RideMATCHain: a Semantic-enhanced Blockchain Marketplace for Ridesharing

Filippo Gramegna*, Arnaldo Tomasino*, Saverio Ieva*, Ivano Bilenchi*, Agnese Pinto*, Giuseppe Loseto†, Floriano Scioscia*, Michele Ruta*
* Politecnico di Bari – Via E. Orabona 4, Bari (I-70125), Italy
{name.surname}@poliba.it
† Università LUM "Giuseppe Degennaro", Casamassima BA I-70010, Italy
loseto@lum.it

***Abstract*—The integration of blockchain platforms with Semantic Web technologies can increase information interoperability and enable knowledge-driven task automation by intelligent software agents. The paper presents a blockchain platform extending Hyperledger Sawtooth with semantic Smart Contracts allowing annotated resource registration, discovery, explanation, and selection. The platform leverages non-standard inferences for semantic matchmaking, endowed with logic-based justification of results. A prototypical semantic-enhanced ridesharing marketplace been built to validate the proposal.**

Index Terms*—Semantic matchmaking, Blockchain, Intelligent Transportation Systems

I. Introduction

The need for a trusted intermediary to ensure transaction integrity can be a strong limitation for several classes of Intelligent Transportation Systems (ITSs) applications open to the general public. To overcome this issue, *blockchain* has emerged as a family of data structures and protocols for peer-to-peer *trustless* distributed transactional systems. Blockchain platforms approve transactions by means of peer-to-peer *consensus* protocols. Many blockchains can host *Smart Contracts* (SCs), programs encoding and enforcing terms of an agreement among two or more parties as executable procedures. This opens up the integration with Artificial Intelligence (AI). In particular, Semantic Web technologies –grounded on formal Knowledge Representation and Reasoning– promote information interoperability and enable intelligent software agents to automate data-driven tasks. In [1] blockchain assets are endowed with a semantic-based structured representation, in order to combine the blockchain benefits of trustworthiness and traceability with semantic-based service/resource discovery capabilities. Smart Grid and Smart Mobility marketplaces [2] are among the leading target applications.

This paper introduces a semantic-enhanced blockchain-based platform for multi-domain service/resource marketplaces. It extends the *Hyperledger Sawtooth* project [3] with semantic SCs for managing the registration, discovery, explanation and selection steps of service/resource lifecycle. Discovery exploits semantic matchmaking via non-standard inference tasks, capable of ranking a set of available resources by semantic affinity w.r.t. a request, where the request and the resources are annotated referring to a common domain ontology. The adopted inferences also grant logic-based explanation of outcomes, thus making requesters not only trust the blockchain platform for transaction security, but also AI facilities through interpretable results. On top of the platform, a prototypical semantic-enhanced *ridesharing* marketplace has been created, supporting detailed specification of vehicle and driver characteristics as well as passenger preferences. Section II describes in detail the proposed framework, while the ridesharing service is in Section III, before conclusion.

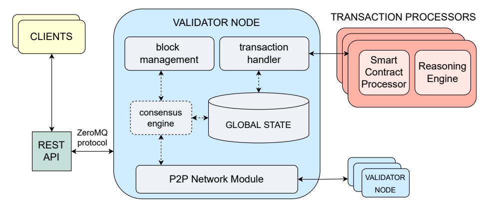

Fig. 1: Semantic-enhanced blockchain architecture

II. Semantic blockchain platform

Figure 1 depicts the extended Hyperledger Sawtooth architecture. *Transaction Processors* (TPs) execute transactions according to the (i) payload format, (ii) state information model and (iii) transaction validation procedure defined for a *Transaction Family* (TF). *Validator Nodes* (VNs) manage the global state by (i) interacting with TPs, (ii) coordinating the communication with their peers to apply the consensus procedure, and (iii) verifying incoming blocks and creating new ones. *Clients* query VNs to know the state of the blockchain and submit transaction batches via a REST API (REpresentational State Transfer Application Programming Interface) over the *ZeroMQ* protocol (https://zeromq.org/). The proposed framework extends Sawtooth TPs, by providing a set of SCs to manage assets –representing physical or digital resources, or service instances– semantically annotated w.r.t. a domain ontology and perform inference procedures. For this purpose, a novel TF has been defined along with the corresponding SC Processor, which embeds the *Tiny-ME* reasoning engine [4] to execute the semantic SCs. Storing semantically annotated resource descriptions as assets enables traceability of discovery sessions and resource updates by means of the blockchain, without resorting to external information sources.

Non-standard inference services are exploited in semantic matchmaking to achieve a fine-grained ranking of matches. Given a request R and a resource S annotated w.r.t. a common ontology $\mathcal{T}$, if the *Satisfiability* check states R and S are not compatible (*i.e.*, their conjunction is unsatisfiable), *Concept Contraction* detects (i) what part G (*Give up*) of R clashes with S, and (ii) a contracted version K (*Keep*) of R such that it is compatible with S. If R and S are compatible, instead, but S does not fully satisfy R (as determined by a failed *Subsumption* check, implying S is not more specific than R), *Concept Abduction* identifies a concept H (*Hypothesis*) representing what is missing in S in order to reach a full match, in the Open World Assumption. Penalty functions based on the Conjunctive Normal Form for Description Logic (DL) concept expressions can be associated to both G and H, in order to compute a semantic distance of each available resource w.r.t. a given request.

The introduced semantic-enabled SCs are detailed hereafter:

– **Registration**. To store a resource into the blockchain, a Client sends a Registration transaction to a VN specifying: (i) the resource Uniform Resource Identifier (URI); (ii) its semantic description, *i.e.*, a concept expression w.r.t. a reference ontology; (iii) the ontology URI; (iv) other context-related attributes (*e.g.*, resource price). Multiple domains and associated ontologies can coexist on the same blockchain.

– **Discovery**. In order to search for (a set of) items, the requester sends a Discovery transaction to a VN with: (i) the URI of the reference ontology; (ii) the semantic description of the request, specifying desired features and constraints; (iii) other contextual information, *e.g.*, minimum semantic relevance threshold, maximum number of results to be returned or the price the requester is willing to pay. Semantic matchmaking is carried out as outlined above and a list of ranked resource URIs is returned to the Client.

– **Explanation**. This optional step allows a Client to get a justification of the matchmaking outcome. It is useful to trigger a request refinement process.

– **Selection**. After receiving a set of results, the Client can choose the desired resource with a Selection transaction, specifying the resource URI and optional context parameters.

Regardless of the matchmaking results, using a blockchain substratum guarantees outcomes are validated by consensus and data is synchronized among all participants.

III. The RideMATCHain ridesharing marketplace

A short example is provided from the prototypical ridesharing marketplace named *RideMATCHain* (https://github.com/sisinflab-swot/ridematchain) to clarify the proposal. The service matches passengers with the ride better fulfilling their requirements, considering user preferences and compatibility issues. Each available ride is an asset –stored in the blockchain by means of the Registration SC in Section II– with a semantic annotation according to a prototypical ontology (not shown due to space constraints). Let us consider this request R: *a mother with a baby is looking for a car with two seats, one of which equipped with baby seat. She would like to travel comfortably in an air-conditioned, quiet and non-smoking environment*. In order to take into account passenger-car and passenger-passenger constraints, the framework invokes the Discovery SC, which exploits the aforementioned semantic matchmaking task implemented in [4]. Figure 2 reports an excerpt of the annotations for R and two available rides in the proposed example, in Manchester syntax (http://www.w3.org/TR/owl2-manchester-syntax). The request and the SUV profile are very similar: only the driver experience requirement is not fulfilled explicitly. Conversely, the request is not compatible with the city car: the Explanation SC produces the $G_{CityCar1}$ expression obtained from *Concept Contraction*, showing the unavailability of the second passenger seat with a baby seat. The platform therefore ranks *SUV1* above *CityCar1* in the result list. Finally, the Selection transaction updates the *SUV1* description on the blockchain to account for the new passenger and triggers semantic compatibility check between any pair of vehicle occupants.

R: (accepts **only** (NonSmoking **and** Quiet)) **and** (has_Feature **only** (Baby_Seat **and** Air_Conditioning)) **and** (comfort_Level **only** High_Comfort) **and** (driver_Experience **only** High_Xp) **and** (available_Seats **min** 2)

SUV1: SUV **and** (accepts **only** NonSmoking) **and** (has_Feature **only** (Car_Radio **and** Air_Conditioning **and** Baby_Seat)) **and** (comfort_Level **only** High_Comfort) **and** (available_Seats **exactly** 6) **and** (available_Capacity **exactly** 650)

CityCar1: Utility_Car **and** (has_Feature **only** (Sunroof **and** No_Baby_Seat)) **and** (comfort_Level **only** Standard_Comfort) **and** (driver_Experience **only** High_Xp) **and** (available_Seats **exactly** 1) **and** (available_Capacity **exactly** 200)

$\mathbf{G_{CityCar1}}$: (has_Feature **only** (Baby_Seat)) **and** (available_Seats **min** 2)

Fig. 2: Annotations in the illustrative example

IV. Conclusion

The paper has presented a semantic-enhanced blockchain platform, based on Hyperledger Sawtooth. Smart Contracts manage the fundamental steps of the lifecycle of semantically annotated services/resources. A prototypical ridesharing marketplace has been built on top of the platform in order to validate the proposal. Future work concerns: experimental performance analysis, even involving resource-constrained IoT nodes; platform extension with further semantic SCs to provide more advanced capabilities, including service/resource composition, substitution, and negotiation; integrate these new features in the ridesharing solution.

Acknowledgments

This work has been supported by Italian Ministry of Economic Development R&D project BARIUM5G (Blockchain and ARtificial Intelligence for Ubiquitous coMputing via 5G).

References

[1] M. Ruta, F. Scioscia, S. Ieva, G. Capurso, and E. Di Sciascio, “Semantic blockchain to improve scalability in the Internet of Things,” *Open Journal of Internet Of Things (OJIOT)*, vol. 3, no. 1, pp. 46–61, 2017.

[2] H. Liu, Y. Zhang, and T. Yang, “Blockchain-Enabled Security in Electric Vehicles Cloud and Edge Computing,” *IEEE Network*, vol. 32, no. 3, pp. 78–83, 2018.

[3] S. Aggarwal and N. Kumar, “Hyperledger,” in *Advances in Computers*. Elsevier, 2021, vol. 121, pp. 323–343.

[4] M. Ruta, F. Scioscia, I. Bilenchi, F. Gramegna, G. Loseto, S. Ieva, and A. Pinto, “A multiplatform reasoning engine for the Semantic Web of Everything,” *Journal of Web Semantics*, vol. 73, p. 100709, 2022.