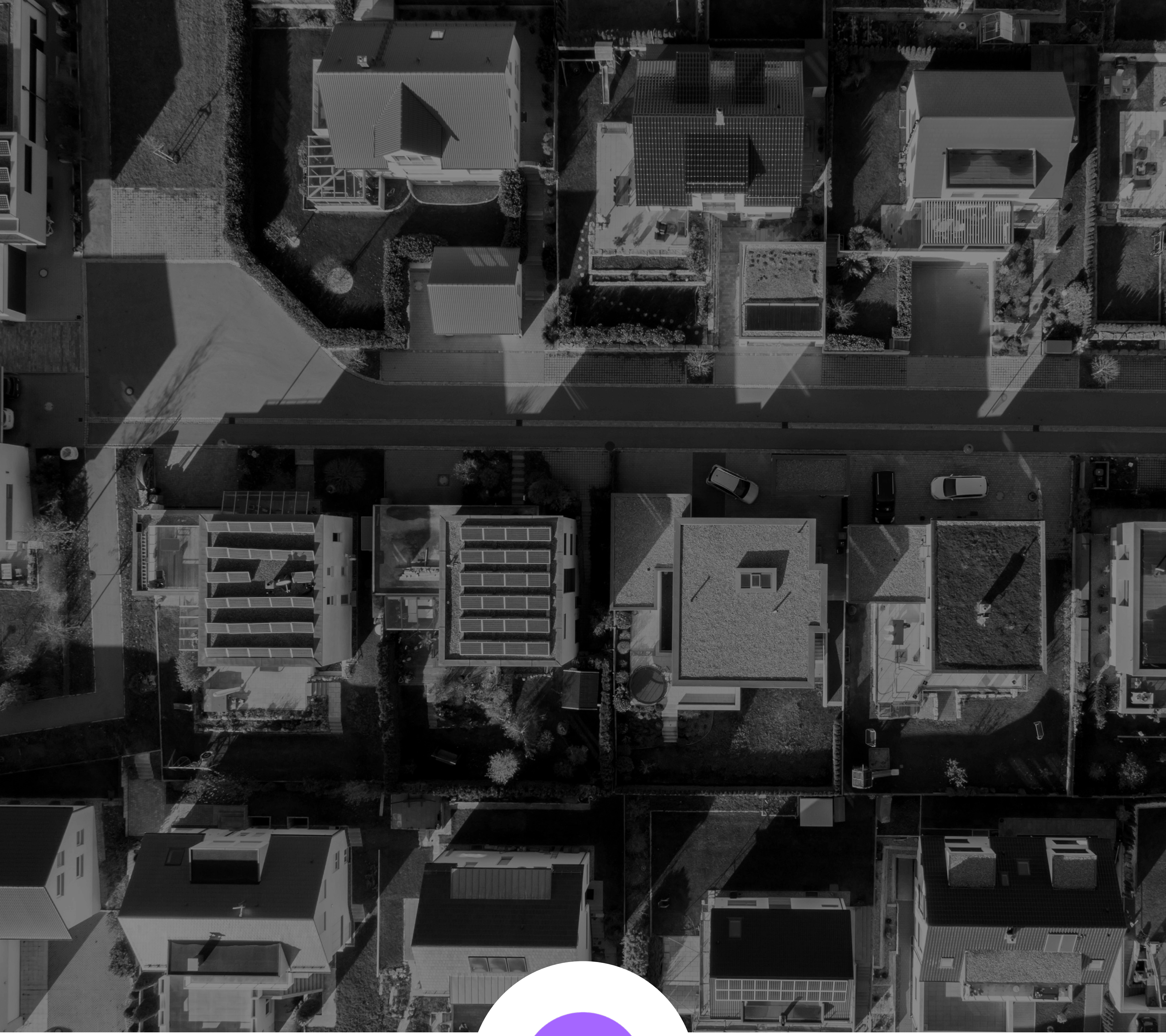


energy web

EW-DOS: The Energy Web Decentralized Operating System

The Open-Source Technology Stack for Accelerating the Energy Transition

PART 2: TECHNOLOGY DETAIL

June 2020

Table of Contents

This paper explains Energy Web's refined vision for leveraging blockchain and decentralized technologies to accelerate the energy transition, based on the past three years of hands-on experience building solutions with our global community of members. It is intended for a more-technical audience. For a high-level description of EW's current technology and roadmap intended for a general audience, please see this paper's companion piece.

Key Takeaways

- By 2030 customer investment in renewable energy, distributed energy resources (DERs), and electric mobility will exceed utility investment in generation, transmission, and distribution. Renewables and DERs will represent two-thirds of global installed electric generating capacity.
- At Energy Web (EW), we believe that **open, public, digital** infrastructure will be as integral as *physical* infrastructure for the secure, reliable operation of a highly decarbonized and distributed electricity grid. Just as grid operators have built and operated the grid's physical infrastructure over the past century, our vision is for grid operators to invest in, build, and operate digital systems that securely integrate millions and eventually billions of customer-owned DERs into core operation and planning functions.
- Based on business and technical requirements from the global energy community, the **Energy Web Decentralized Operating System (EW-DOS)** is a public, open-source stack of technologies (including the Energy Web Chain) for connecting customers, assets, and existing energy-sector IT and OT systems with energy markets and programs. EW-DOS can be applied in any regulatory context or market framework.
- We intend for EW-DOS to become a de-facto global standard for digital infrastructure in the energy sector. When EW-DOS is fully deployed, anyone—utilities, startups, individual customers—will be able to write an application on their laptop and instantly deploy it at enterprise scale without needing any of their own infrastructure. The decentralized network of EW-DOS nodes will provide all the infrastructure needs (such as messaging, storage, and consensus).
- EW-DOS comprises three layers:
  1. **Trust,** which provides consensus and immutability via the public Energy Web Chain;
  2. **Utility,** the "middleware" layer of the EW-DOS stack, which simplifies the experience of creating and using decentralized solutions; and
  3. **Toolkit,** which offers open-source templates to speed the development of applications for renewable energy markets, e-mobility programs, and DER market participation.
- EW-DOS features a universal, hardware-agnostic protocol for connecting customers, physical assets, and existing grid infrastructure with a rapidly growing number of digital applications. Within a defined territory, EW-DOS provides local stakeholders with a shared state of the attributes of operational capabilities of grid resources and participants. EW-DOS leverages self-sovereign digital identity, decentralized identifiers, a series of decentralized registries, messaging services, and integrations with legacy information technology (IT) systems to facilitate transactions between billions of assets, customers, grid operators, service providers, and retailers.
- To achieve our mission, EW is developing and deploying EW-DOS with market participants globally. Since the initial launch of EW-DOS in December 2019, we have worked on more than a dozen implementations around the world, including integrating small-scale customers into wholesale balancing markets with Austrian Power Grid AG, launching next-generation renewables marketplaces in Southeast Asia with PTT and in the U.S. with PJM EIS, supporting virtual power plants in Germany with sonnen, and building an open e-mobility platform with Share&Charge. Other members from the EW community, like The Energy Origin (TEO) by Engie in France and SP Group REC in Singapore, are leveraging EW-DOS for commercial applications as well.

Technology Requirements and Benefits

To achieve mainstream adoption of decentralized technologies, the energy sector requires enterprise-grade tools that simplify the end-user experience and streamline application development and deployment.

EW member organizations have collectively completed dozens of pilot and proof-of-concept projects over the past three years that have proven the business value of blockchain and decentralized technologies. But even as the business case for investing in these digital solutions becomes ever clearer, questions remain about how to move beyond the pilot phase and into core business operations at scale—including adoption of the Energy Web Decentralized Operating System (EW-DOS).

From a technology perspective, there are three categories of barriers (real and perceived) that we believe must be overcome:

- **End-User Experience:** Interacting with an application built on EW-DOS should be no more complicated than sending an email or shopping online. It is unrealistic to expect every customer to become an expert in public-private key management, using tokens to pay transaction fees, and verifying hexadecimal addresses. Services are needed to "hide" some of the more complex elements of decentralized technologies and deliver a seamless customer experience. This is no different than any other software or other digital solution, where good UI/UX masks and simplifies behind-the-scenes complexity.
- **Cross-Platform Interoperability:** For any given EW-DOS application to scale commercially, it will need to interact with myriad different systems, ranging from IT such as utility billing engines and databases, to operational technologies that monitor and control physical grid elements, to other public and private blockchain platforms. APIs and other tools are necessary to transfer data and initiate events between various platforms. Again, this is no different than any other software or other digital solution, where APIs and other tools allow various applications and platforms to interact.
- **Application Performance:** Too often, skepticism of decentralized applications stems from outdated myths about the limitations of blockchain's

> “transactions per second” or privacy features. The reality is that through thoughtful architectural design and integration of complementary technologies for things like messaging and data storage, decentralized applications can be every bit as performant and compliant (with regulations and corporate IT policies) as “conventional” technologies. The barrier is not technical, but rather institutional. The industry needs a simple, integrated way to deliver best practices and development tools.

Over the past three years, through dozens of workshops and projects with our global community of members we’ve identified and developed discrete solutions to each of these challenges. In late 2019, we compiled these various components into the first iteration of EW-DOS.

In the months following the initial EW-DOS publication, we completed multiple projects that applied EW-DOS in a variety of use cases, ranging from e-mobility to DER participation in wholesale markets. These experiences strengthened our conviction in our technology approach while refining our understanding of how to organize and deliver EW-DOS to the global community.

As we approach the one-year anniversary of the EW Chain, we are pleased to re-launch the EW-DOS stack in three distinct, interrelated layers:

- **Trust,** which anchors self-sovereign decentralized digital identities (DIDs) and provides a way to timestamp immutable data-sets and the associated state transitions in smart contracts via the public Energy Web Chain;
- **Utility,** which provides dedicated decentralized solutions for supporting enterprise-scale applications including high-volume messaging, user-experience tools, and back-end application services; and
- **Toolkit,** an expanded layer that now provides enhanced open-source templates for DER integration, energy tracking and trading, IoT and IT integration, and identity access management.

How EW-DOS Works

EW-DOS is a full stack of decentralized technologies and tools built for and operated by energy market participants.

As a nonprofit organization, we don't have shareholders and everything we do is in pursuit of our mission to decarbonize the global energy sector. Accordingly, EW-DOS is not a product, or an application, or a market design. It is an infrastructure.

EW-DOS is an open-source, public, digital infrastructure powered by a network of nodes from the Energy Web member ecosystem. In this decentralized proof-of-authority (PoA) model, the organizations that operate nodes are both EW-DOS users (i.e., application developers) and infrastructure providers (i.e., node operators).

EW-DOS nodes fall into two categories:

- Validator nodes, which provide security and maintain the state of the EW Chain.
- Utility nodes, which provide complementary services within the Utility Layer.

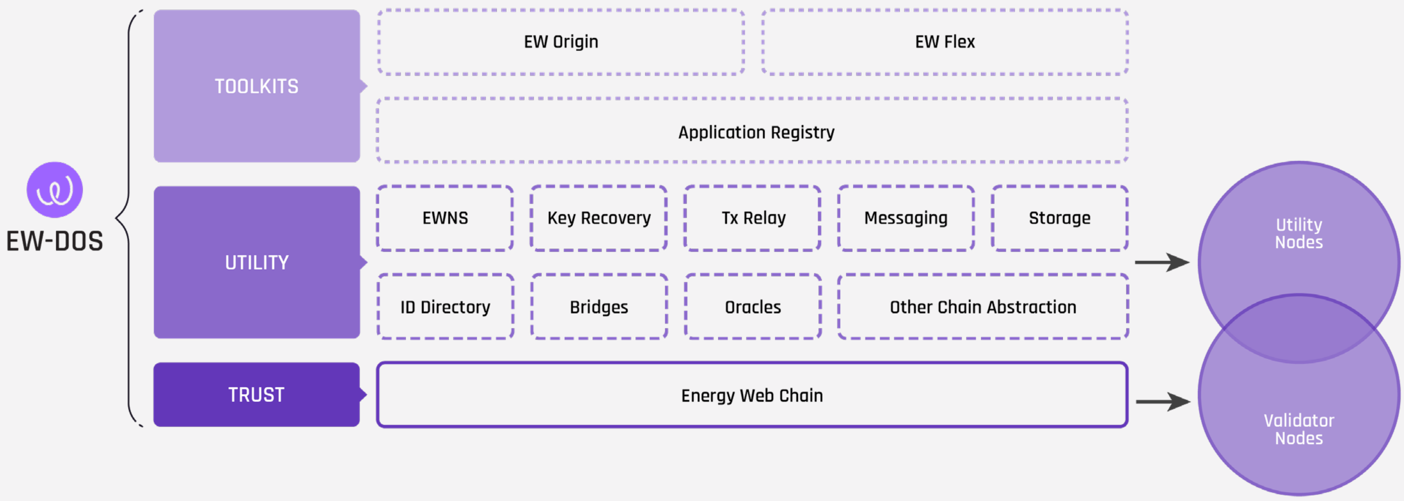
From a technical perspective, validator and utility nodes are separate (i.e., they operate different software, perform different functions, and are hosted on different machines). From an operational perspective, we expect that many organizations will host both types, but there will not be a precise one-to-one relationship. (Operating a validator or utility node requires configuring specific Docker images on a host machine. Each organization will make its own decisions about which containers to run. As a starting point from a governance perspective, all utility nodes must meet the current validator eligibility requirements.)

While there are specific eligibility criteria for operating validator and utility nodes as defined by the current governance mechanism, the EW-DOS infrastructure itself is publicly available for general use. Anyone can create a decentralized digital identity (DID), deploy a smart contract, or build an app on EW-DOS by using the native Energy Web Token (EWT) to pay for services. Toolkits, on the other hand, are simply open-source repositories available on the EW Github page.

The following section describes the three EW-DOS layers in greater detail.

Fig. 1

The EW-DOS Tech Stack



TRUST LAYER

Blockchain technology has two primary value propositions: 1) multi-party consensus about the state of data and 2) trust that a given application or smart contract will behave in a predictable, deterministic way.

In EW-DOS, the EW Chain provides trust. It is a public, proof-of-authority Ethereum Virtual Machine (EVM) based on the AuRA consensus protocol. Energy Web member organizations operate and maintain the EW Chain by hosting validator nodes. They include large utilities, grid operators, and technology companies. As of Q2 2020, more than 25 validators support the EW Chain, with over 30 validators expected by Q4 2020 (the technical limit in AuRA is 150 nodes).

The EW Chain features a native utility token (EWT) that is used to pay for transactions and other EW-DOS services. As an EVM blockchain, the EW Chain also supports all ERC standards. This includes simple fungible ERC20 and non-fungible ERC721 as well as lesser-known transferable certificates ERC1888 and many more. In addition to the main EW Chain, the validators also support a test network (i.e., sandbox and QA environment) called Volta, which can be used to evaluate pilot projects.

Since its launch in June 2019, the EW Chain has executed approximately 1.1 million transactions over 5.5 million blocks. In that same period, the Volta test network (a technical replica of the EW Chain used for proof-of-concept and pilot applications) has executed approximately 30 million transactions over 6 million blocks. To date the upper boundary of transaction fees on the EW Chain is approximately 0.00015 EWT and the vast majority of fees range from ten-thousandths to millionths of one EWT; it costs no more than a few cents to fill a block with transactions.

The EW Chain's PoA consensus mechanism currently enables roughly two to three times greater throughput capacity than the Ethereum mainnet thanks to faster block time (in the future, the EW Chain can potentially enable up to 7.5 times greater throughput capacity vs. Ethereum via a higher gas limit). The EW Chain's current parameters of a five-second block time and a block gas limit of 8 million translate into a rate of approximately 76 transactions per second (or 380 transactions per block), though in general transactions per second is not the best metric for measuring scalability (not all transactions are equal in terms of complexity and computation). A more relevant metric is gas limit (a measure of computation) per unit of time; for example in a 15-second period the EW Chain can execute 24 million gas (over three blocks) which is more than double public Ethereum (currently 10 million gas at a ~15 second block time).

UTILITY LAYER

The Utility Layer, analogous to the "middleware" layer of the EW-DOS stack, streamlines the experience of creating and managing applications by giving developers tools to easily build solutions that gain the advantages of decentralized digital infrastructure yet deliver familiar UX to customers. The services provided by utility nodes in the Utility Layer are priced and paid in EWT.

The Utility Layer addresses three broad categories of user experience / enterprise implementation:

End-User Experience:

- **Energy Web Name Service (EWNS):** Decentralized digital identities (DIDs) form the basis of most interactions in EW-DOS, but as-built, they can be challenging to use (e.g., few will ever memorize the hexadecimal string that represents their DID). EWNS dramatically simplifies the process of managing DIDs and makes it easy for users to interact with contracts, addresses, and applications by mapping human-readable names to DIDs and blockchain objects (the same way people-friendly website addresses point to hard-to-remember, long, numeric IP addresses). Users can simply map an EWNS name (e.g., name.ewc) to their DID address and create a sub-name for an unlimited number of other resources, like an email address, contract interface/address, or any other arbitrary metadata (e.g., resource.name.ewc.). With EWNS, DIDs truly become universal passports on EW-DOS: users gain the ability to interact with their DID and other applications in a way that mimics the existing internet, and developers can reserve EWNS domains for their applications and/or users.
- **DID Key Recovery:** In the EW-DOS DID implementation, the DID is itself a smart contract governed by a key pair. In the event that the DID owner loses control over the original key pair, the Key Recovery solves the "password reset" problem and prevents adversaries from unilaterally gaining control over their DID. This custodial service establishes a multi-signature wallet that governs ownership over the DID. When first creating a DID, the identity owner would not just create the DID contract and associated key pair but also delegate authority to two or more known, trusted parties (e.g., peers, utilities, etc.) to validate any future changes to the DID ownership. In this system, the identity owner simply creates a new key pair and performs the necessary verification steps with the delegated authorities to port the DID ownership to the new key if/when necessary.
- **Transaction Relay:** A transaction relay server that enables fire-and-forget transaction submission, checks whether the transaction is mined, and resubmits it in case of error. This relay also allows users to interact with the EW Chain without needing to hold / deal with EWT directly, in effect creating the same UX as traditional web applications. This enables application developers (e.g., utilities, grid operators, service providers) to build user-facing applications in which users perform transactions (e.g., create DIDs and manage DID claims) while a delegated proxy node pays for transaction fees in EWT.

Cross-Platform Interoperability:

- **Bridges:** As blockchain technology continues to mature, we expect multiple blockchain platforms and protocols to emerge for specific use cases and/or geographies. To enable identities and contracts running on the EW Chain to interact with peers on other blockchain networks, purpose-built smart contracts called bridges are used. The first two production bridges are designed to transfer tokens between the EW Chain and the main Ethereum network; one enables users to transfer native EWT from the EW Chain to an ERC-20 representation on Ethereum and the other enables users to transfer DAI stablecoins from the Ethereum network to a bridged DAI on the EW Chain. Over time functionality will expand to enable any arbitrary data or transaction to occur between networks, for example using the Arbitrary Message Bridge to trigger interactions between smart contracts on EW Chain and Ethereum. This will create a universal base layer for building bridges between EW-DOS applications and applications on any other EVM-based chain.

- **Oracles:** Blockchain-based smart contracts cannot fetch data from external systems, yet in nearly every energy-sector use case smart contracts and events require data inputs from off-chain events or systems. For use cases where it's beneficial to leverage multiple input sources (e.g., monitoring of local voltage for multi-party reconciliation, reporting of distributed solar for renewable portfolio standards accounting), we are building on top of emerging open-source protocols, particularly the Chainlink protocol, for establishing a network of independent nodes to provide event data to on-chain contracts. The EW-DOS decentralized oracle implementation eliminates single points of failure and can reduce friction in market transactions by accepting data from multiple sources concurrently. Setting up an oracle on EW-DOS involves configuring dedicated nodes that execute specific steps to retrieve, interpret, and aggregate external data.

- **Other Chain Abstraction:** Going forward we will continue to develop other services and tools that make it easier for users and applications to interact with the EW Chain. Examples include a smart contract application programming interface (API) generator that provides a standard communication interface between on-chain identities and contracts and off-chain systems and data through usage of GraphQL or RESTFul middleware, the EW Chain station (which enables applications to fetch real-time gas prices (in Gwei) from a public end point), and the EW Wallet, which provides a simple user interface for managing EWT. Additionally, most enterprise implementations will feature custom architectures of orchestration services that securely connect legacy IT systems (e.g., customer relationship management, enterprise resource planning, billing engines) as well as IoT devices to on-chain components. Designs range from server-side integrations into big SCADA systems to small, lightweight implementations that can run on a small IoT device.

Application Performance:

- **Identity Directory:** Decentralized Identifiers (DIDs) are a new type of identifier for verifiable, persistent, resolvable, and secure digital identities that are directly owned and controlled by end users. DIDs are URLs that relate a DID subject (e.g. person, organization, asset) as a means for trustable interactions with that subject and can be implemented independently of any centralized registry, identity provider, or certificate authority. DIDs resolve to DID documents, which describe how to use that specific DID. Each DID Document may express cryptographic material, verification methods, and/or service endpoints. These provide a set of mechanisms which enable a DID controller to prove control of the DID. Service endpoints enable trusted interactions with the DID subject. At the most basic level, a DID is simply a smart contract governed by a public-private key pair controlled by an individual or organization; the key pair resides on a decentralized network that is not controlled by any single party. As such, the end-user is ultimately in control of their digital identity and can grant access or supply information to other actors as needed. The EW Chain DID implementation is based on the W3C DID standard. The identity holder (the "subject") can add as much information as they like to their identity (i.e., claims) and they can get this information verified by authorities (e.g., a government, energy company, bank). Collectively, this process authenticates an identity. These verified claims can then be used to selectively disclose information to third parties, akin to a person proving that they have a valid driver's license by showing the expiry date and the picture but without disclosing their home address, name, or age. The core Identity Directory is a smart contract that contains the universal list of DIDs and associated claims on the EW Chain. This architecture gives end users greater agency over how their digital identity (and associated data, vis a vis DID Documents) is used and stored while also offering seamless interoperability with all decentralized applications running on the EW Chain. DID Documents provide a structured way to define:

- ***Authentication Mechanism:*** By which the DID subject can provide a cryptographic proof of the control over its identity.
- ***Public Keys:*** The DID document can list the authorized (RSA and ECDSA based) keys to provide digital signatures and encryption operations.
- ***Authorization and Delegation:*** A DID can authorize or delegate another DID Subject to perform certain operations on its behalf. The authorized or delegated DID can serve different business process related operations.
- ***Service Endpoint:*** The service endpoints listed in the DID document of the DID subject provide a lookup mechanism for addressable URLs to interact with the DID subject.

- **Messaging:** In general, it is only practical to perform messaging on-chain when the message (or related event) requires establishing multi-party consensus. EW's enterprise-grade decentralized messaging infrastructure enables high-volume, low-latency (e.g., machine-to-machine) communications that do not rely on blockchain transactions. In conjunction with the identity system, the messaging service creates a secure and trustworthy communication layer that doesn't rely on any single party. EW's messaging solution builds on established standards (e.g., AMQP, MQTT, STOMP) and is built in such a way that the infrastructure can be decentralized; any qualified entity can operate a node or set up their own service.

  The core characteristics of the messaging solution include:

  - A DID-based authentication mechanism for participants to establish permissions based on defined criteria (e.g., geographic location);
  - Enforced signing of messages to validate authenticity and implement on-chain settlement;
  - Integration with MQTT, AMQP, and STOMP client and server architecture.

- **Storage:** Given the volume of data in the energy sector, as well as the complicated regulations governing its use, it will be impractical to store data on blockchain at any reasonable scale. EW is developing a decentralized data storage solution for content-addressed and key value data, but in many cases existing storage solutions (e.g., either private cloud or on-premise database) will be used for commercial applications and the messaging and other chain abstraction components will serve as a connective layer to on-chain components.

  The storage system has two types of elements:

  - ***Content-addressed:*** The content-addressed data are those which must not be editable. This includes: services for DID documents, files. We are experimenting with IPFS, which was used for the implementation of the first claims store for the DID library.
  - ***Key Values:*** The key-value data contains things that need to have a predictable key or an arbitrary key. Examples include: state of an EVSE, reference to the latest version of an app. We are working to repurpose existing, centrally managed key value stores like couchdb, Redis, and Ignite.

TOOLKITS

Toolkits are like blueprints for building decentralized applications on EW-DOS. They provide generic functionality common across many different markets and use cases, but need to be customized and implemented within a specific context to create business value.

- **Application Registry:** EWF's application registry reference architecture provides market participants a standardized way to create bespoke registries with administrative features specific to a particular geography, market, or application. Application registries act as an "authorization" layer, setting the rules and roles for DIDs that wish to participate in the given market. For example, a national grid operator may create an application registry that dictates eligibility for participation in a wholesale market (e.g., DID must have verified claim as being a qualified DER and that claim must be signed by another DID that has a verified claim from the national regulator). Every decentralized application (dApp) running on the EW Chain will have at least one application registry, but a given application registry can be applied to multiple applications (in the example above, a distribution utility could coordinate with the national grid operator and use the same registry for a local congestion management program). EW's reference architecture includes a series of open-source smart contracts and dApps for managing changes and updates to the registry and creating an audit trail of all interactions between DIDs within the registry.

- **EW Origin:** Origin is a family of software toolkits that support "proof-of-impact" applications for tracking, trading, and reporting energy attribute certificates (EACs) based on industry standards. Applications built using Origin can be used for 1) creating a digital registry of buyers, sellers, asset owners, and regulators to issue and monitor transfer of digital EACs; 2) creating a digital marketplace to trade, claim, and report EACs that integrate with existing EAC registries, including legacy systems that do not use the EW Chain or Origin (e.g., the I-REC Standard); and 3) enabling generation devices to self-register in appropriate registries, automatically report their operational data, and consequently access relevant EAC markets. Origin leverages the multi-token standard ERC-1155, which combines a non-fungible token with a fungible component that is similar to ERC-20 tokens and can therefore be transferred and bundled in a similar way. In detail, the non-fungible token stores the device information and time frame of the certificate and the fungible part stores the energy volume that has been generated in that time frame. Origin consists of four modules:

    - **Registry:** The registry module stores and manages user and device information to inform subsequent participation in EAC markets/programs. The registry enables both on- and off-chain storage capabilities, ensuring that private information is safely and securely kept away from the public domain while leveraging on-chain proofs to ensure that off-chain data is verifiable and tamper-proof.

    - **Issuer:** The issuer module enables EAC issuers (e.g., regulators, standards bodies) to mint EACs upon request based on provided generation evidence. Besides just minting new EACs, the issuer module also ensures that the certificate lifecycle adheres to relevant local regulations; each regional platform or application has its implementation of the issuer module.

    - **Exchange:** The exchange module facilitates trading between buyers and sellers of EACs via an order book system where sellers post asks and buyers post bids. Once there's a match based on EAC criteria and price, the trade is executed and the EAC ownership is updated.

    - **User Interface:** The UI module is the glue that connects all underlying modules and makes them accessible to the end-user. It's also a demonstration of how easy it is to build an open, transparent, and regulatory-compliant market for EACs around the world.

- **EW Flex**: Flex is an open-source software architecture for coordinating asset (e.g., DERs, electric vehicles) data and operations across organizational and technological boundaries. Flex solves existing pain points in harmonizing transmission- and distribution-system operations, managing an ever-growing fleet of diverse distributed energy resources (DERs), and reconciling data among participants in wholesale markets or across information and operational technology systems within vertically integrated utilities. Flex consists of a series of modules that address the full lifecycle of DER participation in wholesale markets and/or demand-side management programs, from prequalification through settlement. Flex was born out of requirements for integrating one million distribution-level, behind-the-meter batteries into a wholesale frequency regulation market, but can be applied in any regulatory or market context from utility "bring your own device" programs, to deregulated competitive markets, to emerging electric mobility programs. Flex is a modular architecture featuring four modules:

    - **Flex Nodes:** Decentralized clusters that provide and execute the business logic of the market or program. Nodes are responsible for managing the lifecycle of asset offers (or operating schedules), and also implement algorithms for optimizing the matching of offers and requests according to the specific use case / market rule. By varying the functionality of the nodes, grid operators can procure a multitude of different services from capacity, to energy, to ancillary services.

    - **Flex Clients:** Multiple software clients, including Flex IoT, Flex UI, and Flex Mobile, which provide user interaction interfaces with Flex Nodes. The Flex IoT module implements a small and efficient operating system for embedded devices. It allows any IoT asset—including DERs and EVs/EVSE—to have its own digital identity via the use of secure elements, either based on cryptographic chips or mobile SIM cards. It also allows the assets to participate automatically in energy markets by cryptographically signing flexibility offers and, at the same time, verifying incoming requests for flexibility. The Flex UI and Flex Mobile clients provide frameworks for developing web- and mobile-based frontend applications where grid operators and prosumers manage various market processes, from DID creation and registration, to offering/scheduling, to settlement.

- **Flex Bridge:** Fully customizable and pluggable integration modules for Flex Nodes, enabling communication and coordination with grid operator IT (e.g., market trading platform) and OT (e.g., load frequency controller) systems.

- **Flex Governance:** A series of EW Chain components including smart contracts and multi-signature wallets that are used to govern the execution of market or business logic via Flex Nodes. These tools enable a regional consortium (for example, a TSO, several DSOs, and a regulator) to set the rules and criteria for prequalifying prosumers/DERs in specific markets, administering data access / management privileges, establishing constraints and limitations that impact offer optimization, and monitoring DER behavior over time.

Tech Development Pipeline

EW's Plan to Expand and Enhance EW-DOS

There is an unprecedented amount of investment, talent, and innovation pouring into blockchain and related decentralized technologies. We know that EW-DOS will need to continue to grow organically and adopt other solutions from the wider community as time goes on.

As we look ahead to 2021 and beyond, there are a number of items that are high priorities on our development roadmap, including:

- **Preparing for "EW Chain 2.0":** EW will work with the community of validators to experiment with emerging technologies and consensus protocols to support the evolution of the Energy Web Chain. Our research activities include scaling via layer one and two solutions including but not limited to beacon chain architectures (e.g., ETH 2.0), Substrate, Polkadot, and Cosmos architectures, state channels, and bridging solutions that enable the integration and interaction with other chains.

- **Expanding DID:** We plan to extend the existing EW-DOS DID implementation to other DID methods, including DID resolvers for Sovrin and Kilt.

- **Enhancing Privacy:** We will continue to further develop privacy-preserving features including zero-knowledge proofs and verifiable claims to support additional on-chain activity (e.g., financial settlement).

- **New Toolkits:** We are continuing to develop additional toolkits for renewable energy, DERs, and electric vehicle market participation as we gather additional requirements from our global network of members. In-development toolkits include functionality that enables digital identities to settle payment; automatically conduct evaluation, measurement, and verification (EM&V); post value in escrow; and engage in complex transactions (e.g., financial contracts).

Contributors

All contributors from EWF unless otherwise noted. Contributors listed alphabetically.

Nitin Gavhane, Mani Hagh Sefat, Sam Hartnett, Walter Kok, Doug Miller, Jesse Morris, Mario Pavlovic, Micha Roon, Meerim Ruslanova

Suggested Citation

EW-DOS: The Energy Web Decentralized Operating System. *An Open-Source Technology Stack to Accelerate the Energy Transition. PART 2: Technology Detail.* Energy Web. June 2020.
<< https://www.energyweb.org/reports/EWDOS-Technology-Detail >>

About Energy Web

Energy Web is a global, member-driven nonprofit accelerating a low-carbon, customer-centric electricity system by unleashing the potential of blockchain and decentralized technologies. EW focuses on technology integration and development, co-creating standards and architectures, speeding adoption, and building community.

In mid-2019, EW launched the Energy Web Chain, the world's first enterprise-grade, open-source blockchain platform tailored to the sector's regulatory, operational, and market needs. EW also fostered the world's largest energy blockchain ecosystem, comprising utilities, grid operators, renewable energy developers, corporate energy buyers, and others.

Energy Web has become the industry's leading energy blockchain partner and most-respected voice of authority on energy blockchain.

For more, please visit https://energyweb.org.

Learn More

For a more-detailed technical description of EW's current technology and roadmap, see this paper's companion piece, ***EW-DOS: PART 1: Vision & Purpose***.

To explore EW's existing technology stack, visit our Github and Wiki.

To learn how to work with EW, contact us at info@energyweb.org.

To learn more about EW's mission, visit energyweb.org.