high	0 (±0)	2.8 (±1.7)	**5.2 (±1.7)**
	WL SVM			
	low	**11.9 (±0.3)**	0.1 (±0.3)	0 (±0)
	medium	1 (±1.5)	**16.2 (±1.2)**	0.8 (±0.8)
	high	0 (±0)	0.7 (±1.2)	**7.3 (±1.2)**

Table 6.13: Confusion matrix of OWAO popularity without 0 values.

This result can be analysed in more detail by the confusion matrix presented in Table 6.13. It seems that ontowalk2vec can predict most of the labels of low and medium popularity correctly. For example, in the first sub-table showing the BFS Random Forest classification, on average the misprediction of "medium popularity" appears only in 1 data point out of 18, while 17 data points out of the 18 are predicted accurately. In the case of the high popularity values, the SVM classifier provides better predictions in comparison to the Random Forest classifier, which fails in classifying the high-popularity points. As shown in Table 6.13, the BFS SVM predictions for the "high popularity" labels output the correct result for, on average, 7.1 out 8 data points, while the WL SVM result is 7.3.

Regarding the t-SNE visualisation depicted in Figure 6.11, it appears as if the three categories are forming ring-shaped clusters. The low-popularity points (green) tend to appear in the central part of both the BFS and WL plots. The medium-popularity points (blue points) appear to make a cluster around the

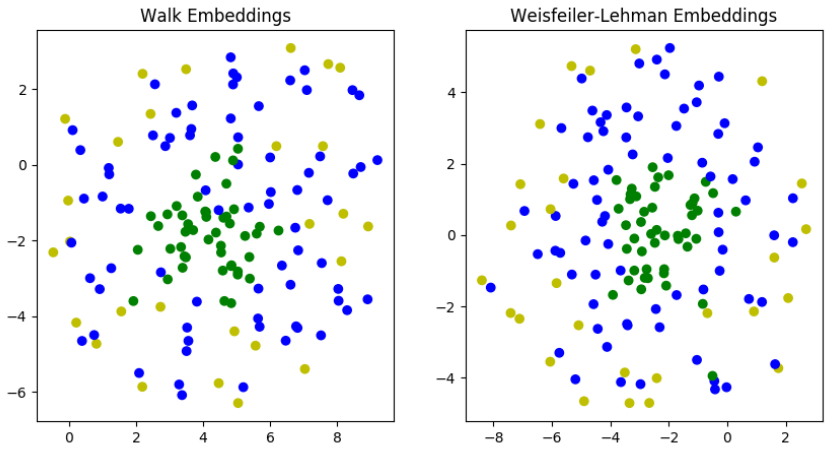


Figure 6.11: t-SNE visualisation for OWAO popularity without 0 values.

low-popularity points, while the high-popularity points (yellow) compose the external ring cluster. However, there is not a clear distinction among them, as we can see some points appearing in an incorrect cluster. This is likely caused by the fact that, in this third experiment, there are much less data points.

6.4.5 Cosine Similarity of OWAO Embeddings

Taking as an example the instance of `owao:User251`, Tables 6.14 and 6.15 exhibit the instances that have the highest similarity values according to their vector embeddings with the selected reference instances. For example, Table 6.14 shows high similarity values for those users, because they have similar UI style preferences, e.g., `owao:User262` prefers UI style `owao:ui_style_light_grey_black_small_left`, the same style as `owao:User251`. Similar results are observed also in Table 6.15.

Another important observation is that the two methodologies output the same four out of the top five instances: `owao:User262`, `owao:User264`, `owao:User259` and `owao:User260`. This means that the model outputs consistent results independently of using the BFS or WL methodologies. The only difference is that the WL methodology is providing a higher similarity value than BFS (0.979).

Instances	Cosine Similarity
`owao:User262`	**0.973**
`owao:User264`	0.962
`owao:User259`	0.957
`owao:User260`	0.951
`owao:User253`	0.950

Table 6.14: Top 5 highest cosine similarities for `owao:User251` using BFS.

Instances	Cosine Similarity
`owao:User262`	**0.979**
`owao:User264`	0.975
`owao:User260`	0.967
`owao:User254`	0.966
`owao:User259`	0.965

Table 6.15: Top 5 highest cosine similarities for owao:User251 using WL.

6.5 Optimisation

The word2vec model integrated in ontowalk2vec for generating the embeddings, but also RDF2Vec and node2vec used for generating random walks, contain several hyperparameters that need to be tuned before efficiently training with these the models. These hyperparameters depend on the type and volume of data and can provide higher accuracy results once they are properly initialised [CDLRL18, Dil19].

Not all the hyperparameters are useful for getting better results. For example a hyperparameter of word2vec is *workers*, which denotes the number of worker threads needed to train the model, obviously aiming for faster training with multi-core machines. Another hyperparameter defines whether the model will use the Skip-gram or the "Continuous Bag of Words" (CBOW) technique, but as already explained in Section 4.3.3; for ontology problems such as ours it is more suitable to use Skip-gram since with the CBOW, the order of the entities appearing in a walk would be lost. In the literature, the most common hyperparameters that are targeted for optimisation are: negative sampling, iterations and window size (described below) [CDLRL18]. In this work, a larger set of hyperparameters is used,to try to provide even better performance. The fine tuning of these hyperparameters is presented below, and illustrated by the use of these optimisation techniques for finding the best combination of hyperparameters for OWAO.

More specifically, the hyperparameters that are being optimised here are:

- **learning rate (alpha)**, i.e., a hyperparamter that controls the amount of change of the weights of the model in each iteration for reaching convergence. In word2vec this hyperparameter denotes the initial value for starting the training of the word2vec model and in each iteration this value changes to a smaller one until reaching the minimum value 0.0001;
- **iterations**, i.e., the number of iterations (epochs) of training over the corpus of data;
- **vector size**,i.e., the dimensionality of the vector space used to represent the data;
- **window**, i.e., the maximum distance between the current and predicted words within a sentence.
- **minimum counts**, such that the words with a total frequency lower than `min_count` are ignored by the model;
- **hierarchical softmax**. Softmax is a normalized exponential function used as the last layer of a neural network to normalise the output. Hierarchical softmax is an optimised softmax funtion representing the words as the leaves of a a binary tree. The hyperparameter hierarchical softmax is a boolean parameter specifying whether hierarchical softmax is used or not;
- **negative** An alternative to hierarchical softmax is the negative sampling approach. The idea behind negative sampling is to distinguish the words that do not contribute in the output (noise) and the useful data. The word2vec hyperparameter indicates whether negative sampling is used or not. If the value is more than 0, negative sampling will be used, while the specified integer is the number of noisy words that should be drawn. If it is set to 0, no negative sampling is used;
- **depth**, i.e., the maximum level to be reached when traversing the ontology-derived graph.

6.5.1 Experimental Setup

In order to better compare the generated embeddings, 10 pairs of OWAO instances were selected for evaluating the cosine similarity value of their vectors according to the output of the ontowalk2vec model. These pairs are listed in Tables 6.16 and 6.17. As explained in the second column of the table, some of the pair instances are relevant to each other, while others are not: in the first case, the cosine similarity should be high, while in the latter, low. For example, the instance `owao:white_black_medium_left` is a UI style very similar to `owao:white_black_medium_left`, and thus the similarity value between these

two vectors should be high. Moreover, the instance `owao:white_black_medium_left` is preferred by user `owao:User103`. Therefore, these instances are relevant, meaning that there is a OWAO-based relation between them. For this reason, their cosine similarity should be high. In contrast to the previous example, `owao:User917` should have a low similarity value, because it has no clear resemblance with the instance `owao:white_black_medium_left`. The same reasoning can be applied to the paired instances of Table 6.16. The three first instances should have high similarity values, because there is a relation among them, while the two last instances should have low similarity values.

instance `owao:white_black_medium_left`			
	Compared instance	Explanation	Cos. Similarity
Relevant	`owao:ui_style_white_black_ medium_center`	Similar UI style	BFS:**0.403** WL:**0.427**
	`owao:white_black_medium_ left_Form1_pref_60`	UI preference related to this UI style	BFS:0.763 WL:0.575
	`owao:User103`	User with the specific UI preference	BFS:0.425 WL:0.208
Irrelevant	`owao:ui_style_greenBack ground _yellow_big_right`	UI style instance that is not similar to the instance	BFS:0.266 WL:0.267
	`owao:User917`	User instance that does not prefer the specific UI style	BFS:0.255 WL:0.275

Table 6.16: Instances for similarity comparison with `owao:white_black_ medium_left`.

However, as it is observed from the tables, the model using the current hyperparameter values does not always provide the expected results and needs to be optimised. For example, the instance `owao:ui_style_white_black_ medium_center` should have high similarity value with `owao:white_black_medium_left` since they are similar UI styles but the results are quite low (BFS: 0.403) and (WL:0.427). For the optimisation process, these pairs will be taken as reference data to optimise the ontowalk2vec model.

instance owao:User251			
	Compared instance	Explanation	Cos. Similarity
Relevant	owao:ui_style_light_grey_black_small_left	UI style preferred by owao:User251	BFS:0.671 WL:0.512
	owao:white_black_medium_left_Form1_pref_251	UI preference related to the user instance	BFS:0.895 WL:0.921
	owao:User263	User instance with the same UI preference	BFS:0.887 WL:0.935
Irrelevant	owao:ui_style_light_grey_black_medium_left	UI style instance that is not related to the user	BFS:0.487 WL:0.319
	owao:User872	User instance that does not share same UI preferences with owao:User251	BFS:0.794 WL:0.859

Table 6.17: Instances for similarity comparison with owao:User251.

6.5.2 Evaluation Metrics and Results

For performing the experiments, the true positives of the selected data are the ones labelled as relevant cases in Tables 6.16 and 6.17, while the true negatives are the irrelevant cases of the same tables. Taking into account the fact that when the model is trained in several iterations, the similarity values of true positives cases/pairs (e.g. owao:ui_style_white_black_ medium_center and owao:white_black_medium_left) is on average 0.7, this value is chosen as the threshold to assess whether the model is predicting that the instance is relevant (positive) or irrelevant (negative).

For the purpose of assessing the model validity using the results provided with this experimental setup, the most common metrics for evaluating a recommender system are utilised [SG11]. The first evaluation metric is precision, defined as:

$$\frac{\|true\,positives\|}{\|true\,positives\| + \|false\,positives\|}. \tag{6.3}$$

where $\|L\|$ denotes the number of elements in set L.

The second evaluation metric is recall,defined as:

$$\frac{\|true\,positives\|}{\|true\,positives\| + \|false\,negatives\|}. \tag{6.4}$$

A good model is the one for which both precision and recall values are high (approaching 1). For this reason, a combined metric is commonly used for evaluating recommender systems; this is the F1 score, providing a normalised average and defined as:

$$2 * \frac{precision * recall}{precision + recall}. \tag{6.5}$$

In order to evaluate how the hyperparameters affect our model and which hyperparameters should be used for maximising both precision and recall, an exhaustive grid search was performed using hyperparameter combinations uniformly sampled from the range of hyperparameter values shown in Table 6.18.

Hyperparameter	Range	Default Value
learning rate (lr)	[0.1, 0.05,0.025,0.01]	0.025
iterations (iter)	[1,5,20,50]	5
vector size (vs)	[20,100, 200, 500]	100
window (win)	[3,5,7]	5
min_count (mc)	[0,1]	5
negative (neg)	[0,1,5,10]	5
hierarchical softmax (hs)	[0,1]	0
depth (dep)	[1,2,3]	1

Table 6.18: Hyperparameter and their range, used to form combinations.

As shown in the histograms of Figures 6.12 and 6.13, most of the combinations result in a F1 score lower than 0.9; 8 hyperparameter combinations for BFS (presented in Table 6.19) and 10 for WL (presented in Table 6.20) provide F1 score higher than 0.9. Since our goal is to maximise both precision and recall, i.e., to find the hyperparameters that make the model give the highest F1 score, only the hyperparameter combinations where the F1 score is higher than 0.9 are selected.

Tables 6.19 and 6.20 show the top hyperparameter combinations among those that provide the best F1 scores and consequently the best combination of precision and recall. While it is difficult to conclude in a single rule taking into account all the hyperparameters, the same combination, highlighted in bold, is found in the top scores of both methodologies BFS and WL. This combination is given by the following hyperparameter setting: `window size = 7`, `vector size = 500`, `learning rate = 0.0025`, `iteration = 1`, `min count = 1`, `depth = 2`, `negative = 10` and thus `hierarchical softmax = 0`.

Table 6.21 illustrates the top recommendations (top similarity predictions) for an example of the UI style instance `owao:ui_style_light_grey_black_medium_center`, when using the BFS method. The instances recommended by the on-

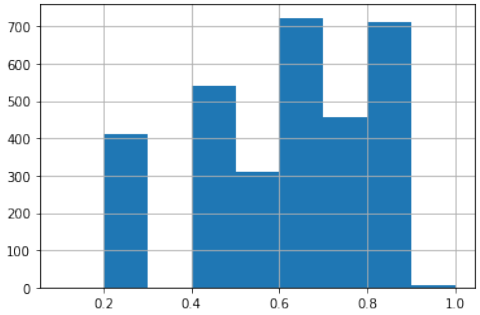


Figure 6.12: BFS F1 score histogram.

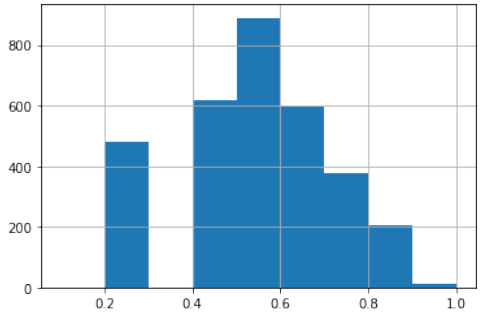


Figure 6.13: WL F1 score histogram.

Hyperparameters								Evaluation metrics		
win	vs	hs	lr	iter	neg	cnt	dep	rec	prec	F1
3	200	1	0.010	1	0	1	1	1	0.85	0.923
7	100	0	0.025	1	5	0	2	1	0.857	0.923
5	500	0	0.025	1	20	0	2	1	0.857	0.923
7	**500**	**0**	**0.025**	**1**	**10**	**0**	**2**	1	0.857	0.923
5	200	1	0.010	1	0	1	1	1	0.857	0.923
5	100	10	0.025	1	5	0	2	1	0.857	0.923
7	500	0	0.010	5	5	1	1	1	0.857	0.923
7	20	0	0.025	50	1	0	2	0.833	1	0.909

Table 6.19: Top F1 score (> 0.9) for BFS.

Hyperparameters								Evaluation metrics		
win	vs	hs	lr	iter	neg	cnt	dep	rec	prec	F1
7	100	0	0.010	5	5	1	1	1	0.857	0.923
7	**500**	**0**	**0.025**	**1**	**10**	**0**	**2**	1	0.857	0.923
3	500	0	0.050	1	5	0	1	1	0.857	0.923
5	500	0	0.025	1	20	1	1	1	0.857	0.923
5	20	0	0.025	1	20	1	1	1	0.857	0.923
7	20	0	0.025	1	10	1	1	1	0.857	0.923
7	100	0	0.025	1	10	0	2	1	0.857	0.923
3	100	0	0.050	1	5	0	1	1	0.857	0.923
7	20	0	0.010	5	5	1	1	1	0.857	0.923
3	20	0	0.050	1	5	1	1	1	0.857	0.923

Table 6.20: Top F1 score (> 0.9) for WL.

towalk2vec model that appear to have high similarity values with this particular

instance do indeed have common UI style elements such as the same font size, similar background colour or font colour and text alignment.

Instances	Cosine Similarity
owao:ui_style_white_black_medium_center	0.956
owao:ui_style_light_grey_black_medium_left	0.955
owao:ui_style_light_grey_black_large_left	0.941
owao:ui_style_white_black_large_left	0.931
owao:ui_style_light_grey_black_medium_center	0.924

Table 6.21: Top cosine similarities for `owao:ui_style_white_black_medium_left` using BFS.

6.6 Summary

This chapter presents the ontowalk2vec algorithm and testing classification experiments. Ontowalk2vec can be used for generating embeddings from OWAO instances and seem to perform on average better than state-of-the-art methodologies. Moreover, an exhaustive search for selecting the best hyperparameters is performed. Several classification methods and evaluation metrics have been used for testing purposes, and the experimental validity of the results is shown through various examples. The code of ontowalk2vec containing all the ontologies used and techniques of hyperparameter optimisation can be found in the online resources[5].

Using high-accuracy vector embeddings allows for building a solid recommender system that can suggest to users UI styles similar to their current preferences or to those of users who have similar UI preferences or features. More generally, this means that our approach can be used for both content-based recommenders system and collaborative filtering, if there are enough data. Finally, though more software development work would be needed, these embeddings could be integrated in GenAppi-generated web applications and used for optimising User Experience by recommending different UI styles.

[5]https://gitlab.cern.ch/irrad/ontowalk2vec

Chapter 7

Conclusion

Version française

Ce travail de thèse vise à combler le fossé entre la sémantique du web et la physique expérimentale des particules dans le but de normaliser les connaissances de ce domaine scientifique spécifique, de faciliter la génération de systèmes de gestion de données homogènes et de proposer une méthodologie pour l'utilisation de recommandations fondées la personnalisation de l'interface utilisateur. Dans ce dernier chapitre, quelques conclusions sur les principales contributions de cette thèse sont présentées, et les perspectives sur les travaux futurs et les applications sont discutées.

7.1 Contributions

Dans les paragraphes suivants, quelques conclusions sur les contributions de cette thèse sont présentées.

Gestion des données d'expériences d'irradiation (IEDM)

Une première contribution de cette thèse est l'ontologie de domaine IEDM dédiée à la représentation et la description des concepts et notions communes de la gestion des données dans les expériences d'irradiation. De plus, IEDM est également utilisé comme ontologie de domaine pour GenAppi, notre nouvel outil qui peut automatiquement dériver des systèmes de gestion de données dédiés aux instances de toute ontologie codée en OWL. L'application web résultante sur IEDM est présentée dans le chapitre 5.

GenAppi

Une autre contribution principale de cette thèse, présentée au chapitre 4, est donc la méthodologie GenAppi pour générer des applications web de gestion de don-

nées en utilisant en entrée toute ontologie de domaine. GenAppi utilise OWAO, une ontologie d'application web que nous introduisons également dans ce travail de recherche et qui inclut les règles d'interface utilisateur pour la génération d'applications web fondées sur GenAppi. OWAO est utilisé pour définir, en particulier, les fonctionnalités qui doivent être implémentées par l'application web, pour définir les noms des fichiers générés et pour stocker les préférences UI des utilisateurs de l'application web générée. GenAppi peut être utilisé pour générer une application web de gestion de données de base sans aucun effort de développement logiciel, sauf pour définir l'ontologie du domaine d'entrée.

Même si GenAppi est testé dans le contexte de la physique expérimentale, les règles de génération d'applications web ne sont pas limitées au domaine EPP. Par conséquent, il peut être utilisé pour d'autres ontologies de domaine. Pour toute entrée de ce type, l'application générée peut être considérée comme une application web de gestion de données de base sur laquelle les utilisateurs peuvent s'appuyer pour l'étendre, dans le cas où des fonctionnalités plus complexes sont nécessaires.

ontowalk2vec

Le modèle ontowalk2vec a été développé pour la génération de noyaux ontologiques qui peuvent être utilisés dans un système de recommandation. Il est fondé sur les modèles node2vec [GL16] et RDF2vec [RRN+19]. Notre approche ontowalk2vec a été comparée aux modèles de pointe et a donné de meilleurs résultats, en moyenne, dans les tâches de classification testées. Le chapitre 6 présente en détail l'algorithme ontowalk2vec, les expériences utilisées pour comparer ontowalk2vec ainsi que les plongements résultants lors de la prise en compte des données de l'ontologie OWAO pour les préférences d'interface utilisateur instanciées grâce aux résultats de la recherche UX décrits dans la section 3.4.1. Après avoir utilisé certaines techniques d'optimisation d'hyperparamètres, ce modèle montre des résultats cohérents et clairs, ce qui suggère qu'il pourrait être utilisé de manière fiable dans un système de recommandation hybride fondé à la fois sur l'analyse de contenu et le filtrage collaboratif.

Contributions à des projets de développement

Cette thèse a également conduit à des contributions / développements plus pratiques qui sont déjà déployés et pleinement fonctionnels.

- *L'application web IRRAD Data Manager (IDM) est utilisée dans l' installation IRRAD du CERN depuis 2018. IDM a été le point de départ et l'inspiration pour développer des applications plus génériques fondées sur les ontologies avec GenAppi. Bien que déjà utilisée de manière opérationnelle, davantage de fonctionnalités liées aux activités d'IRRAD peuvent*

être facilement mises en œuvre dans IDM. IDM peut ainsi être considérée comme une feuille de route pour l'ajout de fonctionnalités similaires plus complexes pour les applications web générées par GenAppi.

- *Les installations d'irradiation et les bases de données et sites web concernant les faisceaux pour la qualification des detecteurs sont un autre "sous-produit" de ce travail de thèse. Après des recherches effectuées sur les installations d'irradiation et de faisceaux d'essai dans le cadre de ce travail de thèse, les données collectées ont été incluses dans deux bases de données accessibles au public. De plus, deux sites web dédiés ont été développés pour permettre l'accès et la modification des données dans le monde entier par les coordinateurs des installations mentionnées.*

7.2 Perspectives

Ce travail de thèse fournit une preuve de concept pour l'idée consistant à générer automatiquement des applications web de gestion de données personnalisées à partir d'ontologies, et en introduisant comme cas d'utilisation l'ontologie IEDM. Au-delà du cadre temporel de cette thèse, les méthodologies, modèles et outils de prototypage proposés ici doivent évidemment être développés pour proposer des systèmes plus complets et plus robustes.

De plus, bien qu'IEDM, l'ontologie actuellement utilisée pour GenAppi, puisse servir de point de départ pour la normalisation de la gestion des données des expériences d'irradiation, elle nécessite une validation supplémentaire par la communauté EPP afin d'être finalement adoptée par d'autres équipes des installations d'irradiation. De plus, cette généralisation pourrait devenir le début d'un développement ultérieur d'infrastructures logicielles communes qui pourraient communiquer de manière transparente entre elles, contribuant ainsi au partage des connaissances scientifiques et technologiques entre les installations d'irradiation, améliorant la traçabilité des équipements testés et contribuant à la transparence des résultats expérimentaux.

Ce travail mérite également d'être testé au-delà de la communauté EPP, et d'être intégré et utilisé dans d'autres domaines scientifiques et industriels, apportant ainsi une preuve supplémentaire de sa généralité. Dans cette thèse, certains tests préliminaires utilisant d'autres ontologies de domaine telles que SOSA [JHC+19], EXPO [SK07] et Marinetlo, une ontologie pour le domaine marin [TAB+13], ont suggéré que GenAppi peut générer des applications web indépendamment du domaine. De plus, le volume et la structure des ontologies sous-jacentes peuvent changer avec le temps. Ainsi, plusieurs configurations de l'ontologie doivent également être prises en compte dans le temps, mettant ainsi en évidence la question du maintien de la cohérence sur les systèmes web générés, qui doivent évoluer également de manière compatible. Ce problème se

répercute de la même façon à un niveau supérieur, car des fonctionnalités supplémentaires pour les applications générées pourraient être intégrées ultérieurement dans OWAO, telles que des fonctionnalités de gestion de la communication avec d'autres systèmes logiciels.

En ce qui concerne les préférences d'interface utilisateur, davantage d'éléments d'interface utilisateur devraient être intégrés dans OWAO et dans les applications générées afin d'améliorer l'expérience utilisateur et de collecter davantage de données concernant les préférences réelles d'interface utilisateur choisies; celles-ci pourraient être utilisées comme entrée pour ontowalk2vec. De plus, les plongements ("embeddings") créés par ontowalk2vec pourraient être facilement déployés dans n'importe quel système de recommandation existant. Une autre application de ces méthodologies pourrait être la génération de plongements pour l'ontologie de domaine elle-même, si un intérêt pour une telle extension de la part des utilisateurs se manifeste. De plus, le chapitre décrivant ontowalk2vec s'est concentré davantage sur la partie méthodologique de la génération des plongements, tandis que les applications possibles font toujours partie des travaux futurs. Des recherches supplémentaires sont donc nécessaires pour tester, évaluer et adapter le modèle ontowalk2vec sur des ontologies différentes et des ensembles de données plus importants. Enfin, on pourrait envisager d'intégrer encore davantage certaines des technologies sémantiques web discutées dans cette thèse, notamment pour le déploiement d'un système de recommandation fondé sur les plongements fournis par ontowalk2vec.

7.3 Épilogue

La vision de ce travail de thèse est de servir de catalyseur pour l'apparition de systèmes de gestion de données plus interconnectés dans le domaine de l'EPP; ceux-ci faciliteraient la communication entre les équipes et les projets des installations d'irradiation et empêcheraient l'isolement des données et des connaissances, voire leur perte potentielle. À plus long terme, ce travail ouvre de nouvelles voies en combinant la sémantique du Web et l'intelligence artificielle pour la construction d'outils et d'infrastructures communs pour une meilleure croissance scientifique en physique expérimentale des particules et, espérons-le, également dans d'autres domaines scientifiques.

English Version

This thesis work aims at bridging the gap between Web Semantics and Experimental Particle Physics for the purpose of standardising the knowledge of this specific scientific field, facilitating the generation of homogeneous data management systems and suggesting a methodology for using embedding-based recommendations that can be used for UI interface personalisation. In this final chapter, some conclusions about the main contributions of this thesis are presented, and perspectives about future work and applications are discussed.

7.4 Thesis Contributions

In the following paragraphs, some conclusion about this thesis research contributions are presented.

Irradiation Experiment Data Management (IEDM)

A first contribution of this thesis is the domain ontology IEDM dedicated to the representation and description of common concepts and meanings of data management in irradiation experiments. Moreover, IEDM is also used as a domain ontology for GenAppi, our new tool that can automatically derive data management systems dedicated to the instances of any OWL-encoded ontology, and the resulting web application on IEDM is presented in Chapter 5.

GenAppi

Another main contribution of this thesis, presented in Chapter 4, is thus the GenAppi methodology for generating data management web applications by using as an input any domain ontology. GenAppi uses OWAO, the Ontology-based Web Application Ontology that we also introduce in this research work and that includes the UI rules for the generation of GenAppi-based web applications. OWAO is used to define, in particular, the functionalities that need to be implemented by the web application, to map the names of the generated files and to store the UI preferences of the generated web application users. GenAppi can be used to generate a basic data management web application without using any software development effort, except for defining the input domain ontology.

Even though, GenAppi is tested in the context of Experimental Physics, the rules for the generation of web applications are not restricted to the EPP field. Therefore, it can be used for other domain ontologies. For any input of that type, the generated web application can be considered as a baseline data management web application that manages and/or that users can build upon and extend in the case that more complex functionalities are required.

ontowalk2vec

The ontowalk2vec model was developed for the generation of ontology embeddings that can be used for a recommender system. It is based on the node2vec [GL16] and RDF2vec [RRN+19] models. Our ontowalk2vec approach was compared to the state-of-the-art models and has exhibited better results, on average, in classification tasks. Chapter 6 presents in detail the ontowalk2vec algorithm, the experiments used to compare ontowalk2vec as well as the resulted embeddings when taking as input the OWAO ontology data for UI preferences, instantiated by the UX research findings described in Section 3.4.1. After using some hyperparameter optimisation techniques, this model shows clear consistent results, which suggest that it could reliably be used in a hybrid recommender system based on both content analysis and collaborative filtering.

Contributions in Development Projects

This thesis also led to some more practical contributions/developments that are already deployed and fully functional.

- The IRRAD Data Manager (IDM) web application has been used in the CERN IRRAD facility since 2018. IDM was the initial point and inspiration for developing more generic ontology-based applications with GenAppi. Despite being already used in day-to-day operations, more functionalities related to IRRAD activities can be easily implemented in IDM. IDM can thus be seen as a road-map for adding similar more complex functionalities for GenAppi-generated web applications.
- Irradiation Facilities and Test-Beam Databases and Websites are another "by-product" of this thesis work. After research performed on the irradiation and test-beam facilities as part of this thesis work, the data that were collected have been included in two databases available publicly. Also, two dedicated websites were developed for enabling world-wide access and modification of the data by the mentioned facility coordinators.

7.5 Perspectives

This thesis work provides a proof of concept for the idea of automatic generation of personalised data management web applications by introducing as a use case the IEDM ontology. Going beyond this thesis' time frame, the methodologies, models and prototyping tools provided here should obviously be first further developed in order to offer more complete and robust systems.

Also, although IEDM, the ontology currently used for GenAppi, can serve as a starting point for the standardisation of the data management of irradiation

experiments, it requires further validation by the EPP community in order to be finally adopted by other irradiation-facility teams. Moreover, this generalisation could become the beginning of further development of common software infrastructures that could seamlessly communicate with each other, thus contributing in sharing scientific and technological knowledge among the irradiation facilities, advancing the traceability of the tested equipment and aiding in the transparency of the experimental results.

This work also deserves to be tested beyond the EPP community, and be integrated and used in other scientific and industrial domains, thus providing additional proof of its generality. In this thesis, some preliminary tests using other domain ontologies such as SOSA [JHC+19], EXPO [SK07] and Marinetlo, an ontology for the marine domain [TAB+13], have suggested that GenAppi can generate web applications independently of the domain. Moreover, the volume and structure of the underlying ontologies can change over time. Thus, several configurations of the ontology need to be taken into consideration over time too, thus highlighting the issue of maintaining consistency over the generated web systems, which need to evolve in a compatible fashion. This issue also percolates at a higher level, since additional functionalities for the generated applications can be foreseen in OWAO, such as features for managing communication with other software systems.

Concerning the UI preferences, more UI elements such as flow/grid layout, order of fields, etc.) should be integrated in OWAO and in the generated applications in order to enhance User Experience and collect more data regarding actual UI preferences; these could be used as input to ontowalk2vec. Moreover, the ontowalk2vec embeddings could be easily deployed in any existing recommender system. Another application for these methodologies could be the generation of embeddings for the domain ontology itself, if there is interests for such extension from users. Also, the chapter describing ontowalk2vec focused more on the methodological part of generating the embeddings while the possible applications are still part of future work. More research is thus needed to test, evaluate and adapt the ontowalk2vec model on different ontologies and larger datasets. Finally, one could envision integrating more of the web semantic technologies discussed in this thesis, in particular for the deployment of a recommender system based on the embeddings provided by ontowalk2vec.

7.6 Epilogue

The vision of this thesis work is to act as a trigger for more interconnected data management systems in EPP; these would facilitate the communication among irradiation facilities teams and projects and prevent data and knowledge isolation and potential loss. Longer term, this work opens new roads in combining web semantics and artificial intelligence for building common tools and infras-

tructures for achieving better scientific growth in EPP and, hopefully, also in other scientific domains.

Bibliography

[A+08a] K. Aamodt et al. The ALICE experiment at the CERN LHC. *JINST*, 3:S08002, 2008.

[A+08b] A. Augusto Alves et al. The LHCb Detector at the LHC. *JINST*, 3(LHCb-DP-2008-001. CERN-LHCb-DP-2008-001):S08005, 2008. Also published by CERN Geneva in 2010.

[AAB+98] F. Arfelli, M. Assante, V. Bonvicini, A. Bravin, G. Cantatore, et al. Low-dose phase contrast x-ray medical imaging. *Physics in Medicine and Biology*, 43(10):2845–2852, oct 1998. Available: https://doi.org/10.1088%2F0031-9155%2F43%2F10%2F013.

[ABBR19] V. Ashok, S.M. Billah, Y. Borodin, and I. Ramakrishnan. Auto-Suggesting Browsing Actions for Personalized Web Screen Reading. In *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '19, page 252–260, New York, NY, USA, 06 2019. Association for Computing Machinery. Available https://doi.org/10.1145/3320435.3320460.

[AC18] M. Acencio and The Consortium. The Gene Ontology Resource: 20 years and still GOing strong. *Nucleic Acids Research*, 49:gky1055, 11 2018.

[AFO] A+ FontSize Changer Lite. Available: https://chrome.google.com/webstore/detail/a%20-fontsize-changer-lite/ckihgechpahhpompcinglebkgcdgpkil?hl=en.

[AID] AIDA-2020 Project Homepage.
Available: http://aida2020.web.cern.ch/.

[AKP+18] S. Auer, V. Kovtun, M. Prinz, A. Kasprzik, M. Stocker, and M. E. Vidal. Towards a Knowledge Graph for Science. In *Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics*, WIMS '18, pages 1:1–1:6, New York, NY, USA, 2018. ACM. http://doi.acm.org/10.1145/3227609.3227689.

[ALB] ALBA Homepage. Available: https://www.cells.es/.

[AW05] B. Alain and H. J. Weyer. The Digital User Office (DUO). *Nuclear Instruments and Methods in Physics Research Section A:*

Accelerators, Spectrometers, Detectors and Associated Equipment, 546:591–603, 2005.

[BAL] Balsamiq wireframes. Available: https://balsamiq.com.

[BC83] D. H. Bauer and C. R. Cavonius. Improving the legibility of visual display units through contrast reversal. E. Grandjean, E. Vigliani (Eds.), Ergonomic Aspects of Visual Display Terminals, pages 137–142, London, 1983.

[BLH17] C. Barki, S. Labidi, and B. R. Hanene. Ontology-driven generation of radiation protection procedures. *World Academy of Science, Engineering and Technology International Journal of Environmental, Chemical, Ecological, Geological and Geophysical Engineering*, 11:237–242, 01 2017.

[BLHL01] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, May 2001. Available: https://www.scientificamerican.com/article/the-semantic-web/.

[BM14] D. Brickley and L. Miller. FOAF Vocabulary Specification 0.99, 2014. Available: http://xmlns.com/foaf/spec/.

[Boy09] S. A. Boyer. *Scada: Supervisory Control And Data Acquisition*. International Society of Automation, USA, 4th edition, 2009.

[Bra18] S. Bradley. *Design Fundamentals*. Vanseo Design, Boulder, Colorado, 2nd edition, 2018.

[BRB+17] M. Broseta, D. Roldan, A. Burgos, G. Cuní, D. Fernandez-Carreiras, and S. Rubio-Manrique. A Web-Based Report Tool for Tango Control Systems via WebSockets. In *16th Int. Conf. on Accelerator and Large Experimental Control Systems, Barcelona, Spain*, October 2017.

[Bre01] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[BSRF+17] S. Balakirsky, C. Schlenoff, S. Rama Fiorini, S. Redfield, M. Barreto, H. Nakawala, J. L. Carbonera, L. Soldatova, J. Bermejo-Alonso, F. Maikore, P. J. S. Goncalves, E. De Momi, Veera R. Sampath K., and T. Haidegger. Towards a Robot Task Ontology Standard. In *Manufacturing Science and Engineering Conference (MSEC 2017)*, volume 1 of *International Manufacturing Science and Engineering Conference*, 06 2017. V003T04A049 Available: https://doi.org/10.1115/MSEC2017-2783.

[C+08a] S. Chatrchyan et al. The CMS Experiment at the CERN LHC. *JINST*, 3:S08004, 2008.

[C+08b] The ATLAS Collaboration et al. The ATLAS Experiment at the CERN Large Hadron Collider. *Journal of Instrumentation*, 3(08):S08003–S08003, aug 2008. Available: https://doi.org/10.1088%2F1748-0221%2F3%2F08%2Fs08003.

[C+17] M.J. Clarke et al. Live Visualisation of Experiment Data at ISIS and the ESS. In *16th Int. Conf. on Accelerator and Large Experimental Control Systems, Barcelona, Spain*, October 2017.

[CCDT+15] D. Carral, M. L. Cheatham, S. Dallmeier-Tiessen, P. Herterich, M. D. Hildreth, P. Hitzler, A. Krisnadhi, K. Lassila-Perini, E. Sexton-Kennedy, C. Vardeman, and G. Watts. An Ontology Design Pattern for Particle Physics Analysis. In *WOP*, 2015.

[CDLRL18] H. Caselles-Dupré, F. Lesaint, and J. Royo-Letelier. Word2vec applied to recommendation: hyperparameters matter. *Proceedings of the 12th ACM Conference on Recommender Systems*, 2018.

[CER] CERN Homepage. Available: https://cern.ch.

[CFG+17] A. Curioni, R. Froeschl, M. Glaser, E. Iliopoulou, F.P. La Torre, F. Pozzi, F. Ravotti, and M. Silari. Single- and multi-foils 27al(p,3pn)24na activation technique for monitoring the intensity of high-energy beams. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 858:101 – 105, 2017. Available: http://www.sciencedirect.com/science/article/pii/S0168900217304199.

[CIF] CERN Irradiation Facilities Online Database and Website. Available: http://cern.ch/irradiation-facilities.

[Coh16] S. M. Cohen. Aristotle's Metaphysics. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, winter 2016 edition, 2016.

[DES] Deutsches Eletron-Synchrotron (DESY) Homepage. Available: http://www.desy.de/.

[Dil19] J. Dillenberger. Evaluation of model and hyperparameter choices in word2vec, 2019. Bachelor's Thesis, Koblenz University, Landau, Germany. Available: https://west.uni-koblenz.de/assets/theses/evaluation-model-hyperparameter-choices-word2vec.pdf.

[DJA] Django Framework Homepage. Available: https://www.djangoproject.com.

[DLL] DL-Learner Homepage. Available: http://dl-learner.org/.

[DOC] Docker Homepage. Available: https://www.docker.com.

[DS13] A. Darejeh and D. Singh. A Review on User Interface Design Principles to Increase Software Usability for Users with Less Computer Literacy. *JCS*, 9:1443–1450, 2013.

[dV13] Gerben K. D. de Vries. A fast approximation of the weisfeiler-lehman graph kernel for rdf data. In Hendrik Blockeel, Kristian Kersting, Siegfried Nijssen, and Filip Železný, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 606–621, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[E+19] R. K. Ellis et al. Physics Briefing Book: Input for the European Strategy for Particle Physics Update 2020. Technical Report arXiv:1910.11775, CERN, Geneva, Oct 2019. Available: https://cds.cern.ch/record/2691414.

[EB08] L. Evans and P. Bryant. LHC machine. *Journal of Instrumentation*, 3(08):S08001–S08001, aug 2008. Available: https://doi.org/10.1088%2F1748-0221%2F3%2F08%2Fs08001.

[EBCTCG05] (EBCTCG) Early Breast Cancer Trialists' Collaborative Group. Effects of radiotherapy and of differences in the extent of surgery for early breast cancer on local recurrence and 15-year survival: an overview of the randomised trials. *The Lancet*, 366(9503):2087 – 2106, 2005. Available: http://www.sciencedirect.com/science/article/pii/S0140673605678877.

[Ein11] D. Einfeld, on behalf of the CELLS - Commissioning Team. ALBA Synchrotron Light Source Commissioning. In *2nd International Particle Accelerator Conference, San Sebastián, Spain*, September 2011.

[Eng18] R. S. Engelschall. *Hierarchical User Interface Component Architecture*. PhD thesis, Augsburg University, Augsburg, Germany, 2018.

[ER04] G. Erkan and D. R. Radev. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *J. Artif. Intell. Res.*, 22:457–479, 2004.

[ESS] European Spallation Source (ESS) Homepage. Available: https://europeanspallationsource.se/.

[FCC] FCC Homepage. Available: https://fcc.web.cern.ch.

[FEDB00] F. Fonseca, M. Egenhofer, C. Davis, and K. Borges. Ontologies and knowledge sharing in urban GIS. *Environ. Urban. Syst.*, 24(3):232–251, 2000.

[Fin07] D. J. S. Findlay. ISIS - pulsed neutron and muon source. In *2007 IEEE Particle Accelerator Conference (PAC),Albuquerque, NM, USA*, June 2007.

[G+03] A. Gőtz et al. Tango a CORBA Based Control System. In *9th Int. Conf. on Accelerator and Large Experimental Control Systems, Gyeongju, Korea*, October 2003.

[Gan18] F. Gandon. A Survey of the First 20 Years of Research on Semantic Web and Linked Data. *Ingénierie des systèmes d'information*, 23:11–38, 08 2018.

[Gar17] D. Garijo. WIDOCO: A Wizard for Documenting Ontologies. In C. d'Amato, M. Fernandez, V. Tamma, F. Lecue, P. Cudré-Mauroux, J. Sequeda, C. Lange, and J. Heflin, editors, *The Semantic Web – ISWC 2017*, pages 94–102, Cham, 2017. Springer International Publishing.

[GARW19] B. Gkotse, S. Azimova, F. Ravotti, and H. Wilkens. AIDA-2020 Test Beam facilities website and database user manual. Technical Report AIDA-2020-NOTE-2020-001, CERN, Geneva, Dec 2019. Available: http://cds.cern.ch/record/2706474.

[GAT] GATE Homepage. Available: https://www.helmholtz-berlin.de/user/gate.

[GBC+17] B. Gkotse, M. Brugger, P. Carbonez, S. Danzeca, A. Fabich, R. G. Alia, M. Glaser, G. Gorine, M. R. Jaekel, I. M. Suau, G. Pezzullo, F. Pozzi, F. Ravotti, M. Silari, and M. Tali. Irradiation Facilities at CERN. In *2017 17th European Conference on Radiation and Its Effects on Components and Systems (RADECS)*, pages 1–7. IEEE, 2017. Available: https://ieeexplore.ieee.org/document/8696163.

[GG16] B. Gkotse and G. Gorine. Specifications for IRRAD sample & user management system and online database fixed, AIDA-2020-MS16, CERN, 2016. Available: https://cds.cern.ch/record/2159521.

[GG17] B. Gkotse and G. Gorine. AIDA-2020 Irradiation Facilities Website and Database User Manual. Technical Report AIDA-2020-NOTE-2017-002, CERN, Geneva, Feb 2017. Available: http://cds.cern.ch/record/2244674.

[GGJ+17] B. Gkotse, M. Glaser, P. Jouvelot, E. Matli, G. Pezzullo, and F. Ravotti. Towards a Unified Environmental Monitoring, Control and Data Management System for Irradiation Facilities: the CERN IRRAD Use Case. In *2017 17th European Conference on Radiation and Its Effects on Components and Systems (RADECS)*, pages 1–8. IEEE, 2017. Available: https://ieeexplore.ieee.org/document/8696209.

[GGMR15] B. Gkotse, M. Glaser, M. Moll, and F. Ravotti. IRRAD: The New 24 GeV/c Proton Irradiation Facility at CERN. In *Proceedings, 12th International Topical Meeting on Nuclear Applications of Accelerators (AccApp 2015): Washington, D.C., United States, November 10-13, 2015*, pages 182–187, 2015. Available: http://accapp15.org/wp-content/data/62105-ans-1.3388085/t006-1.3388638/f006-1.3388639/15627-1.3388652.html.

[Gir19] J. Girard. How to Contrast Background and Foreground Colors in Web Design, 2019. Available: https://www.lifewire.com/contrasting-foreground-background-colors-406136.

[GJPR19] B. Gkotse, P. Jouvelot, G. Pezzullo, and F. Ravotti. The IRRAD Data Manager (IDM). In *Proceedings of ICALEPCS2019, New York, NY, USA*, pages 318–322. JACoW Publishing, 2019. Available: http://icalepcs2019.vrws.de/papers/mopha048.pdf.

[GJR19a] B. Gkotse, P. Jouvelot, and F. Ravotti. Automatic Web Application Generation from an Irradiation Experiment Data Management Ontology (IEDM). In *17th International Conference on Accelerator and Large Experimental Physics Control Systems, ICALEPCS 2019*, pages 687–693. JACoW Publishing, 2019. Available: http://icalepcs2019.vrws.de/papers/tubpl01.pdf.

[GJR19b] B. Gkotse, P. Jouvelot, and F. Ravotti. IEDM: An Ontology for Irradiation Experiments Data Management. In *The Semantic Web: ESWC 2019 Satellite Events*, pages 80–83, Cham, 2019. Springer International Publishing.

[GKG] Google Knowledge Graph. Available: developers.google.com/knowledge-graph/.

[GL16] A. Grover and J. Leskovec. node2vec: Scalable Feature Learning for Networks. *KDD : proceedings. International Conference on Knowledge Discovery & Data Mining*, 2016:855–864, 2016.

[GMR13] M. Glaser, M. Moll, and F. Ravotti. Installation of new equipment: Movable irradiation tables operational, AIDA-MS31, 2013. Available https://cds.cern.ch/record/1594787.

[Gol08] J. Golbeck. Linking Social Networks on the Web with FOAF: A Semantic Web Case Study. In *Twenty-Third AAAI Conference on Artificial Intelligence*, volume 2, pages 1138–1143, 01 2008.

[GPM+18] G. Gorine, G. Pezzullo, I. Mandic, A. Jazbec, L. Snoj, M. Capeans, M. Moll, D. Bouvet, F. Ravotti, and J. Sallese. Ultrahigh Fluence Radiation Monitoring Technology for the Future Circular Collider at CERN. *IEEE Transactions on Nuclear Science*, 65(8):1583–1590, Aug 2018.

[Gru93] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199 – 220, 1993. Available: http://www.sciencedirect.com/science/article/pii/S1042814383710083.

[GS59] J. P. Guilford and P. C. Smith. A System of Color-Preferences. *The American Journal of Psychology*, 72(4):487–502, 1959. Available: http://www.jstor.org/stable/1419491.

[Har] J. Harrison, Sensory Perception and Interaction Research Group, University of British Columbia. White background is better than black background here is why. Available: https://www.dpreview.com/forums/thread/3997600.

[HEA] Healthy seeds - treated environmentally friendly. Available: https://www.fraunhofer.de/en/press/research-news/2013/february/healthy-seeds-treated-environmentally-friendly.html.

[HH04] R. Hall and P. Hanna. The impact of web page text-background colour combinations on readability, retention, aesthetics and behavioural intention. *Behaviour & IT*, 23:183–195, 05 2004.

[HKP17] M. Hitz, T. Kessel, and D. Pfisterer. Towards Sharable Application Ontologies for the Automatic Generation of UIs for Dialog based Linked Data Applications. In *5th International Conference on Model-Driven Engineering and Software Development (MODELSWARD 2017)*, pages 65–77, 02 2017. 10.5220/0006137600650077.

[HS19] A. Harms and S. Spinder. A comprehensive view of machine learning techniques for CPI production. *Statistics Netherlands*, nov 2019. Available:

https://www.cbs.nl/en-gb/background/2019/47/machine-learning-techniques-for-cpi-production.

[HUH10] A. Haller, J. Umbrich, and M. Hausenblas. RaUL: RDFa User Interface Language – A Data Processing Model for Web Applications. In L. Chen, P. Triantafillou, and T. Suel, editors, *Web Information Systems Engineering – WISE 2010*, pages 400–410, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[HZB] Helmholtz-Zentrum Berlin (HZB) Homepage. Available: https://www.helmholtz-berlin.de/.

[ISI] ISIS Neutron and Muon Source. Available: https://www.isis.stfc.ac.uk/.

[JH74] K. W. Jacobs and F. E. Jr. Hustmyer. Effects of Four Psychological Primary Colors on GSR, Heart Rate and Respiration Rate. *Perceptual and Motor Skills*, 38(3):763–766, 1974. PMID: 4842431 Available: https://doi.org/10.2466/pms.1974.38.3.763.

[JHC+19] K. Janowicz, A. Haller, S. J.D. Cox, D. L. Phuoc, and M. Lefrançois. SOSA: A lightweight ontology for sensors, observations, samples, and actuators. *Journal of Web Semantics*, 56:1 – 10, 2019. Available: http://www.sciencedirect.com/science/article/pii/S1570826818300295.

[JIN] Jinja2 Templates. Available: https://jinja.palletsprojects.com.

[Jin18] Zhi Jin. Chapter 4 - Ontology-Oriented Interactive Environment Modeling. In Zhi Jin, editor, *Environment Modeling-Based Requirements Engineering for Software Intensive Systems*, pages 45 – 67. Morgan Kaufmann, Oxford, 2018. Available: http://www.sciencedirect.com/science/article/pii/B9780128019542000042.

[JLA] Jefferson Lab Homepage. Available: https://www.jlab.org/.

[JQU] JQuery Library. Available: https://jquery.com.

[KBMA+13] M. P. Kepinski, C. Bruno, L. Delamare, S. Mallon Amerigo, P. Martel, S. Petit, T. Schmittler, M. J S Tavlet, and D. Widegren. TREC: Traceability of Radioactive Equipment at CERN. In *4th International Particle Accelerator Conference*, page THPEA042. 3 p, 2013. Available: https://cds.cern.ch/record/2010978.

[KMBS12] S. Kayali, W. McAlpine, H. Becker, and L. Scheick. Juno radiation design and implementation. In *2012 IEEE Aerospace Conference*, pages 1–7, March 2012.

[KUB] Kubernetes Homepage. Available: https://kubernetes.io.

[Kul87] S. Kullback. Letters to the Editor: The Kullback–Leibler distance. *The American Statistician*, 41(4):338–341, 1987. Available: https://doi.org/10.1080/00031305.1987.10475510.

[L+11] M. Lindroos et al. The European Spallation Source. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 269:3258–3260, 2011.

[Lam17] JB Lamy. Owlready: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies. *Artificial Intelligence in Medicine*, 80, 08 2017.

[LHB10] W. Lidwell, K. Holden, and J. Butler. *Universal Principles of Design*. Beverly, Massachusetts, USA: Rockport, 2010.

[Li08] Q. J. Li. *Design and implementation of a user-adaptive website with information palettes*. PhD thesis, Massachusetts Institute of Technology. Department of Electrical Engineering and Computer Science, 2008.

[LOV] Linked Open Vocabularies (LOV). Available: https://lov.linkeddata.es/dataset/lov.

[LPW17] A. Lossent, A. Rodriguez Peon, and A. Wagner. PaaS for web applications with OpenShift origin. *Journal of Physics: Conference Series*, 898:082037, oct 2017. Available: https://doi.org/10.1088%2F1742-6596%2F898%2F8%2F082037.

[LR11] C. Leroy and PG Rancoita. *Principles of Radiation Interaction in Matter and Detection*. WORLD SCIENTIFIC, 3rd edition, 2011. Available: https://www.worldscientific.com/doi/abs/10.1142/8200.

[LUI] Linked Open Vocabularies (LOV) User Interface ontology. Available: https://lov.linkeddata.es/dataset/lov/vocabs/ui.

[MAN] Mantid homepage. Available: https://www.mantidproject.org.

[Mar13] P. Martel. An Equipment Hub for Managing a Small Town and a Complex Machine THPEA043. In *4th International Particle Accelerator Conference, IPAC2013, Shanghai, China, 2013*, pages 3237–3239, 2013. Available: http://accelconf.web.cern.ch/AccelConf/IPAC2013/papers/thpea043.pdf?n=IPAC2013/papers/thpea043.pdf.

[MAX] MAX IV Homepage. Available: https://www.maxiv.lu.se/.

[MCCD13] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. *CoRR*, abs/1301.3781, 2013.

[MH08] L. van der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[MIN] Minimal Reading Mode. Available: https://chrome.google.com/webstore/detail/minimal-reading-mode/peoapnglceoafojobbkpohnojniabmkd?hl=en.

[MLA18] K. Mahmudi, M. M. Inggriani Liem, and S. Akbar. Ontology to relational database transformation for web application development and maintenance. *Journal of Physics: Conference Series*, 971:012031, mar 2018.

[Mob16] E. Mobs. The CERN accelerator complex. Complexe des accélérateurs du CERN. Jul 2016. General Photo.

[MPT15] M. A. Musen and the Protégé Team. The Protégé Project: A Look Back and a Look Forward. *AI Matters*, 1(4):4–12, 2015. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4883684/.

[MRZP08] J. Malone, T. Rayner, X. Zheng, and H. Parkinson. Developing an application focused experimental factor ontology: embracing the obo community. 01 2008.

[MSJ19] B. Massoni Sguerra and P. Jouvelot. "An Unscented Hound for Working Memory" and the Cognitive Adaptation of User Interfaces. In *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '19, page 78–85, New York, NY, USA, 2019. Association for Computing Machinery. Avaialable: https://doi.org/10.1145/3320435.3320443.

[Nie] J. Nielsen. Children's websites: Usability issues in designing for kids. nielsen norman group. Available: https://www.nngroup.com/articles/childrens-websites-usability-issues/.

[Nie99] J. Nielsen. *Designing Web Usability: The Practice of Simplicity*. New Riders Publishing, Indianapolis, 7th edition, 1999. Available: https://www.nngroup.com/books/designing-web-usability/.

[NM01] N. F. Noy and D. L. McGuinness. Ontology development 101: A guide to creating your first ontology. Technical report, Stanford Knowledge Systems Laboratory, March 2001. Available: http://www-ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness-abstract.html.

[Noy04] N. F. Noy. Semantic Integration: A Survey Of Ontology-Based Approaches. *SIGMOD Record*, 33:65–70, 2004. Available: https://doi.org/10.1145/1041410.1041421.

[ORA] Oracle Database Service. Available: https://cern.service-now.com/service-portal/service-element.do?name=oracle-database-service.

[OWL] Web Ontology Language (OWL). https://www.w3.org/OWL/.

[OXF] Oxford English Dictionary. Available: https://www.oed.com/view/Entry/131551?redirectedFrom=ontology.

[Pau17] H. Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, 8 (3):489–508, 2017.

[Per11] C. Perciballi. Real-time Adaptive Morphing Website Modeled Per User and Optimized Across Users, 03 2011.

[PFOL18] H. J. Pandit, K. Fatema, D. O'Sullivan, and D. Lewis. GDPRtEXT - GDPR as a Linked Data Resource. In A. Gangemi, R. Navigli, ME Vidal, P. Hitzler, R. Troncy, L. Hollink, A. Tordai, and M. Alam, editors, *The Semantic Web*, pages 481–495, Cham, 2018. Springer International Publishing.

[PNL02] A. Pease, I. Niles, and J. Li. The Suggested Upper Merged Ontology: A Large Ontology for the Semantic Web and its Applications. In *AAAI*, 01 2002.

[PRS15] J. Preece, Y. Rogers, and H. Sharp. *Interaction Design: Beyond Human-Computer Interaction*. John Wiley & Sons, 2015.

[PSI] Paul Scherrer Institut (PSI) Homepage. Available: www.psi.ch.

[PVG+11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011. Available: http://jmlr.org/papers/v12/pedregosa11a.html.

[PWNG14] K. Pernice, K. Whitenton, J. Nielsen, and Nielsen Norman Group. *How People Read on the Web: The Eyetracking Evidence*. Nielsen Norman Group, 2014. Available: https://www.nngroup.com/reports/how-people-read-web-eyetracking-evidence/.

[Rav06] F. Ravotti. *Development and Characterisation of Radiation Monitoring Sensors for the High Energy Physics Experiments of the CERN LHC Accelerator*. PhD thesis, Montpellier University, 2006. Presented on 17 Nov 2006, Available: http://cds.cern.ch/record/1014776.

[Rav18] F. Ravotti. Dosimetry Techniques and Radiation Test Facilities for Total Ionizing Dose Testing. *IEEE Transactions on Nuclear Science*, 65(8):1440–1464, Aug 2018.

[RB13] K. Reinecke and A. Bernstein. Knowing What a User Likes: A Design Science Approach to Interfaces That Automatically Adapt to Culture. *MIS Q.*, 37(2):427–454, June 2013. Available: https://doi.org/10.25300/MISQ/2013/37.2.06.

[RDF] Resource Description Framework (RDF). Available: https://www.w3.org/RDF/.

[RDS] Resource Description Framework Schema (RDFS). Available: https://www.w3.org/RDF/.

[Row09] D. Rowse. Light or Dark Blog Backgrounds?, 2009. Available: https://problogger.com/light-or-dark-blog-backgrounds-poll-results/.

[RP] Radiation Protection Homepage. Available: https://hse.cern/services-support/radiation-protection.

[RPKC11] C. Roussey, F. Pinet, M. A. Kang, and O. Corcho. *An Introduction to Ontologies and Ontology Engineering*, pages 9–38. Springer London, London, 2011.

[RRN+19] P. Ristoski, J. Rosati, T. Di Noia, R. De Leone, and H. Paulheim. RDF2Vec: RDF graph embeddings and their applications. *Semantic Web*, 10(4):721–752, 2019. Available: https://madoc.bib.uni-mannheim.de/50498/.

[RvAT13] H. Rijgersberg, M. van Assem, and J. Top. Ontology of Units of Measure and Related Concepts. *Semant. web*, 4(1):3–13, January 2013. Available: http://dl.acm.org/citation.cfm?id=2595053.2595055.

[S+07] B. Smith et al. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat Biotechnol.*, 25:1251, November 2007. Available: https://doi.org/10.1038/nbt1346.

[SB10] S. Sayago and J. Blat. Telling the story of older people e-mailing: An ethnographical study. *International Journal of Human-Computer Studies*, 68(1):105 – 120, 2010. Available: http://www.sciencedirect.com/science/article/pii/S107158190900158X.

[SG11] G. Shani and A. Gunawardana. *Evaluating Recommendation Systems*, pages 257–297. Springer US, Boston, MA, 2011. Available: https://doi.org/10.1007/978-0-387-85820-3_8.

[SGBST17] P. J. Seweryn, M. Gonzalez-Berges, J. B. Schofield, and F. M. Tilaro. Data Analytics Reporting Tool for CERN SCADA Systems. In *Proceedings of ICALEPCS2017, Barcelona, Spain*, 2017.

[Sha11] S. K. Shahzad. Ontology-based User Interface Development: User Experience Elements Pattern. *J. UCS*, 17:1078–1088, 01 2011.

[SIR] SIRIUS: Soft Interfaces and Resonant Investigation on Undulator Source - SIRIUS Homepage. Available: https://www.synchrotron-soleil.fr/en/beamlines/sirius.

[SK07] L. Soldatova and R. King. An ontology of scientific experiments. *Journal of the Royal Society, Interface / the Royal Society*, 3:795–803, 01 2007.

[SL19] R.J. Slominski and T. Larrieu. Web Extensible Display Manager 2. In *17th Int. Conf. on Accelerator and Large Experimental Control Systems, New York, NY, USA*, October 2019.

[SLR] SOLARIS Synchrotron Homepage. Available: https://synchrotron.uj.edu.pl/.

[Smi04] B. Smith. Ontology. In Luciano Floridi, editor, *Blackwell Guide to the Philosophy of Computing and Information*, pages 155–166. Oxford: Blackwell, 2004.

[SNL62] C. F. Schmidt, W. K. Nank, and R. V. Lechowich. Radiation Sterilization of Food. *Journal of Food Science*, 27(1):77–84, 1962. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-2621.1962.tb00062.x.

[SOL] Soleil synchrotron homepage. Available: https://www.synchrotron-soleil.fr/.

[SP17] J. Szota-Pachowicz. Building a synchrotron ontology: An analysis of a synchrotron control system in a collaborative environment. *Computer Science*, 18:53, 01 2017.

[SPG+07] E. Sirin, B. Parsia, B. Cuenca Grau, A. Kalyanpur, and Y. Katz. Pellet: A practical owl-dl reasoner. *Journal of Web Semantics*, 5(2):51 – 53, 2007. Software Engineering and the Semantic Web. Available: http://www.sciencedirect.com/science/article/pii/S1570826807000169.

[SS17] J. Shariat and C. S. Saucier. *Tragic Design: The Impact of Bad Product Design and How to Fix It*. O'Reilly Media, Inc., 1st edition, 2017. Available: https://www.tragicdesign.com/.

[Sta] ASTM Standard. E170, Standard Terminology Relating to Radiation Measurements and Dosimetry. *Annual Book of ASTM Standards*, 12:19103–1187.

[STM+18] A. A. Salatino, T. Thanapalasingam, A. Mannocci, F. Osborne, and E. Motta. The Computer Science Ontology: A Large-Scale Taxonomy of Research Areas. In D. Vrandečić, K. Bontcheva, M. C. Suárez-Figueroa, V. Presutti, I. Celino, M. Sabou, L-A Kaffee, and E. Simperl, editors, *The Semantic Web – ISWC 2018*, pages 187–205, Cham, 2018. Springer International Publishing.

[SUI] Semantic UI framework. Available: https://semantic-ui.com.

[SUN] Soleil user net set homepage. Available: http://sunset.synchrotron-soleil.fr/sun.

[SUO] Semantic User Interface ontology. Available: https://old.datahub.io/dataset/ui.

[Szy17] T. Szymocha,*et al.* SOLARIS Digital User Office. In *16th Int. Conf. on Accelerator and Large Experimental Control Systems, Barcelona, Spain*, October 2017.

[T+18] Tanabashi et al. Review of Particle Physics. *Phys. Rev. D*, 98:030001, Aug 2018. Available: https://link.aps.org/doi/10.1103/PhysRevD.98.030001.

[TAB+13] Y. Tzitzikas, C. Alloca, C. Bekiari, Y. Marketakis, P. Fafalios, M. Doerr, N. Minadakis, T. Patkos, and L. Candela. Integrating Heterogeneous and Distributed Information about Marine Species through a Top Level Ontology. In *Proceedings of the 7th Metadata and Semantic Research Conference (MTSR'13)*, Thessaloniki, Greece, November 2013.

[TBD] Test Beam Database Homepage. Available: http://cern.ch/tbdb.

[TOR] Tornado Documentation. Available: http://www.tornadoweb.org.

[TR06] D. Taniar and J. W. Rahayu. *Web semantics and ontology*. Idea Group Publishing, USA, 2006.

[Vje17] V. Vjetkovic. Web Physics Ontology: Online interactive symbolic computation in physics. In *4th Experiment International Conference, IEEE, Faro, Portugal*, 2017.

[VSA+20] G. Vandewiele, B. Steenwinckel, T Agozzino, M. Weyns, P Bonte, F Ongenae, and F De Turck. pyRDF2Vec: A python library for RDF2Vec. IDLab, 2020.

[W3C] Semantic Web (W3C) Homepage. Available: https://www.w3.org/standards/semanticweb/.

[WCC] SIMATIC WinCC. Available: https://new.siemens.com/global/en/products/automation/industry-software/automation-software/scada.html.

[Wei] Hyperplane. From MathWorld–A Wolfram Web Resource. https://mathworld.wolfram.com/Hyperplane.html.

[Wil66] G. D. Wilson. Arousal properties of red versus green. *Perceptual and Motor Skills*, 23(3, PT. 1):947–949, 1966.

[WJV] WebJive Repository. Available: https://gitlab.com/MaxIV/webjive.

[WLA18] V. Wiens, S. Lohmann, and S. Auer. Webvowl editor: Device-independent visual ontology modeling. In *17th Int. Semantic Web Conference (ISWC 2018) Posters & Demonstrations, Industry and Blue Sky Ideas Tracks*, 2018.

[WLW04] TF Wu, CJ Lin, and R. C. Weng. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5(Aug):975–1005, 2004.

[Yu13] CH Yu. *Web page enhancement on desktop and mobile browsers*. PhD thesis, Massachusetts Institute of Technology. Department of Electrical Engineering and Computer Science, 2013.

Glossary

Angular is a TypeScript-based open-source web-application framework led by the Angular Team at Google and by a community of individuals and corporations. It is used to build applications with the web and combines declarative templates, dependency injection and end-to-end tooling. 29

Cold Start is the problem that occurs in recommender systems when there are not enough user ratings or data in order to classify a specific item. 43, 53

composites are high-level fragments of a user interface (e.g., panel, interacting dialog, etc.. 35

GraphQL is a query language for APIs and a runtime for fulfilling queries with existing data. Available: https://graphql.org/. 29

Java is a general-purpose programming language that is class-based and object-oriented. 28

JavaScript is a language for client-side web development. Available: https://www.javascript.com/ . 29

Mantid provides a framework that supports high-performance computing and visualization of material science data. 30

MySQL is an open-source relational database-management system. 29

Oracle is a proprietary multi-model database management system produced and marketed by Oracle Corporation. 28

React is a JavaScript library for building user interfaces. Available: https://reactjs.org/. 29

Spring Boot is an open-source Java-based framework used to create micro Services. It is developed by Pivotal Team and is used to build stand-alone and production-ready Spring applications. 28

WebSocket is a computer communication protocol providing full-duplex communication channels over a single TCP connection. 31

widgets are fragments or a set of fragments of a user interface (e.g., form, table, list, etc.) . 34, 35

Acronyms

AI The modern definition of Artificial Intelligence (or AI) is "the study and design of intelligent agents", where an intelligent agent is a system that perceives its environment and takes actions that maximises its chances of success. 11

CRUD Basic operations that Create, Read, Update and Delete data. 34

CSV-LD CSV-LD is defined within a standard JSON-LD document where string values may take the form of a pattern. A pattern is composed of one or more field references, which are used to replace the field reference with the value of the field for each record.

DevOps Practices combining software development (Dev) and information-technology operations (Ops) aiming at shortening the system development life cycle of software and proving continuous delivery with high software quality. 65

DUO Digital Users Office. 28

EPICS Experimental Physics and Industrial Control System (EPICS) is a set of open-source software tools, libraries and applications developed collaboratively and used worldwide to create distributed soft-real-time control systems for scientific instruments such as particle accelerators, telescopes and other large scientific experiments. 29

EPP Experimental Particle Physics. 2, 6

GRDDL Gleaning Resource Descriptions from Dialects of Languages.

GUI Graphical User Interface. 30

HTML HyperText Markup Language.

HTTP HyperText Transfer Protocol.

IE Irradiation Experiment.

IRI Internationalized Resource Identifier.

JSON JavaScript Object Notation.

JSON LD JavaScript Object Notation for Linked Data.

MVC Model-View-Controller architecture. 35

NLP Natural Language Processing is a specific domain of Computer Science aiming to understand and process human-created text. 3, 7, 13

N-Quad N-Quads is a line-based, plain text format for encoding an RDF dataset. The main distinction from N-Triples is that N-Quads allow encoding multiple graphs.

N-Triple N-Triples is a line-based, plain text format for representing RDF/XML [RDFMS] data.

OWL Web Ontology Language.

PaaS Platform-as-a-Service. 65

PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open-source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. 28

PyQt PyQt is a Python binding for the Qt cross-platform C++ framework. 30

R2RML Language for expressing customised mappings from relational databases to RDF datasets.

RDF Resource Description Framework. 34

RDFa Resource Description Framework in Attributes. 34

RDFS Resource Description Framework Schema.

RGB Every colour can be described as a combination of the 3 three colours Red, Green and Blue. 39

SCADA Supervisory Control and Data Acquisition is a control system architecture that uses computers, networked data communications and graphical user interfaces to monitor and control large processes. 31

SHACL Shapes Constraint Language.

SPARQL SPARQL Protocol And RDF Query Language.

SSO Single Sign-On. 65

TriG TriG is a textual syntax for RDF that allows an RDF dataset to be completely written in a compact and natural text form, with abbreviations for common usage patterns and datatypes. TriG is an extension of the Turtle format.

Turtle Turtle provides levels of compatibility with the N-Triples [N-TRIPLES] format as well as the triple pattern syntax of SPARQL.

UI User Interface.

URI Uniform Resource Identifier.

URL Uniform Resource Locator.

UX User Experience. 35

XML Extensible Markup Language. 16

Appendix A

Irradiation Facilities Database and Website

This website contains the basic functionalities for users seeking information about irradiation facilities and for coordinators who want to promote their facilities and capabilities to the HEP community.

A.1 Functionalities

The main functionalities of the irradiation facilities website are:

- display of the CERN and worldwide facilities;
- search feature by country, source or radiation field;
- irradiation facilities worldwide map;
- update of irradiation facilities information (only by the facility coordinators);
- creation of a new irradiation facility entry (only by the facility coordinator);
- self-maintenance (regular reminders sent to the facility coordinators in order to keep the information up to date).

This website and its content are visible to anybody. However, a user, in order to add or update an irradiation facility entry, has to log in through the CERN Authentication System (SSO).

Common use cases for a facility coordinator are: creating a new entry, and editing or deleting a current entry in the database, through the following steps. The facility coordinator first logs-in into the irradiation facilities website and has the permission to see and edit the facility under his/her responsibility. In the case that he/she needs to add a new irradiation facility, he/she can click

the button `Add Facility` and fill in the corresponding web-form that appears. Otherwise, for updating an existing facility data, the coordinator will see which entries he/she can edit. Once the facility coordinator presses `Confirm`, the data are saved but are no longer publicly visible in the website until the irradiation-facilities-database administrator checks and approves the changes. When the data have been approved, the entry of the specific facility is again available online. In the case that a facility coordinator needs to delete a facility entry (it may no longer exist), a specific delete request is sent via e-mail to the database administrator. In this case, the administrator makes this entry no longer visible on the public website, but he/she does not delete it from the database, since the information must be archived.

A.2 Database Content

Every irradiation facility has its own entry, which includes the following information (Fig. A.1):

- facility coordinator contact information;
- institute/ organisation details;
- facility data;
- irradiation conditions;
- safety
- accessibility.

Figure A.1 shows all the details provided to the users about an irradiation facility, which in this case is the IRRAD facility.

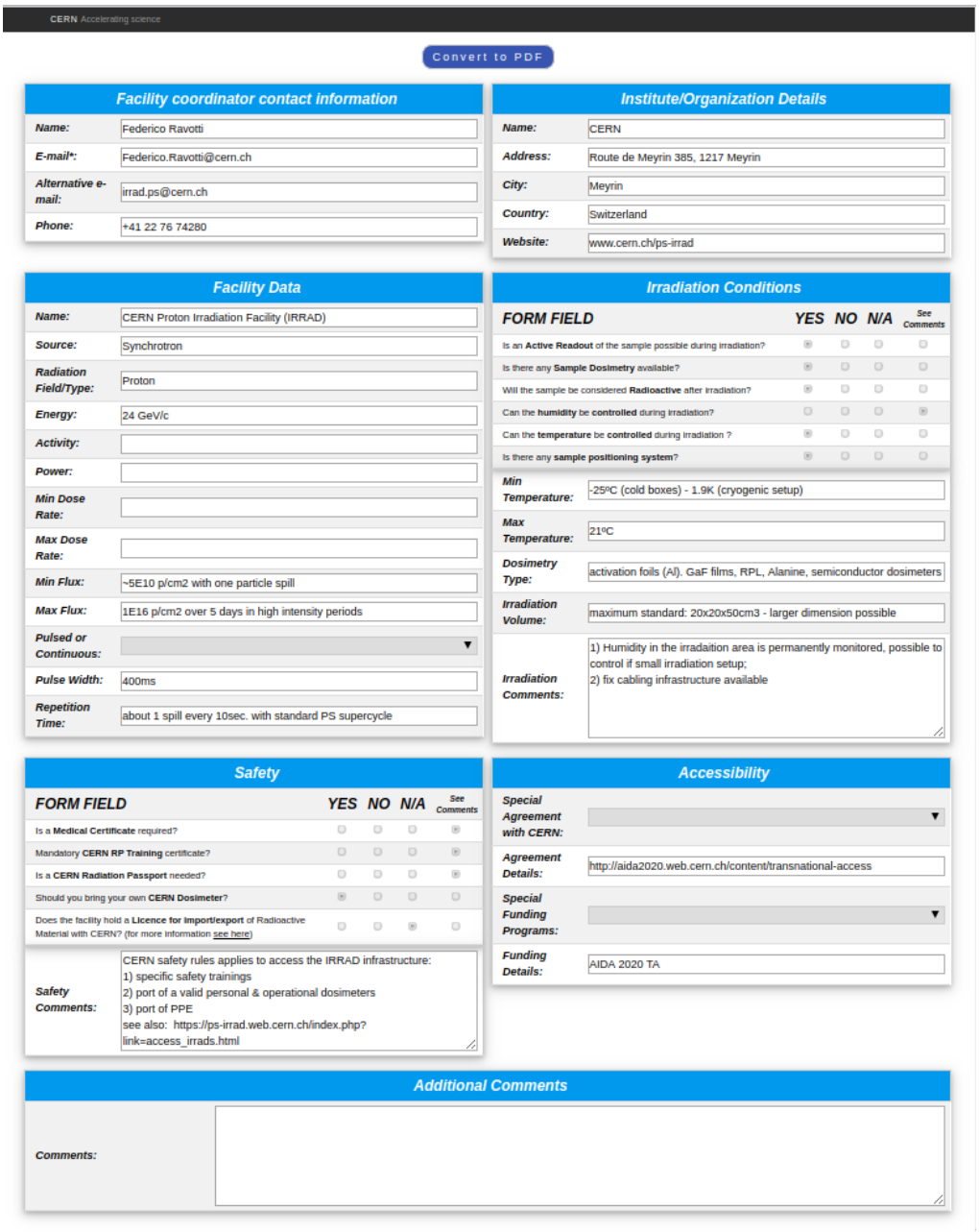


Figure A.1: Screenshot of the irradiation facility details provided to users.

Appendix B

Test Beam Facilities Database and Website

The Test beam facilities website and database is a customised and updated version of the Irradiation facilities database and website, customised according to the test-beams community's requirements.

B.1 Functionalities

The functionalities are similar to the ones of the Irradiation facilities website; however, there are some changes and additions:

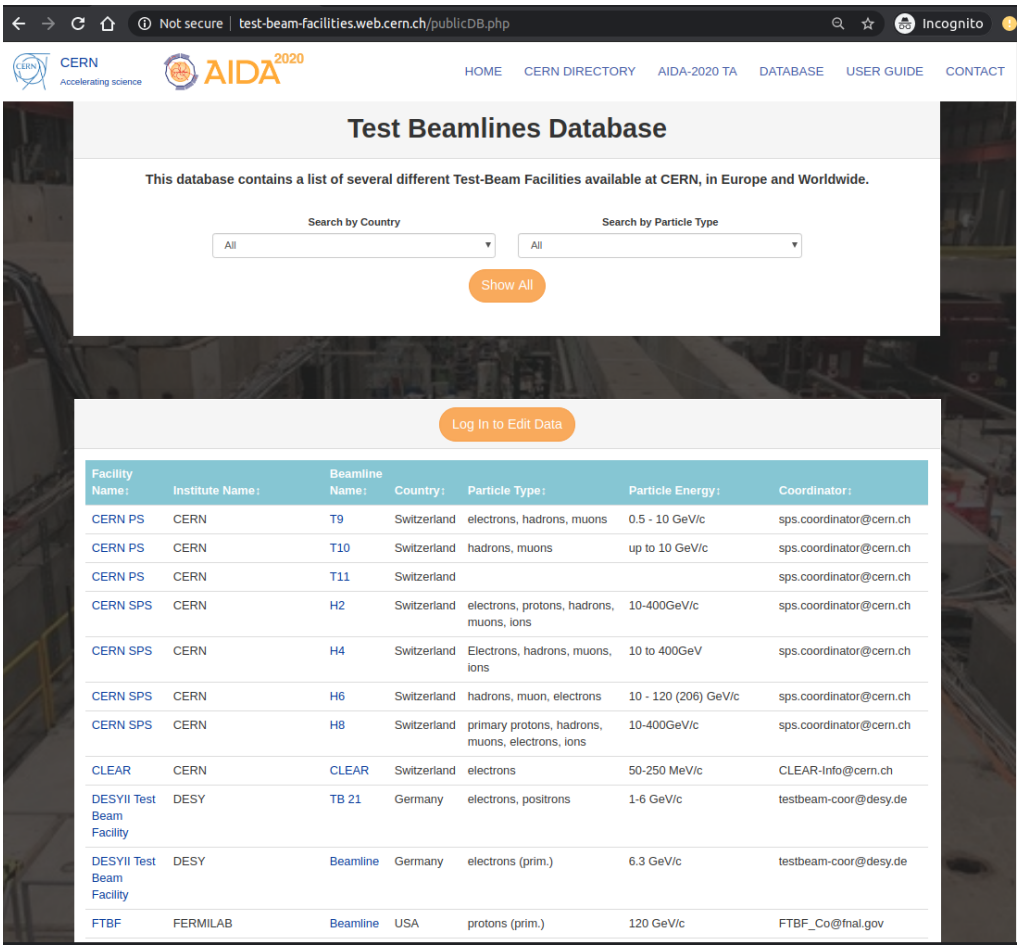
- search feature by country or radiation field;
- information about the facilities and beam lines separated in two different views, since more than one beam line can correspond to one test-beam facility.

B.2 Database Content

As for an irradiation facility, the data included in the database are related to:

- facility coordinator contact information;
- institute/ organisation details;
- facility data;
- irradiation conditions;
- safety
- accessibility.

Figure B.2 shows all the information contained in these fields, which are related to a particular test-beam facility; in this case the Super Proton Synchrotron facility is taken as an example.



Facility Name	Institute Name	Beamline Name	Country	Particle Type	Particle Energy	Coordinator
CERN PS	CERN	T9	Switzerland	electrons, hadrons, muons	0.5 - 10 GeV/c	sps.coordinator@cern.ch
CERN PS	CERN	T10	Switzerland	hadrons, muons	up to 10 GeV/c	sps.coordinator@cern.ch
CERN PS	CERN	T11	Switzerland			sps.coordinator@cern.ch
CERN SPS	CERN	H2	Switzerland	electrons, protons, hadrons, muons, ions	10-400GeV/c	sps.coordinator@cern.ch
CERN SPS	CERN	H4	Switzerland	Electrons, hadrons, muons, ions	10 to 400GeV	sps.coordinator@cern.ch
CERN SPS	CERN	H6	Switzerland	hadrons, muon, electrons	10 - 120 (206) GeV/c	sps.coordinator@cern.ch
CERN SPS	CERN	H8	Switzerland	primary protons, hadrons, muons, electrons, ions	10-400GeV/c	sps.coordinator@cern.ch
CLEAR	CERN	CLEAR	Switzerland	electrons	50-250 MeV/c	CLEAR-Info@cern.ch
DESYII Test Beam Facility	DESY	TB 21	Germany	electrons, positrons	1-6 GeV/c	testbeam-coor@desy.de
DESYII Test Beam Facility	DESY	Beamline	Germany	electrons (prim.)	6.3 GeV/c	testbeam-coor@desy.de
FTBF	FERMILAB	Beamline	USA	protons (prim.)	120 GeV/c	FTBF_Co@fnal.gov

Figure B.1: A screenshot from the test beam facilities and beamlines view.

In addition to the facility data, dedicated information about each beamline of a test beam facility are shown:

- beamline characteristics;
- infrastructure.

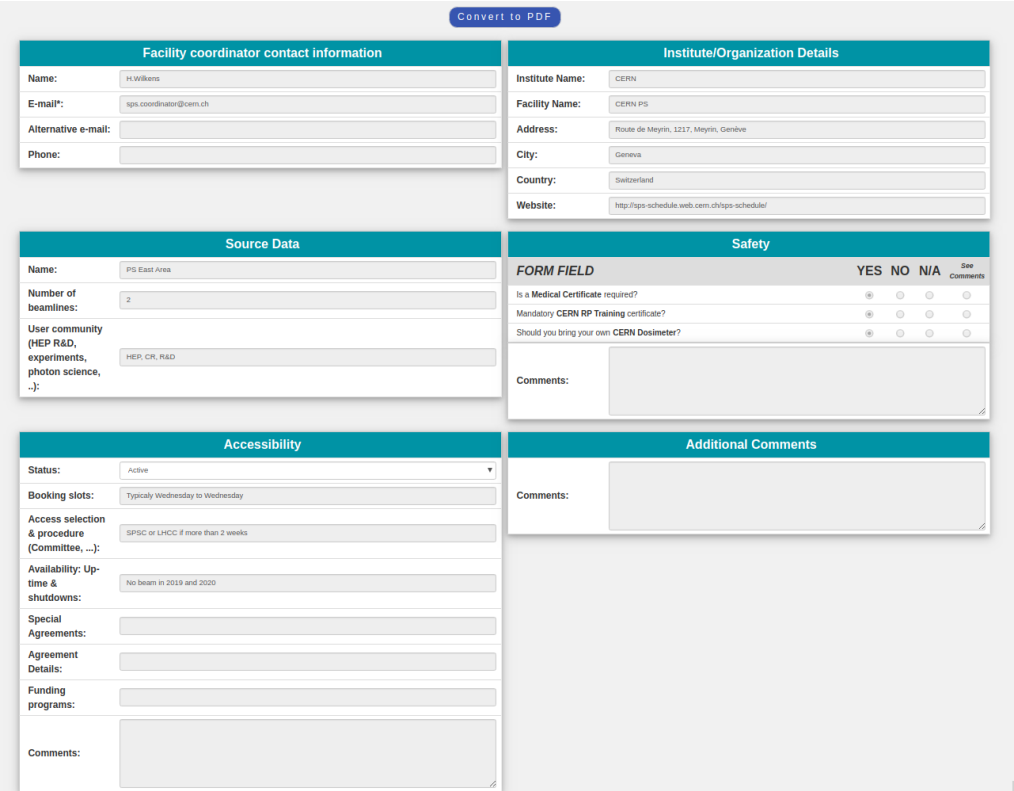


Figure B.2: A screenshot from the Super Proton Synchrotron (SPS) test beam facility data.

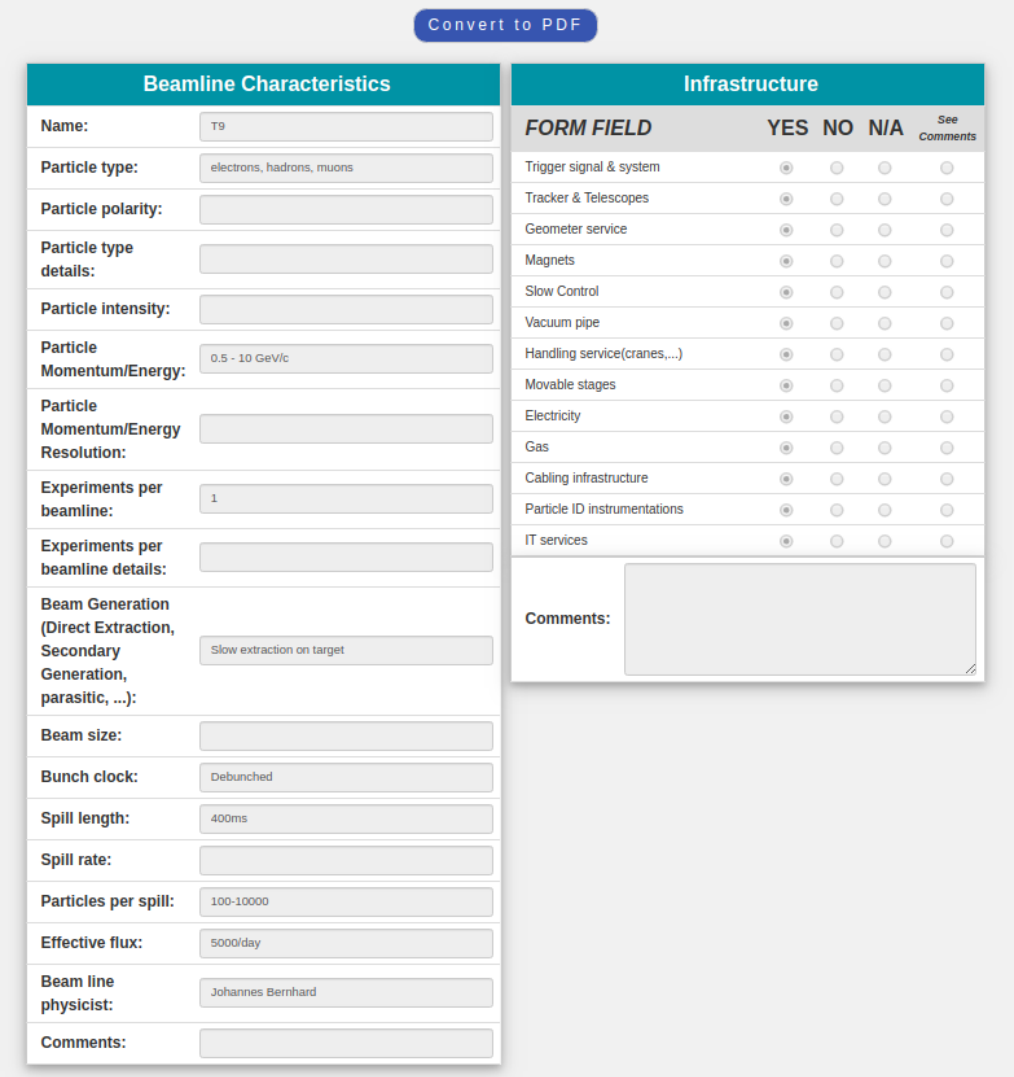


Figure B.3: A screenshot from the entry data of the T9 beamline of SPS.

Appendix C

IRRAD Data Manager (IDM) Installation

This appendix provides a guide for installing IDM on `localhost`. In order to achieve that, the following steps should be performed.

Gitlab project

Fork the gitlab project https://gitlab.cern.ch/irrad/samples-manager.git and clone it to the local PC. (On the gitlab, page instructions are given for deploying the web application of Openshift).

Conda Installation

Conda is used for resolving dependency conflicts among python packages: https://docs.conda.io/projects/conda/en/latest/user-guide/install/. Once Conda is installed, one can create an environment named `idm_env` (other names can be used) using the following command, and press `Yes` to the questions:

```
conda create -name idm_env python=3.6
```

In order to activate this environment, use one of this OS-dependent commands:

`activate idm_env`, if Windows;
`conda activate idm_env`, if Linux.

For more details on the Conda commands, see the link https://docs.conda.io/projects/conda/en/latest/user-guide/getting-started.html.

Requirements Installation

Navigate to the directory `/samples_manager` and install the `requirements.txt` file with this command:

```
pip install -r requirements.txt
```

Django commands

The first command is used for storing the model changes as migrations:

```
python manage.py makemigrations samples_manager
```

This will allow the changes to take place on the database and form the necessary tables:

```
python manage.py migrate
```

In order to run the server, use:

```
python manage.py runserver
```

RÉSUMÉ

Ce travail de thèse vise à combler le fossé entre les domaines de la sémantique du Web et de la physique des particules expérimentales. En prenant comme cas d'utilisation un type spécifique d'expérience de physique, les expériences d'irradiation utilisées pour tester la résistance des composants au rayonnement, un modèle de domaine, ce qui, dans le domaine de la sémantique du Web, est appelé ontologie, a été créé pour décrire les principaux concepts de la gestion des données des expériences d'irradiation. Puis, en s'appuyant sur ce type de formalisation, une méthodologie a été conçue pour réaliser automatiquement la génération de systèmes de gestion de données fondés sur des ontologies ; elle a été utilisée pour générer des interfaces utilisateur pour l'ontologie IEDM introduite précédemment. Dans la dernière partie de ce travail de thèse, nous nous sommes penchés sur l'utilisation des préférences d'affichage des interfaces-utilisateur (UI), stockées en tant qu'instances d'une ontologie de description d'interfaces que nous avons développée pour enrichir IEDM. Nous introduisons une nouvelle méthode d'encodage de ces données, instances d'ontologie, en tant que vecteurs de plongement ("embeddings") qui pourront être utilisés pour réaliser, à terme, des interfaces utilisateur personnalisées.

MOTS CLÉS

Web sémantique, Ontologie, Physique expérimentale des particules, Expérience d'irradiation, Gestion de données, Interface utilisateur, Application Web, Plongements de mots.

ABSTRACT

This thesis work aims at bridging the gap between the fields of Web Semantics and Experimental Particle Physics. Taking as a use case a specific type of physics experiments, namely the irradiation experiments used for assessing the resistance of components to radiation, a domain model, what in Web Semantics is called an ontology, has been created for describing the main concepts underlying the data management of irradiation experiments. Using such a formalisation, a methodology has been introduced for the automatic generation of data management systems based on ontologies and used to generate a web application for IEDM, the previously introduced ontology. In the last part of this thesis work, by the use of user-interface (UI) display preferences stored as instances of a UI-dedicated ontology we introduced, a method that represents these ontology instances as feature vectors (embeddings) for recommending personalised UIs is presented.

KEYWORDS

Web Semantics, Ontology, Experimental Particle Physics, Irradiation Experiment, Data Management, User Interface, Web Application, Embeddings.