



HAL
open science

Towards User Profile Meta-Ontology

Ankica Barisic, Marco Winckler

► **To cite this version:**

Ankica Barisic, Marco Winckler. Towards User Profile Meta-Ontology. Human-Centered Software Engineering. HCSE 2023. Lecture Notes in Computer Science. LNCS, 2023. hal-04210148v2

HAL Id: hal-04210148

<https://hal.science/hal-04210148v2>

Submitted on 6 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards User Profile Meta-Ontology

Ankica Barišić¹[0000–0001–7513–7907] and Marco Winckler¹[0000–0002–0756–6934]

Université Côte d’Azur, CNRS, Inria, I3S
{Ankica.Baristic,Marco.Winckler}@univ-cotedazur.fr

Abstract. With the rapid advancement of digital transformation, a large amount of data related to human activities is being gathered. Whilst knowing and understanding users is an essential activity for designing and evaluating interactive systems, describing users is a daunting task. This paper investigates the challenges for describing users’ profile and how using Ontology for creating and representing user profiles might be a suitable solution for ensuring interoperability and facilitating data federation. This paper proposes a solution in the form of a meta-ontology for user profiling and explores its creation, use, and effective federation. As we shall see, Ontology for describing users’ profile is not a new idea but it required an update to cope with the advent of new practices for collecting users’ data, the variety of attributes used nowadays to describe users, and the huge amount ammassed informaiton about user profiles.

Keywords: User profiling · Persona · Ontology · Modeling

1 Introduction

The ongoing global technological transformation has revolutionised our interactions with software and systems. As the user base becomes increasingly diverse, it is crucial to enhance development support for creating interactive systems that adapt to users’ varied characteristics and needs. To achieve this, a deep understanding of user mindset, interaction objectives, preferences and awareness is crucial.

Developers require approaches that enable them to empathise with and address the challenges faced by different end users, incorporating software solutions that cater to a wide range of human characteristics. Design support is also necessary, equipping developers with advanced techniques and tools to effectively address the complexities posed by diverse end users [21].

User-centred design approaches [17], considering factors like knowledge expertise and performance monitoring, have proven essential in understanding end users’ profiles. A *user profile* is the main component needed for the evaluation, personalisation and adaptation of interactive systems. It has played an active role in various domains such as healthcare sectors, banking sectors, social media, e-commerce, security, access control, and social networking [18]. Moreover, establishing a strong link between human characteristics and the application domain provides valuable insights into user behaviour and preferences [13].

With the rise of digital transformation, an abundance of data on human activities, and emotional, mental and physical states can be collected. This data, captured through mobile and wearable devices or biosensor cameras, holds immense potential. However, to harness its power, data sharing, reuse, and understanding patterns of unwanted human states are crucial. This data can be used to train AI models, adapt systems to individuals' needs, and create personalised experiences.

To effectively leverage this data and assist developers in modelling, reusing, and sharing user profiles, there is a pressing need for a general ontology for creating user profiles [26], [18]. Such an ontology offers a standardised framework for creating and representing user profiles, ensuring interoperability and facilitating data federation. Model-driven engineering is a promising approach to enable developers to design user profiles, generate software models, reuse existing user data, and receive guidance in modelling profiles from scratch.

A standardised user profile is expected to enable the implementation of interoperable adaptive systems sharing modelling information. Ontology-based user profiling is particularly vital for systems reasoning across multiple profiles, so-called social adaptive systems, or benefiting from complex inference on multiple ontologies representing different knowledge [34].

This paper presents the User Profile Meta Ontology (UPMO) as a proposed solution. In Section 2, we examine various techniques for acquiring user profiles. In Section 3, we present an overview of the existing ontologies in the field of user profile creation. Section 4 explores the effective design, utilization, and federation of UPMO. Additionally, in Section 5, we analyze the User Dimension, which serves as a guiding factor for the implementation of UPMO. Finally, in Section 6, we conclude the paper and outline our ideas for future research directions.

2 Techniques for Obtaining User Profiles

The concept of a *User profile* refers to the representation of an individual user, real or fictional, or a group of users. This profile is essential for understanding and catering to individual user needs in various applications [18]. User profiles are instantiated as a *User model*, a data structure to capture specific characteristics about an user, which serves as software component/artefact. These models can be static or dynamic, with static models representing unchanging user data and dynamic models evolving based on user interactions and interests.

To meet the demands of human-centered socio-technical systems, *highly adaptive user models*, representing one particular user, are required. These models focus on personalized solutions allowing high adaptability of the system, rather than relying solely on demographic data, like stereotype based models [34]. Gathering extensive information is crucial for highly adaptive models to reach their full potential.

Thematic analysis and content analysis are common techniques used to extract valuable insights for creating user profiles and models. The research methods to gather relevant information about the user are divided to qualitative and

quantitative. *Qualitative* methods involve understanding and interpreting subjective experiences, behaviors, and perceptions of individuals or groups. These methods focus on gathering non-numerical data through techniques such as interviews, observations, focus groups, case studies, and user storytelling. They provide detailed insights into the meanings, contexts, and interpretations of the target user. *Quantitative* research methods, such as surveys, experiments, statistical analysis, machine learning, and data mining, offer ways to analyze large datasets, identify patterns, and make predictions. Combining qualitative and quantitative methods provides a comprehensive understanding of users.

We highlight three common approaches to obtaining user profiles and models: user profiling, persona development, and role-based approach. The *role-based approach* creates user profiles based on the specific roles and tasks of users when interacting with a system. This approach focuses on understanding task sequences, interactions, and user roles in accomplishing those tasks. *User profiling* aims to create detailed representations of real users or user groups, capturing individual characteristics based on actual user data. *Persona development*, on the other hand, involves creating fictional characters that represent target user archetypes, guiding design decisions and representing user groups.

Role-based approach: In the role-based approach, user profiles are created based on the roles and tasks that users perform within the system, regardless their individuals differences. This approach focuses on capturing the user’s responsibilities, goals, and behaviors related to particular tasks or activities. The process typically begins by identifying the key tasks or activities that users engage in during their interactions with the system. To gain insights into the underlying cognitive mechanisms involved, cognitive task analysis methods are often employed [35]. These methods analyze and capture the cognitive aspects of tasks, such as problem-solving, decision-making, and information processing. Understanding how individuals think, reason, and make decisions during task execution can be leveraged to inform the design of various interventions aimed at enhancing task performance and supporting cognitive workload. Once the tasks are identified, the associated roles are defined. A uniform task meta-model is developed to facilitate the transformation of individual task models into their standardized counterparts while minimizing the loss of information [27].

Persona development: Personas are fictitious, stereotyped representations of target users that resemble real people while telling a meaningful story describing their behaviour and needs. They provide shared mental models, facilitate communication among team members, and help empathize with end-users [30]. Personas are useful for segmenting diverse online audiences and integrating human aspects into the development process. They are commonly used in requirements engineering, often represented as text-based descriptions obtained through qualitative approaches like focus groups and workshops [24]. Personas enhance stakeholder satisfaction, prioritize human-centric design, support developers, and are often combined with scenarios to depict user behaviour. Salminen et al. [31] investigated the application of personas in different domains, highlighting their predominant usage in software development (40% of cases) and other fields such

as healthcare, higher education, marketing, and robotics. Personas were found to be valuable in organizing user information, and adapting to both short-term tasks and long-term goals. They also proved useful in expert verification during testing, usability testing, and guiding semi-structured interviews. In a separate study [30], the authors focused on quantitative persona creation, emphasizing the potential of algorithmic methods to generate accurate and up-to-date personas from numerical and textual data. They noted the absence of a unified metric for evaluating persona quality but highlighted clustering as the most commonly employed technique.

User profiling: User profiling helps organizations better understand their users, anticipate their needs, and make informed decisions during product design and personalization [19]. It focus on differences among individuals in population. The process involves data collection, analysis, and interpretation. Recent studies have focused on authorship attribution, social annotation profile modeling, data acquisition, feature extraction, modeling techniques, and performance metrics. These studies highlight the need for user profiling approaches that support general-purpose user profiles, are language-independent, have ontological representation, and address dynamic, distributed, and secure user profiles with predictive and temporal features [18].

3 Toward Ontology for creating User Profile

User profiles encompass a wide range of user information, from demographics to personal preferences, and are essential for achieving user-centred design and personalization. However, they also raise concerns about data privacy, ethical considerations, and algorithmic bias. Furthermore, user profiles have economic implications, enabling targeted advertising but also posing risks of data exploitation for financial gain. Responsible and ethical use of user profiles is essential. This includes providing user empowerment and control over their data, continuous adaptation and feedback mechanisms, and addressing issues related to cross-platform and cross-device compatibility. While user profiles offer great potential for enhancing user experiences, their use requires a thoughtful approach that respects individuality and privacy while mitigating the risks associated with data collection and utilization.

To facilitate the exchange and sharing of user profiles, it is imperative to establish standardized domain representations and common vocabularies. Although numerous user-adaptive systems rely on ontologies for semantic representation and knowledge acquisition, these ontologies frequently exhibit a pronounced tethering to particular domains, impeding their reusability. Furthermore, conventional taxonomies in the realm of software engineering often exhibit limitations in adequately encompassing human values, needs, and characteristics. This underscores the necessity for ontological structures capable of describing universal user profiles, disentangled from domain-specific constraints [21].

This section analyse in detail the existing ontologies dedicated to user profiles, offering an exploration of both general and domain-specific user attributes through a comparative alignment with these pre-existing ontological frameworks.

In order to gain a comprehensive understanding of the contemporary landscape concerning methodologies for constructing user profiles and to procure extant ontological frameworks, a systematic review of the existing scholarly literature pertaining to this subject matter was undertaken. The review was conducted through the utilization of the Google Scholar platform, with a deliberate selection of keywords, namely: "User Profile, User Profiling, User Group, User Model, User Modelling, Persona," in conjunction with "Survey, Literature Review, Ontology." A total of 23 papers were analysed within the temporal scope spanning from 2005 to 2022. Among these, 14 papers were identified as secondary studies, presenting a synthesis of existing research, while the remaining 9 publications were identified as primary research contributions that specifically addressed the development of ontologies for the purpose of user profile creation.

3.1 Existing Ontologies for creating User Profile

The General User Modelling Ontology (GUMO) [23] defines user dimensions and analyses in detail properties such as heartbeat, age, position, birthplace, or swimming ability. This ontology identifies 1000 groups of auxiliaries, predicates and ranges. The User Profile Ontology (UPO) [20] aims to model user profiles at a high level, incorporating relevant concepts and properties. The Ontology-based User Modeling framework (OntobUMf) [28], [29] focuses on analyzing user behaviour in knowledge management systems. The Grapple ontology [10] supports the rule-based creation of domain-related statements for users. The Holistic Persona Ontology (HPO) [11] proposes a persona ontology framework with five dimensions, encompassing factual, personality, intelligence, knowledge, and cognitive process aspects. The Persona Ontology (PO) [32] provides a comprehensive set of concepts related to personal characteristics and environment, by having separate ontologies for requirements and behavioural aspects which are connected in a model-driven way. The User Modelling Meta-Ontology [38] offers a top-level classification of the user modelling field, analyzing diverse data collection and interpretation approaches.

As demonstrated in Table 1, we found five ontologies that group various user characteristics into different user profile dimensions. Three ontologies show how to use the proposed dimensions by giving a complete instance model, while two introduce partial instantiation. Finally, four of the ontologies introduce limited implementation details based on OWL syntax. although they provide the links to their implementation, they are not functional at the time of writing this article, and ontologies can not be retrieved. Currently, there is a broadly used ontology for social networks *Friend of a Friend (FOAF)*¹, which serves to represent relations of user with other people. Further, there is a vocabulary class that characterizes generic RDF resources within a Knowledge Graph², which focuses

¹ <http://xmlns.com/foaf/0.1/> (Access date 30, September 2023)

² <https://schema.org/Person>, (Access date 30, September 2023)

Table 1: Overview of analysed ontologies
 X - exist in the article, O - was partially presented

Ontology	<i>Domain</i>	<i>Dimension</i>	<i>Instance</i>	<i>Implemented</i>
General User Modeling Ontology (GUMO) [23]	General	X	O	OWL
User Profile Ontology (UPO) [20]	Personal information management	X		
Ontology-based User Modeling Framework (OntobUMf) [28], [29]	Knowledge-management system	X	O	KAON (OWL)
Holistic Persona Ontology (HPO) [11]	User Centered Design	X	X	
Persona Ontology (PO) [32], [33]	Requirement engineering	X	X	Protege (OWL)
Grapple ontology [10]	User model pipes		X	OWL
User Modeling Meta-Ontology [38]	User modeling field			

on representing the notion of the person with the particular instance for the medical field.

It is important to note that analysed ontologies are predominantly static and do not address how the analysed property can change with time or new data retrieval. There is a lack of support for dynamic and temporal features (for instance the features that can change in a matter of ms, like one captured by IoT devices and often stored in time-series databases. The existing approaches primarily adopt a top-down approach, requiring developers to start from scratch. There is a need for bottom-up approaches that enable the generation of user profiles from existing datasets, particularly ones that are domain-independent and applicable to various applications. Analyzing general user profile characteristics and establishing mappings to domain-specific profiles is the crucial starting point in addressing these requirements.

3.2 Extracting User Profile dimensions

The core idea of adaptation is based on the assumption that differences in some user characteristics affect the usefulness of the services or information provided to the individuals. Thus if a system’s behaviour is tailored according to such individual characteristics, its value to individuals will be increased. We propose fixed user dimensions in Table 2 to guide user profile ontology implementation.

We differentiate between domain-specific and general dimensions. Domain-specific are highly dependent on the application domain, e.g. they are not valid in the other domain. General, or domain-independent, characteristics are ones which we can reuse in between domains. They can represent the general user characteristics and abilities, which later we can relate to the context of the study to understand their impact. They can also represent the existing data from

Table 2: User Profile Dimensions and Encapsulated Concepts

<i>Dimension</i>	<i>Description</i>
Demographic [18], [34], [23]	encompass the fundamental information commonly used for segmentation and categorization purposes. This includes basic user details such as name, date of birth, email, and contact information [23], as well as information related to the user’s place of residence and type of housing [20]
Characteristic [18], [34], [23], [20]	represent a range of attributes that can be categorized into general physical traits and personality traits. General characteristics include physical features such as eye colour, height, and weight, while Personality traits encompass qualities such as talkativeness, dominance, reservedness, shyness, kindness, warmth, and openness to experience. Commonly used models like the Five-Factor Model define a set of properties that describe personality, including extroversion/introversion, thinking, feeling, and sensing [11].
Role [23]	encompass various aspects such as user Profession [20] and Occupation [32]. Additionally, roles refer to specific positions or responsibilities that users hold within a particular system or context. These roles can include administrator, manager, employee, customer, guest, moderator, contributor, and other relevant roles based on the specific application or domain.
Ability [23], [20], [32]	represent the user’s Proficiency [23] or Expertise [20] in performing various tasks or activities. These abilities can vary widely including user’s level of Education [20], [32], Qualifications, Certifications, and Licenses [28], user’s Knowledge which comprises factual, procedural, conceptual, and metacognitive aspects providing insights into their cognitive abilities, such as learning process (remembering, understanding, applying, analyzing) [11] or their Intelligence [11], technical skills, physical capabilities, and language proficiency.
Relation	represents the connections or associations between users and other entities or users within a system or context. It signifies the relationships to Thing (Ownership), Contact (Other persons) [20] or to certain Organisations.
Human state	encompasses various aspects of the user’s well-being, including their nutritional status, emotional state (such as worry, boredom, hope, joy), mental state (including sleep, drowsiness, stress), physiological indicators (such as heartbeat, blood pressure, temperature), and even their motion or facial expressions [23]. These states are observed and captured to gain insights into the user’s Experience and overall Condition.
Behaviour [28], [32]	include user activity patterns, interaction sequences, frequency of actions, duration of engagement, and other behavioural indicators exhibited by the user while interacting with a system or engaging in specific activities.
Interest [28], [32], [20]	refer to the topics or subjects that attract the user’s attention or curiosity or areas of expertise that they find engaging or enjoyable
Preference [20]	refer to the individual choices, inclinations, or options that a user favours or selects in various contexts. They encompass a wide range of aspects, like preferences for specific features, functionalities, or settings of a system or application. They can also include preferences for certain types of content, styles, layouts, or presentation formats.
Goal/Task [28], [32]	represent the specific objectives, aims, or activities that a user intends to accomplish within a system or context. They reflect the user’s intended outcomes or desired results while interacting and provide insights into the user’s motivations, intentions, and priorities.
Environment [32]	include physical location where the user interacts with a system (e.g., home, office, public space), the social context in which the user operates (e.g., social networks, online communities), and the technological infrastructure and devices available to the user (e.g., IoT devices, internet connectivity)
Requirement [32]	include functional requirements, which describe the specific features, functionalities, or capabilities that the user expects from a system or application and non-functional requirements, which focus on aspects such as performance, reliability, usability, security, or accessibility.

which the human state is interfered with for the user. In most cases when the user profile dimensions are measured person has a rough idea about the expected expiry [23]. For instance, human states are highly dynamic, and some of them can change within a few seconds, while user demographics and characteristics, in general, don't change too often.

	Property Change: Rarely ←————→ Often					
	Demographics	Characteristic	Role	Relation	Ability	Human State
Survey	■ +	■ +			■ + Knowledge Background Beliefs	- Emotional State
PO	- Persona		■ + Occupation		■ + Knowledge Education Language	
GUMO	■ + Contact Info	■ +	■ +		■ + Proficiency	- Emotional, Physiological & Mental State, Nutrition, Motion, Facial Expression
UPO	- Person, Living conditions	■ + Personality	- Profession	- Thing	■ + Expertise Education Activity	
OntobUMf			- Affiliation	- Contact	- Competency QCL	
HPO	- Factual	- Personality			- Knowledge Intelligent Cognitive	

■ + same dimension name
 ■ + same dimension name plus encapsulate additional attributes
 dimension is not addressed
 - different dimension name/similar content

Fig. 1: Linking general dimensions with existing ontologies

We introduce the mapping of the dimensions introducing general properties with one proposed by the existing ontologies and the existing surveys ([18], [34]) in Figure 1. We sort the dimensions by frequency of how likely it is that their inner property will change; on the left *Demographics* being rare to change, while on the right ones that more often change, especially the dimension encapsulating *Human State*. We can note that the user *Ability* dimensions were widely analysed by all proposed ontologies, while the *Relation* and *Human State* were examined only by two proposed ontologies.

We derived the existing mapping table, and proposed dimensions, by linking the dimensions proposed by existing ontologies (Table 1 based on the provided definitions, as well as, the given instance examples. In Figure 2 we can see an example of linking of existing dimensions proposed by Persona ontology [32]. In this case, in addition to the Ability dimension, dimensions related to Knowledge, Education and Language are derived. When taking a look at the given model instance, we can see that all of these dimensions actually represent the user's

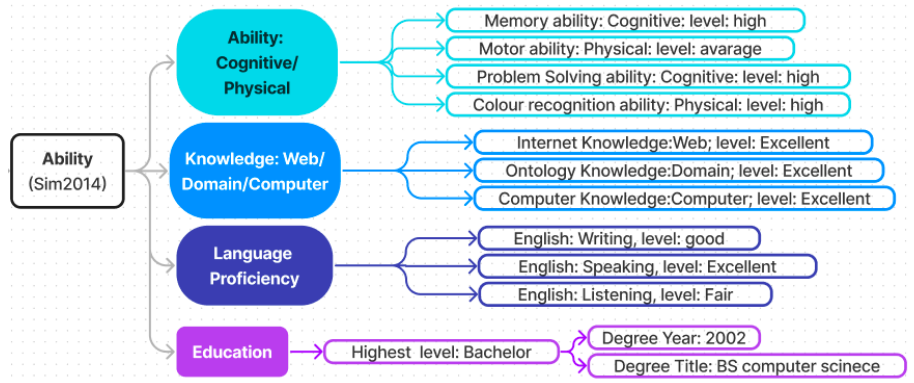


Fig. 2: Mapping Ability dimension to the one proposed by PO [32]

ability or expertise in something. It is possible to reuse the proposed dimensions as subcategories that guide the developer when creating the ontology and selecting existing properties if needed.

	Behaviour	Interest	Preference	Goal/Task	Environment	Requirement
Survey	■	■	■	■ + Plans Needs		
PO	■	■+ Concern	- Usability Preference	■ + Scenario	■	■
GUMO						
UPO		■	■			
OntobUMf	■ + Identification Accessibility Activity	■		■		
HPO						

■ same dimension name ■+ same dimension name plus encapsulate additional attributes
 □ dimension is not addressed - different dimension name/similar content

Fig. 3: Linking domain-specific dimensions with existing ontologies

Further, in Figure 3 we introduce the domain-specific dimensions. These dimensions can't be ordered by the expectations on how often their properties will change (e.g. to differentiate between transitory and long-lasting states), as this is highly domain-dependent. We can see that Sim and Brouse [32] cover this dimension widely in their PO ontology, although focusing only on Usability preferences.

4 User Profile Meta-Ontology (UPMO)

The ontology representation of user profiles serves as a means to define and represent the knowledge contained within them, enabling the sharing and reuse of domain knowledge [18]. It facilitates the analysis and extraction of domain-specific information from operational knowledge and supports the explicit sharing and exchange of profiles based on an agreed model.

Ontology engineering is a subfield of model-driven engineering (MDE) [12]. In this paper, we propose the development of User Profile Meta-Ontology (UPMO), based on the dimensions which we analysed in detail in the previous Section 3. The idea is to rely on Knowledge Graphs (RDF) [25] for describing user models as they allow us to apply reasoning techniques and enable finding relationships between data. On the other hand, it is to use the MDE environment to create ontology-based domain-specific modelling language.

In the rest of this section, we draw from the existing surveys on user profiles ([26], [18], [34]) to establish the current requirements. We also delineate the essential role that UPMO plays in the user model development process, alongside providing an overview of the envisioned modeling environment.

4.1 UPMO requirements and role during system development

According to presented research trends, there are several requirements that contribute to the wide dissemination of user profiles [26], [18], [34], and which are defined in Table 3. These requirements aim to ensure the versatility, effectiveness, and adaptability of UPMO, promoting their widespread implementation and usability across different contexts and applications.

UPMO is expected to play a significant role in designing, implementing, and testing emerging socio-technical systems. We highlight here how it can contribute to each of these development phases:

Design: UPMO is expected to help in the design phase by providing a standardized vocabulary and a conceptual model that captures the key concepts, relationships, and constraints regarding user linked to a specific domain. It helps designers understand and model the characteristics and needs of the target users, by capturing user preferences, interests, demographics, and other relevant factors, designers can create user personas, profiles and role/task models. Obtained artefacts serve as a foundation for designing tailored interactions, content recommendations, and personalized features while ensuring consistency and interoperability between the user profiling model and other system components.

Implementation:: UPMO is expected to guide the implementation of personalized features and adaptive functionalities. Developers can integrate the ontology into the system architecture, allowing user data to be collected, stored, and processed in a structured manner. It serves as a common representation that facilitates communication between different software modules and enables the effective utilization of user profile information. Developers can use UPMO to define rules, algorithms, and mechanisms for personalization and adaptivity.

Table 3: Requirements for User Profile modelling

<i>Requirement</i>	<i>Description</i>
Generality	UPMO should exhibit generality by explicitly representing domain-independent properties of user profiles. They should be designed to be compatible with various applications and domains, enabling their utilization in different contexts.
Expressiveness	UPMO should demonstrate a high degree of expressiveness, enabling it to effectively represent a wide array of facts and rules pertaining to the user. This includes accommodating diverse types of information, allowing for the comprehensive and flexible representation of user attributes, characteristics, and behaviours.
Import of external data	UPMO should possess the capability to integrate data from multiple heterogeneous sources into a unified data source, regardless of their formats, structures, or locations. This data federation is expected to provide a unified and consistent view of data, enabling querying of integrated data as if it were stored in a single centralized database.
Integrating multiple ontologies	UPMO should support combining ontologies from different sources to enhance reuse, knowledge sharing and interoperability. Ontology federation allows for the integration of ontologies that capture different perspectives of user profiles or specialize in specific domains. It involves establishing mappings between the concepts and relationships which enable the knowledge exchange.
Extensibility	UPMO should provide the interface for the modelling of new features at the meta-level and APIs that allow for the exchange of user information between user modelling tools, such as data collection methods, analysis algorithms, and modelling approaches. It should be capable of accommodating evolving user modelling techniques and supporting a wide range of user modelling tasks and applications.
Privacy	It is essential for the user model to adhere to privacy policies, conventions, and both national and international privacy legislation. To ensure user privacy, the UPMO should incorporate robust tools and services that safeguard sensitive user information.
Reasoning abilities	The UPMO should possess capabilities to perform diverse forms of reasoning and effectively handle conflicts that may arise from contradictory facts or rules. Model verification enables the user modeller to effectively handle ambiguity and reconcile conflicting information.
Usability	The UPMO should prioritize comprehensibility, efficiency, effectiveness, and learnability. This can be achieved through clear and thorough documentation that explains the ontology's purpose, structure, and how it is used. It's also important to adhere to standard naming conventions and organize the hierarchy of classes logically. Alignment with existing user representations and the availability of user-friendly tools are essential. It should foster collaboration and actively seek feedback from the user community. This feedback loop can result in valuable improvements and a broader perspective on the ontology's usability.

Testing: Testers can use the generated user model to simulate different user profiles and interactions, evaluate the data collection, interpretation mechanisms and system’s ability to deliver personalized content, adapt to user preferences, and provide relevant recommendations. It helps in defining test cases, expected outcomes, and criteria for assessing the system’s performance in terms of user modelling and personalization accuracy.

4.2 Envisioned UPMO development environment

To develop the UPMO we propose using the interrelated domain-specific languages (DSLs) provided by openCAESAR platform [37]. This platform represents ontology DSL precisely using the semantics of web ontologies while integrating information in engineering tools based on code generation and model transformation. It supports information authoring, federation, linking change management, and configuration management while making the system engineering process more agile with DevOps (CI/CD) practices.

The core part of this opensource platform is Ontological Modeling Language (OML)³, which is inspired by the Ontology Web Language 2 (OWL2) [4] and the Semantic Web Rule Language (SWRL) [2]. The abstract syntax is defined with Ecore, and it generates Java API published as a Maven dependency and Eclipse feature. Default serializations are in XMI and it works with all standard Eclipse Modeling Frameworks (EMF) [3] like OCL, ATL, QVT etc.). The textual syntax is defined with Xtext [9], and the graphical syntax with Sirius framework [6].

OML natively support merging and validating of datasets, tools for reporting differences and Describe Logic (DL) [1] reasoning over OWL datasets, as well as sending query to SPARQL [7] endpoints or rules to SHACL endpoints [5]. There are several adapters which are supported like Ecore to OML, OML to OWL and transformations with UML profile and model [8].

In summary, this framework enables smooth integration of the tools commonly used by data scientists to define the ontology and to work with the data integration and analysis, and ones used by software and system engineers to create system architecture, enable code generation and create different editors empowering all stakeholders to work collaboratively on the same model.

In Fig 4 we illustrate the envisioned user profile modeling environment, an ontological DSL, interpreted from UPMO, which supports common techniques for creating the user profile, namely Personas, User Profiling and task/role modeling (see Section 2. The environment enables linking of the existing user data, which can be retrieved from the web, empirical studies, IoT devices as well as public institutions (e.g. medical, educational etc.). Also, it is to enable linking of the existing domain ontologies, like existing user ontologies and the ontologies describing the part of the system to be related to the domain-specific characteristics. The provided model is to generate data storage and interpret the User Model component which is to be integrated into the runtime environment or to the analysis and simulation platform. The resulting ontological DSL for modeling

³ <https://www.opencaesar.io/oml/>, (Access date 30, September 2023)

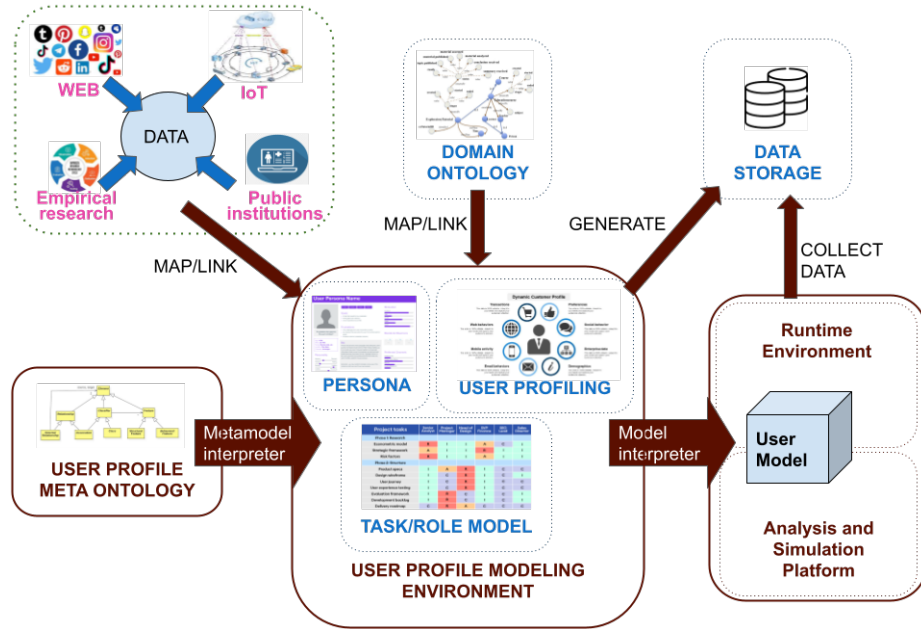


Fig. 4: Envisioned UPMO-based modeling environment

of User Profiles can also natively be combined with other modeling languages, like for instance with the one for usability evaluation [13], or for the requirements gathering [14].

Using a UPMO is expected to provide developers and domain experts with a solid foundation, standardized representation, and guidance throughout the development process. It can help them align their understanding of user profiles with established best practices, facilitating knowledge sharing and collaboration. The interpreted modelling environment is expected to save time, promote consistency and interoperability, facilitate integration with external data sources and domain ontologies, and allow for iterative improvements based on feedback and evolving user modelling needs.

5 Illustration of modelling process with UPMO

In this section, we partially illustrate the iterative and incremental modeling process with UPMO-based ontological DSL on the driver model described in [15]. In the given case study the objective was to design the driver model, the so-called Fallback Ready User, which captures the minimum awareness necessary from the driver to detect and react to a Take-Over Request in an autonomous vehicle.

1. **Identify User Profiling Needs** - UPMO is envisioned to provide a comprehensive set of concepts and relationships, guiding developers in identifying user profile

requirements specific to their system and determining the types of user information needed for personalization and automation, such as their abilities, characteristics, roles, or system specific preferences, behaviours or objectives.

To design the driver model, the idea is to use the set of proposed dimensions and identify the high-level properties important in their scenario, starting from general ones to domain-specific ones. For instance, the driver’s state is important in the given case study as it should be monitored inside the cabin. The biosensor camera is used to capture the state of the driver, and the important system requirement is that this is captured in real-time. However, to have the means to interpret the relevant state of the driver in their scenario, like stress or sleepiness, it is necessary also to think of the means of reusing the existing related data if the same one can be obtained. Further, demographic information, as well as driver abilities are relevant for the driver to react to autonomous transfer requests from the vehicle. As the process is iterative and incremental, the developers are advised not to go too wide, and instead to capture in the first iteration the most critical properties of the user profile to be tested in the given scenario.

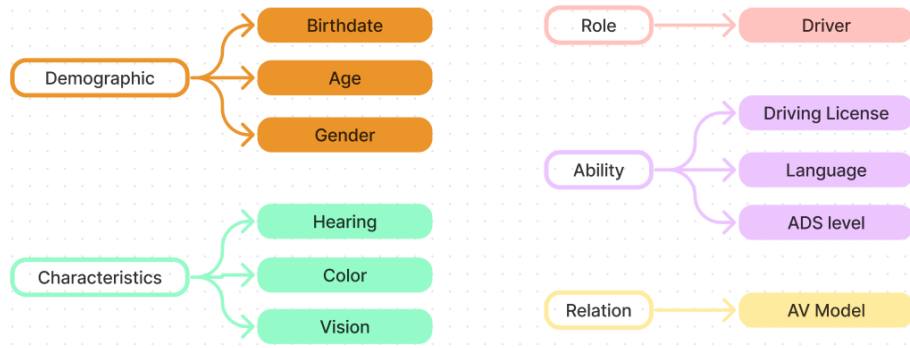


Fig. 5: High level features and long term properties

2. **Create/Extend the User Profile Ontology** - UPMO serves as a starting point for developers to extend and customize the ontology, and to link existing domain ontologies necessary for capturing user profiles in the given context. This saves significant effort and time compared to starting from scratch, as developers can build upon the existing ontology structure and relationships by using the ontology alignment approach [36].

In the case of the driver model, the developers examine the properties regarding their user which they want to test in the first iteration. For instance to capture high-level features and long-term properties they introduce them in the correlated dimensions (see Figure 5. Users are encouraged to engage in thoughtful reasoning about the properties within each dimension that align with their specific requirements. For instance, they may seek information regarding substantial, long-term attributes within demographics, such as age, gender, or a person’s role as a driver, including their connection to the vehicle they use, any potential auditory or visual impairments, and their driving capabilities. The profile designer should have the flexibility to select property definitions from established ontologies or introduce

new ones. If a newly introduced property already exists, the framework should offer suggestions for existing properties. Furthermore, understanding how existing properties are interrelated and characterized can serve as a guide for users when implementing their own properties.

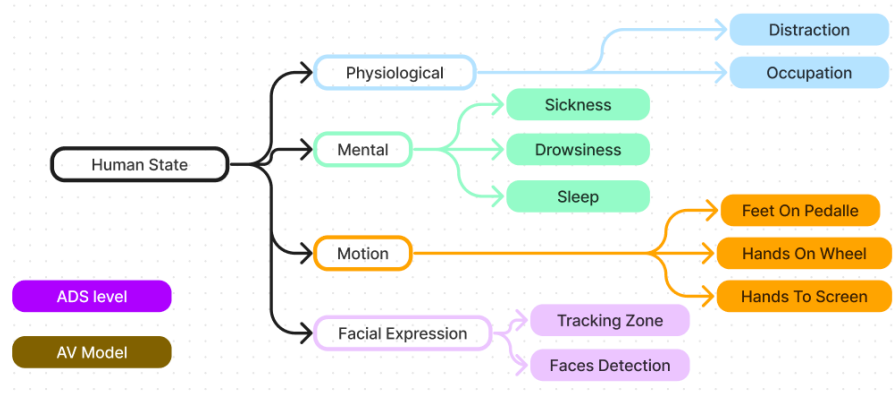


Fig. 6: Real-time properties

Additionally, we encounter scenarios where real-time properties need to be captured, such as time-sensitive data from signals that indicate whether the driver has their foot on the pedal or hands on the wheel, their gaze direction, distractions, engagement in driving, or other activities. This data can also help determine if the driver is fatigued, drowsy, or experiencing stress (see Figure 6). In such cases, it is beneficial to leverage categorizations and mental state definitions that have been thoroughly analyzed in GUMO [23]. Reusing static property definitions for naming and employing their expiration reasoning can prove particularly useful.

3. **Capture and map User Profile Data** - Identify the data sources available from IoT devices (e.g. cameras, motion sensors, security devices), web (e.g. social networks, e-commerce platforms), public institutions (e.g. education institution, healthcare institutions) or existing empirical studies (e.g. surveys, user logs, usability studies). Having a UPMO helps developers map the ontology's predefined concepts and relationships with collected data to ensure data consistency and interoperability.

In the case of the driver model, the user dynamic properties are gathered by the biosensor camera device inside the vehicle. The raw data which is gathered doesn't have the format, or naming which can be easily interpreted, used or analysed. The tendency of developers when using this data is to keep the raw structure that was found as the output of given devices, which prevents the reuse and often stores a lot of information which is never used by a system. In Table 4 we illustrate the resulting data format and interpreted meaning, which are represented as Triple (variable, timestamp, confidence). This information is typically stored in a time-series database, and the idea is to incorporate existing approaches like [16] to include this type of data linking to be supported by UPMO .

Table 4: Dynamic properties (taken from [15])

variable	value	description
distracton	boolean	indicate if the driver is occupied with non-driving task
occupation	boolean	indicate if the driver is occupied with driving task
sleep	boolean	indicates if driver is in sleep state
sickness	boolean	indicates if a driver is experiencing sudden sickness
drowsiness	boolean	indicates if driver is tired or sleepy
feet_on_pedalle	boolean	Indicates if driver's feet are on the pedals
hands_on _wheel	hand_none hand_left hand_right hand_both	Indicates which hand(s) the driver has on the wheel
hand_to_screen	boolean	indicates if driver's hand is at the touch screen
tracking_zone	string	Indicates where the driver is looking
faces_detection	.front: int .rear: int .occupation: int	N of detected humans in front N of detected humans in rear N of seat occupied

4. **Implement Data Storage and Retrieval** - UPMO provides guidance on how to organize and store user profile data, enabling developers to create efficient and scalable storage systems that are compatible with the ontology.

Based on the previously linked data and the vocabulary definitions the modelling environment generates the data structure for data collection, which we call User Model. To retrieve the data from this artefact, the developer needs to create SPARQL queries over the RDF data to get results, for instance in JSON format, which answers certain questions, like for instance: 'Is driver looking at the road?'; 'Is driver sleepy?', 'Does the driver have hearing problems?', 'Is the driver ready to take over the control of the vehicle (e.g. his hands are on the steering wheel, the feet are on the paddle, he is looking at the road and is mentally ready)?'. In this phase, the developer is also encouraged to think about the methods for deducting user properties, like data mining techniques (e.g. clustering, classification, association rules, sequential patterns, latent variable models).

5. **Test and Integrate User Model** - UPMO is generating a user model component which can be integrated with a simulation environment. It supports the development of test cases and scenarios to validate the user profiling functionality of the system. It helps to verify that the system correctly collects, stores, and retrieves user profile data and that it effectively uses ontology to personalize interactions.

UPMO allows developers to integrate user profiling capabilities with system components. They can design modules that utilize user model components to access and leverage user profile data enabling personalized automation, adaptive interfaces, and decision-making processes based on standardized representations.

In our case study, we want to generate the User Model component in a simulation environment, a cockpit equipped with seat shakers, a biosensor camera and a front screen running the simulation for a driving experience. First, it is to perform

Unit testing to verify that based on the existing or automatically generated data, SPARQL queries return correct reports (i.e. answers to the questions). Further, it is to perform *Integration testing*, where it is to test that data collection is working properly (e.g. camera captures correctly information in the user model) or that the cabin monitoring system receives the analysis reports in the correct format and is able to use the provided information as expected from system. For instance, it is to test a time delay from data collection, analysis and returning a report to another component. Finally, it is to perform a *System testing*, basically validating the complete takeover scenario, while having a human driver in the cockpit, and focusing on testing the performance, usability and user experience of the system.

6. **Iterate and Refine** - By leveraging the UPMO as a starting point developers benefit from the iterative refinement process when extending the user profiling ontology [22]. It helps them to continuously iterate and refine the ontology and its implementation based on feedback, user evaluations, and system performance.

In our example, in the following interaction, the developers should focus on analysing domain-specific dimensions. For this purpose, they are encouraged to map domain ontologies, like the ones describing the autonomous vehicle, or introduce this domain concept from existing UML models or DSLs. It will be also necessary to analyse the rules and relations between properties and dimensions introduced in the proposed model.

For instance, in the Behaviour dimension, it is to introduce the driver's reaction when the system alert is transmitted. For instance to collect the sequence of the relevant actions. For Interest, it can be that the driver wants to make customized alerts. Further drivers could have preferences regarding the auditive signal for instance because the person has hearing problems. The task the driver wants to perform is to take over the control of the vehicle when the alert is transmitted. Regarding the Environment, it is to introduce a specific system environment which can be imported from existing UML models. For instance, ones describing the driving condition simulation component, cabin monitoring system or cabin experience manager. Finally, some of the relevant requirements could be the definitions of the response time related to the specific reports, e.g., when the driver is in the fallback ready state, the frequency of checking if the driver is sleepy.

6 Conclusions and future work

This paper presents the design of the User Profile Meta Ontology (UPMO) and conducts an analysis of the user dimensions, which serve as high-level concepts guiding the development of UPMO. We explore the anticipated development process for adaptive systems and examine the potential benefits for developers and domain experts through UPMO-supported artefacts. UPMO provide unified standardized terminology for representing User Profiles, enable sharing of User Models and collected data, and support federation and linking of existing user data

As part of our future work, we are planning to achieve several key objectives. First, we aim to align the User Profile Meta-Ontology (UPMO) with existing ontologies for user profiles and propose a systematic alignment process. This alignment will enhance interoperability and data sharing across different ontologies.

Additionally, we intend to introduce user profile-guided semantics to support integrative data analysis of existing data. This will allow for more effective and comprehensive data analysis, driven by user profile information.

Another important aspect of our future work is to validate the feasibility of the proposed dimensions by mapping them to existing user and data models introduced by previous researchers. This will help us ensure that the dimensions are well-defined and compatible with established practices in the field.

We also plan to delve deeper into the detailed structure of the envisioned metamodel and analyze the interrelationships among its components. Understanding these relationships is crucial for the effective utilization of the ontology.

Furthermore, we will enhance the iterative development process of the user profile ontology by introducing collaborative workflows. Collaborative development will involve multiple stakeholders, including users, developers, and domain experts, to ensure that the ontology remains relevant and effective.

In parallel, we will consider various system-related aspects, such as how data is captured, analyzed, and represented in UPMO components. This holistic approach will enable a more comprehensive understanding of how the user model informs the system and, in turn, how the system serves the user.

Overall, our future work is focused on aligning, expanding, and validating UPMO to make it a more powerful and user-centric tool for developing user profiles and supporting user data analysis.

References

1. The Description Logic Handbook: Theory, Implementation and Applications, 2nd ed. Cambridge: Cambridge University Press, 2007.
2. “A Semantic Web Rule Language Combining OWL and RuleML (SWRL).” [Online]. Available: <https://www.w3.org/Submission/SWRL/>, <http://www.w3.org/Submission/SWRL/>
3. “Eclipse Modeling Framework (EMF).” [Online]. Available: <https://www.eclipse.org/modeling/emf/>, <https://www.eclipse.org/modeling/emf/>
4. “OWL2 Computational Properties.” [Online]. Available: https://www.w3.org/TR/owl2-profiles/#Computational_Properties, http://www.w3.org/TR/owl2-profiles/#Computational_Properties
5. “Shapes Constraint Language (SHACL).” [Online]. Available: <https://www.w3.org/TR/shacl/>, <http://www.w3.org/TR/shacl/>
6. “Sirius UI Framework for Eclipse.” [Online]. Available: <https://www.eclipse.org/sirius/overview.html>, <https://www.eclipse.org/sirius/overview.html>
7. “SPARQL Query Language for RDF.” [Online]. Available: <https://www.w3.org/TR/rdf-sparql-query/>, <http://www.w3.org/TR/rdf-sparql-query/>
8. “Unified Modeling Language (UML).” [Online]. Available: <https://www.uml.org/what-is-uml.htm>, <https://www.uml.org/what-is-uml.htm>
9. “XText.” [Online]. Available: <https://www.eclipse.org/Xtext/>. [Accessed: 12-Feb-2021]., <https://www.eclipse.org/Xtext/>

10. Abel, F., Heckmann, D., Herder, E., Hidders, J., Houben, G.J., Krause, D., Leonardi, E., van der Sluijs, K.: A framework for flexible user profile mashups. In: G. Balint, Antala, B., Carty, C., Mabieme, J.M.A., Amar, I.B., Kaplanova, A. (eds.) Proceedings of the International Workshop on Adaptation and Personalization for Web 2.0 (APWEB 2.0 2009) in conjunction with the International Conference on User Modeling, Adaptation, and Personalization (UMAP 2009). vol. 7, pp. 1–10. CEUR-WS.org (2009). <https://doi.org/10.2/JQUERY.MIN.JS>, <https://research.tue.nl/en/publications/a-framework-for-flexible-user-profile-mashups>
11. Anvari, F., Tran, H.M.T.: Persona ontology for user centred design professionals. In: G. Balint, Antala, B., Carty, C., Mabieme, J.M.A., Amar, I.B., Kaplanova, A. (eds.) Proceedings of the ICIME 4th International Conference on Information Management and Evaluation. vol. 7, pp. 35–44. Academic Conferences and Publishing International (2013). <https://doi.org/10.2/JQUERY.MIN.JS>, <https://researchers.mq.edu.au/en/publications/persona-ontology-for-user-centred-design-professionals>
12. Atkinson, C., Kühne, T.: Model Driven Development: A Meta-modeling Foundation. IEEE software **20**(5), 36–41 (9 2003). <https://doi.org/10.1109/MS.2003.1231149>, <http://portal.acm.org/citation.cfm?id=942589.942704>
13. Barišić, A., Amaral, V., Goulão, M.: Usability Driven DSL development with USEME. Computer Languages, Systems and Structures (ComLan) **51**, 118–157 (2018). <https://doi.org/10.1016/j.cl.2017.06.005>
14. Barišić, A., Blouin, D., Amaral, V., Goulão, M.: A Requirements Engineering Approach for Usability-Driven DSL Development. In: 10th International Conference on Software Language Engineering (SLE). pp. 115–128. ACM, Vancouver, British Columbia, Canada (2017). <https://doi.org/10.1145/3136014.3136027>, <https://dl.acm.org/citation.cfm?id=3136027>
15. Barisic, A., Sigrist, P., Oliver, S., Sciarra, A., Winckler, M.: Driver Model for Take-Over-Request in Autonomous Vehicles. UMAP 2023 - Adjunct Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization pp. 317–324 (6 2023). <https://doi.org/10.1145/3563359.3596994>, <https://dl.acm.org/doi/10.1145/3563359.3596994>
16. Donkers, A., Yang, D., De Vries, B., Baken, N.: Real-time building performance monitoring using semantic digital twins. Proc 9th Linked Data Archit Constr Work [Internet] **12** (2021)
17. Earthy, J., Jones, B.S., Bevan, N.: The improvement of human-centred processes—facing the challenge and reaping the benefit of ISO 13407. International Journal of Human-Computer Studies **55**(4), 553–585 (10 2001). <https://doi.org/10.1006/IJHC.2001.0493>
18. Eke, C.I., Norman, A.A., Shuib, L., Nweke, H.F.: A Survey of User Profiling: State-of-the-Art, Challenges, and Solutions. IEEE Access **7**, 144907–144924 (2019). <https://doi.org/10.1109/ACCESS.2019.2944243>
19. Farid, M., Elgohary, R., Moawad, I., Roushdy, M.: User Profiling Approaches, Modeling, and Personalization. SSRN Electronic Journal (10 2018). <https://doi.org/10.2139/SSRN.3389811>, <https://papers.ssrn.com/abstract=3389811>
20. Golemati, M., Katifori, A., Vassilakis, C., Lepouras, G., Halatsis, C.: Creating an Ontology for the User Profile: Method and Applications. In: In Proceedings AI* AI Workshop RCIS. pp. 407–412 (2007)

21. Grundy, J., Mueller, I., Madugalla, A., Khalajzadeh, H., Obie, H.O., McIntosh, J., Kanij, T.: Addressing the Influence of End User Human Aspects on Software Engineering. *Communications in Computer and Information Science* **1556 CCIS(c)**, 241–264 (2022). https://doi.org/10.1007/978-3-030-96648-5_11
22. Hannou, F.Z., Charpenay, V., Lefrançois, M., Roussey, C., Zimmermann, A., Gandon, F.: The ACIMOV Methodology: Agile and Continuous Integration for Modular Ontologies and Vocabularies. In: *MK 2023-2nd Workshop on Modular Knowledge associated with FOIS 2023-the 13th International Conference on Formal Ontology in Information Systems* (2023)
23. Heckmann, D., Schwartz, T., Brandherm, B., Schmitz, M., Von Wilamowitz-Moellendorff, M.: GUMO - The general user model ontology. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **3538 LNAI**, 428–432 (2005). https://doi.org/10.1007/11527886_58/COVER
24. Karolita, D., Kanij, T., Grundy, J., McIntosh, J.: Use of Personas in Requirements Engineering : A Systematic Literature Review Use of Personas in Requirements Engineering : A Systematic Literature Review (February) (2023)
25. Klyne, G.: Resource description framework (RDF): Concepts and abstract syntax. <http://www.w3.org/TR/rdf-concepts/> (2014)
26. Kobsa, A.: Generic user modeling systems. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* pp. 136–154 (2007). https://doi.org/10.1007/978-3-540-72079-9_4/COVER
27. Limbourg, Q., Pribeanu, C., Vanderdonckt, J.: Towards uniformed task models in a model-based approach. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **2220 LNCS(April)**, 164–182 (2001). https://doi.org/10.1007/3-540-45522-1_10
28. Razmerita, L.: Ontology-Based User Modeling. In: Sharman, R., Kishore, R., Ramesh, R. (eds.) *Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems*, pp. 635–664. Springer US, Boston, MA (2007). https://doi.org/10.1007/978-0-387-37022-4_23
29. Razmerita, L.: An ontology-based framework for modeling user behavior-A case study in knowledge management. *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans* **41(4)**, 772–783 (7 2011). <https://doi.org/10.1109/TSMCA.2011.2132712>
30. Salminen, J., Guan, K., Jung, S.G., Chowdhury, S.A., Jansen, B.J.: A Literature Review of Quantitative Persona Creation. *Conference on Human Factors in Computing Systems - Proceedings* (4 2020). <https://doi.org/10.1145/3313831.3376502>, <https://dl.acm.org/doi/10.1145/3313831.3376502>
31. Salminen, J., Guan, K., Jung, S.G., Jansen, J.: Use Cases for Design Personas: A Systematic Review and New Frontiers. pp. 1–21 (2022). <https://doi.org/10.1145/3491102.3517589>
32. Sim, W.W., Brouse, P.: Towards an Ontology-based Persona-driven Requirements and Knowledge Engineering. *Procedia Computer Science* **36(C)**, 314–321 (1 2014). <https://doi.org/10.1016/J.PROCS.2014.09.099>
33. Sim, W.W., Brouse, P.: Developing Ontologies and Persona to Support and Enhance Requirements Engineering Activities – A Case Study. *Procedia Computer Science* **44(C)**, 275–284 (1 2015). <https://doi.org/10.1016/J.PROCS.2015.03.060>
34. Sosnovsky, S., Dicheva, D.: Ontological technologies for user modelling. *International Journal of Metadata, Semantics and Ontologies* **5(1)**, 32–71 (2010). <https://doi.org/10.1504/IJMSEO.2010.032649>

35. Stanton, N.A.: The Handbook of Task Analysis for Human-Computer Interaction THE HANDBOOK OF TASK ANALYSIS FOR HUMAN-COMPUTER INTERACTION Edited by Dan Diaper Bournemouth University (September 2003) (2014)
36. Tounsi Dhouib, M., Faron Zucker, C., Tettamanzi, A.G.B.: An ontology alignment approach combining word embedding and the radius measure. In: Semantic Systems. The Power of AI and Knowledge Graphs: 15th International Conference, SEMANTiCS 2019, Karlsruhe, Germany, September 9–12, 2019, Proceedings 15. pp. 191–197. Springer International Publishing (2019)
37. Wagner, D.A., Chodas, M., Elaasar, M., Jenkins, J.S., Rouquette, N.: Ontological Metamodeling and Analysis Using openCAESAR. Handbook of Model-Based Systems Engineering pp. 1–30 (2022). https://doi.org/10.1007/978-3-030-27486-3_78-1, https://link.springer.com/referenceworkentry/10.1007/978-3-030-27486-3_78-1
38. Yudelson, M., Gavrilova, T., Brusilovsky, P.: Towards user modeling meta-ontology. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) **3538 LNAI**, 448–452 (2005). https://doi.org/10.1007/11527886_62/COVER, https://link.springer.com/chapter/10.1007/11527886_62