Programme Area: Smart Systems and Heat

Project: WP3 Case Study Development Hybrid Heat Pumps

Title: ICT Architectural Considerations for Future Energy Systems.

Abstract:

This report describes the specific considerations and approaches taken for developing ICT slices (bounded areas of functionality that can be specified, designed and implemented in manageable stages) through the architecture. This report is based on the exemplar Energy System Architecture is as described in the "Energy Systems Architecture Methodology: Enabling Multi-Vector Market Design," paper.

Context:

The Case Study Development project was commissioned by the ETI in Nov 2015 as part of Work Package 3 (WP3) of the Smart Systems and Heat Phase 1 programme. The project was intended to develop Market, Business and ICT Integrated Solutions through system architectures to provide evidence and guidance for business strategy and policy to enable the UK low carbon heat transition. This was achieved through understanding the inter-relationships between market frameworks, business process, asset management and ICT solutions. Primary focus was on the implementation at the local level, but in the context of a national energy system transition.

ICT Architectural Considerations for Future Energy Systems.

Methodology and Exemplars

Sub-Deliverable ID WP3-19

Document Control

ESC programme name	Smart Systems and Heat Phase 1 Work Package 3
ESC project number	SSH Phase 1 WP3 EPO Use-Cases (ESC00053)
Version*	1.0
Status	**Draft**: Contains preliminary information only. **Approved**: Contains reviewed and approved content. (delete as appropriate)
Restrictions*	
Release date	13/Jul/2018
External release ID	n/a

* Refer to the Information Classification Policy.

Review and Approval

	Name	**Position**
Author	Peter Brookes / Daniel Mee	ICT Architect / Energy Systems Architect
Reviewer(s)	Daniel Mee	Energy Systems Architect
Approver	John Batterbee	Head of Architecture and Transformation

Revision History

Date	**Version**	**Comments**
13/Jul/2018	V1.0	First Release

Document Protection

Arising IP

Description	Owner	Category
See IP Registers: ESC_Intellectual_Property_Register WP3 ESC00050 IP Register FINAL ESC_Intellectual_Property_Register WP3 ESC00053 IP Register FINAL	ETI	See Register

Background IP

Description	Owner	Category
See IP Registers: ESC_Intellectual_Property_Register WP3 ESC00050 IP Register FINAL ESC_Intellectual_Property_Register WP3 ESC00053 IP Register FINAL	ETI	See Register

Trademarks, licenses and disclaimers

Description	Owner	Category
EnergyPath is a registered trademark of the Energy Technologies Institute LLP	ETI	Trademark

Contents

Smart Systems and Heat 1 Programme

Heating accounts for almost one third of total UK carbon emissions; to achieve the 2050 target of an 80% reduction in carbon emissions, the UK must decarbonise the domestic heating market at the rate of 20,000 homes a week by 2025 – the current rate is less than 20,000 homes a year.

The Smart Systems and Heat (SSH) programme is designed to help innovators address this market failure and unlock the commercial opportunity of low carbon heating, by:

- Addressing the technical, regulatory, economic and social barriers that block new low carbon heat products, services and business models getting to market,
- Establishing a range of platforms, insights and modelling tools to help innovators discover new low carbon heating solutions that consumers value,
- Bringing innovators, businesses, local authorities, networks, policy-makers, regulators and consumers together to create new markets that deliver low carbon heating solutions at scale.

The Energy Technologies Institute (ETI) launched SSH and funded Phase 1 of the programme, which was delivered by the Energy Systems Catapult and its partners.

Introduction

The deployment of low-carbon technologies, at a large enough scale to impact decarbonisation of heat in the UK, is driven by complex issues, the least of which are technological. There are sufficiently mature products that could decarbonise much of the housing stock. What is not clear are the market arrangements, business models, service propositions, financing approaches, policies and regulations that are needed to support the introduction and integration of such products.

To find solutions to the barriers of deployment of low-carbon technology requires systems thinking to analyse problems from a holistic perspective. Systems thinking often creates significant benefits; such as lower overall costs, better cyber-security resilience, more effective markets and greater equity between different social groups. But these benefits have so far been nearly impossible to move from theory into everyday industry practice. This is because it is too abstract, too big and too long-term for most private investors' horizons.

SSH1 WP3 responded to this issue by demonstrating how a high-level concept for a future energy market structure can be developed into enough detail to implement and how objective design choices can be made, using insight from whole system simulation.

This report describes the specific considerations and approaches taken for developing ICT slices (bounded areas of functionality that can be specified, designed and implemented in manageable stages) through the architecture. This report is based on the exemplar Energy System Architecture is as described in the "Energy Systems Architecture Methodology: Enabling Multi-Vector Market Design," paper (see further reading below).

The exploration of possible ICT architectures, herein, follows the methodology described in that paper starting with Section 2, in this report, which describes the key dimensions of difference for ICT architectures.

Section 3 discusses the relative requirements of the different components of the exemplar Energy System Architecture.

Section 4 describes how the approaches might be applied to the different components and what the resulting advantages and disadvantages might be.

This paper does not consider different technologies for implementation. For example, technologies like blockchain may be valid for incorporation as a solution within the discussions below although this paper is solution-agnostic.

Further reading

This report is a summary of the studies, considering ICT architectures, from the SSH1 WP3 projects. This report is one of the documents and deliverables submitted to the Energy Technologies Institute Ltd as part of the SSH1 programme.

The ETI has published a paper entitled "Tools for Future Energy Systems" [1] (Energy Technologies Institue, 2017) to explain the purpose of architecture and supporting tools.

The ESC, as part of the SSH phase 1 programme, has published a paper entitled "Energy Systems Architecture Methodology: Enabling Multi-Vector Market Design," (Energy Systems Catapult, 2017) which is available on both the ESC and ETI's websites[2] which summarises the methodology for developing architectures and explains, in some detail, some potential candidate architectures, herein referred to as the exemplar architecture. However, this paper provides further explanation for pursuing that exemplar architecture and explains the alternative implementation approaches explored by the ESC which may be returned to later.

WP3-11, a report titled: "Facilitating the mass deployment of Hybrid Heat Pumps" should be read as a pre-requisite to this document. It explains the wider rationale behind this paper and sets the architectural context within which this ICT picture is framed.

[1] Paper available at https://d2umxnkyjne36n.cloudfront.net/insightReports/Tools-for-Future-Energy-Systems.pdf?mtime=20171218110743

[2] Paper available at: https://es.catapult.org.uk/publications/energy-systems-architecture-methodology-enabling-multi-vector-market-design/

Fundamental ICT Principles

This section describes the key considerations of complex ICT systems in the sense of the characteristics that need to be balanced with the specific needs of the system in question. In general, some systems need to be more secure than others, for example banking software, but other systems might place scalability needs above cost e.g. social media platforms which start small and attract more users.

Performance Parameters

Scalability

Scalability is the ability of a system to deliver a uniform level of service as the load on it increases. If, by autonomously adding resources, a system can increase its capacity to deal with additional data processing requirements it can be called auto-scaling.

Equally as important as scaling, is the ability of a system to automatically de-scale as the demand for its (expensive) services reduces; without this crucial de-scaling capability, a system will consume ever-increasing amounts of resource and will not be cost-effective to run.

Scalability is often defined in 2 dimensions, namely: Horizontal scaling (in/out) is the ability to add or remove more nodes to the system, for example adding a new server centre or another provider to the existing infrastructure. Vertical scaling is the process of increasing resources (e.g. processing cores, memory) to an existing node.

Predicting quantitative scalability in a design, rather than in an implemented system, is not straightforward. Furthermore, the scalability of performance (adding more power to solve a problem) often suffers from diminishing returns as described by Amdahl's law (Amdahl, 1967). Despite this there are some guidelines to improve scalability, for example:

- Asynchronous systems which have an effective queuing mechanism rather than sitting and waiting (i.e. a synchronous approach) tend to scale more readily.
- Design for horizontal scalability is preferred over vertical. Costs increase linearly with adding more nodes, but approach an exponential increase with boosting the power of an existing node.
- Maintenance must also scale so designing systems for upgrade without being taken off-line is essential.
- Avoid single-point failures. Failures happen and designing for redundancy will lead to a mind-set of distributed functions (not necessarily distributed systems), which will later scale.

Performance

Performance is a measure of the ability of a system to do a certain amount of work within a set interval.

There are many factors to be taken into consideration for performance such as:

- Processing speed, the number of instructions that can be handled in each time frame.
- Throughput, the rate of processing of information.
- Response time, the total time to process an instruction and reply.
- Latency, the time delay between the cause and effect of an event.

All the parameters listed above, alongside other measures of performance are easily quantitatively assessed after development, many are also predictable at the time of design. That said, the performance of a complex set of interconnected systems is very difficult to predict.

Availability

Availability measures the amount of time that a service/resource is accessible. Reliability *contributes* to availability, but availability can be achieved in a well architected system even when components fail. In a system that is designed with redundant components and failover in mind, any single component can fail, impacting reliability, but the service will still be available due to the redundant design.

Reliability

Reliability is:

1. A measure of the ability to exhibit consistency, transactional integrity and accuracy.
2. A measure of resistance to failure.

Most often, reliability drives maintenance costs rather than the performance perceived by customers since component failure leads to maintenance but systems with sufficient redundancy will continue to operate with little or no external visible impact.

Reliability is probabilistically predictable prior to implementation and measurable following the development.

Interoperability

Defined by the European Interoperability Framework as, "The ability of disparate and diverse organisations to interact towards mutually beneficial and agreed common goals, involving the sharing of information and knowledge between the organisations, through the business processes they support, by means of the exchange of data between their respective ICT systems"

If a system can flexibly and easily communicate and exchange information with other external (and/or internal) systems it can be classified as an interoperable system. An interoperable system can easily exchange and reuse information internally as well as externally. Openly interoperable systems (those with published interface definitions to allow any interested party to connect) can be open technically but still have commercial controls on each side of an interconnection to ensure that value moves between organisations.

Extensibility

If a system can have additional functionality added, or existing functionality adapted, without impacting its existing features, then it is extensible. Extensibility can be measured whenever new features or services need to be added to system but not as a global variable. Instead a system may be judged to be extensible to some requirements and not others. It is, therefore, very important to identify potential areas of high-value extensibility early on in design to maximise the chances of inclusion at a later stage.

Security

A secure system is resilient to attempts to compromise its function and performance. Some of the core considerations for cyber security are:

- **Boundary firewalls:** Minimising incursions at boundaries to the system
- **Human support and responsibilities:** Accepting that human fallibility is a risk and building systems which are most resilient and mitigating for that risk.
- **Patch management:** Ensuring updates are maintained and identified vulnerabilities closed as soon as possible.
- **Secure configuration:** Closing unsecured ports and services across routers, switches and firewalls.
- **Access control:** Account management and closure, controlled access (need to know) restrictions.
- **Malware protection:** Defences against attack and recovery following incursion, prevention of spread of malware and detection reporting.

The Centre for Internet Security (Centre for Internet Security , 2018) describes 20 controls to be considered for implementing more cyber secure infrastructure. The US National Institute of Standards and Technology (NIST) have developed a Framework for Improving Critical Infrastructure Cybersecurity (CSF) which provides a common taxonomy and plan for groups to assess and manage their current and future cyber security provisions.

In addition, mitigations for incursions, might exist outside of the ICT domain. Accepting the adage, "if it can be hacked, it will be hacked" means that localised, independent, robust measures are likely to be necessary and complimentary to usual cyber defences.

Manageability

Manageability defines how easy it is to administer a system, ensuring its continued health with respect to scalability, reliability, availability, performance, and security. Manageability deals with system monitoring and the ability to modify its configuration to dynamically improve the quality of service it provides. Manageability is often improved as complex problems are decomposed into smaller challenges which can then be dealt with independently.

Maintainability

A maintainable system can have flaws and bugs in functionality rectified without detrimentally affecting other system services. Designing a system to have loosely-coupled, encapsulated components assists in delivering maintainability.

Costs

Broadly costs are separated in to 4 main areas:

Capital Expenses: The purchasing, installation and set-up costs.

Running Costs: Regular costs for running and upkeep, including monthly service fees, energy usage, physical security etc.

Upgrade Costs: The costs to scale the solution (more capital outlay and scalable running costs depending on the size of upgrade).

Decommissioning Costs: The costs to shut-down and dispose of any equipment.

The major differentiator of costs is dependent on the decision of self-ownership versus renting "cloud" services. Capital outlay, upgrade and decommissioning being significant for ownership options but running costs being the issue for cloud offerings.

Design Considerations

Self-Healing

In a complex system, things often go wrong – communications unexpectedly stop working, hardware fails, power is interrupted. It is therefore a good approach to expect these failures, and to design applications to fix themselves when problems arise by building the platform to:

- identify issues and failures
- smoothly and predictably handle issues
- output descriptive and useful error messages to searchable logs that can be used to diagnose root causes and related factors

The way an application handles failure depends on how critical the application is, for example, if the system must always be available then – in the event of a regional failure - an identical "failover" system, running on a separate hosting and communications infrastructure in a different region, might be "swapped" in to replace the failing one.

Usually, though, failures are not so wide-reaching as regional outage, and involve the loss of database connections, power, network communication or hardware and are short-lived and localised. They can, however, propagate into bigger systemic issues.

Steps to help self-healing include:

Retry failed actions: Transitory problems might take place because of a brief loss of network service, a database connection issue, or a service that is temporarily too busy to respond to a request. The first step in building a self-healing platform is to use code that retries failed operations so handling momentary failures does not always generate bigger issues.

Guard remotely failing services: If a transitory failure persists, then continually retrying it can result in a failing service being put under ever-increasing load, which can lead to progressively damaging failures, as requests for the service increase in volume and frequency.

Implement load smoothing: Sudden jumps in traffic or processing can overwhelm system services, to avoid this, a load smoothing pattern can be implemented to smooth peaks.

Failover: If a service is unavailable, then failing over to a different service instance is an easy way to solve transient errors.

Tally prolonged transactions: Keeping a running record of any protracted operation means the entire transaction does not need to be recalculated in the event of a failure. The

record can be used to continue the operation from the point of failure when another process continues the calculation.

Staged functionality degradation: It is usually better for a service to provide some functionality, so a degraded mode of operation is preferable.

Independence

Most large-scale applications are composed of multiple services - web front ends, databases, business logic, reporting etc. To realise scalability and reliability, multiple instances of each service should run concurrently, with a load balancer in front of them distributing requests in an asynchronous manner. Considerations include:

Command Query Responsibility Segregation: is an architectural model which separates multiple operations and allows for scalable, asynchronous systems.

Eventual Consistency: is a view that databases eventually become synchronised even if out-of-date information is sometimes returned.

Partition Data: each database / system maintains is responsible for maintaining its own persistent data set.

Evolutionary

All application change over time, and if an application's services are tightly-coupled, it makes it difficult to make significant changes, because a change in one part of the code may affect code elsewhere. Services are a popular way of building an evolutionary architecture, because they deal with many of those concerns. Considerations for systems that are evolutionary include:

High cohesion & loose coupling: A cohesive service provides only functionality that is logically related. *L*oosely coupled services can each be independently altered without having to change the others. Well-designed services should have high cohesion and loose coupling.

Abstract domain knowledge: A service should abstract the domain knowledge that it implements, so that clients do not need to understand its low-level concepts.

Asynchronous messaging: In asynchronous messaging, the message publisher and the message subscriber do not have to be online together at the same time. Eventually, a sent message will be delivered whenever the recipient is able to receive it.

Permissions and Access Rights (Medium) (InfoQ, 2018)

Permissions and access rights pertain to the setting of role-based authorities to view, create and edit data or functions within the operating environment.

Across self-contained systems, managing permissions and access rights, is a relatively straightforward technical activity with the biggest complications often being around revoking access once users change roles or leave an organisation and in monitoring authorised users for unauthorised behaviours.

Across distributed or decentralised applications then such access becomes more difficult to manage. In multitenancy systems, for example in public clouds, threats emanate from untrusted users crossing from areas, where they have legitimate access, to sub-systems they have no access to. Side-channel attacks, and there are many methods to promulgate them, refer to using some inherent knowledge of the computer system to breach and access to part of a distributed system makes it easier to learn about, and exploit, such information. Distributed and decentralised systems need a sophisticated security architecture to ensure secure access whilst maintaining / improving interoperability between adjacent systems.

Multitenancy: Multitenancy refers to the presence of multiple parties sharing resources (exactly as described in a cloud infrastructure).

Federated Identity Management (FIM): Decentralised administration means that individual sub-systems or components have a level of autonomy and are responsible for their own access rights. However, given that users need to access information / data / services across multiple sub-systems, layers or components, FIM is used to ensure that identification credentials and permissions can be communicated and/or transferred.

Virtualisation: Different parts of distributed or decentralised systems often have different security or access policies which can make interference between them a particular risk.

Collaboration Approaches: Federated collaboration requires adherence to an overarching policy that multiple clouds or services ascribe to and follow leading to a federated system which experiences mutual trust and shared access rights. Loosely coupled systems are, generally, more independent, flexible and autonomous, sharing resources according to service level agreements leading to access rights governed by local policies. Systems which follow ad hoc collaboration approaches might not be aware of all the shareable services available but rather discover as part of authentication processes. These authorisation mechanisms need to be defined to allow other systems / clouds to join and leave in pseudo-random ways.

Secure Digital Identifiers

Secure digital identifiers (SDI) allow for trusted 'proof of identity' to be established. They work at multiple levels and in more integrated systems are essential for ensuring that communications and commands are passed between 2 or more intended systems.

Currently digital identification is not fully mature and though there are examples of transitory identification going 'digital' such as boarding passes, gig and movie tickets, banking transactions overall digital identity is still in trial stages (for example the UK digital driving licence initiative (The Guardian, 2017)).

Secure digital identification will be increasingly important in distributed future systems to ensure that the correct user is receiving the services they've paid for and are billed accordingly. Furthermore, secure digital identifiers will be essential for identifying pieces of equipment offering external services, such as demand side response, such that the devices are sent compatible control signals and report their consumption / savings as required.

For all forms of SDI, cyber security is a vital consideration as the risks associated with spoofing identity can be damaging (for example incorrect billing) or from data-privacy breaches (for example people's whereabouts can be tracked without permission).

Components, for example, smart appliances will also need unique Secure Digital Identifiers to enable communication routing to take place.

Risk Mitigation

Black Start Provision: Black-start, or the ability to restart after a partial or total shutdown of the grid, needs to be considered from 2 dimensions in future smart grids:

1. What assistance can be given from connected digital 'smart' equipment?
2. How can 'smart' equipment not make things worse and contribute to the grid re-tripping?

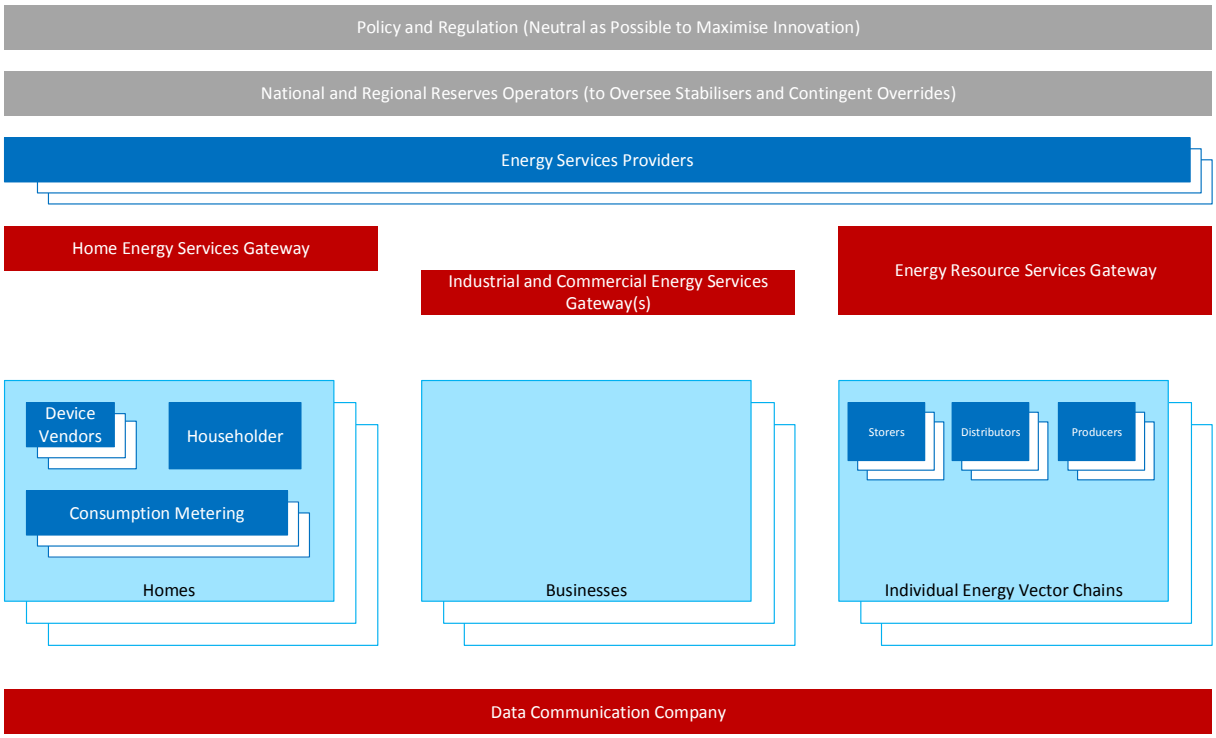
In most occurrences the responses to these questions need to be pre-determined and pre-installed into equipment to ensure that the procedure is followed when power is restored, even if the communication networks are down (power will return before most communications infrastructure following a power outage).

Frequency Responsiveness: Localised responses to extreme frequency excursions may provide a last line of defence against cyber-attacks or failures which affect a lot of controllable demand or supply. To be effective this is likely to need to be independent from software control to prevent circumvention.

ICT Considerations within Exemplar Architecture

Summary of Exemplar Architecture

As discussed in the WP3-11 (Energy Systems Catapult, 2018) report an exemplar architecture was chosen and developed, specifically with the purpose of having an example to develop the architecture and design process around. Therefore, this paper explores that exemplar architecture but this method would need to be repeated for future alternative architectures.



Red indicates shared platforms (single instances which require collaboration to develop, operate and benefit from) whilst blue indicates proprietary (many competing instances developed and owned by one or more private institutions)

Figure 1 - Conceptual Future GB Energy System Architecture Diagram

The new elements in the architecture above are the gateways, namely the Home Energy Services Gateway (HESG), plus the industrial and commercial counterparts, and the Energy Resource Services Gateway (ERSG) alongside the Energy Services Provider (ESP).

For an understanding of the functionality of these components refer to the "Energy Systems Architecture Methodology: Enabling Multi-Vector Market Design," paper. (Energy Systems

Catapult, 2017). A summary is provided below to highlight their key functions that would drive ICT functionality:

The ICT needs of the new items are given in detail below whilst the other elements are expected to be familiar with a wide audience but a summary of their ICT functions is provided to give some context for the interfaces they have / might need in the future to communicate with the other actors described.

Policy and Regulation

A future regulator and/or policy maker are likely to need to be more technologically connected to examine information and to satisfy themselves that the markets and systems are operating as intended. In a highly digitalised market place it is anticipated that there will be new ways to try to 'game the system' and therefore the regulators will need to stay one step ahead. Policy decisions might be improved as machine learning and 'big data' analysis provides more insights into consumers (industrial, commercial and domestic), into resource utilisation (output from generators, distributed resources) and into performance of networks (congestion, voltage excursions etc.) can lead to useful evidence.

National and Regional Reserves Operators

In the future, higher levels of digitalisation and high-speed control might change the dynamic control of energy networks. Take, for example, an in-home digital solution that can respond to market price signals. Assuming half-hourly (HH) pricing then an increase in price in a new half-hour time slot would result in mass load shedding as those digital solutions sought to minimise consumers' costs. This might prove difficult to manage in, for example, an electricity system where some physical systems are just not capable of ramping up and down that quickly. More digital monitoring and control systems will enable operators at regional, national and possibly across national borders to respond to more dynamic control.

Home Energy Services Gateway (HESG) and similarly for Industrial and Commercial Energy Services Gateways

The Home Energy Services Gateway is an open intermediary to connect any Householder to any Energy Services Provider; and any Energy Services Provider to any device from any Device Vendor.

- For Householders / customers, it enables comparison of offers from competing service providers with a comparable language of service attributes and performance levels; and it enables use of their data to drive a market instead of being tied-in to a limited range of services locked to closed devices
- For Energy Services Providers, it establishes a common language with which to understand, shape and bound a Householder's service expectations; and it enables access to the critical data and devices Services Providers require to design, price and deliver innovative high value services
- For Device Vendors, it enables access to a new revenue stream in return for making their devices available to Energy Services Providers for the purposes of executing new services. Device performance, usage permissions and any fees are defined in standard device class Service Level Agreements
- For product and service developers it enables revealed consumer preferences to drive investment
- Provides an interoperable platform between competing digital Home Energy Management offerings
- Provides open, independent data of interactions and actions taken to allow for feedback and corrective-action with authorised parties

The key enterprise architecture priorities for the HESG are:

Performance Parameter	**Considerations**	**Priority / Importance**
Scalability	Must be able to scale dynamically in several directions 1) to handle more traffic as more commands, need to be moved around at peak times and; 2) to support more customers as they are included and; 3) to support more device classes of equipment	High

Performance	Architecture to be developed to not require real-time centralised signals due to cyber security threats. Many functions, such as; setting new service plans, collecting usage statistics, suggesting new service offerings, can happen in non-real time.	Low
Availability	Consumers will want access when they want access and service outages, within the HESG, would prevent consumer use and, would likely, cause significant frustration.	High
Reliability	Most equipment will be in industrial / commercial setting and so component failure will be expected and managed. Availability is more critical so redundancy is important.	Low
Interoperability	Early definition to support interoperability will be critical.	High
Extensibility	The addition of new device classes, customers, suppliers etc. is a function of the tool – they are really a measure of scale. True extensibility requirements will be limited.	Low
Security	The ability to control consumer's appliances could open up misuse both at an individual level (control of equipment could provide nuisance outcomes) or at an aggregate level (control of equipment could be used to rapidly alter demand and lead to network instabilities etc). Need to ensure that unauthorised usage is minimised and managed.	High
Manageability	Continuing to update and scale is important, however, much of the ICT systems with HESG are likely to be owned by businesses and corporations meaning that this is a lesser consideration than, for example, components in people's homes.	Mid
Maintainability	It is critical that systems are easily maintained and that bugs / problems are easily rectified without having to affect multiple, distributed components	High
Costs	Capital costs are less critical than the operating costs since the HESG will handle very high volumes of low-value transactions and therefore the cost per transaction needs to be minimised.	Mid (capital) High (operating)

Energy Resource Services Gateway (ERSG)

The Energy Resource Services Gateway is an open intermediary to connect any Resource or Asset Owner to any Energy Services Provider.

- For Service Providers, it enables comparison of offers from competing resource providers
- For Service Providers, it facilitates bi-directional status and control information flows between Resource Providers to enable the various actors to optimise their operations.
    - The resource status feeds include: for storage, factors such as current volume of energy stored; for distribution, factors such as current network headroom in a given network area; and for production, factors such as standby status, wind forecasts, etc.
    - The resource control requests include: for storage, factors such as time profile of energy to take-in/-out; for distribution, factors such as network operator requests to curtail demand; for production, factors such as the time profile for energy production, standby preparedness and short-notice requests to increase or decrease production.
- Consumption Metering collated via the Data Communications Company enables Service Level Agreement compliance and usage to be traced for the settlement of transactions between Energy Services Providers and their Resource Providers
- For all energy users the ERSG, allows for trading different parameters such as capacity, carbon intensity, flexibility, availability of resources alongside kWh
- For investors, monitoring market utilisation of resources sends investment signals to inform on what to build

The key enterprise architecture priorities for the ERSG are:

Performance Parameter	**Considerations**	**Priority / Importance**
Scalability	New operators / customers are likely to be added infrequently and in smaller volumes.	Low
Performance	Time-critical requests will likely involve the ERSG so high-performance communications systems are necessary.	High
Availability	Customers will want access when they want access and service outages, within the ERSG, might cause system control issues.	High

Reliability	Most equipment will be in industrial / commercial setting and so component failure will be expected and managed. Availability is more critical so redundancy is important.	Low
Interoperability	Many organisations will have bespoke control equipment, developed over many decades so new interfacing systems must be developed to be compatible with as many as possible.	High
Extensibility	As new technology arrives, ERSG will need to adapt.	Mid
Security	ERSG might interface with systems which are considered critical national infrastructure and must therefore conform to national cyber security rules.	High
Manageability	Much of the ICT systems with ERSG are likely to be owned by businesses and corporations meaning that this is a lesser consideration than, for example, components in people's homes.	Mid
Maintainability	It is critical that systems are easily maintained and that bugs / problems are easily rectified without having to affect multiple, distributed components	High
Costs	ERSG will handle low volumes of data transactions but each one being relatively high-value.	Mid

Energy Services Provider (ESP)

Energy Services Providers essentially take responsibility for ensuring agreed service outcomes are delivered to Householders and/or Businesses. They assemble the required supply chains and optimise their day-to-day operations to drive-up customer satisfaction while driving-down costs. They are the counterparty for energy Resource Provider capacity contracts.

- Deliver differentiated experiences based outcomes to consumers (e.g. time to warm a home/room etc)
- The ESP uses data to underpin its offerings to consumers and to reveal preferences on ability and willingness to pay for services
- The ESP becomes a technology agnostic channel to market for low carbon products
- The ESP works to optimise the supply chain by considering the various constraints and limitations and working within those to maximise customer satisfaction, using various control options
- Acts as the integrator of devices on behalf of consumers to deliver outcomes

The key enterprise architecture priorities for an ESP are:

Performance Parameter	**Considerations**	**Priority / Importance**
Scalability	Consumers will come and go frequently so the ability to scale when necessary (in both directions) will be key to managing costs.	High
Performance	Most ESP operations should not need to be real time so low-latency, high bandwidth should be relatively unnecessary. A centralised form of control which	Mid
Availability	ESPs may miss out on business if they fail to respond to consumers in a timely fashion although this won't always be mission critical.	Mid
Reliability	Most equipment will be in industrial / commercial setting and so component failure will be expected and managed. Availability is more critical so redundancy is important.	Low
Interoperability	An ESP uses the HESG and ERSG specifically to manage the multiple paths and systems. Therefore, the ESP's systems mostly have to be interoperable with only a small number of specific interfaces.	Low

Extensibility	An ESP uses the HESG and ERSG specifically to manage the addition / changes of interfacing systems. As functionality evolves some extensibility will be required.	Mid
Security	ESP control and monitoring are commercially sensitive and therefore, of significant business importance.	High
Manageability	Much of the ICT systems with ESP are likely to be owned by businesses and corporations meaning that this is a lesser consideration than, for example, components in people's homes.	Mid
Maintainability	It is critical that systems are easily maintained and that bugs / problems are easily rectified without having to affect multiple, distributed components	High
Costs	The ESPs will, potentially, interact with huge volumes of data. Scaled and transactional cost minimisation are most critical.	High

Device Vendors

Manufacturers of devices may range from:

1. Suppliers of devices as today, where a user, homeowner or integrator selects the device and installs through;
2. Creators of equipment capable of connecting to other providers digital equipment providing direct insight into usage and performance information to;
3. Smart device manufacturers with their own eco-systems providing access to the collected data and learning obtained.

Each possibility, in the list above, requires different levels of ICT engagement, the most extreme being number 3 which requires comprehensive, sophisticated ICT approaches commensurate with the description of the HESG above.

Consumption Metering

Though largely expected to be in line with the current smart-meter deployment, additional requirements may present themselves following roll-out.

Householders

If current trends continue, consumers will have greater access to digital products, improving choice, access and convenience. This will create opportunities for more engagement in the energy system and, likely an increased expectation for lifestyle improvements. Distributed IoT sensors, AI and machine learning give industry the opportunity to understand individual consumers better, to segment markets efficiently, to tailor products and services to those consumers and to perform price discovery in a way that finds what consumers are willing and able to pay for.

Storers

Storers have an opportunity to store energy when it is plentiful and return it to the system when it is in demand but these actions must be coordinated with the operation of smart appliances, vehicle to grid (V2G) services, aggregator actions and those of balancing and reserves operators to ensure that dynamic changes don't cause grids / networks to become unstable. Such coordination is likely to need significant control and ICT approaches between each of the actors described.

Distributors

Distributors of different vectors, i.e. the owners and operators of the pipes and wires, can avoid costly upgrades in areas which don't yet need them thereby saving consumers unnecessary costs. Dynamic control under different load conditions is one way to achieve that, either directly or through commercial arrangements with other actors (given that in today's architecture, distributors usually don't own a relationship with an end consumer).

Producers

Given the changing demands of consumers and actors involved, in the different vectors of the energy system, and if, as predicted, the vectors become more-tightly coupled, then the behaviours of producers may need to adapt. Today supply follows demand as closely as possible but the possible increase in size of swings (possible peak to trough demand) as operational, commercial or weather changes drive behaviour may require faster, more dynamic, coordinated responses may be needed. This gives rise to the need for enhanced ICT and enterprise level performance facilitated by connected platforms.

Considering the likely shift to more distributed supply, especially with electricity generation, then more interconnectivity, interoperability and coordination may be necessary.

Architectural Implementation Approaches

Hosting & Cloud Computing

Shared cloud computing resources can be bought, on-demand, from a public cloud provider such as Amazon (AWS) or Microsoft (Azure), or a company can build a private cloud that is solely for their own use. The sections below discuss the differences between public and private clouds.

Private

A private cloud is, for the purposes of this document, a set of computing and networking resources that are purchased, provisioned and maintained exclusively for a single business user of the system. The compute resources are physically hosted in their own building (or buildings), which are have access controlled by the same company (or an agency under their control).

Private hosting like this can be a great option for companies who already own and run their own data centres because they can use their current infrastructure.

The main disadvantage of a private cloud, is that all management, maintenance and updating of the data centre infrastructure is the responsibility of the company that runs it. Over time, servers, networking, firewalls, air-conditioning, fire-prevention systems, electrical conduits, wiring and communication will all need to be replaced, which can get very expensive. The other main disadvantage when compared with a public cloud, is that compute, storage and memory costs are fixed and (other than power consumption) independent of usage – so there's no cost saving in scaling down; there is, however, a limit to scaling *up* which is bounded by the amount of hardware in the datacentre

The main advantage that private clouds offer is an increased level of security as they share no resources with any other organisations.

Public

The main differentiator between public and private clouds is that the management of a public cloud is entirely the responsibility of a third party. Data is stored in the cloud provider's data centre, and the cloud provider is responsible for the management, maintenance, backup, redundant storage of the data inside that datacentre (or the backup datacentre if the primary datacentre fails). The advantage of this type of cloud environment that it reduces lead times (effectively to zero) in providing whatever scale of infrastructure is

required, and can scale out practically infinitely to cope with any current or future demand; the public cloud can also scale back down when demand reduces, and because costs are on a usage basis, the overall bill is reduced in line with decreased utilisation.

The major disadvantages of a public cloud are that:

a) security could be lacking, or could be *perceived* to be lacking because physical access to servers is controlled by the cloud provider and not by the service user and;
b) it can be very expensive for a high-load business using a lot of system resources

Multicloud

A multicloud refers to the use of multiple cloud computing platforms provided by multiple suppliers. The advantages of a multicloud approach include:

- Risk spreading – not having to rely on a single supplier, disaster mitigation / recovery, increased flexibility through choice and being able to select the best supplier for each function
- Workload spreading – multiple clouds could spread work evenly, different clouds might handle different aspects of the work, or one cloud could deliver all the functionality whilst another provides back-up

Whilst disadvantages include:

- Increased complexity in security, governance and access
- Resiliency may be compromised due to number of complex interfaces

Comparing Hosting Approaches

For any implementation of the exemplar architecture, security is a key issue, and if the solution utilises public cloud hosting (such as Azure) it has much less control over the hosted physical infrastructure compared to a private hardware and network.

If all other factors are equal, then using a shared public cloud infrastructure increases the likelihood that malicious attackers can infiltrate data and services, and so the easy assumption to make is that it is safer to use private, self-hosted infrastructure. This view should be balanced, however, by considering the amount of ongoing 'housekeeping' that public cloud providers undertake, some of which are technological, but many others of which are contractual, jurisdictional, organisational and security focused; some security and housekeeping measures have very high initial and ongoing costs, and are difficult to justify unless they are amortised at cloud scale across many customers.

Given that public cloud providers host, 24x7, millions of users, running millions of applications in multiple data centres across different geographic regions, they already have robust, tested and scalable security procedures in place; it is likely that the *perceived* increase in security that a private cloud solution provides is illusory, and a public cloud provider would be more secure.

To build, provision and maintain a private cloud is very expensive in terms of capital expenditure for equipment and operating expenditure for staff, premises, power, communications, insurance etc. Accurately predicting the computing requirements of a private cloud is also very difficult, and even in the best case – overcapacity provisioning – it adds to capital and operating costs, and leads to underutilised services and relatively inefficient use of capital.

Combining the security of public platforms with multi-cloud redundancy, is the best available option for hosting the exemplar architecture. Using replicated, multi-cloud platforms provides enormous resilience, scalability and redundancy for the exemplar architecture, but will be more expensive and complex than using a single public cloud approach, and leveraging the years of multibillion investment in always-on, infinitely scalable computing, and the availability of cutting-edge tools and products delivers very good value.

It should be noted that any future digital infrastructure might be considered to be a piece of critical national infrastructure and therefore it is possible that regulatory requirements might dictate the choice of platform hosting.

Implementation Strategy

Implementing ICT systems involves integrating complex hardware and software components. Often there is a need for make 'vs' buy decisions around both elements, which can be facilitated by the implementation strategy.

Monolithic Development

Traditional ICT implementation follows a standard simple engineering development approach of defining the problem, designing, implementing and testing. Said designs are often monolithic and driven top down, but are also self-contained and focused on the problem at hand.

For self-contained, rarely changing applications, this approach can reduce development costs.

Service Orientated Approach

Similar to traditional development in that SOA are often top-down, the key differentiator is that this development approach concentrates on breaking components into discrete services (or micro-services) which are simple and perform a single, specific function, with the idea that reusability is the goal. These services (hardware and/or software) are then created and integrated together to perform the system objectives.

Generally, an SOA system uses an application server or another form of infrastructure component around which the services are developed. This provides structure and defined interfaces but can represent a significant initial outlay.

Enterprise Service Bus

An ESB approach also focuses on building out specific functions but centres around an ESB and translator functions which can take any developed service and link into the integrated application. This is a bottom-up approach which seeks to accept any developed software component / service. In that sense an ESB approach prevents vendor lock-in or constraining architectural decisions.

The ESB is, in essence a piece of middleware which integrates data and communications amongst multiple systems without those applications having to worry about compatibility.

Comparing Implementation Approaches

For self-contained, rarely changing applications, monolithic approaches can reduce complexity and thereby development costs.

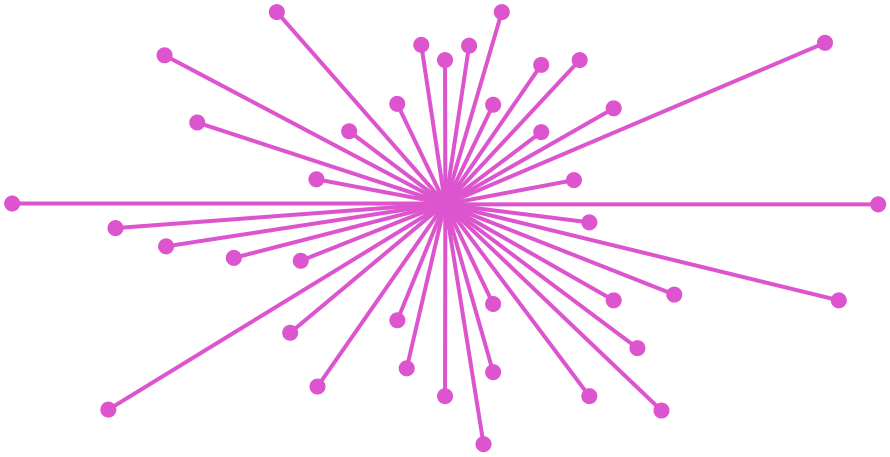
From a re-use point of view an SOA development allows newly developed items to be re-used whereas an ESB requires a highly flexible central integration point with custom translators being developed to move from existing implementations of hardware and software.

It's important to consider the availability of suitable applications prior to settling on the ESB approach since its value is dependent on reusing already created components.

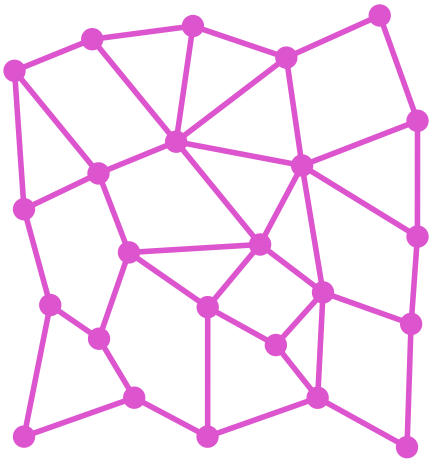
Control Architecture Strategy

In this sense, control architectures refer to the methods of sending signals to change the behaviour of one or multiple aspects of the system. Smart grids are highly sophisticated and tightly integrated control environments but they are made up of many independent systems driven to maximise each individual actors' business models. This section describes the methods of interconnection between already complex systems.

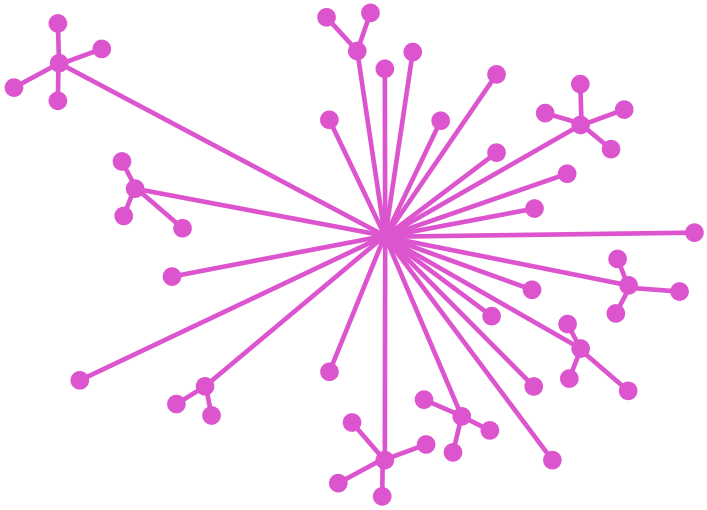
Control is often grouped into three primary types:



Centralised: A single node makes decisions, with all possible knowledge of the current system state(s) and sends control signals to outlying nodes.



Distributed: Both decisions and information are shared across nodes but the decisions may still require complete systems knowledge to understand.



Decentralised: No single node has all the information; every node decides its own behaviour and the resulting system performance is the aggregated behaviour.

Comparing Control Strategy Approaches

In ICT architectures all types are of approach are common and come with relative advantages and disadvantages.

	Advantages	Disadvantages
Centralised	System Actions are Co-ordinated Can be cheaper (Single server with many cheap clients, for example)	Slowest (highest latency) Threats from interception of commands Risk of single point failures
Distributed	More scalable Some co-ordination between nodes	Complex Threat from failures in adjoining nodes.
Decentralised	Easiest to scale Resilient to failure Each node can be simpler	Most complex Each node is less capable

Open versus Closed Systems

The Data and Communications Company exists to connect the smart meter infrastructure in the UK to the business systems of energy companies through a national secure communications network. Access to the network is restricted to those who follow the terms and conditions the DCC, therefore, whilst it is open to many users, the network is considered closed. There are advantages to closed systems such as:

- Auditable data is trusted from end-to-end and forms the basis for billing millions of transactions.
- Security is much more easily controlled within a smaller community of authorised users.
- Has guaranteed performance characteristics (e.g. coverage, up-time etc.) rather than traditional internet provision.

Some disadvantages include:

- Compliance with the rules and procedures comes with an overhead.
- Each DCC transaction has a cost to pay.
- At some point systems have to co-operate and closed systems run the risk of being included in something bigger.

Fully open systems, on the other hand, are:

- Cheaper (or free) per transaction.
- Increased pace of innovation as more developers support progress.
- Reduced costs to a single company as maintenance and upgrades can be shared.

But the downsides of open systems are that they may also be:

- Open to unauthorised access and/or unauthorised use.
- Leads to bigger design trade-offs to maximise usability for broader set of users.
- Each actors' business faces threats from others over market share with a reduced barrier to compete.

Communications

There a number of choices for delivery method for communicating between various pieces of equipment and between actors within an energy system.

Fixed Line: Generally referring to wired connections joining two or more parties together. Generally expensive to install fixed line networks have the advantage of being reliable and robust.

Power-Line Comms: This requires passing data down the power line by super-imposing signals on to the power line, often using a carrier wave. The advantage is that it is available wherever there is power. The major disadvantage is that it is usually desirable to stop the propagation of the signal across boundaries (e.g. building to distribution network) for many reasons including nuisance noise for other users, security and as a result of transformers degrading signal to noise ratios beyond usefulness. Finally, the physical construction of power lines has been optimised for 50-60Hz which generally limits the ability for high-frequency propagation and therefore the bandwidth available.

Mobile/GPRS: Despite the popularity and penetration of mobile communications over the past few decades coverage is still not universal and consistent across the country meaning that mobile solutions often need to be supplemented with additional technologies in remote areas.

DCC: The Data Communications Company (DCC) has the ambition to cover the UK in an independent, secure digital network, initially to support the operation of smart meters but with the potential to be used for additional services.

Routing Approaches

With devices some important hierarchical decisions will need to be made. Figure 2 below illustrates an example of a choice that needs to be made. There are two approaches to trust domains shown. In (a) a demand side response (DSR) control signal would be sent from an originating source, let's say a distribution network operator (DNO) to the components in its sphere of control. In the second example the same originator would send a signal to the properties in its sphere of influence (b) who in-turn would control the device level (c).

The first approach is quicker, simpler and requires less complexity at the domestic property level but requires visibility of components inside a person's home. The second approach requires more complexity in the domestic arena but needs to know less of the about the consumer, an enhancement to their privacy.

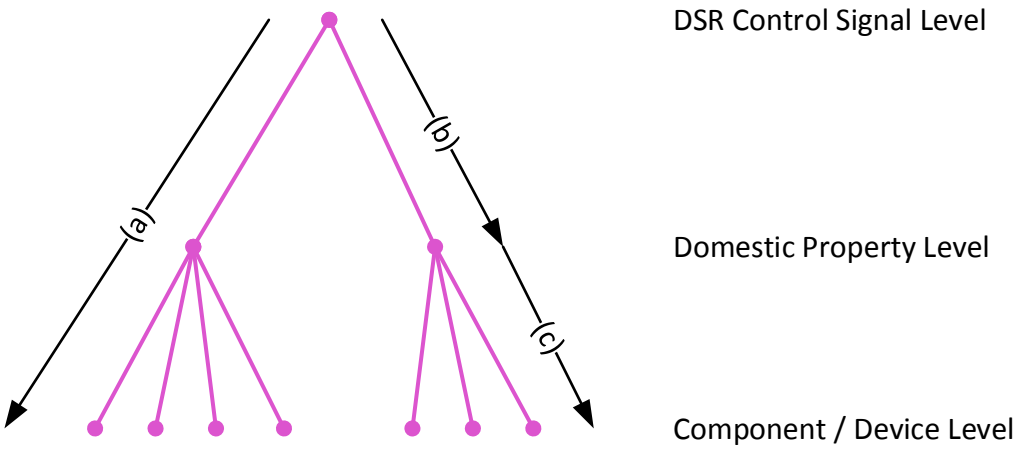


Figure 2 - Trust Domains

Trust is vital for all forms of identification, as evidenced from the UKs attempt to bring in identity cards. A report (Accenture, 2018) defined the four keys to digital trust as security, privacy, benefit/value and accountability. Countries where trust in government is high have found it easier to implement SDI schemes versus those with lower levels (The Guardian, 2017).

Conclusion

This paper provides a generalists overview of the challenges facing actors within the exemplar architecture described.

This paper articulates that different actors and their ICT systems have different needs and business priorities and it is these that will shape their choices of systems as they see fit. There is no one right answer for all systems.

Architectural robustness means that a joined up, whole-systems approach is required to develop enabling platforms since reliance on ICT principles of security will not be sufficient alone.

No single commercial business is motivated to deal with emergent properties of collective action such as cyber security threats.

Different architectures create different stability and security risks so government forethought, from a whole systems perspective, is key to making good choices.

Key elements of the exemplar architecture would, ideally be developed individually but possibly concurrently and with the requirements of the interfaces firmly in mind. In all likelihood such an approach is likely to require multiple iterations and trials at scale to mature.

The principles and approaches described in this paper could form the basis for a collaborative exploration with stakeholders. Comprehensive modelling of different ICT approaches and the business models behind their operation would likely yield insights for the actors within the energy system. Followed up with a practical demonstration could provide the basis of a transition to new ways of prospering from the energy revolution. Energy Systems Catapult looks forward to working with interested innovators in pursuit of these objectives.

Bibliography

Accenture. (2018, July 13). *Latest Thinking*. Retrieved from DIGITAL TRUST IN THE INTERNET OF THINGS ERA: https://www.accenture.com/us-en/insight-digital-trust

Amdahl, G. M. (1967). Validity of the single processor approach to achieving large scale computing capabilities. *AFIPS spring joint computer conference.* California.

Centre for Internet Security . (2018, June 27). *CIS Controls*. Retrieved from Center For Internet Security: https://www.cisecurity.org/controls/

Cointelegraph. (n.d.). *Bitcoin Mining Uses More Power Than Most African Countries.* Retrieved from https://cointelegraph.com/news/bitcoin-mining-uses-more-power-than-most-african-countries

Docker. (n.d.). *What is a container.* Retrieved from https://www.docker.com/what-container

Electricchain. (n.d.). *ElectriCChain* . Retrieved from http://www.electricchain.org/

Energy Systems Catapult. (2017). *Energy Systems Architecture Methodology: Enabling Multi-vector Market Design.* Birmingham, UK: ESC.

Energy Systems Catapult. (2018). *Enabling Mass Market Deployment of Hybrid Heat Pumps.* Birmingham, UK: ESC.

Energy Technologies Institue. (2017). *Tools for Future Energy Systems.* Loughborough, UK: ETI.

HCash. (n.d.). *HCash.* Retrieved from https://h.cash/

HCLTech. (n.d.). Retrieved from https://www.hcltech.com/blogs/everything-you-need-know-about-enterprise-service-bus-esb

http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.165.2908&rep=rep1&type=pdf. (n.d.).

Hyperledger. (n.d.). *Hyperledger blockchain framework.* Retrieved from https://www.hyperledger.org/

InfoQ. (2018, July 13). *A Distributed Access Control Architecture for Cloud Computing*. Retrieved from InfoQ: https://www.infoq.com/articles/distributed-access-control-architecture-for-cloud-computing

IOTA. (n.d.). *IOTA* . Retrieved from https://iota.org/

Medium. (n.d.). *What are the possibilities for DLTs like Blockchain, Tangle and other related technologies in the electric mobility infrastructure?* Retrieved from https://medium.com/@harmvandenbrink/what-are-the-possibilities-for-dlts-like-blockchain-tangle-and-other-related-technologies-in-the-40c8f9f90890

Microsoft. (n.d.). Retrieved from https://azure.microsoft.com/en-gb/services/sphere/

Microsoft. (n.d.). *Microservices.* Retrieved from https://docs.microsoft.com/en-us/azure/architecture/microservices/index

Microsoft. (n.d.). *Queue based load levelling.* Retrieved from https://docs.microsoft.com/en-us/azure/architecture/patterns/queue-based-load-leveling

Microsoft. (n.d.). *Service fabric overview.* Retrieved from https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-overview-microservices

Micrososft. (n.d.). *Circuit breaker pattern.* Retrieved from https://docs.microsoft.com/en-us/azure/architecture/patterns/circuit-breaker

MuleSoft. (n.d.). Retrieved from https://www.mulesoft.com/resources/esb/enterprise-application-integration-eai-and-esb

Mulesoft. (n.d.). *Microservices best practices.* Retrieved from https://www.mulesoft.com/ty/wp/best-practices-microservices

Mulesoft. (n.d.). *Microservices vs. ESB.* Retrieved from https://blogs.mulesoft.com/dev/microservices-dev/microservices-versus-esb/

Mulesoft. (n.d.). *What is an ESB.* Retrieved from https://www.mulesoft.com/resources/esb/what-esb

OpenSourceForU.com. (n.d.). Retrieved from https://opensourceforu.com/2016/07/getting-started-with-docker-swarm/

Solarcoin. (n.d.). *Solarcoin.* Retrieved from https://solarcoin.org/en/node/6

Solcrypto. (n.d.). *Solcrypto.* Retrieved from https://www.solcrypto.com/

Tangleblog. (n.d.). *The Tangler.* Retrieved from http://www.tangleblog.com/what-is-iota-what-is-the-tangle/

Techtagret.com. (n.d.). Retrieved from https://searchcio.techtarget.com/definition/distributed-ledger

The Guardian. (2017, March 30). *The Guardian*. Retrieved from Driving Licences could be on phones by 2018: https://www.theguardian.com/money/2017/mar/30/driving-licences-could-be-on-phones-by-2018

Wikipedia. (n.d.). Retrieved from https://en.wikipedia.org/wiki/Directed_acyclic_graph

CATAPULT
Energy Systems

Energy Systems Catapult
7th Floor, Cannon House
18 Priory Queensway
Birmingham
B4 6BS

es.catapult.org.uk
0121 203 3700