Semantic-based Adaptation of Quality of Experience in Web Multimedia Streams

Giuseppe Loseto
LUM University "G. Degennaro"
Casamassima, Italy
loseto@lum.it

Floriano Scioscia
Polytechnic University of Bari
Bari, Italy
floriano.scioscia@poliba.it

Michele Ruta
Polytechnic University of Bari
Bari, Italy
michele.ruta@poliba.it

Filippo Gramegna
Polytechnic University of Bari
Bari, Italy
filippo.gramegna@poliba.it

Ivano Bilenchi
Polytechnic University of Bari
Bari, Italy
ivano.bilenchi@poliba.it

ABSTRACT

Video streaming accounts for the majority of worldwide Internet traffic, and HTTP-based multimedia has the largest share among technologies and protocols. The wide availability of mobile devices and wireless broadband networks currently leads to wider heterogeneity of fruition contexts and frequent condition changes during a streaming session. *MPEG-DASH* is the reference standard for Dynamic Adaptive Streaming over HTTP: a provider defines several representations for a segmented multimedia source, with different bit rates, allowing a client to dynamically select the best one based on current conditions, and to download the corresponding sequence of segments for smooth playback. MPEG-DASH does not mandate specific bit rate adaptation schemes; conventional approaches are divided in buffer-based, bandwidth-based and hybrid. Nevertheless, Quality of Experience (QoE) can be influenced by many additional factors. This paper proposes a novel QoE adaptation approach based on dynamic ontology-based annotation of streaming context and mobile matchmaking with DASH representation profiles in a Web Ontology Language (OWL) fragment, exploiting a WebAssembly port of an embedded reasoning engine. The proposed framework enables adaptation based not only on network status, but also on client device capabilities, ambient conditions and multimedia content type. A case study validates the proposal, while early experiments support its sustainability.

CCS CONCEPTS

• **Computing methodologies** → **Knowledge representation and reasoning**; • **Information systems** → *Web applications*; **Multimedia streaming**.

KEYWORDS

Semantic matchmaking, Web Ontology Language (OWL), Quality of Experience, Multimedia streaming, MPEG-DASH

SAC '23, March 27-31, 2023, Tallinn, Estonia

ACM ISBN 978-1-4503-9517-5/23/03…$15.00
https://doi.org/10.1145/3555776.3577686

ACM Reference Format:
Giuseppe Loseto, Floriano Scioscia, Michele Ruta, Filippo Gramegna, and Ivano Bilenchi. 2023. Semantic-based Adaptation of Quality of Experience in Web Multimedia Streams. In *The 38th ACM/SIGAPP Symposium on Applied Computing (SAC '23), March 27-April 2, 2023, Tallinn, Estonia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3555776.3577686

1 INTRODUCTION

According to recent statistics, video streaming accounts for over 60% of worldwide Internet traffic [33]. Among different protocols and services, standards-based HTTP multimedia streaming has the largest share (17.3%), surpassing IPTV (Internet Protocol television), Netflix and YouTube. Besides an ever-increasing consumption – further boosted in the last years by the COVID-19 pandemics– a major trend concerns the adoption of new client devices in addition to desktop and laptop computers, including Smart TVs, tablets and smartphones. Mobile devices, in particular, are the primary video streaming clients for large population segments in many countries, due to socio-technical factors. The penetration of wired broadband connectivity is still insufficient not only in developing countries, but also in rural areas of developed ones, while mobile broadband coverage is widespread with 3G, 4G and upcoming 5G networks [19]. The diversification of client platforms and mobile usage habits are currently leading to a wider heterogeneity of multimedia fruition contexts, with increasingly frequent condition changes during a streaming session.

MPEG-DASH [37] is the reference standard of the Motion Picture Experts Group for Dynamic Adaptive Streaming over HTTP, published by ISO/IEC in 2012 and updated in 2019 [20]. It is based on the segmentation of multimedia resources and the timely download of segments via HTTP for smooth playback. In MPEG-DASH, a *provider* can define several *representations* for a segmented multimedia *source*, with different bit rates, allowing a *client* to dynamically select the highest quality which can guarantee smooth playback, given the current conditions. The latter, in fact, is a primary requirement, as playback interruptions for *re-buffering* degrade the user's Quality of Experience (QoE) [18] more severely than temporary drops in bit rate.

MPEG-DASH does not mandate sophisticated bit rate adaptation techniques; typical approaches are divided in buffer-based, bandwidth-based and hybrid. Besides network parameters, factors of the fruition *context* (as per Dey's seminal definition [11]) influencing the QoE are not taken into account. In order to overcome this limitation and to grant higher flexibility, this paper proposes a novel knowledge-oriented QoE adaptation approach. It relies on dynamic ontology-based annotation of streaming context and on a semantic matchmaking with DASH reference profiles. The Web Ontology Language (OWL) 2 [26] fragment corresponding to the Attributive Language with unqualified Number restrictions ($\mathcal{ALN}$) Description Logic (DL) [2] is chosen as formalism. Particularly, main contributions of the paper are:

- MPEG-DASH bit rate adaptation based not only on network parameters, but also on: client device capabilities and state (screen features, computational and memory load), ambient conditions (light and noise levels as derived by on-device sensors or *e.g.*, by interfacing a client with home and building automation systems), and multimedia content type. They may influence both the perceptual and usefulness (*e.g.*, information assimilation) aspects making up QoE [1, 25].
- A Knowledge Graph (KG) comprising (i) a terminology (*i.e.*, ontology) for real-time annotation of streaming context and (ii) a set of reference profiles. Both KG elements are easily extensible to increase information granularity and include additional context parameters.
- Specialization for bit rate adaptation of a general-purpose matchmaking framework [29]. Standard and non-standard inferences are used to compare service profiles with current context description, in order to retrieve the closest one and adapt bit rate accordingly.
- Integration of inference services in a Web client by means of a complete WebAssembly port of the lightweight matchmaker in [30]. This avoids dependency on remote reasoners, which might compromise user privacy and introduce latencies incompatible with real-time QoE adaptation. To the best of our knowledge, this is the first OWL-based reasoning engine for the WebAssembly runtime: the integration of inferential capabilities within the client side of Web applications opens up many opportunities for Web developers and users.

A complete prototype of the proposed framework has been implemented to demonstrate the feasibility of the approach and a case study has been devised to validate the concept. Furthermore, computational sustainability has been evaluated through experimental tests.

The remainder of the paper is as follows. Section 2 briefly recalls the state of the art in adaptive multimedia streaming, while related work is discussed in Section 3. The proposed framework is described in detail in Section 4, and Section 5 illustrates the case study. Early performance results are in Section 6, before conclusion.

2 ADAPTIVE MULTIMEDIA STREAMING: STATE OF THE ART AND PERSPECTIVES

HTTP Adaptive Streaming (HAS) refers to dynamic streaming algorithms and frameworks aiming to adapt the quality of HTTP-based video streams, mainly according to network conditions (*e.g.*, buffer status, packet loss ratio and bandwidth) [34]. HAS methods require video content to be composed of small *segments*, each containing a few seconds of playtime, available in multiple quality levels and bit rates. The streaming client downloads the upcoming parts of a video by selecting a quality level which complies with detected network parameters, in order to avoid video stalling and optimize the playback buffer with respect to the available bandwidth.

Several HAS solutions have been proposed in the Web industry: *Microsoft Silverlight Smooth Streaming* (MSS) [39] by Microsoft Corporation, *HTTP Live Streaming*[1] (HLS) by Apple Inc., and *HTTP Dynamic Streaming* [2] (HDS) by Adobe Systems Inc. are among the most popular ones. Despite the fact that they all exploit a similar technological approach, these proposals are vendor-specific and mutually incompatible, thus fragmenting the market and dampening their acceptance and adoption among providers. To overcome these limitations, in 2012 the Moving Picture Experts Group (MPEG) defined the open-source Dynamic Adaptive Streaming over HTTP (DASH) [37] standard for dynamic and adaptive media streaming over HTTP. Also known as MPEG-DASH, it aims to address the interoperability needs between client devices and servers of various vendors.

The DASH specification is composed of four parts, defining (i) the *Media Presentation Description* (MPD) and *segment* formats, (ii) conformance and reference software, (iii) implementation guidelines, and (iv) segment encryption and authentication. An MPD file is an XML document acting as manifest for a streaming resource. The MPD XML Schema divides a resource in `Periods` (*e.g.*, chapters in a movie), with given start time and duration, each one containing one or more logically correlated `AdaptationSets`. For example, a movie may have an AdaptationSet for the video stream and an additional one for each available audio track in a different language. Subtitles and other metadata (*e.g.*, for accessibility) can also be associated to each AdaptationSet. Each `AdaptationSet` element in MPD further contains `Representations` providing a different codec, resolution and/or bit rate for the same content. For each `Representation`, the MPD document contains segment URLs to be downloaded for playback. Specifically, either a `SegmentList` including individual `SegmentURL` elements, or a single `SegmentTemplate` defining the URLs of all segments in a parametric way (*e.g.*, by including a progressive number), so that the client can determine the actual URLs at run time.

To play a video, the DASH client first obtains and parses the MPD to extract useful information. Then, it selects the most suitable representation by exploiting an adaptive bit rate (ABR) algorithm, and starts streaming the content by fetching the related segments via HTTP. Delivery of media-encoding formats of segments, client behavior for fetching, adaptation heuristics, and content playback are outside of the scope of the MPEG-DASH specification, and can be independently defined by each implementation. In latest years, novel client-side ABR algorithms supporting the MPEG-DASH standard have been defined, in order to adapt video delivery according to the user experience and the perceived level of satisfaction [3]. Basically, this vision extends conventional methods relying on technical network parameters only, thus moving from traditional frameworks based on Quality of Service (QoS) to approaches based on QoE, aiming to suggest the optimal streaming conditions according to user-oriented contextual features.

An important challenge currently faced by the research community involved in HAS applications is to correctly take into account

[1] *https://developer.apple.com/documentation/http_live_streaming*
[2] *https://business.adobe.com/products/primetime/adobe-media-server/hds-dynamic-streaming.html*

the end-user experience. Every experience-based streaming algorithm should be grounded on a clear and accurate QoE model, where both network-level and application-level features have to be considered. The *Qualinet White Paper on Definitions of Quality of Experience* defines QoE as *"the degree of delight or annoyance of the user of an application or service"* [5]. The International Telecommunication Union (ITU), subsuming this definition, specifies influencing factors *"include the type and characteristics of the application or service, context of use, the user's expectations with respect to the application or service and their fulfilment, the user's cultural background, socio-economic issues, psychological profiles, emotional state of the user, and other factors"* [18]. Starting from this definition, the Application, Resource, Context, User (ARCU) model [36] categorizes main factors influencing the QoE into four multi-dimensional spaces: (i) *system* parameters concern the technical aspects of a multimedia service (*e.g.*, network performance, display and media configurations); (ii) *human* characteristics refer to the individual experiencing the fruition (*e.g.*, demographics or personal interests); (iii) *context* features are related to the environment where the experience is consumed (*e.g.*, physical features of the context, economic factors, home functionalities); (iv) *content* factors include information about the content being offered by the service/application under consideration. ARCU is one of the most widely adopted models for QoE-based ABR algorithms.

Nevertheless, some non-negligible weaknesses are still evident in HAS frameworks. Particularly, QoE models define simplistic rules based on numeric constraints and opaque labels. Classical Machine Learning (ML) techniques have been largely used to improve HAS [3], but these solutions are seldom fully manageable in practical streaming applications. They basically carry out no more than a prediction task (*e.g.*, classification, regression) whose algorithms are black boxes, both for humans and automatic systems, and relationships between input features and output results are very difficult to understand for non-expert users. The lack of a machine-understandable description or human-understandable explanation of the adaptation results is still a considerable limit of state-of-the-art HAS techniques.

The exploitation of a QoE modeling approach grounded on Knowledge Representation (KR) techniques should overcome these limitations. Context features and QoE profiles are annotated through standard Semantic Web languages (*e.g.*, OWL 2 [26] or Resource Description Framework (RDF) [28]). In this way, a typical bit rate adaptation problem can be treated as a resource discovery task leveraging automated logical inferences. A key benefit of such ABR approaches is that outputs also provide human-understandable and computer-processable information, while explanation of matchmaking outcomes improves the user confidence on results and may identify actionable factors to further enhance the overall QoE.

3 RELATED WORK

Integrating Semantic Web technologies within adaptive systems (*e.g.*, HAS frameworks) is a relevant research trend. Several approaches have been proposed, exploiting reference ontologies as vocabularies to annotate context parameters, resource/service content and user requirements. A comprehensive survey on ontology-based solutions in adaptive systems is in [27], which proposes the Requirements-based Adaptive Systems Ontology (RASO) to model both domain and adaptation requirements. It aims to improve understanding, knowledge sharing and creation of requirement-based approaches for developing adaptive systems. A knowledge-based approach for distributed network applications is also in [9], where different QoE policies are defined for each provided service, and an inference engine is adopted to make a comparison with expected quality and detect possible degradation. Semantic Web Rule Language (SWRL) [16] is used as representation language, making the approach simply viable, albeit preventing articulated QoE optimization. [6] proposes a service-oriented and context-aware monitoring tool able to autonomously react to context changes. It includes different monitors which can be easily reconfigured and adapted to support dynamic environments and improve the QoE, and an ontology modeling features and capabilities of the system. A dedicated module manages the knowledge base of resources (modeled through context information, *e.g.*, time, location, environment) and actions to be taken on them.

With particular reference to the video streaming field, Exposito and Gómez-Montalvo [13] exploit an ontology based on the ITU-T F.700 framework recommendation for multimedia services [17] to model user preferences and application priorities when using shared network resources. Particularly, preferences are translated to service classes for enabling a QoS-based streaming adaptation in home environments, whereas global user satisfaction is evaluated through a utility function. Knowledge representation techniques are also adopted in [10] where an IPTV monitoring service is modeled in RDF. Network data are enriched through semantic attributes and processed using event aggregation procedures in order to identify network state transitions. Well-known network conditions are also modeled and annotated, and decoupled into atomic SWRL rules, which could trigger a system reconfiguration. Finally, in [8] KR is used to perform a context-driven service and content adaptation. Available services are described by means of OWL profiles following the Inputs, Outputs, Preconditions and Effects (IOPE) model. SWRL rules dynamically drive adjustments of delivered multimedia content according to the context changes and trigger bit rate variations. The solution also includes a rule-based inference engine, with the ensuing limitation that the context should fully match a rule head in order to trigger its body.

Table 1 sketches a specific comparison of the approach proposed here with related adaptive QoE systems leveraging Semantic Web technologies, in order to highlight the main features of existing semantic-based frameworks within the wide QoE literature, which includes proposals exploiting a wide range of methods and technologies. It should be pointed out that neither potential/partial correspondences nor explanations of outcomes are provided yet in existing approaches. In fact, full matches rarely occur in real-world scenarios, whose entities are featured by detailed, heterogeneous and often contradictory information, unless one uses very basic rules. On the contrary, the proposed framework supports non-monotonic inferences allowing also approximate matches, which can yield suitable results whenever full matches are not available. Moreover, reasoning results also include a logic-based numeric score and user-oriented annotations aiming to increase the awareness of the user about the applied QoE configuration.

Table 1: Comparison of the proposed approach with related semantic-based adaptive systems

Work	Year	QoE Metrics*	Use case	Representation Language	Adaptation Mechanism	Match Types	Explanation & Ranking
Exposito and Gómez-M. [13]	2010	U, N	Multimedia and distributed applications	OWL	User-defined static policies	Exact only	No
Chellouche *et al.* [8]	2014	U, N, A, D, CX	Home multimedia services	OWL	SWRL rules	Exact only	No
de Fréin *et al.* [10]	2015	N, A, D	IPTV services	RDF/OWL	SPARQL query + SWRL rules	Exact only	No
da Silva *et al.* [9]	2018	U, N, A, CT, CX	Ambient Assisted Living home services	OWL	SWRL rules	Exact only	No
Cabrera *et al.* [6]	2021	U, CT, CX	Context-aware monitoring systems	OWL	Service orchestration	Exact only	No
Proposed approach	2022	N, D, CT, CX	Fruition of multimedia contents	OWL	Semantic matchmaking	Exact and approximated	Yes

*Available metrics: user profile (U), network conditions (N), application constraints (A), device features (D), media content (CT), context (CX).

Other works in literature have shown how non-standard inferences introduce relevant benefits in terms of explanation capabilities, also increasing the confidence of the user in system outcomes. The proposed approach is basically general-purpose and it has been applied in different scenarios, *e.g.*, healthcare [24], home and building automation [32], cyber-physical systems [31]. These properties make the exploited matchmaking approach more flexible and accurate with respect to classical frameworks by Shu *et al.* [35] –which could not evaluate matches between pairs of disjoint concept descriptions– and by Li and Horrocks [23], which considered approximate match categories in the same way as our proposal but was not able to rank matches within each category. More recent proposal either adopt variants of Li and Horrocks' approach inheriting its limitations, such as [12], or mix semantic matchmaking with stochastic techniques like latent Dirichlet allocation [7] or random forests [21], making explanation of outcomes less clear.

4 QOE ADJUSTMENT VIA NON-STANDARD REASONING

In order to find the best QoE profile, all contextual and network features characterizing a QoE assessment are expressed as semantic annotations by using a proper set of logic operators. Non-standard inference services enable a matchmaking between the fruition context and QoE template profiles. The outcome is a ranking which enables to adjust quality and settings automatically, while providing a machine-understandable and human-readable explanation.

In the following subsections, the proposed framework is outlined: both system architecture and the theoretical scheme for semantic-based QoE adaptation are described in detail.

4.1 Framework architecture

The reference architecture for the proposed system is depicted in Figure 1. The *resource server* is a common HTTP-based Web server hosting the media content segmented according to MPEG-DASH specification. Different video quality profiles and encodings are supported. The *streaming client* application runs a MPEG-DASH compliant *media player*. In a Web-oriented scenario, the client is a browser endowed with a plug-in player typically implemented in JavaScript. Futhermore, this architectural style allows QoE adaptation functionalities to be directly integrated into applications for several platforms, *e.g.*, desktop PCs, notebooks, smartphones, tablets and smart TVs.

In greater detail, the client also includes: (i) a *context manager* aiming to collect information characterizing fruition conditions; (ii) an *HTTP daemon* managing the communication with the resource server; (iii) the *QoE adaptation module*, including the matchmaking engine providing non-standard inferences for QoE profile selection. The latter component receives both context and network parameters from the streaming client and builds an annotated description according to the modeling guidelines in Section 4.2.

The QoE adaptation module has been implemented in WebAssembly 1.0 (Wasm; https://webassembly.org/specs/) [14], a binary application format proposed as portable compilation target for different programming languages. Wasm enables the deployment of Web components for both client and server applications and introduces several benefits with respect to plug-in runtime environments: full compatibility with major browser engines; browser-Wasm interaction based on standardized Web APIs (https://www.w3.org/TR/wasm-web-api-1/); optimized processing directly exploiting the host hardware environment; compliance with same-origin and permissions security policies of the browser. The client-side graphical user interface (GUI) is simple and straightforward: it shows inference results and allows managing manual user selection of video quality settings. In this case, if the selected quality appears inappropriate with respect to the context conditions, the user receives a warning message along with an explanation of contrasting features.

Figure 1 also sketches the basic workflow for semantic-based QoE adaptation. The main steps are summarized hereafter:

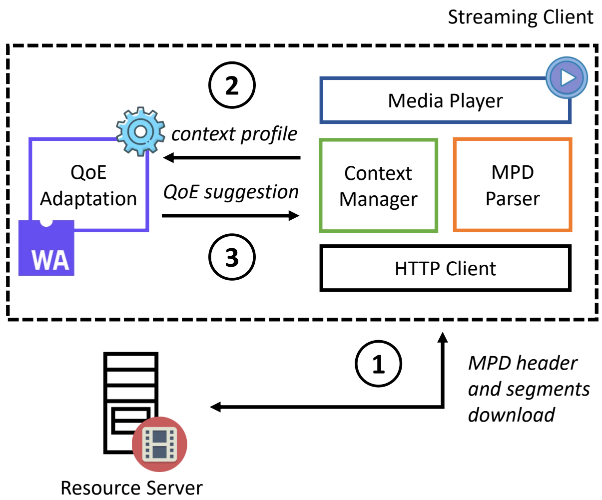


Figure 1: Framework architecture

(1) the streaming client loads the reference Web page and the media application starts playing video, either triggered by the user or autonomously in case of `autoplay` enabled;
(2) the player downloads the MPD file from the resource server and parses it to get information about segments composing the multimedia resource;
(3) contextual and network parameters are collected and sent to the QoE adaptation module. An OWL-based annotation is dynamically created according to the reference ontology and submitted to a matchmaking process against a set of QoE template profiles (see Section 4.3). To avoid unnecessary computation, a reasoning task is only started if the client detects a variation in the observed parameters;
(4) the matchmaking process returns the most suitable profile along with: (i) a relevance score depending on the semantic distance between the current context and the selected QoE profile; (ii) values of settings associated to the selected profile (*e.g.*, resolution, brightness, contrast, saturation); (iii) a list of compatible and contrasting features as identified by the applied inference;
(5) according to the above results, the media player retrieves the next segment with the appropriate resolution, while also adjusting the settings of the video panel if needed. An additional MPD header is downloaded from the resource server when a new video quality is suggested for the first time;
(6) information about compatible and conflicting characteristics contributing to the matchmaking output are displayed on the explanation section of the interface.

Steps 3-6 are repeated for all segments until the video playback ends.

4.2 Knowledge Graph modeling

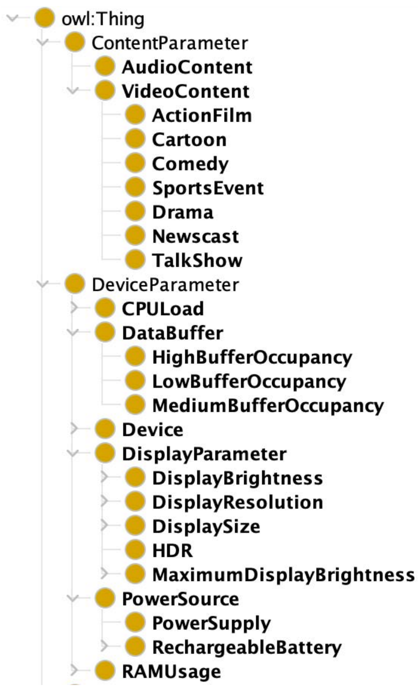
In order to support semantic-based data annotation and interpretation, a reference ontology $\mathcal{T}$, available at http://swot.sisinflab.poliba.it/onto/semantic-qoe, has been defined to model the domain conceptualization detailed hereinafter. In particular, two well-known RDF vocabularies are used as upper-ontologies, *i.e.*, DBpedia [22] (modeling basic concepts about media content and device/network parameters) and DogOnt [4] (concepts for home environments, home/building automation and appliances). $\mathcal{T}$ is assumed as acyclic and expressed in the moderately expressive $\mathcal{ALN}$ (Attributive Language with unqualified Number restrictions) DL. This is required by the adopted non-standard inference services for matchmaking settings [30]. The resulting Knowledge Graph (KG) is composed by: (i) a formal conceptual model of the media streaming domain (Terminological Box $\mathcal{T}$); (ii) assertions about *instances* (Assertion Box, ABox), including streaming profiles for ideal QoE in given fruition conditions.

As shown in Figure 2, the proposed ontology models QoE elements following the conceptualization proposed in [36]. Concepts are grouped in four categories: content, device, environment and network parameters. For each measurable feature (*e.g.*, buffer occupancy, network delay, battery power), $\mathcal{T}$ includes a hierarchy forming a partonomy of the topmost concept. In this way, each parameter is modeled through a taxonomy representing all significant value ranges and configurations it can take in the domain of interest. While expanding the model with additional features, the depth of the hierarchy and the breadth of each level will be chosen by the knowledge modeler, and they are typically proportional to either value ranges or specific subclasses suitable for each parameter, as well as to the needed degree of detail in data representation.

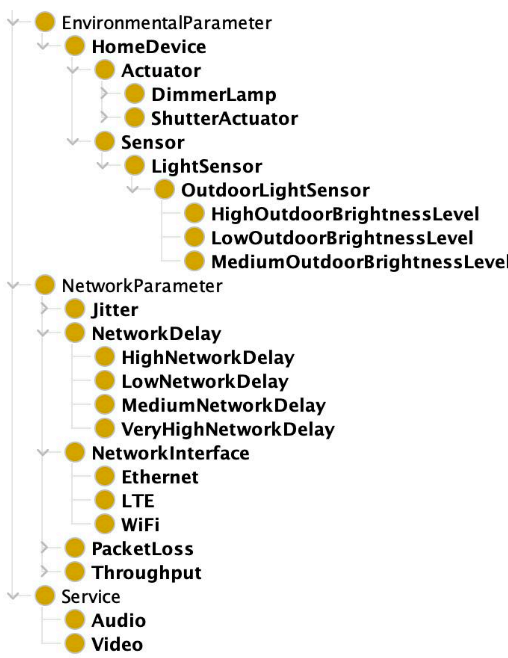
According to the proposed modeling, QoE guidelines can be translated to a set of OWL individuals where each streaming profile is associated to an instance in the reference KG. As depicted in Figure 3, each individual is defined by means of: (i) an OWL-based description expressed as the conjunction of *concepts*, usually combined using different *object properties* (*e.g.*, hasDeviceParameter) through existential role quantification, universal role quantification and number restrictions; (ii) a set of *annotation properties* representing extra-logical attributes related to the suggested video settings (*e.g.*, brightness, resolution) ensuring optimal QoE. The KG currently includes an initial set of configuration parameters, which have provided control over the modeling effort while enabling a full effectiveness evaluation of the proposed knowledge-based approach. By extending the KG, additional concepts and annotation properties can be modeled in order to further detail QoE profiles without changing the overall framework or the inference-based matchmaking.

4.3 Automated reasoning for QoE adaptation

The matchmaking engine in [30] has been adapted to run in a WebAssembly environment. Its port has been straightforward as it is implemented in the C programming language, therefore it has been cross-compiled to Wasm with minor configuration of the *CMake* (https://cmake.org) build system, exploiting the *Emscripten* toolchain (https://emscripten.org). While it would be possible to just compile the C core of the matchmaker and expose its procedural API to JavaScript, better interoperability has been attained by



(a) Content and device parameters



(b) Context and network parameters

Figure 2: Conceptual model of the media streaming domain

implementing a thin object-oriented C++ wrapper and generating bindings to JavaScript classes through the *Embind* tool[3].

[3]Related documentation: https://emscripten.org/docs/porting/connecting_cpp_and_javascript/embind.html

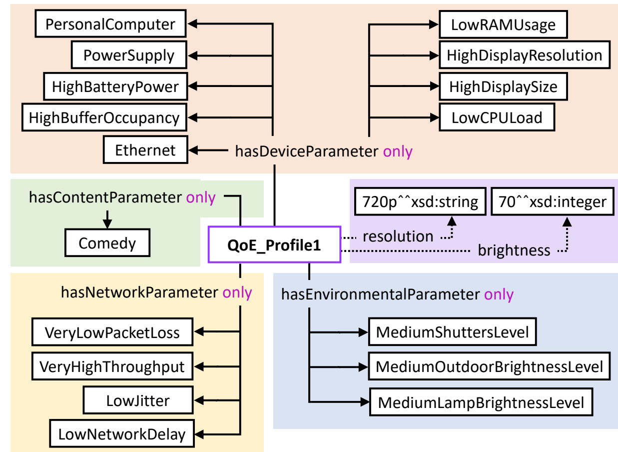


Figure 3: Example of QoE profile annotation

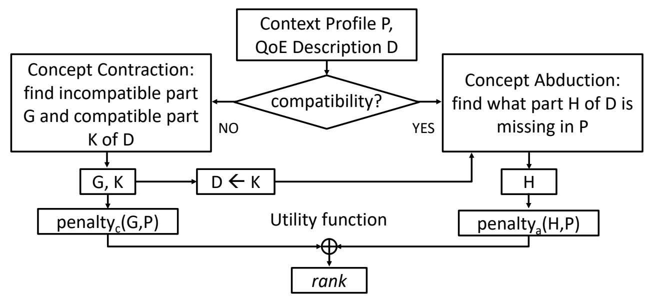


Figure 4: Semantic-based matchmaking for QoE adaptation

The matchmaker computes a score based on the semantic *affinity* between an OWL-based annotation of current client-side context and available QoE profiles. Basically, a DL concept expression of contextual profile P is compared to QoE description D as in the flowchart in Figure 4, by means of the *Concept Contraction* (CC) and *Concept Abduction* (CA) non-standard, non-monotonic reasoning services [29]. In case D and P contain clashing concepts, there is a *partial match* between them, and CC is applied to determine a pair $\langle G, K \rangle$ representing what has to be retracted G (for *Give-up*) and what can be kept K (for *Keep*) in D, in order for K to reach a *potential match* with P. Basically, G includes the elements of D conflicting with P, while K is the (best) contraction of D compatible with P. In case P and D are not conflicting, but P does not completely satisfy D, there is a *potential match*, and CA is applied to find the concept H (for *Hypothesis*) specifying what has to be hypothesized in P to reach a *full match* with D (or its contracted version K). Both G and H can be also used to enable a refinement process where the user could act to adapt the context conditions and improve the media streaming experience, by *e.g.*, closing a concurrent application or switching from cellular to Wi-Fi connection.

Inferences are implemented via structural algorithms based on *Conjunctive Normal Form* (CNF) normalization of concept expressions, as detailed in [29]. Since a concept CNF is unique, a semantic-based *penalty* can be associated to Concept Contraction and Concept Abduction and computed for every (D, P) pair on a "norm" on the solutions G and H. Formal minimality criteria exist to give up or hypothesize as little as possible [29]. Finally, the following *rank* formula translates the semantic distance measure to a $0 - 100\%$ ascending scale of *relevance*:

$$penalty(D, P) = \frac{\alpha * penalty_c(G, P) + \beta * penalty_a(H, P)}{penalty_a(P, \top)}$$

$$rank(D, P) = 100 * [1 - penalty(D, P)]$$

where $penalty_a(H, P)$ and $penalty_c(G, P)$ measure the Abduction-induced and the Contraction-induced semantic distances, respectively. These values are also normalized dividing by the penalty between P and the universal concept *Thing* (a.k.a. *Top*, $\top$), which depends exclusively on axioms in the reference ontology, since concept descriptions are *unfolded* before reasoning [29]. According to the inference outcome shown in Figure 4, the embedded matchmaker computes $penalty_a$ between D and P if the two annotations are compatible, *i.e.*, their conjunction is satisfiable. If it detects incompatibility, Concept Contraction is performed to extract the conflicting *Give-up* part of the request and evaluate the induced $penalty_c$ value. Then it computes $penalty_a$ again on the remaining *Keep* part. Two tunable weighting factors (α and β) in the *rank* formula combine both contributions and enable a ranking mainly based on either conflicting or missing features.

The above procedure is repeated for all QoE descriptions defined within the KG, comparing each one to the current context profile P. The QoE description with the highest rank is selected. Hence, the adopted approach: (i) produces a fine-grained logic-based profile ranking identifying the most appropriate QoE settings; (ii) detects possible incompatibilities between a proposed streaming profile and contextual conditions; (iii) provides matchmaking outcomes which are both human-readable and machine-understandable, along with a rigorous logic-based explanation consisting of the concept expressions G and H.

5 CASE STUDY

A fully functional Web-based prototype has been developed to highlight peculiarities of the proposed framework. An illustrative example is proposed hereafter to clarify the semantic-based inferences by presenting the QoE adaptation procedure in a basic usage scenario. Semantic-based annotations related to context profiles and QoE individuals are always verified by the matchmaking engine in order to detect inconsistencies within and between concept descriptions.

The system implementation is based on the following tools. The open-source Apache HTTP server (https://httpd.apache.org, version 2.4.46) has been used as resource server hosting the *Big Buck Bunny* (https://peach.blender.org) short movie, an open video project developed by the Blender Foundation and freely available under the Creative Commons Attribution 3.0 license. In particular, the server exposes a segmented version of the movie available at http://ftp.itec.aau.at/datasets/mmsys12/BigBuckBunny/, structured according to the MPEG-DASH specification. It consists of segments with a length of 2 seconds and an increasing bit rate from 200 to 2000 kbps. Mozilla Firefox desktop edition (https://www.mozilla.org/firefox, version 88.0.1) has been selected as reference browser for the client application.

The user displays the GUI in Figure 5 through the browser: it consists of an HTML page linking the modules described in Figure 1 (*i.e.*, HTTP client, MPD parser and context manager), developed in JavaScript. The QoE adaptation module is implemented in WebAssembly as described above.

The GUI consists of two main sections. On the left hand side, it includes: a Web player supporting the MPEG-DASH protocol; two sets of icons (in violet) denoting resolution and brightness related to the QoE profile selected through the matchmaking procedure; a rank bar representing the semantic-based similarity score. On the right hand side, a simple explanation section summarizes parameters of the current context profile. They are associated to illustrative icons and grouped by category (blue background) following the same taxonomy of the ontology. Only the device type is shown as a separated element, in order to improve its visibility on the GUI. The explanation section has a twofold purpose. Basically, it shows the status of fruition parameters exploiting a color-coded legend: icons with a light blue background indicate active on/off functionalities (*e.g.*, Wi-Fi network access) or equipment/stream properties (*e.g.*, device type), whilst a gradient color palette (ranging from light green to red) highlights detected features with a value range from very low to very high. In both cases, the *context manager* manages the correspondences between GUI elements and concepts modeled in the KG. Moving the mouse pointer over an icon, a tooltip text also appears showing the name of the related OWL class. For example, following the scenario in Figure 5, the context profile P is expressed (in OWL 2 Manchester syntax [15] and with respect to the reference KG) as:

```
P ≡ (hasContentParameter only Comedy) and (hasDeviceParameter
only (MediumBatteryLevel and LTE and HighCPULoad and
HighDisplayResolution and HighRAMUsage and LowBufferOccupancy
and MediumDisplaySize and Notebook)) and
(hasEnvironmentalParameter only (HighShutterLevel and
HighLampBrightnessLevel and MediumOutdoorBrightnessLevel)) and
(hasNetworkParameter only (HighPacketLoss and HighNetworkDelay
and HighJitter and LowThroughput))
```

Basically, the client detects high usage of local processing and memory and there is a network congestion and high brightness within the room where the user is watching a comedy video. The original QoE individual reported in Figure 3 includes several missing or contrasting features, resulting inadequate for such detected context. Hence the semantic-based matchmaking is invoked and retrieves the following QoE annotation D presenting the highest logic affinity (about 89%) with the context situation.

```
D ≡ (hasContentParameter only Comedy) and (hasDeviceParameter
only (LowBatteryLevel and LTE and HighCPULoad and
HighDisplayResolution and HighRAMUsage and LowBufferOccupancy
and LowDisplaySize and Smartphone)) and
(hasEnvironmentalParameter only (MediumOutdoorBrightnessLevel
and MediumLampBrightnessLevel and MediumShutterLevel)) and
```

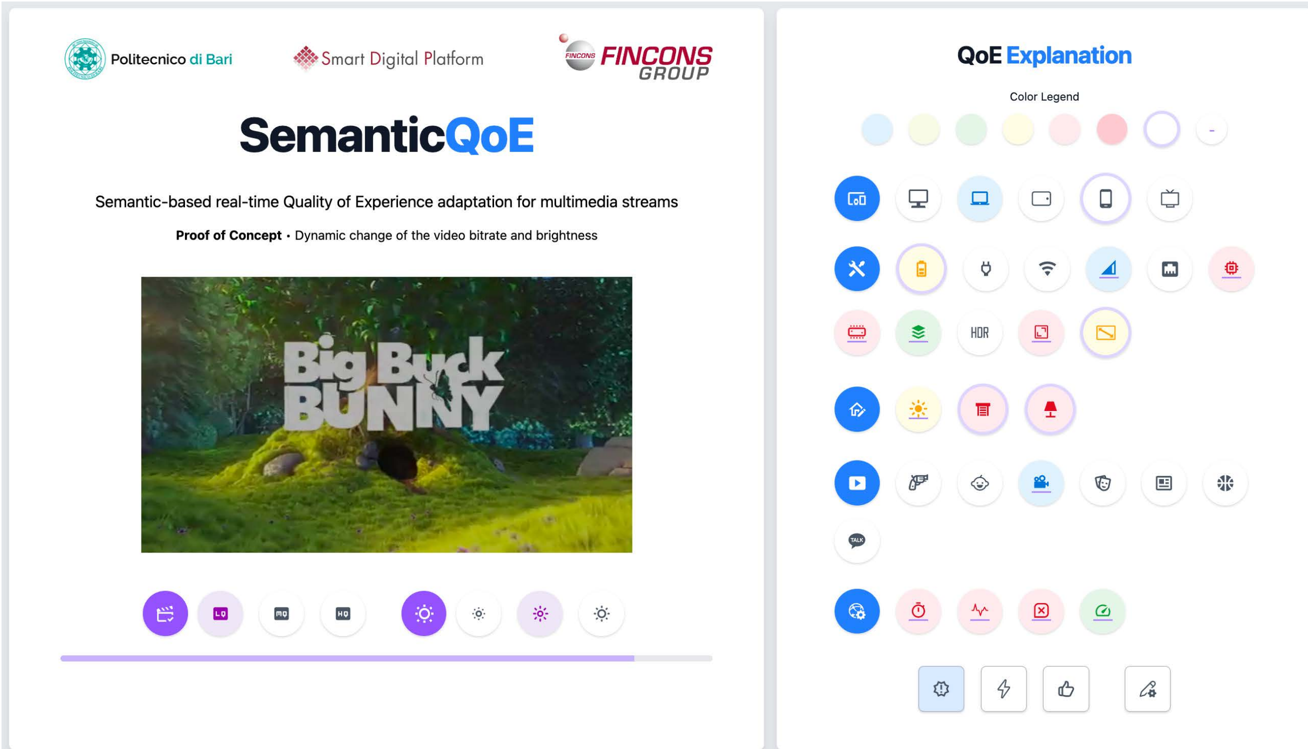


Figure 5: Streaming client GUI

(hasNetworkParameter **only** (HighPacketLoss **and** HighNetworkDelay **and** HighJitter **and** LowThroughput))

Notice an appropriate video experience is granted now, although a partial match is obtained also in this case. In particular, *D* proposes a low video quality suitable for smartphones and devices with constrained hardware performance and limited network bandwidth. A medium video brightness is also suggested considering the internal/outdoor light conditions and window shutters partially opened. Matchmaking outcomes, detailed below, are also reported in the GUI of Figure 5 through the aforementioned components.

G (give-up) ≡ (hasEnvironmentalParameter **only** (MediumShutterLevel **and** MediumLampBrightnessLevel)) **and** (hasDeviceParameter **only** (LowBatteryLevel **and** LowDisplaySize)

K (keep) ≡ (hasContentParameter **only** Comedy) **and** (hasDeviceParameter **only** (LTE **and** HighCPULoad **and** HighRAMUsage **and** LowBufferOccupancy **and** HighDisplayResolution) **and** (hasNetworkParameter **only** (HighNetworkDelay **and** HighJitter **and** HighPacketLoss **and** LowThroughput)) **and** (hasEnvironmentalParameter **only** MediumOutdoorBrightnessLevel)

H (hypothesis) ≡ (hasDeviceParameter **only** Smartphone)

In the GUI icons, a horizontal bar underlines active features which are compatible with the selected QoE profile, *i.e.*, the *K* part of the related OWL annotation. For example, in Figure 5 the icons for CPU load and memory occupancy are both red –as a high level has been detected by the client– and underlined, since they match the reference profile. Buffer occupancy icon is underlined as well, but it is colored in green as the observed level is low. Missing or contrasting features (*i.e.*, *H* and *G*, respectively) detected through the Concept Abduction or Concept Contraction algorithms are depicted using a bold violet border. Also in this case, a tooltip shows inference results reporting on concepts which would ensure a full match with the QoE annotation. The meaning of all graphical elements is summarized by means of a legend displayed in the upper part of the explanation section.

For experimentation purposes, the right-hand section of the GUI doubles as an interactive configuration panel in the prototype. By clicking on a feature, the user can modify the related value through a simple popup window listing available alternatives. This makes it possible to simulate variations of the context parameters and verify how the QoE adaptation module reacts according to the updated condition. In the bottom part of the section, four buttons allow activating a set of predefined scenarios (requiring different QoE configurations) or to apply a given context profile customized by the user (button with a pencil icon).

Finally, a user can also set a particular video resolution or brightness by clicking on the related icons on the left-hand section. In this case, the logic-based automatic adaptation of the QoE will be temporarily deactivated and the matchmaking process will be only used to compare the configuration selected by the user with the client context conditions, displaying the relevance rank outcome in the bottom horizontal bar. In real scenarios, users often select inappropriate streaming settings; this approach provides useful support by immediately highlighting the motivations of the configuration mistake via the explanation GUI section. The automatic QoE adaptation is reactivated by simply clicking on the pencil icon.

6 PERFORMANCE

A preliminary assessment of the implemented prototype has concerned compatibility with modern browsers. The following browser/platform combinations have been tested: Chrome (version 90.0.4430) and Firefox (version 88.0.1) on Windows 10 May 2020 Update, Ubuntu 20.10, macOS Big Sur and Android 10; Edge (version 88.0.-705.74) on Windows 10; Safari (version 14.0) on macOS, iPadOS 14 and iOS 14. Full compatibility has been observed in all tests, with the only exception of Safari on iOS, which is currently missing significant portions of the *Media Source Extensions* [38] standard API. The computational performance of the QoE Adaptation module has been also evaluated in an early experimental campaign. Tests have been carried out on an *Apple MacBook Pro (2017)*[4], using the same software setup as the one in Section 4.3, *i.e.*, using an Apache HTTP server (version 2.4.46) and Mozilla Firefox browser (version 88.0.1). All performance results are the average of ten cold runs, *i.e.*, restarting the browser between each run.

The first test concerns turnaround time of the matchmaking task, which has been dissected into the following sub-tasks:

(1) KG parsing and loading time;
(2) context profile P creation and modeling with respect to the reference KG;
(3) semantic matchmaking between the context description P and each QoE profile in the KG. Each pair of individuals has been processed with the algorithm in Figure 4, with the aim of selecting the most suitable profile.

It is important to point out that step (1) only happens once per session, while (2) and (3) are performed multiple times during playback, with a frequency that depends on how often the context changes.

Figure 6 shows the time to perform each individual step. The most relevant result is that context profile creation and semantic matchmaking take less than 3 ms cumulatively, *i.e.*, they can be executed over 300 times per second, which means QoE adaptation can be carried out with per-frame accuracy. This is of course overabundant, though it confirms the suitability of the proposed system for real-life multimedia stream QoE adaptation.

The second and last test deals with memory usage of the Wasm matchmaker, profiled via the snapshotting capabilities embedded into the Firefox developer tools. By default, the Emscripten compiler statically allocates a 16 MB contiguous chunk of memory for the Wasm module, which is problematic because it does not allow to know how much memory is actually needed at runtime by the matchmaking task. Therefore for testing purposes the initially allocated memory has been decreased to 192 kB, the minimum amount for which the code would compile, and the Wasm module has been recompiled with the `ALLOW_MEMORY_GROWTH` flag[5] enabled, which lets the browser allocate additional memory for the module if necessary. Given the above configuration, the following memory

[4]Intel Core i7-7820HQ quad-core CPU at 2.9 GHz, 16 GB LPDDR3 RAM at 2133 MT/s, 512 GB SSD, macOS Big Sur 11.3.1
[5]Related documentation: https://emscripten.org/docs/optimizing/Optimizing-Code.html

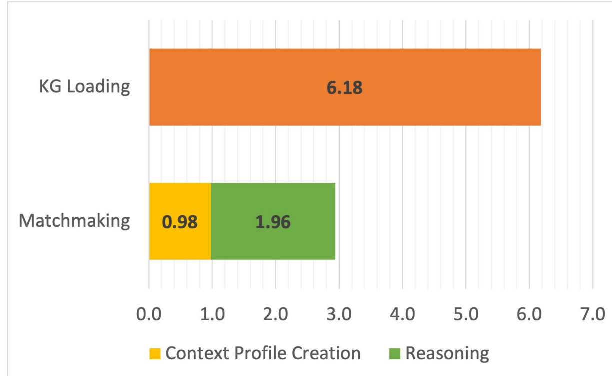


Figure 6: Turnaround time (ms)

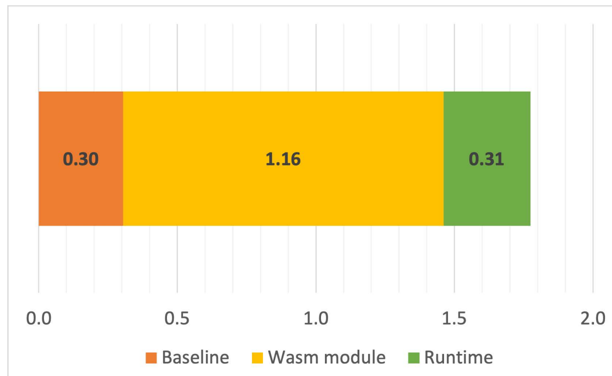


Figure 7: Wasm memory usage (MB)

snapshots have been taken: (i) *baseline:* blank page, before loading the Wasm module; (ii) *wasm module:* after loading the Wasm module; (iii) *runtime:* memory usage peak, while matchmaking.

As evidenced by Figure 7, loading the Wasm module introduces a 1.16 MB memory overhead over the baseline, which includes the Wasm memory buffer (initially sized at 192 kB, as said). While matchmaking, an additional maximum of 0.31 MB of memory is allocated, and the Wasm memory buffer is dynamically resized by the browser to 320 kB. These results evidence a tolerable memory overhead for the matchmaking process, which corroborates the feasibility of the proposed approach.

7 CONCLUSION

This paper has introduced a novel semantic-based framework for real-time adaptation of Quality of Experience in Web multimedia streaming. A Knowledge Graph provides the vocabulary for annotating dynamically both fruition context and reference service profiles. Moreover, non-standard $\mathcal{ALN}$ DL inferences allow (i) comparing profiles with the current fruition scenario by supporting both exact and approximate matches, (ii) ranking profiles by semantic affinity, and (iii) explaining matchmaking outcomes, evidencing the missing or conflicting elements in the streaming context, which might be fixed to improve the overall QoE. Profile selection enables bit rate adaptation in the MPEG-DASH standard to grant smooth playback at the most suitable quality level option. A Web application prototype fully implements the framework, which is compatible with major Web browsers and standard HTTP servers endowed with MPEG-DASH resources. Reasoning is performed by a Description Logics inference engine for the WebAssembly runtime embedded in the client Web application. This is a notable novel feature, which may have a wide range of applications in the future. A case study has been devised to validate the proposal, while early experimental results support its computational feasibility.

Future work will expand the KG by including further context parameters as well as specializing the service profile descriptions. A comparative experimental evaluation with other state-of-the-art QoE adaptation techniques, based on subjective and/or metric-based approaches, is a relevant research goal. Ongoing investigations also concern the integration with IoT and home and building automation solutions to gather context characterization features. Finally, extensive compatibility and performance tests on mobile devices, Smart TVs and other device categories are planned.

ACKNOWLEDGMENTS

The work has been funded by Fincons Group S.p.A. "Smart Digital Platform" grant, award no. 4500008490-10-20-30. The authors are grateful to Riccardo Vescovi for his continued support.

REFERENCES

[1] Zahid Akhtar and Tiago H Falk. 2017. Audio-visual Multimedia Quality Assessment: A Comprehensive Survey. *IEEE Access* 5 (2017), 21090–21117.
[2] Franz Baader, Diego Calvanese, Deborah L McGuinness, Daniele Nardi, and Peter Patel-Schneider. 2002. *The Description Logic Handbook.* Cambridge University Press, Cambridge, United Kingdom.
[3] Nabajeet Barman and Maria G Martini. 2019. QoE Modeling for HTTP Adaptive Video Streaming–A Survey and Open Challenges. *IEEE Access* 7 (2019), 30831–30859.
[4] Dario Bonino and Luigi De Russis. 2018. DogOnt as a viable seed for semantic modeling of AEC/FM. *Semantic Web* 9, 6 (2018), 763–780.
[5] Kjell Brunnström, Sergio Ariel Beker, Katrien De Moor, Ann Dooms, Sebastian Egger, Marie-Neige Garcia, Tobias Hossfeld, Satu Jumisko-Pyykkö, Christian Keimel, Mohamed-Chaker Larabi, et al. 2013. *Qualinet White Paper on Definitions of Quality of Experience.* Technical Report. Qualinet consortium.
[6] Oscar Cabrera, Marc Oriol, Xavier Franch, and Jordi Marco. 2021. A context-aware monitoring architecture for supporting system adaptation and reconfiguration. *Computing* published online (mar 2021), 1–35. https://doi.org/10.1007/s00607-021-00923-z
[7] Gilbert Cassar, Payam Barnaghi, Wei Wang, and Klaus Moessner. 2012. A hybrid semantic matchmaker for IoT services. In *2012 IEEE International Conference on Green Computing and Communications.* IEEE, IEEE Computer Society, Los Alamitos, CA, USA, 210–216.
[8] Soraya Ait Chellouche, Daniel Negru, Julien Arnaud, and Jordi Mongay Batalla. 2014. Context-aware multimedia services provisioning in future Internet using ontology and rules. In *2014 International Conference and Workshop on the Network of the Future (NOF).* IEEE, Piscataway, NJ, USA, 1–5.
[9] Madalena Pereira da Silva, Felipe Volpato, and MAR Dantas. 2018. Proactive Autonomic Semantic Engine to User Experience-Aware Service Provision. In *2018 IEEE Symposium on Computers and Communications (ISCC).* IEEE, Piscataway, NJ, USA, 00836–00842.
[10] Ruairí de Fréin, Cristian Olariu, Yuqian Song, Rob Brennan, Patrick McDonagh, Adriana Hava, Christina Thorpe, John Murphy, Liam Murphy, and Paul French. 2015. Integration of QoS Metrics, Rules and Semantic Uplift for Advanced IPTV Monitoring. *Journal of Network and Systems Management* 23, 3 (2015), 673–708.
[11] Anind K Dey. 2001. Understanding and using context. *Personal and ubiquitous computing* 5, 1 (2001), 4–7.
[12] Giuseppe Di Modica and Orazio Tomarchio. 2016. Matchmaking semantic security policies in heterogeneous clouds. *Future Generation Computer Systems* 55 (2016), 176–185.
[13] Ernesto Exposito and Jorge Gómez-Montalvo. 2010. An ontology-based framework for autonomous QoS management in home networks. In *2010 Sixth International Conference on Networking and Services.* IEEE, Piscataway, NJ, USA, 117–122.
[14] Andreas Haas, Andreas Rossberg, Derek L Schuff, Ben L Titzer, Michael Holman, Dan Gohman, Luke Wagner, Alon Zakai, and JF Bastien. 2017. Bringing the Web up to speed with WebAssembly. In *Proceedings of the 38th ACM SIGPLAN Conference on Programming Language Design and Implementation.* ACM, New York, NY, USA, 185–200.
[15] Matthew Horridge and Peter Patel-Schneider. 2012. *OWL 2 Web Ontology Language Manchester Syntax (Second Edition).* W3C Note. World Wide Web Consortium. http://www.w3.org/TR/owl2-manchester-syntax/.
[16] Ian Horrocks, Peter F Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, Mike Dean, et al. 2004. *SWRL: A semantic web rule language combining OWL and RuleML.* W3C Member submission. World Wide Web Consortium. https://www.w3.org/Submission/SWRL/.
[17] International Telecommunication Union. 2000. *F.700: Framework Recommendation for multimedia services.* Technical Report. ITU.
[18] International Telecommunication Union. 2017. *Vocabulary for performance, quality of service and quality of experience.* Technical Report. ITU-T.
[19] International Telecommunication Union. 2020. *Measuring digital development: Facts and figures 2020.* Technical Report. ITU.
[20] ISO/IEC. 2019. *ISO/IEC 23009-1:2019 Information technology — Dynamic adaptive streaming over HTTP (DASH) .* Technical Report. ISO/IEC.
[21] Wei Jiang, Junyu Lin, Huiqiang Wang, and Shichen Zou. 2020. Hybrid semantic service matchmaking method based on a random forest. *Tsinghua Science and Technology* 25, 6 (2020), 798–812.
[22] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. DBpedia – A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic web* 6, 2 (2015), 167–195.
[23] L. Li and I. Horrocks. 2004. A Software Framework for Matchmaking Based on Semantic Web Technology. *International Journal of Electronic Commerce* 8, 4 (2004), 331–339.
[24] Giuseppe Loseto, Floriano Scioscia, Michele Ruta, Filippo Gramegna, Saverio Ieva, Agnese Pinto, and Crescenzio Scioscia. 2020. Knowledge-based Decision Support in Healthcare via Near Field Communication. *Sensors* 20, 17 (2020), 4923.
[25] Andreea Molnar. 2017. Content type and perceived multimedia quality in mobile learning. *Multimedia Tools and Applications* 76, 20 (2017), 21613–21627.
[26] Bijan Parsia, Sebastian Rudolph, Markus Krötzsch, Peter Patel-Schneider, and Pascal Hitzler. 2012. *OWL 2 Web Ontology Language Primer (Second Edition).* W3C Recommendation. World Wide Web Consortium. http://www.w3.org/TR/owl2-primer.
[27] Cássio Capucho Peçanha, Bruno Borlini Duarte, Ricardo de Almeida Falbo, and Vitor E Silva Souza. 2018. RASO: an Ontology on Requirements for the Development of Adaptive Systems. *Cadernos do IME-Série Informática* 41 (2018), 57–76.
[28] Yves Raimond and Guus Schreiber. 2014. *RDF 1.1 Primer.* W3C Note. W3C. https://www.w3.org/TR/rdf11-primer/.
[29] Michele Ruta, Eugenio Di Sciascio, and Floriano Scioscia. 2011. Concept Abduction and Contraction in Semantic-based P2P Environments. *Web Intelligence and Agent Systems* 9, 3 (2011), 179–207.
[30] Michele Ruta, Floriano Scioscia, Ivano Bilenchi, Filippo Gramegna, Giuseppe Loseto, Saverio Ieva, and Agnese Pinto. 2022. A multiplatform reasoning engine for the Semantic Web of Everything. *Journal of Web Semantics* 73 (2022), 100709.
[31] Michele Ruta, Floriano Scioscia, Saverio Ieva, Giuseppe Loseto, Filippo Gramegna, Agnese Pinto, and Eugenio Di Sciascio. 2017. Knowledge discovery and sharing in the IoT: the Physical Semantic Web vision. In *32nd ACM SIGAPP Symposium On Applied Computing.* Association for Computing Macinery, Inc. (ACM), ACM, New York, NY, USA, 492–498.
[32] Michele Ruta, Floriano Scioscia, Giuseppe Loseto, Filippo Gramegna, Saverio Ieva, Agnese Pinto, and Eugenio Di Sciascio. 2018. Social Internet of Things for Domotics: a Knowledge-based Approach over LDP-CoAP. *Semantic Web Journal* 9, 6 (2018), 781–802.
[33] Sandvine. 2019. *Global Internet Phenomena.* Technical Report. Sandvine. https://www.sandvine.com/phenomena.
[34] Michael Seufert, Sebastian Egger, Martin Slanina, Thomas Zinner, Tobias Hoßfeld, and Phuoc Tran-Gia. 2014. A survey on quality of experience of HTTP adaptive streaming. *IEEE Communications Surveys & Tutorials* 17, 1 (2014), 469–492.
[35] Gao Shu, Omer F Rana, Nick J Avis, and Chen Dingfang. 2007. Ontology-based semantic matchmaking approach. *Advances in engineering software* 38, 1 (2007), 59–67.
[36] Lea Skorin-Kapov and Martin Varela. 2012. A multi-dimensional view of QoE: the ARCU model. In *2012 Proceedings of the 35th International Convention MIPRO.* IEEE, Piscataway, NJ, USA, 662–666.
[37] Iraj Sodagar. 2011. The MPEG-DASH Standard for Multimedia Streaming Over the Internet. *IEEE MultiMedia* 18, 4 (2011), 62–67.
[38] Matthew Wolenetz, Jerry Smith, Mark Watson, Aaron Colwell, and Adrian Bateman. 2016. *Media Source Extensions).* W3C Recommendation. W3C. https://www.w3.org/TR/media-source/.
[39] Alex Zambelli. 2009. IIS smooth streaming technical overview. *Microsoft Corporation* 3 (2009), 40.