Future Generation Computer Systems 134 (2022) 236–246

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Trustable service discovery for highly dynamic decentralized workflows

Iain Barclay [a,*], Chris Simpkin [a], Graham Bent [a], Tom La Porta [c], Declan Millar [b], Alun Preece [a], Ian Taylor [a], Dinesh Verma [d]

[a] School of Computer Science and Informatics, Cardiff University, UK
[b] IBM Research Europe, UK
[c] The Pennsylvania State University, USA
[d] IBM Thomas J. Watson Research Center, USA

ARTICLE INFO

Article history:
Received 3 June 2021
Received in revised form 23 March 2022
Accepted 25 March 2022
Available online 9 April 2022

Keywords:
Decentralized workflows
Vector Symbolic Architectures
Workflow orchestration
Self-sovereign identity
Trusted services
Dynamic wireless networks

ABSTRACT

The quantity and capabilities of smart devices and sensors deployed as part of the Internet of Things (IoT) and accessible via remote microservices is set to rise dramatically as the provision of interactive data streaming increases. This introduces opportunities to rapidly construct new applications by interconnecting these microservices in different workflow configurations. The challenge is to discover the required microservices, including those from trusted partners and the wider community, whilst being able to operate robustly under diverse networking conditions. This paper outlines a workflow approach that provides decentralized discovery and orchestration of verifiably trustable services in support of multi-party operations. The approach is based on adoption of patterns from self-sovereign identity research, notably Verifiable Credentials, to share information amongst peers based on attestations of service descriptions and prior service usage in a privacy preserving and secure manner. This provides a dynamic, trust-based framework for ratifying and evaluating the qualities of different services. Collating these new service descriptions and integrating with existing decentralized workflow research based on vector symbolic architecture (VSA) provides an enhanced semantic search space for efficient and trusted service discovery that is necessary to support a diverse range of emerging edge-computing environments. An architecture for a dynamic decentralized service discovery system, is designed, and described through application to a scenario which uses trusted peers' reported experiences of an anomaly detection service to determine service selection.

1. Introduction

The quantity and capabilities of smart devices and sensors deployed as part of the Internet of Things (IoT) and accessible via remote microservices is set to rise dramatically as the provision of interactive data streaming, AI-based analysis and decision-making becomes increasing viable at the network edge. Data analytics requirements in military and civilian domains, are increasingly being met through the adoption of self-describing modular services that can be used to construct new applications by dynamically linking services and devices into different workflow configurations. As the number of such services increase, this represents a burgeoning requirement for rapid service discovery and a capability to configure service workflows that can be adapted to perform a variety of tasks using assets distributed at the network edge.

Some high-stakes environments, such as military deployments or disaster response scenarios [1], introduce a diverse and challenging set of requirements for constructing and then running the workflows necessary to support operations, specifically when using edge devices and supporting microservices to provide distributed analytics. In such environments communication is made via a Mobile Ad Hoc Wireless Network (MANET) [2], an interconnected set of wireless (and satellite) nodes that form a multi-hop network infrastructure. This can often result in low bandwidth communications links, where endpoint stability, network bandwidth, and connectivity remain limited and transient and it becomes impractical, if not impossible, to support centralized service registries and management of workflows executing at the edge. Time-critical applications operating in such environments introduce further complexity requiring a capability for applications to rapidly reconfigure themselves in the event of change, as individual edge services become inaccessible, or otherwise unavailable due to changing environments and conditions.

As a consequence, there is a need for decentralized application construction and workflow orchestration that can operate

* Corresponding author.
E-mail address: BarclayIS@cardiff.ac.uk (I. Barclay).

https://doi.org/10.1016/j.future.2022.03.035
0167-739X/

effectively without the need for a central point of control [3]. It will also be critical for stakeholders to be able to develop trust in previously unknown services and their providers, and be assured of qualities such as strong authentication, confidentiality, availability and privacy. As a result, dynamic workflow configuration needs to be able to provide rapid and autonomous discovery and configuration of *suitable* resources from a plethora of available devices and services, based not only on functional factors such as capabilities, but also non-functional attributes such as quality of service and provider reputation.

To meet these demands, there is a need for robust service orchestration capabilities that can provide a configuration capability that combines rapid configuration and adjustment afforded by a decentralized solution, with mechanisms that provide selection priority to trustworthy services—based on fluid notions of what can be trusted at any given point in time: as risk increases, the act of placing trust becomes harder to justify [4]. In such circumstances, those who place trust intelligently, seek out evidence to provide assurance that their judgement is appropriate. This evidence must be considered for its accuracy and relevance, which requires the ability to attribute claims to their sources whose authority and trustworthiness must also be assessed [5,6].

The authors' previous work has shown that decentralized workflow mechanisms, such as those based on a Vector Symbolic Architectures (VSA), can be used to facilitate efficient service discovery and workflow formulation in decentralized collaborative environments [7–9], without requiring central control [3]. VSA uses hyper-dimensional vector spaces in which the vectors can be real-valued, such as in Plate's Holographic Reduced Representations (HRR) [10], typically having dimension $512 \leq D < 2048$, or they can be large binary vectors, such as Penti Kanerva's Binary Spatter Codes (BSC) [11], typically having $D \geq 10^4$.

This paper describes the use of a VSA to architect a mechanism that can be used for distributed discovery and orchestration of edge devices and microservices, where device and service descriptions are derived from interoperable linked data, semantic web technologies, and emerging open web standards, such that pre-existing descriptive resources can be utilized as far as possible. In addition, a dynamic layer of trust is added to service descriptions through the employment of certified credential documents, using design patterns which provide cryptographic proof of issue and integrity adopted from research into self-sovereign identity. These credentials are used to provide assurance on service qualities as experienced by trusted peers. The service descriptions and credentials are collated to seed service and sensor descriptions in the following way:

- Interfaces (APIs) to microservices are provisioned with service descriptions, in formats such as SPARQL [12], which offer self-describing mechanisms for interactions and exploit a capability for "Continuous Acquisition of Behaviours" [13]. RDF triples underlying SPARQL descriptions can be further augmented by service descriptions provided as Web of Things (WoT) 'Thing Descriptions' (TD) [14] or other JSON or JSON-LD [15] format Linked Data descriptions.
- Verifiable Credentials (VC) enable service providers to attest to the *specifications* of their service offerings, and additionally allow peers to digitally sign and issue assertions about their service *experiences*, based on actual use of the service. E.g., "The service was effective at identifying street fighting". As a result, JSON-LD documents are created describing both service specifications and service experiences.
- SPARQL and JSON-LD service descriptions, service specification JSON-LD VCs, and experiential JSON-LD VCs are encoded using VSA techniques, such that they can be integrated and used by a VSA workflow system, and provide a capability for efficient searching for services based on the semantic properties of VSA.

As a result, mechanisms can be provided which are able to take advantage of lightweight, interoperable service description technologies to enable efficient service discovery and orchestration across dynamic, contested decentralized environments. VSA enables service discovery in a semantic space, seeded by descriptions provided by service providers and users. Using a semantic search provides service discovery that is decoupled from the precise language used by service operators and users, whilst experience reports from users can be used to refine service selection, prioritizing those services which have been successfully used by trusted peers.

The proposed method is presented in the context of a scenario developed in the context of the International Technology Alliance in Distributed Analytics and Information Sciences (DAIS ITA) [16] project, which is aiming to enable the creation of a distributed *cognitive computer system* that can perform analytics on demand across heterogeneous networks of interconnected devices in support of coalition operations, where multiple partners share sensing and information processing assets. In such an environment, clients want to be able to ensure services are trustworthy, but also to prioritize selections based on the experiences of their peers through *social transparency* [17] — e.g., "has this service been endorsed by a member of our defence force or by a coalition member?"

The main contribution of this paper is the incorporation into the VSA approach of a method that introduces trustworthiness as a factor in service selection and workflow configuration. The method provides rapid configuration of services in a workflow that takes into account claims made about service capabilities and reports of experiences of peers in using services. This enables the trustworthiness and authority of peers and information they provide to be considered in service ranking criteria. To support peer-to-peer trust relationships, the method adopts decentralized, self-sovereign identity concepts. This provides a framework for service providers to express the capabilities of their services, and for trusted peers to provide service experience reports, as signed digital credentials which are collated and encoded as VSA vectors. This method facilitates efficient service matching and workflow configuration, based on dynamic trust relationships between service providers, users and those seeking services. The method, and a supporting software architecture, are described in the context of a scenario in which a remote anomaly detection service is selected for adoption in a workflow based on the reported usage experiences of trusted coalition partners.

The rest of the paper is organized as follows. Section 2 describes related work in the field of service discovery, and identifies shortcomings in support of providing evidence that can foster the development of trust in services. Section 3 describes a method that can provide a means to provide such evidence, based on the use of signed digital credentials in vector encoded service descriptions. The proposed method is instantiated in Section 4, in the context of a case study based on the dynamic selection of a remote anomaly detection service in a military scenario. The paper concludes in Section 5, which identifies further research to extend this work.

2. Related work

Service discovery is a key component in a transient distributed networked environment, but is challenging, as services are developed and deployed independently or developed by loosely cooperating developers in open environments. This has led to a complex mix of disparate service architectures employing different methodologies for the description of their inputs, outputs, and configurations. In support of such situations, vector based representations can be employed as a means of encoding service

descriptions, so that they can be semantically compared within particular contexts in an extremely resource efficient way. Using such vectors, semantically rich queries in the form of vectors, can be sent out to the network, using protocols such as multicast for efficient querying in a complex space.

Recent literature has described experiences in bridging web service APIs with semantically searchable interfaces through the provision of SPARQL service descriptions. Michel et al. [18] developed an interface to a web-hosted dataset in which "a SPARQL microservice evaluates a query against an RDF graph built at run-time from data obtained from the Web API", which was later extended [19] to provide a SPARQL front-end to a service which aggregated results of the previous experiment with those from a REST API. A PHP server application bridges a SPARQL client and the web APIs, converting SPARQL queries into search parameters. Such a service could be provisioned to provide access an interface to a web service, an ML model, a sensor or actuator, or a dataset [18]. As such, SPARQL interfaces can be created for a range of resources offered by service providers, third parties or open source, providing adaptable semantic search interfaces where none currently exists.

For some services, semantically rich Linked Data [20] (LD) service descriptions are made available by the service provider. This might be the case for a physical asset, where a Web of Things 'Thing Description' document could be provided on the asset itself or via a proxy to give a JSON-LD Linked Data description of the affordances of a device, or its capabilities and how it is to be used, "in order to increase interoperability between connected devices and develop arbitrarily complex mash-ups" [21]. Similarly, a dataset might be accompanied by a Linked Data description. Where such information is supplied by a service provider, this can be rendered as RDF triples to provide the SPARQL microservice's description. Bienz et al. [22] have demonstrated the use of SPARQL endpoints to provide approximate search capabilities for physical devices, seeded from published WoT TDs.

LD principles provide mechanisms that Mayer et al. [23] describe as having the ability "to underpin systems that integrate multiagent planning and acting with semantic technologies and with interoperable mixed reality interfaces", enabling "the creation of highly augmented environments…where physical and digital things coexist and interact with one another". Suri et al. [24] provide an analysis of the applicability of these 'physical and digital things' in a decentralized environment, and conclude that technical challenges enumerated by Zheng [25] in regards to connectivity, digital analytics, and interoperability of assets in decentralized environments can be addressed through the use of semantic web [26] technologies, which are identified as providing "(I) Open integration standards; (II) Reasoning support; (III) Support for data provenance management". and state that "one of the many applications of IoT would be shared sensing among mobile devices / sensor nodes in an area of interest. Sensing – e.g., of the environment, people, and devices – is at the core of IoT…Combined with robust short range communication, IoT devices would be able to utilize placement or sensing modality of other sensors to supplement their own sensing methods" [24].

Zschorn et al. [27] reflect that "Information requirements of Defence operational staff are …many, varied and changing, and sometimes unpredictable. These various operations require at times communication and coordination with coalition military partners, federal and state police forces, other government agencies, and non-government agencies". As such, it is important to be able to develop trust in providers and sources of data, which is raised as a concern by Michaelis et al. [28]: "…information derived from IoT sources may have varying degrees of integrity, possibly making it unfit for analyst/commander usage" who identify a need for "methods to associate records of provenance with information, sufficiently detailed for a collection of (possibly unforeseen) assessment tasks".

Indeed, Evdokimov et al. [29] provide a Comparison of Discovery Service Architectures for IoT outlining various key high-level features that must be addressed in order to adequately fulfil a robust/industrial-strength decentralized architecture (suitability, accurateness, interoperability, compliance, support for independent design, organic growth, scalability, quality of service, data ownership, and security). Further, Ververidis et al. [30] identify key areas where the integrity and provenance of services and data must be maintained during service discovery and orchestration in MANET environments, namely; (a) during service registration and deregistration for directory-based discovery mechanisms, (b) during service discovery for both client/requesters and service providers, in both directions, and (c) during service delivery. Ververidis et al. [30] conclude that "Unfortunately service discovery approaches fail to address most of the above requirements and often overlook the problems of security, privacy and trust". The VSA architecture scheme described in [7–9] details a peer-to-peer mechanism for efficient on-the-fly workflow discovery and orchestration in dynamic MANET environments capable of addressing many of these requirements, however, data and service ownership and trust relationships between service providers and those seeking service fulfilment is not addressed.

The importance of the role of trust in service discovery and selection for workflow configuration has been recognized [30,28], but there is a gap in published research in providing methods by which information and metadata about services can be used as trustable evidence. Claims about service capabilities and qualities need to be able to be judged for accuracy and relevance, as well as on the trustworthiness and authority of those providing the claims in order to offer assurance to those making judgements about service suitability in high-stakes settings. This paper seeks to close this gap by designing a method that uses decentralized trust models to augment service descriptions and reported service experiences, within the structure of a VSA.

3. Enabling trustable service configurations

Self-sovereign identity (SSI) [31] is terminology used to describe the ability of an individual to take ownership of their personal data and to control who has access to that data, without the need for a centralized infrastructure, or any control or authorization being required by any third party. To date, the focus of effort of SSI researchers has been on personal identity and data privacy for individuals [32], however the underlying computer science techniques can be applied to any type of entity, including digital assets such as datasets [33], and devices [34]. SSI is decentralized, and is built upon well-established cryptographic techniques whereby a user holds a private and shares a public key [35]. The private key is used to sign documents, whilst the public key can be used by anybody with access to it to verify that the document has indeed been signed, and has not been tampered with. SSI uses a system built on decentralized identifiers (DID) to identify parties involved, with the DIDs resolving to documents which explain, in machine-readable format, how to locate the public key needed to validate claims made about that DID, in the same way as web addresses resolve to provide web pages.

3.1. Verifiable credentials as evidenced claims

The SSI research community has developed data models and protocols [36] that provide mechanisms for any party identified by a DID to issue cryptographically verifiable sets of credentials to any subject entity, also identified by a DID. In this way, a party which believes something to be true about another party

Table 1
Service description sources populated by providers and coalition peers.

	Type	Description
SD	Service Description	Service described by provider in RDF triples or Web of Things JSON-LD
VC_S	Service Specification	Deployment specifications for service issued by provider as credentials
VC_XP	Service Experiences	Experience reports issued by peers as credentials

can declare this in a standardized way using a JSON-LD formatted document, and sign this attestation using asymmetric cryptography techniques, based on the DIDs used being able to be resolvable to validate the assertions made. This cryptographically signed document is known as a Verifiable Credential, and will be held by the subject of the credential, or in the case of a dataset or physical asset, by an authorized holder.

At a later date, when the holder seeks to enter into a transaction, a service provider may request proof of status or entitlements. The VC document provides a means for this proof to be provided, as the holder of the credential can generate a Verifiable Presentation (VP) containing assertions from the VC document. By processing the VP document, the Verifier can use the accessible public keys to check that (i) the presented proof pertains to the subject it is being presented on behalf of, (ii) the presented proof contains assertions signed by the original Issuer, and finally (iii) that the presented documents have not been tampered with. As such, triangles of trust [37] can be leveraged to enable parties to issue, hold and verify credentials without reliance on any central authority. Systems based on the paradigm of the self-sovereignty of human participants, data resources and devices are inherently decentralized, with attributes held at the edges of the ecosystem.

3.2. Credentials for edge service descriptions

Three different types of service metadata are proposed (Table 1). These can be collated to provide an overall service description which can be VSA encoded, resulting in a semantically searchable vector describing the service's capabilities.

3.2.1. Service descriptions and specification credentials

Use of interoperable semantic web technologies and emerging standards for describing microservices such as APIs, datasets and sensor devices, offers an opportunity to adopt semantic mechanisms and tools from a wider community, and provides a route for service configurations to ingest open source resources such as video infrastructure in a city [38], as well as those provided privately within a consortium or collaboration. To provide further context on services, LD documents, in the form of VCs, are used to augment information from Service Description documents (or RDF triples) provided by manufacturers, with further contextual information about the service deployment. In the first instance, solution vendors, systems integrators or service providers can issue attestations relating to the specifications or qualities of the service, which can be stored and made available for inspection. These are cryptographically signed JSON-LD documents (VC_S), which can be verified against a public key known to be owned by the signer and credential issuer, which can prove that the document was signed by the issuer and has not been tampered with. As an example, a service provider deploying a video camera at a certain location could issue a signed VC asserting the coordinates of the location of the camera. Any parties subsequently interested in using the camera could request and inspect the credential and (provided they trusted the signing party) could be assured of the location of the camera.

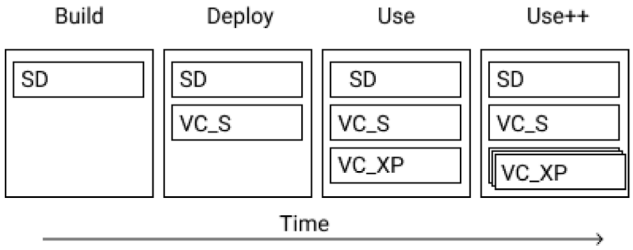

Fig. 1. Experiences are added as service is used.

3.2.2. Service experience credentials

Additionally, it is possible for any service user to create and issue a signed credential relating to their own experiences in using the service, which would be held as part of the service's metadata corpus. For example, if an AI service offers vehicle identification [39] and a user has had good success using the service for identifying Sports Utility Vehicles (SUV), then they are able to create a JSON-LD document attesting to this and sign it, resulting in an experience credential (VC_XP) being held by the service. Subsequent parties seeking SUV identification capabilities would be able to inspect this credential, and determine its suitability as additional information, based on trust they have in the party signing and issuing the claim.

3.3. Service descriptions augmented by usage

As microservices and devices are deployed in different workflows, users are able to contribute reports about their own experiences with the service or device. These reports, structured as JSON-LD VC documents, can be cryptographically signed and issued to the service. As such, new metadata based on actual experiences of services becomes increasingly available as services are adopted (Fig. 1), and is able to provide deeper semantic information about service qualities.

Furthermore, as cryptographic signatures based on DID properties are used to sign these experience credentials, they can be linked to pseudonymous identities known by potential service users. As a result, service selection can begin to include the presence of trusted experience reports as a criteria in workflow configuration—reverting back to a military context, a UK commander would be more likely to trust a service ratified by his US counterpart, than one not ratified at all, or indeed one ratified by a more transient ally.

3.4. Service description encoding and workflow orchestration

Vector Symbolic Architectures are a family of bio-inspired methods for representing and manipulating concepts and their meanings in a high dimensional vector space [11]. They are a form of 'brain like' distributed representation that enables large volumes of data to be combined into a fixed size feature vector such that the semantic meaning of the data and relationships that they represent is preserved. Such vector representations were originally proposed by Hinton [40] who identified that they have recursive binding properties that allow for higher level semantic vector representations to be formulated from, and in the

same format as, their lower level semantic vector components. Eliasmith, in his book 'How to Build a Brain' [41], shows how these vector representations can be used to perform 'brain like' neuromorphic cognitive processing. He coined the phrase 'semantic pointer' for such a vector since it acts as both a 'semantic' description of the concept, which can be manipulated directly and a 'pointer' to the concept. As such they are said to be semantically self-describing. VSAs are also capable of supporting a large range of cognitive tasks such as: Semantic matching; Representing meaning and order; Analogical mapping; and Logical reasoning [42]. Consequentially they have been used in natural language processing [42,43] and cognitive modelling [41,44].

The approach for creating semantically rich representation of services and workflows adopted is to represent them as high level semantic concept vectors that are themselves constructed from semantic vectors representing their sub features in a recursive manner using vector binding and superposition operations [8,9]. Given a role vector $\boldsymbol{r}$ and filler (value) vector $\boldsymbol{v}$, these can be bound together using the exclusive or (XOR) operation $\boldsymbol{r} \cdot \boldsymbol{v}$.

The high-level semantic vector representation $\boldsymbol{z}$ of a sensor device or a service object is made up of a nested superposition of its sub-feature vectors,

$$\boldsymbol{z} = \boldsymbol{r}_{\text{SD}} \cdot \boldsymbol{v}_{\text{SD}} + \boldsymbol{r}_{\text{VC\_S}} \cdot \boldsymbol{v}_{\text{VC\_S}} + \boldsymbol{r}_{\text{VC\_XP}} \cdot \boldsymbol{v}_{\text{VC\_XP}}; \tag{1}$$

where $\boldsymbol{v}_{\text{VC\_S}}$, $\boldsymbol{v}_{\text{VC\_SD}}$, $\boldsymbol{v}_{\text{VC\_XP}}$ are themselves high-level concept vectors built from RDF triple sets or parsed JSON-LD.

As an example, Listing 1 shows a Web of Things 'Thing Description' for a web camera [45], encoded in JSON-LD, as $\boldsymbol{v}_{\text{VC\_S}}$ of the $\boldsymbol{z}$ object description. This in turn is converted to a flattened collection of sub-features [8].

Listing 1: WoT Thing Description for Camera Sensor

```
{
 "@context": "https://iot.mozilla.org/schemas/",
 "@type": ["Camera", "VideoCamera"],
 "name": "Web Camera",
 "description": "Mobile web camera",
 "properties": {
   "video": {
     "@type": "VideoProperty",
     "title": "Stream",
     "links": [{
       "href": "rtsp://eg.com/video.mp4",
       "mediaType": "video/mp4"
     }]
   },
   "image": {
     "@type": "ImageProperty",
     "title": "Snapshot",
     "links": [{
       "href": "http://eg.com/image.jpg",
       "mediaType": "image/jpg"
     }]
   }
 }
}
```

The $\boldsymbol{v}_{\text{VC\_S}}$ and $\boldsymbol{v}_{\text{VC\_XP}}$ components can be encoded in a similar manner. Listing 2 shows an example of a Verifiable Credential, used as a VC_S, issued by a service provider when deploying a camera device. The credential stores information about the deployed location of the camera, along with a cryptographic proof which can be used by relying parties to verify that the credential document has not been tampered with. The credential could be deleted or revoked when the camera is moved to a new location.

Listing 2: An example VC_S for a deployed Camera Sensor

```
{
 "@context": [
   "https://www.w3.org/2018/credentials/v1",
   "https://schema.org/"
 ],
 "id": "http://eg.com/credentials/e94a16cb",
 "type": [
    "VerifiableCredential",
    "DeployedDeviceCredential"
 ],
 "name": "Camera Deployment",
 "description": "Roadside camera deployed.",
 "issuer": "did:v1:nym:z6Mk..63oP39k",
 "issuanceDate": "2020-04-09T21:13:13Z",
 "credentialSubject": {
   "deviceIdentifier": "3a185b8f",
   "deployedLocation": {
     "address": "Kirkegata, Anglova",
     "latitude": "58.145",
     "longitude": "7.998"
   }
   ...
 },
 "proof": {
   "type": "Ed25519Signature2018",
   "created": "2020-04-09T21:13:28Z",
   "proofPurpose": "assertionMethod",
   "verificationMethod": "did:v1::nym:z6Mkhdm"
 }
}
```

All sensor, device or service objects are encoded in this fashion by the object on the owning physical device, hence each object must be VSA aware or able to provide its RDF or JSON-LD descriptions to a VSA cognitive wrapper service that can be executed on the device object. Such metadata can then be encoded into a VSA service description that can be injected into the MANET to dynamically discover and connect the appropriate components together as described in the authors' previous work [7–9].

Once sensor and functional microservices have been encoded into VSA vectors, a VSA workflow vector can be constructed. This will be capable of discovering, connecting, and orchestrating such objects together as described in [9, pages 28-31]. When injected into a MANET, the VSA workflow vector will automatically locate appropriate devices and service objects and connect them together into the required analytics chain. If different sensor chains are required to be constructed in parallel, then each sensor chain would be encoded into a VSA vector in a similar manner and then these can be combined into a new single workflow vector. In the previous work a simple three stage processing chain was described but longer and more complex workflow configurations can also be constructed. Eq. (2) shows a generalized sensor chain encoding using the hierarchical binding notation defined in [8,9], for example,

$$\begin{aligned}\boldsymbol{z}_{\text{chain}} = {} & \boldsymbol{p}_0^0 \cdot \boldsymbol{z}_{\text{sensor}}^1 \\ & + \boldsymbol{p}_0^0 \cdot \boldsymbol{p}_1^0 \cdot \boldsymbol{z}_{\text{analysis}}^2 \\ & + \boldsymbol{p}_0^0 \cdot \boldsymbol{p}_1^0 \cdot \boldsymbol{p}_2^0 \cdot r_{\text{stream}}^3 \\ & + \boldsymbol{p}_0^0 \cdot \boldsymbol{p}_1^0 \cdot \boldsymbol{p}_2^0 \cdot \boldsymbol{p}_3^0 \cdot \boldsymbol{p}_4^0 \cdot \boldsymbol{r}_{\text{collector}}^4 \\ & + \boldsymbol{p}_0^0 \cdot \boldsymbol{p}_1^0 \cdot \boldsymbol{p}_2^0 \cdot \boldsymbol{p}_3^0 \cdot \boldsymbol{p}_4^0 \cdot \boldsymbol{p}_5^0 \cdot \boldsymbol{z}_{\text{results}}^5; \end{aligned} \tag{2}$$

where $\boldsymbol{z}_{\text{sensor}}$ and $\boldsymbol{z}_{\text{analysis}}$ are VSA encodings of the particular sensor and sensor analysis object descriptions, respectively.

Multiple sensor chains may be combined as

$$\boldsymbol{z}_{\text{start}} = \boldsymbol{p}_0^0 \cdot [\boldsymbol{z}_{\text{chain01}} + \boldsymbol{z}_{\text{chain02}} + \boldsymbol{z}_{\text{chain03}} + \cdots]^1; \tag{3}$$

Note, $[\cdots]$ indicates the normalized *majority* sum of terms. and multicast into the network.

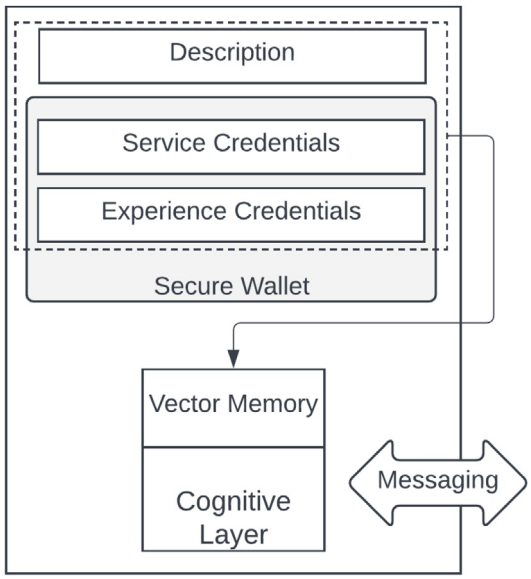

Fig. 2. Architecture of a service node.

VCs provide a mechanism for service users to issue reports about service use experiences to the services themselves, through VC_XP 'Experience Credentials'. These reports, which are expressed in JSON-LD formatted documents, are semantically searchable through the VSA mechanisms detailed above, and also provide additional contextual information as they are signed by the party that is making the claim. The identity of the signing party is expressed only as a pseudononymous decentralized identifier. This DID may or may not be known to parties checking the experience reports, and as such, a varying amount of regard can be given to it.

The use of VC_XPs allow services to be categorized by trust relationships, and selected with a degree of granularity. An absence of VC_XPs, means that no other party has left a signed experience report for the service, and where VC_XPs are available, configurations can be made such that priority is given to VC_XPs signed by close peers ahead of those signed by unknown parties. It can be envisaged that an ordering of service selection might be made where priority is given such that services with known VC_XPs are selected ahead of services with unknown VC_XPs, with services with zero VC_XPs chosen as a last resort. The selection field could vary depending on the urgency of the situation, and the quantity of resource required versus resource availability.

3.5. System architecture

The architecture of a node in the ecosystem is illustrated in Fig. 2. This shows how an SSI agent that adopts standard protocols for credential exchange is to integrated with the VSA service layer, such that a service node can respond to service request matches based on VSA encodings of service descriptions and VCs provided by peers. Secure storage of VCs is provided by a wallet sub-component within each SSI agent. This architecture could be implemented directly on a service node, in the case of a software microservice, or could be used as a proxy for an underlying IoT sensor.

4. Case study: Anomaly detection service selection

Military scenarios provide an extreme environment for the application of workflow configuration architectures. They consist of coalition partners or peer organizations exhibiting varying and fluid levels of trust. Services need to be deployed rapidly in fragile battlefield environments, often based on MANETs, with sensors and other devices coming in and out of range, and network fragmentation occurring frequently. It is not possible to rely on availability of centralized registries, or even to know the IP location of objects and devices as they come in and out of service. As such, a mechanism is needed that can locate and orchestrate the required workflow in the face of these challenges. As described in [7–9], the VSA architecture can be used for peer-to-peer (P2P) discovery of appropriate devices and functional microservices without service and device registries because the VSA representation is able to act as both the object description and the address of the object. To extend this previous work, and introduce trust into service selection, new types of service descriptions can be added, based on the design approach presented in Section 3.

Information Systems artefacts, such as methods, can be evaluated by considering their role in the context of the proposed deployment environment and assessing how it well they would operate as a solution to the problems identified. An artefact interacts with its intended deployment context as a *treatment*, and instantiating and demonstrating treatments based on the designed artefact provides opportunity for evaluation of how effective the treatment might be in the context of the scenario. The demonstration present here provides an instance of a "single-case mechanism experiment" [46]. This contributes to a technical risk and efficacy evaluation [47], showing that the technical approach can be effective.

To demonstrate the proposed approach to introducing trust into service selection, a scenario based upon a deployment of a monitoring system [48] in the NATO Anglova urban setting [49] is considered. Sensors and analytic services are owned and provided by partners in a military coalition, and procure and analyse multimodal sensor data during times of situational uncertainty and rapidly-evolving circumstances. In the scenario, a data-driven system monitors and evaluates a situation where events indicate growing threats to, and attacks on, a section of the Anglova civilian population, the 'Capulet' community. Each system is configured to provide an event notification when it senses that an identified event has taken place. These notifications are delivered to a UK Intelligence Analyst, who can decide what steps to take in response. Relationships between coalition members typically exhibit asymmetric power balances, and fragile trust, yet partners must collaborate to deliver a solution to a problem in a fast-changing, high-stakes environment.

The services are distributed in a MANET environment, and the operations to construct the workflow are performed using a completely decentralized discovery and connection protocol. In previous work [8,9], the required sensor and services were discovered based on VSA vectors that encoded the required functional capabilities using an assumption that the component services could be trusted, and that no additional checks were required. This paper shows that the adoption of patterns from self-sovereign identity research, notably VCs, to share service descriptions and prior service usage in a trustable, privacy preserving and secure manner, can enhance the VSA vector description and lead to discovery of services that meet service workflow requirements, and have been endorsed by trusted peers.

The architecture is evaluated in the context of a scenario, which requires a trusted workflow configuration to be developed in order to detect potential insurgency events in a remote location in a coalition partner's country [48]. The scenario was seeded with metadata from a survey paper [50] which describes services that offer road traffic anomaly detection from video scenes, and provides an overview of the systems, detailing the datasets used for their training, and the type of anomalies detected [50, Table

7. Representative Work on Scope of Applied Areas, Page 119:17]. This information maps well to a formal specification of the service – that it was designed to do – which can be augmented with experience reports from users who have used it and wish to state what they have found it works well for.

4.1. Service descriptions as verifiable credentials

To instantiate the solution design each service is represented by a software agent, which is capable of receiving and storing VCs issued to it by other entities or agents in the ecosystem, and rendering state from the VCs as VSA encoded vectors in order to support service configuration. A core principle of SSI is that entities hold their own data, and reveal information at their discretion. This is true for individuals in SSI systems, as well as other entities. A third party – the Verifier – can request a credential holder to provide a proof of a credential being held. The Holder, at their own discretion, will generate a Verifiable Presentation (VP), which the Verifier can determine is genuine through SSI protocols. This principle offers confidentiality to service providers, giving them choice over the granularity of service metadata they reveal to other parties. They will make some information public in order to advertise their services in response to service requests, but some metadata will be selectively disclosed. As such, a combination of publicly shared service capabilities and an interaction required to request further proof in the form of a VP, provides both flexibility and speed in service matching, with the ability for high integrity trust-based verification to be performed when required.

To provide this behaviour, the system operates in the following way. A Syndicate.id SSI agent is instantiated to represent the Anomaly Service (AS), with a second SSI agent representing the Service Owner (SO). In the first instance, the SO creates a credential schema to describe the features of the deployment of the AS, and issues a VC to the AS SSI agent. When the AS receives this credential, it stores it securely in a digital wallet component, which in this case is provided by the underlying ACA-Py infrastructure. However, AS will also use some elements of the VC to describe its capabilities, such that it can be discovered by other parties. In the architecture, the AS writes a JSON file with the subset of capabilities from the VC that it wishes to make publicly known—in effect, the capabilities in this file operate as an advertisement for the service. The AS agent may not necessarily publish everything that was included in the VC, some elements can remain hidden within the VC and revealed selectively as part of a subsequent request for a VP. In this way, the AS can publicise information about its service capabilities, whilst retaining control over more sensitive information, which is only revealed – at the AS agent's discretion – to trusted parties who make a subsequent request. The information that the agent wishes to make public is written as JSON data, and can be encoded in a VSA format, as Fig. 3 illustrates, with publicly shared information shown in green, and privately held information shown in pink.

Service users are also able to issue credentials to AS, based on their experiences using the service, and in doing so will enrich the knowledge available about a particular service. In the anomaly detection scenario a service user may have found that one of the monitoring systems had successfully detected a street fight. They would be able to issue an experience report VC with this capability to the AS, so that future users with a requirement for detecting street fighting would be more likely to have the service recommended to them. The AS can take elements of this VC's content and use it to extend its self-attested JSON record, and encode the content into a VSA vector to enrich the ecosystem's knowledge of the service and its capabilities. As more peers use

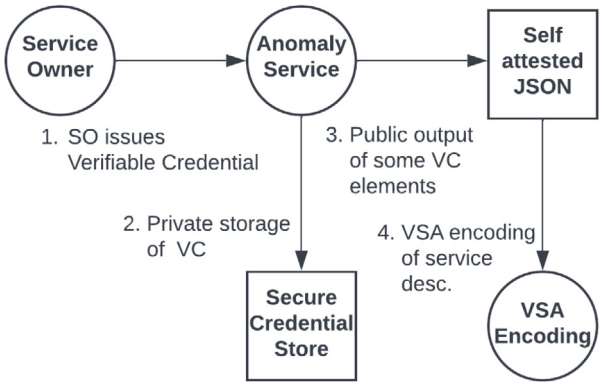

Fig. 3. Publication of capabilities from credentials.

the service, and issue experience reports based on their experiences, the depth of information that the AS can provide about the practical use of its services increases, as previously shown in Fig. 1. Listings 3 to 5 show the body of knowledge that might exist for an example instance of the AS service, derived from the service specification, a deployment credential, and an experience report credential respectively.

Listing 3: Service Description

```
{
  "name": "Nguyen_2015",
  "class": "road traffic anomaly detection",
  "speciality": "Junctions",
  "training_dataset": "MIT_2009",
  "realtime": "YES",
  "explanations": "YES"
}
```

Listing 4: Deployment Credential

```
{
  "deployed": "08/10/18",
  "latency": "0.1"
}
```

Listing 5: Experience Report Credential

```
{
  "xp_report": "Detected street fight"
}
```

The scenario requires detection of public disorder or disturbance, so the experience report credential offers a good indication that this service is a candidate for this specific task. How this is considered in service selection is described in the following section.

4.2. Service description encoding and semantic matching

Simpkin et al. [8] describes an algorithm that uses VSA roll-filler pair binding and bundling to create VSA description vectors directly from a nested JSON service description file. The method applies equally well to the encoding of any format of key–value plain text data files. Unique role vectors are built from the sequence of nested keys. These are then combined with their associated JSON value fields to create a bag of roll-filler pairs that is majority summed into a binary service description vector. Since the experience report credentials shown in Listing 5

Table 2
Top ten nearest words to *'violent'*, *'disturbance'* from 1M word GoogleNews word2vec corpus and corresponding Hamming Similarity for the same 1M words converted to a BSC database.

Rank	Name	Cosine similarity	Name	HSim BSCs
0	Violence	0.8122	Violent	0.8408
1	Destructive	0.7995	Disturbance	0.8350
2	Disturbances	0.7854	Violence	0.7568
3	Severe	0.7730	Destructive	0.7503
4	Disruption	0.7706	Disturbances	0.7477
5	Chaotic	0.7495	Disruption	0.7360
6	Discomfort	0.7467	Severe	0.7354
7	Anxiety	0.7452	Discomfort	0.7334
8	Escalating	0.7450	Criminality	0.7294
9	Criminality	0.7444	Chaotic	0.7288

Table 3
Top 5 nearest words to *'workflow'* from 1M word GoogleNews word2vec corpus and corresponding Hamming Similarity for the same 1M words converted to a BSC database.

Rank	Name	Cosine similarity	Name	HSim BSCs
0	workflows	0.880287	workflow	1.0000
1	automated_workflow	0.758400	workflows	0.8439
2	workflow_automation	0.757980	workflow_automation	0.7759
3	Workflow	0.735511	automated_workflow	0.7706
4	automated_workflows	0.726034	automated_workflows	0.7592

use textual descriptions, some method of semantic matching is required for vectors to match on these descriptions. This can be achieved by using VSA vectors that represent semantic word vectors of the type produced by word2vec [51,52]. Real valued vectors can be converted to VSA binary vectors (BSC) by using randomized binary projection [53]. Using these vectors results in better semantic matching not only on the experience report credentials but also on other VC JSON values and keys. In Natural Language Processing words and sentences are often vectorized for further manipulation and analysis. Word2Vec [51,52] takes a text corpus as input and generates real number output vectors for each word in the corpus such that the words having similar meaning have vectors that are closer together in the output vector space. This means that ready made, leading edge, semantic word representation such as the pre-trained Google News corpus [51] can be leveraged for the purpose of BSC VC descriptions.

Table 2 gives a comparison of the cosine distance word-rank search on the normalized sum of $\boldsymbol{v}_{\text{violent}} + \boldsymbol{v}_{\text{disturbance}}$ from the Google News word2vec corpus when loading the first 10^6 words compared to the same database converted to BSC. The real number input vectors are 300-dimensional and the BSC output vectors are 10^4-dimensional. Table 3 shows the five nearest words to the word 'workflow'. A single word is used for the search because JSON keys are single entry fields, whereas values can be encoded as a list of keywords. To avoid clashes, a different random projection (a random matrix of 10^4 dimensional mapping vectors) is used for generating the JSON keys and values. Table 4 shows the results of a sample of semantic comparisons where both the key name and value are treated as semantically comparable. The requested key–value vector, ranked zero in the table, is created as follows,

$$\boldsymbol{v}_{\text{key\_value}} = \boldsymbol{v}_{\text{workflow}} \cdot [\boldsymbol{v}_{\text{violent}} + \boldsymbol{v}_{\text{disturbance}}] \tag{4}$$

Where, $\boldsymbol{v}_{xyz}$ should be read as the BSC conversion of the semantic word vector retrieved from the Google News Corpus and '*xyz*' is the string used for lookup. (Note, in Eq. (4), '[]' indicates a normalized *majority* sum of terms.) As can be seen using BSC VSA matching works well for both semantically similar keys and values. This approach can be used to encode both service description JSONs and VC JSONs. For multi-entry nested JSON, each key–value chain is encoded separately producing a semantic sub-vector using XOR chaining after which an unordered semantic bag vector is created using VSA majority-vote addition, see [8, page 121]. Experience reports as shown in Listing 5 can be created as per Eq. (4) example. Since it is expected that many VC_XPs will be created these are stored separately for later searching within an associative vector memory.

Table 4
Semantic matching examples using semantic key and value encoding.

Rank	Key	Value	HSim BSCs
0	$\boldsymbol{v}_{\text{workflow}}$	$[\boldsymbol{v}_{\text{violent}} + \boldsymbol{v}_{\text{disturbance}}]$	0.8361
1	$\boldsymbol{v}_{\text{workflow}}$	$\boldsymbol{v}_{\text{disturbances}}$	0.7431
2	$\boldsymbol{v}_{\text{automated\_workflow}}$	$[\boldsymbol{v}_{\text{violent}} + \boldsymbol{v}_{\text{disturbance}}]$	0.6601
3	$\boldsymbol{v}_{\text{workflows}}$	$\boldsymbol{v}_{\text{criminality}}$	0.6521
4	$\boldsymbol{v}_{\text{workflow\_automation}}$	$[\boldsymbol{v}_{\text{severe}} + \boldsymbol{v}_{\text{criminality}}]$	0.6331

4.3. Trust, but verify

When a VSA service search request is matched, the requesting agent can choose to take the encoded (self-attested) service description data at face value, or can initiate an engagement with the AS agent to request a VP providing solid evidence backing the attestation. This choice can be made dynamically, based on circumstances which might include the perceived risk and the timeliness of the requirement.

The VSA distributed workflow architecture described in [7–9] uses multicast for all VSA messages. Since it is likely that multiple matches will occur when using semantic matching on service descriptions and VCs a '*local arbitration*' mechanism is used to resolve multiple replies. The mechanism has two components. First, a delayed response mechanism (inversely proportional to match quality) ensures that services that calculate better quality matches respond faster than those calculating weaker matches. Due to the use of multicast, should a weakly matching responder '*hear*' a better match sent from some other responder then it cancels its response thereby saving bandwidth. Second, when multiple replies are received the requester acts as the final arbitrator, collecting matches and comparing the responders' match values. Using multicast the requester sends a *winner_selected* message back to the network. During this multicast request/response handshake the requester/responders unicast IP-address' are exchanged which can then be used to carry out verification of a responder's credentials using the SSI architecture stack to request a VP to provide proof of the claims made.

The returned VP would include the DID of the party who issued the VC, and if that DID was known, this would provide a strong basis for the claim made in the self-attested JSON to be well-regarded. As a result, the reputation of the service might increase, and the requesting agent may decide it does not need to request a VP in some subsequent engagements. Conversely, if the AS is unable to provide a VP to back up the self-attested claim, the requesting party will be able to disregard the service, and will be less likely to trust it and consider it for use in the future, and can downgrade it in future service match analysis. As a result, dynamic networks of trust can be developed and used in consideration for shaping of workflows, depending on circumstantial priorities for speed of response or depth of trust required in service selection.

This handshaking mechanism also provides support for different levels of information to be provided, and provides both parties with opportunities to have granularity on information shared. In the first instance, the service node making the request for the VP to be provided could also include a request for data from attributes not contained in the publicly available VSA-encoded JSON file, which might be more sensitive, and revealed by the AS at its discretion based on its knowledge of the service making the request and any level of confidence it had in that service. Examples might be a request for provenance, or an explanation, which have value that may not be openly shared.

Instantiating the method in the context of a typical deployment scenario has supported iterative improvements in the approach, and provided initial design validation.

5. Conclusions and future research

Adopting semantic web technologies and open web standards allows service providers to describe their service offerings using interoperable data schemas, with the potential to improve service discovery and orchestration. Providing mechanisms for service users to describe their actual experiences in using services provides a new opportunity for trusted metadata to be added to service descriptions, backed by verifiable assurance of the identity of the party leaving it. Pseudonymous identifiers introduce an opportunity for parties to build networks of trust and develop policies to make service selection choices based on fluid trust relationships with their peers. Converting these data structures into VSA vectors and building on previous work in semantic-based service discovery and orchestration via multicast service requests provides efficient and flexible construction of workflows in fragile and unstable environments. Potentially suitable service matches can be identified as a result of an efficient semantic search, bringing a wider pool of services into consideration. The field can then be narrowed by policies which prioritize selection based on the availability of experience or quality of service reports from trusted partners, resulting in dynamic and situation-aware selection of the most suitable service to perform a particular task.

This paper has introduced a method and a software architecture that supports the role of trust in service selection, and provided design validation through application to a scenario based on selection of services in a remote military surveillance setting. Instantiation of the design in the scenario has crystallized the advantages of the proposed approach of combining JSON service descriptions and service experience credentials with VSA encoding to support rapid – and trusted – service request resolution. The approach combines benefits of decentralized VSA-based workflow configuration, with additional layers of trust and selective information disclosure provided by the self-sovereign patterns of DIDs and VCs. This enables service providers to maintain confidentiality and information privacy by not publishing all available metadata, and to selectively prove or reveal more information as it determines is appropriate.

The system architecture can provide rapid response to service requests based on VSA encoded "self-attested" claims of functional capability taken from VCs issued by operators and peers. The requesting service can, if their policies determine, perform additional interaction with the service agent to request and verify proof. As a result of the successful provision of satisfactory proof, or otherwise, requesting services can build a model of the reputation of the service, and the likelihood of its self-attested claims being valid. This leads to trust becoming a selection criteria in assembling service configurations, which is of increasing importance in military and commercial settings where bad actors can seek to disrupt operations or misappropriate information through rogue sensors and service offerings. Whilst the current trust validation process is based on standard SSI message exchange protocols, future work will seek to exploit the flexibility of the VSA vector representation to perform the verification process using a vector exchange protocol.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence under Agreement Number W911NF–16–3–0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

[1] J.P. Macker, I. Taylor, Orchestration and analysis of decentralized workflows within heterogeneous networking infrastructures, Future Gener. Comput. Syst. 75 (2017) 388–401.
[2] S. Corson, J. Macker, RFC2501: Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations, RFC editor, 1999.
[3] D. Verma, G. Bent, I. Taylor, Towards a distributed federated brain architecture using cognitive IoT devices, in: 9th International Conference on Advanced Cognitive Technologies and Applications, COGNITIVE 17, 2017.
[4] P.J. Nickel, K. Vaesen, Risk and trust, in: Handbook of Risk Theory: Epistemology, Decision Theory, Ethics and Social Implications of Risk, Springer, 2012, pp. 857–876.
[5] O. O'Neill, A Question of Trust: The BBC Reith Lectures 2002, Cambridge University Press, 2002.
[6] J.A. Carter, M. Simion, The ethics and epistemology of trust, in: Internet Encyclopedia of Philosophy, 2020.
[7] C. Simpkin, I. Taylor, G.A. Bent, G. de Mel, R.K. Ganti, A scalable vector symbolic architecture approach for decentralized workflows, in: COLLA 2018 the Eighth International Conference on Advanced Collaborative Networks, Systems and Applications, IARIA, 2018, pp. 21–27.
[8] C. Simpkin, I. Taylor, D. Harborne, G. Bent, A. Preece, R.K. Ganti, Dynamic distributed orchestration of node-RED IoT workflows using a vector symbolic architecture, in: 2018 IEEE/ACM Workflows in Support of Large-Scale Science, WORKS, IEEE, 2018, pp. 52–63.
[9] C. Simpkin, I. Taylor, G.A. Bent, G. de Mel, S. Rallapalli, L. Ma, M. Srivatsa, Constructing distributed time-critical applications using cognitive enabled services, Future Gener. Comput. Syst. 100 (2019) 70–85.
[10] T.A. Plate, Distributed Representations and Nested Compositional Structure, University of Toronto, Department of Computer Science, 1994.

[11] P. Kanerva, Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors, Cogn. Comput. 1 (2) (2009) 139–159, URL http://dblp.uni-trier.de/db/journals/cogcom/cogcom1.html#Kanerva09.
[12] W3C SPARQL Working Group, SPARQL 1.1 Overview, World Wide Web Consortium, 2013, W3C Recommendation. https://www.w3.org/TR/sparql11-overview/.
[13] D. Vachtsevanou, P. Junker, A. Ciortea, I. Mizutani, S. Mayer, Long-lived agents on the web: Continuous acquisition of behaviors in hypermedia environments, in: Companion Proceedings of the Web Conference 2020, 2020, pp. 185–189.
[14] L. Sciullo, C. Aguzzi, M. Di Felice, T.S. Cinotti, WoT store: Enabling things and applications discovery for the W3C web of things, in: 2019 16th IEEE Annual Consumer Communications & Networking Conference, CCNC, IEEE, 2019, pp. 1–8.
[15] M. Lanthaler, C. Gütl, On using JSON-LD to create evolvable RESTful services, in: Proceedings of the Third International Workshop on RESTful Design, 2012, pp. 25–32.
[16] T. Pham, G. Cirincione, A. Swami, G. Pearson, C. Williams, Distributed analytics and information science, in: IEEE International Conference on Information Fusion, Fusion, 2015.
[17] H.C. Stuart, L. Dabbish, S. Kiesler, P. Kinnaird, R. Kang, Social transparency in networked information exchange: A theoretical framework, in: Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work, 2012, pp. 451–460.
[18] F. Michel, C. Faron Zucker, O. Gargominy, F. Gandon, Integration of web APIs and linked data using SPARQL micro-services—Application to biodiversity use cases, Information 9 (12) (2018) 310.
[19] F. Michel, C. Faron-Zucker, O. Corby, F. Gandon, Enabling automatic discovery and querying of web APIs at web scale using linked data standards, in: Companion Proceedings of the 2019 World Wide Web Conference, 2019, pp. 883–892.
[20] C. Bizer, T. Heath, T. Berners-Lee, Linked data: The story so far, in: Semantic Services, Interoperability and Web Applications: Emerging Concepts, IGI Global, 2011, pp. 205–227.
[21] V. Charpenay, S. Käbisch, On modeling the physical world as a collection of things: The W3C thing description Ontology, in: European Semantic Web Conference, Springer, 2020, pp. 599–615.
[22] S. Bienz, A. Ciortea, S. Mayer, F. Gandon, O. Corby, Escaping the streetlight effect: Semantic hypermedia search enhances autonomous behavior in the web of things, in: Proceedings of the 9th International Conference on the Internet of Things, 2019, pp. 1–8.
[23] S. Mayer, A. Ciortea, A. Ricci, M.I. Robles, M. Kovatsch, A. Croatti, Hypermedia to connect them all: Autonomous hypermedia agents and socio-technical interactions, Internet Technol. Lett. 1 (4) (2018) e50.
[24] N. Suri, M. Tortonesi, J. Michaelis, P. Budulas, G. Benincasa, S. Russell, C. Stefanelli, R. Winkler, Analyzing the applicability of internet of things to the battlefield environment, in: 2016 International Conference on Military Communications and Information Systems, ICMCIS, IEEE, 2016, pp. 1–8.
[25] D.E. Zheng, W.A. Carter, Leveraging the Internet of Things for a more Efficient and Effective Military, Rowman & Littlefield, 2015.
[26] T. Berners-Lee, J. Hendler, O. Lassila, The semantic web, Sci. Am. 284 (5) (2001) 34–43.
[27] A. Zschorn, H.-W. Kwok, W. Mayer, Microservice API Design to Support C2 Semantic Integration, Tech. Rep., Australian Government Department of Defence, Science and Technology, 2019.
[28] J.R. Michaelis, M. Tortonesi, M. Baker, N. Suri, Applying semantics-aware services for military IoT infrastructures, in: 21st International Command and Control Research and Technology Symposium: C2 in a Complex Connected Battlespace, 2016.
[29] S. Evdokimov, B. Fabian, S. Kunz, N. Schoenemann, Comparison of discovery service architectures for the Internet of Things, in: 2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, IEEE, 2010, pp. 237–244.
[30] C.N. Ververidis, G.C. Polyzos, Service discovery for mobile ad hoc networks: A survey of issues and techniques, IEEE Commun. Surv. Tutor. 10 (3) (2008) 30–45.
[31] C. Allen, The path to self-sovereign identity, 2016, http://www.lifewithalacrity.com/previous/2016/04/the-path-to-selfsoverereign-identity.html.
[32] F. Wang, P. De Filippi, Self-sovereign identity in a globalized world: Credentials-based identity systems as a driver for economic inclusion, Front. Blockchain 2 (2020) 28.
[33] I. Barclay, S. Radha, A. Preece, I. Taylor, J. Nabrzyski, certifying provenance of scientific datasets with self-sovereign identity and verifiable credentials, in: Proceedings of 12th International Workshop on Science Gateways, 2020.
[34] P.C. Bartolomeu, E. Vieira, S.M. Hosseini, J. Ferreira, Self-sovereign identity: Use-cases, technologies, and challenges for industrial IoT, in: 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA, IEEE, 2019, pp. 1173–1180.
[35] B. Preneel, Cryptographic hash functions, Eur. Trans. Telecommun. 5 (4) (1994) 431–448.
[36] M. Sporny, G. Noble, D. Longley, D.C. Burnett, B. Zundel, Verifiable credentials data model, 2019, URL https://www.w3.org/TR/vc-data-model/.
[37] M. Davie, D. Gisolfi, D. Hardman, J. Jordan, D. O'Donnell, D. Reed, The trust over IP stack, IEEE Commun. Stand. Mag. 3 (4) (2019) 46–51.
[38] T.S. Perry, San Diego's streetlights get smart, IEEE Spectr. 55 (1) (2018) 30–31.
[39] I.S. Ahmad, B. Boufama, Automatic vehicle identification through visual features, in: Proceedings of the 17th International Conference on Advances in Mobile Computing & Multimedia, 2019, pp. 185–194.
[40] G.E. Hinton, Mapping part-whole hierarchies into connectionist networks, Artificial Intelligence 46 (1–2) (1990) 47–75.
[41] C. Eliasmith, How to Build a Brain: A Neural Architecture for Biological Cognition, Oxford University Press, 2013.
[42] M.N. Jones, D.J.K. Mewhort, Representing word meaning and order information in a composite holographic lexicon, Psychol. Rev. 114 (1) (2007) 1–37.
[43] G. Recchia, M. Sahlgren, P. Kanerva, M.N. Jones, Encoding sequential information in semantic space models: Comparing holographic reduced representation and random permutation, Comput. Intell. Neurosci. 2015 (2015) 58.
[44] C. Eliasmith, T.C. Stewart, X. Choo, T. Bekolay, T. DeWolf, Y. Tang, D. Rasmussen, A large-scale model of the functioning brain, Science 338 (6111) (2012) 1202–1205, http://dx.doi.org/10.1126/science.1225266, http://www.sciencemag.org/content/338/6111/1202.
[45] B. Francis, Cameras, sensors & what's next for Mozilla's things gateway, 2019, URL https://hacks.mozilla.org/2019/01/cameras-sensors-whats-next-for-mozillas-things-gateway. (Accessed 11 August 2020).
[46] R.J. Wieringa, Design Science Methodology for Information Systems and Software Engineering, Springer, 2014.
[47] J. Venable, J. Pries-Heje, R. Baskerville, FEDS: A framework for evaluation in design science research, Eur. J. Inf. Syst. 25 (1) (2016) 77–89.
[48] A. Preece, D. Braines, F. Cerutti, G. Pearson, L. Kaplan, Coalition situational understanding via adaptive, trusted and resilient distributed artificial intelligence analytics, 2021.
[49] N. Suri, K.M. Marcus, C. van den Broek, H. Bastiaansen, P. Lubkowski, M. Hauge, Extending the Anglova scenario for urban operations, in: 2019 International Conference on Military Communications and Information Systems, ICMCIS, IEEE, 2019, pp. 1–7.
[50] K.K. Santhosh, D.P. Dogra, P.P. Roy, Anomaly detection in road traffic using visual surveillance: A survey, ACM Comput. Surv. 53 (6) (2020) 1–26.
[51] Google Code Archive: Word2vec, https://code.google.com/archive/p/word2vec/. (Accessed 29 September 2020).
[52] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, 2013, arXiv preprint arXiv:1301.3781.
[53] D.A. Rachkovskij, Estimation of vectors similarity by their randomized binary projections, Cybernet. Systems Anal. 51 (5) (2015) 808–818.

Iain is a Ph.D. researcher at Cardiff University. Iain is interested in applications of decentralized technologies, particularly where they improve privacy, assurance and sustainability. Current research focus includes the use of self-sovereign identity (SSI) in providing provenance and access control to AI & data assets. Iain is also interested in value tokenisation through blockchain technologies, and particularly how these concepts can help to motivate behaviour change towards meeting SDG goals.

Chris Simpkin has worked as a design engineer on high speed control systems including design of analogue control loops, dedicated digital computer control systems. Chris worked for IBM for 10 years in various areas including stress testing of IBM's flagship S/390 G5 Parallel main frames, IBM CICS and IBM Message Queuing products. In 1998 Chris qualified as an Optometrist and spent 10 years managing an Optometry business. Chris is currently writing up his Ph.D. at Cardiff University, UK, focusing in the use of machine learning algorithms for distributed data analytics applications.

Graham Bent was formally an IBM Senior Technical Staff Member and Master Inventor. He retired from IBM in January 2016. He now works for Cardiff University as a consultant. Over the past 10 years Graham has been undertaking research on large scale distributed databases; new encryption techniques for distributed secure computing using fully homomorphic encryption. He is currently involved in a new International Technology Alliance program on Distributed Analytics and Information Science (DAIS ITA). His current research is in the development of intelligent agents for distributed analytics using brain inspired neuromorphic computation.

Thomas F. La Porta is the Director of the School of Electrical Engineering and Computer Science and Penn State University. He is an Evan Pugh Professor and the William E. Leonhard Chair Professor in the Computer Science and Engineering Department and the Electrical Engineering Department. He received his B.S.E.E. and M.S.E.E. degrees from The Cooper Union, New York, NY, and his Ph.D. degree in Electrical Engineering from Columbia University, New York, NY. He joined Penn State in 2002. He was the founding Director of the Institute of Networking and Security Research at Penn State. Prior to joining Penn State, Dr. La Porta was with Bell Laboratories for 17 years. He was the Director of the Mobile Networking Research Department in Bell Laboratories, Lucent Technologies where he led various projects in wireless and mobile networking. He is an IEEE Fellow, Bell Labs Fellow, received the Bell Labs Distinguished Technical Staff Award, and an Eta Kappa Nu Outstanding Young Electrical Engineer Award. He also won two Thomas Alva Edison Patent Awards. His research interests include mobility management, signalling and control for wireless networks, security for wireless systems, mobile data systems, and protocol design.

Declan Millar is a Research Scientist based in Hursley, England. His research centres on artificial intelligence and includes deep reinforcement learning, geometric deep learning, and hyperdimensional computing. He also has an interest in theoretical quantum computing. Declan holds a Ph.D. in theoretical particle physics, jointly awarded by the University of Southampton and Queen Mary University of London. His doctoral research explored quantum field theory models beyond the Standard Model that embed new particles and how these might be discovered at the Large Hadron Collider.

Alun is Co-Director of Cardiff University's Crime & Security Research Institute and Deputy Head of the School of Computer Science and Informatics. Alun is the UK Academic Technical Area Lead for the US/UK Distributed Analytics and Information Sciences International Technology Alliance (DAIS ITA, 2016–2026) funded by the US & UK Governments and led by IBM, in which Alun also leads the project Anticipatory Situational Understanding involving a team from Airbus, BAE Systems, IBM, UCL, and UCLA. Previously, Alun served in the same role for the Network and Information Sciences International Technology Alliance (NIS ITA, 2006–2016).

Ian is a Professor at the University of Notre Dame and at Cardiff University. He has a degree in Computing Science, a Ph.D. studying neural networks applied to musical pitch and he designed/implemented the data acquisition system and Triana workflow system for the GEO600 gravitational wave project. He now specializes in Blockchain, and is co-Founder and CTO of SIMBA Chain. Ian has published over 180 papers (h-index 41), 3 books and has won the Naval Research Lab best paper award in 2010, 2011 and 2015.

Dinesh is an IBM Fellow leading a team working in the area of Distributed AI, which performs research in technology areas at the intersection of Internet of Things, Artificial Intelligence and Distributed Systems & Networks. Dinesh received his undergraduate degree from Indian Institute of Technology in 1987 and doctorate degree from University of California, Berkeley in 1991. Dinesh also holds a degree in Management of Technology from NYU Polytechnic. Dinesh is an IEEE Fellow, and a Fellow of the Royal Academy of Engineering, UK. Dinesh served as the U.S. Principal Investigator for the International Technology Alliance in Network Sciences from 2006–2016, and is the U.S. Principal Investigator for the International Technology Alliance in Distributed Analytics which was initiated in 2016.