Blockchain Playbook for the U.S. Federal Government

Emerging Technology Community of Interest

Blockchain Working Group

Initial Release: April 2, 2018
Updated: June 5, 2019

Synopsis

This Playbook comes right after the Primer[1] and proposes a process and a series of phases to support the United States Federal Government in its understanding and application of blockchain and distributed ledger technologies for its mission. Each phase contains a set of key activities organized in functional areas that go beyond just the technical aspects of blockchain but include management, people, process, and acquisition areas.

Blockchain has the potential to help government mitigate fraud, reduce errors, and lower the cost of paper-intensive processes, while enabling collaboration across multiple divisions and agencies to provide more effective and efficient services to citizens. Moreover, the adoption of blockchain may also allow governmental agencies to provide new value-added services to businesses and others which can generate new sources of revenue for these agencies.

This page is intentionally blank

American Council for Technology-Industry Advisory Council (ACT-IAC)

The American Council for Technology (ACT) is a nonprofit educational organization established to create a more effective and innovative government. ACT-IAC provides a unique, objective, and trusted forum where government and industry executives are working together to improve public services and agency operations through the use of technology. ACT-IAC contributes to better communications between government and industry, collaborative and innovative problem solving, and a more professional and qualified workforce.

The information, conclusions, and recommendations contained in this publication were produced by volunteers from government and industry who share the ACT-IAC vision of a more effective and innovative government. ACT-IAC volunteers represent a wide diversity of organizations (public and private) and functions. These volunteers use the ACT-IAC collaborative process, refined over thirty years of experience, to produce outcomes that are consensus based. The findings and recommendations contained in this report are based on consensus and do not represent the views of any particular individual or organization.

To maintain the objectivity and integrity of its collaborative process, ACT-IAC does not accept government funding.

ACT-IAC welcomes the participation of all public and private organizations committed to improving the delivery of public services through the effective and efficient use of IT. For additional information, visit the ACT-IAC website at www.actiac.org.

Emerging Technology Community of Interest

ACT-IAC, through the Emerging Technology Community of Interest, formed a Blockchain Working Group to give voice to and provide an authoritative resource for government agencies looking to understand and incorporate blockchain technology and functionality into their organizations. This working group includes government and industry thought leaders incubating government blockchain use cases. The ACT-IAC Emerging Technology Community of Interest (ET COI) mission is to provide an energetic, collaborative consortium comprised of leading practitioners in data science, technology, and research, engaged with industry, academia, and public officials and executives focused on emerging and leading technologies which transform public sector capabilities.

Disclaimer

This document has been prepared to contribute to a more effective, efficient, and innovative government. The information contained in this report is the result of a collaborative process in which a number of individuals participated. This document does not – nor is it intended to – endorse or recommend any specific technology, product, or vendor. Moreover, the views expressed in this document do not necessarily represent the official views of the individuals and organizations that participated in its development. Every effort has been made to present accurate and reliable information in this report. However, ACT-IAC assumes no responsibility for consequences resulting from the use of the information herein.

Copyright

Further Information

For further information, contact the American Council for Technology-Industry Advisory Council at (703) 208-4800 or www.actiac.org.

Table of Contents

Introduction

As noted in the ACT-IAC Blockchain Primer[1], blockchain may be applied to help government to reduce fraud, errors, and the cost of paper-intensive processes, while enabling collaboration across multiple divisions and agencies to provide more efficient and effective services to citizens. The adoption of blockchain may enable government agencies to provide new value-added services and modernize IT. How can agencies turn that potential into reality?

- **Understand the technology using the Blockchain Primer:** Over one dozen federal agencies and a variety of industry partners collaborated to develop the ACT-IAC Blockchain Primer which provides the government workforce with an introduction to blockchain and its related technologies, as well as its many potential use cases.

- **Incorporate blockchain functionality using the Blockchain Playbook:** The ACT-IAC Blockchain Working Group developed this playbook to guide the government workforce in taking the appropriate steps and developing the necessary plans to implement the right technology to achieve the goals of their mission.

- **Blockchain and decentralized organizations:** Blockchain has the potential to significantly impact both business processes as well as the fabric of the organization. The usual linear value chain, where value is added in strict sequential order, is being replaced by networked value chain where entities and the entire environment are networked together with automated code (e.g. smart contracts). In a networked value chain enabled by blockchain, there is a more efficient use of resources and process execution, which leads to cost savings and reductions in cycle times. In terms of the organizational structure, classical hierarchical layers may be replaced by a new model emerging from implementations of blockchain and other digital transformation technologies – decentralized organization. These decentralized organization leverages intelligent and distributed nodes that are empowered to execute various processes without human intervention or central oversight.

This playbook applies the concepts of the General Services Administration's Modernization and Migration Management (M3) unified shared services framework to help the government achieve successful outcomes and reduce risk during a blockchain deployment. It involves the modernization of information systems, as well as the migration of data and/or other capabilities. The progression of this framework ensures the government will be able to optimize its resources to deliver the most effective solution.

Users may leverage this playbook during each iteration of a blockchain solution's implementation: minimally viable product, proof of concept, pilot/limited fielding, initial operational capability, full operational capability, etc. Not all topics in each phase may apply to all iterations. However, this playbook should remain useful as a solution moves through its lifecycle.

It is also important to note that at scale and fully implemented, a distributed ledger technology will probably not be a party of one or even two – more like a party of 5, 25, 100, 1,000. A blockchain solution is not like any other organizational technology solution because among other things no one organization alone will be in control of the final product (if it's implemented as intended). Therefore, it is important that at the onset, any organization interested in leveraging blockchain will need to define the appropriate stakeholders and the group (network peers) that will participate in the steps outlined in the playbook.

To address the current high level of government interest and desire to begin deploying blockchain solutions, launch of this playbook occurs in two stages. The next stage will begin in April 2018 to continue supporting these efforts as they evolve.

As government efforts move through implementation of this new and rapidly developing technology, contributions to this playbook (e.g. additional best practices, lessons learned, and other information) are appreciated to ensure this resource is current, comprehensive, and effective in meeting the needs of government.

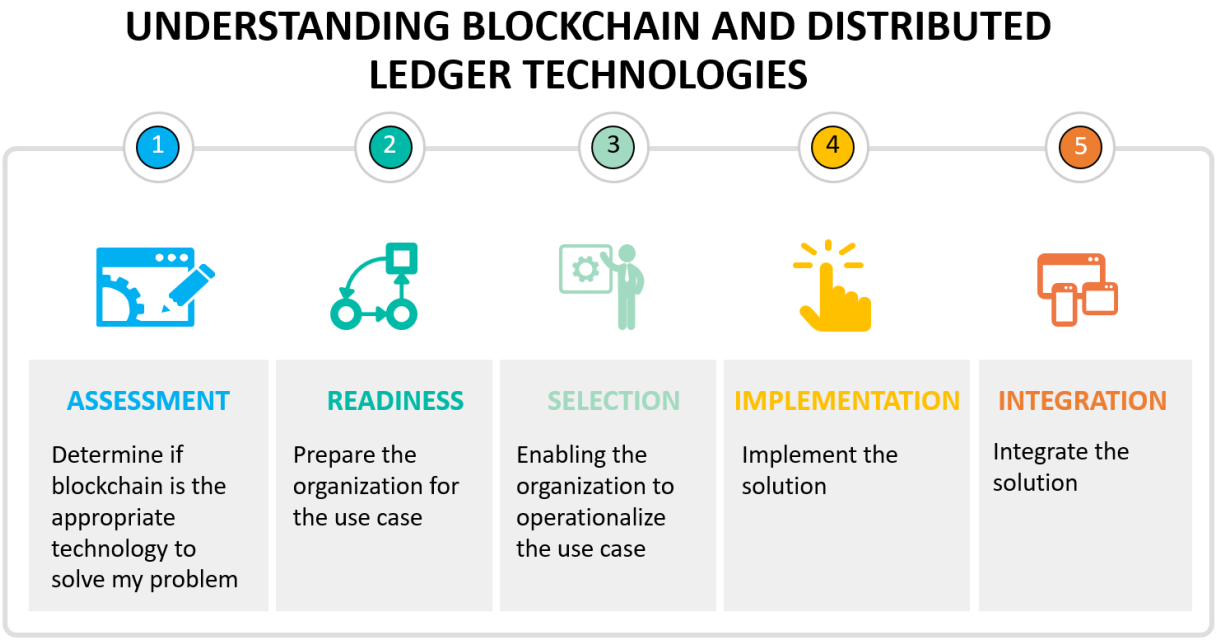

Figure 1: Blockchain Playbook phases

The first four phases of a five-phased approach, which includes key activities and outcomes for each phase, are addressed in this edition of the playbook.

Phase 1 Problem Assessment: Develop a vision and business objectives through various assessments to ensure the blockchain solution addresses a specific use case and delivers results that optimize services and operational delivery.

Phase 2 Organizational Readiness: Engage blockchain subject matter experts and consider the nuances that accompany a blockchain solution to prepare the organization(s). This includes creating a project management office, as well as the establishment of blockchain-tailored business, functional and technical requirements, and implementation plans.

Phase 3 Technology Selection: Conduct a thorough investigation of the business considerations (e.g. consensus mechanism, transaction costs, and on chain/off-chain data requirements), types of blockchains, digital asset and distributed ledger technology (DLT) requirements and considerations, deployment models, and procurement options to enable optimal provider selection to achieve the desired end state.

Phase 4 Blockchain Implementation: Do the implementation, customization and configuration of the blockchain solution.

Phase 5 blockchain Integration: Integrate the blockchain solution into the organization(s)' infrastructure.

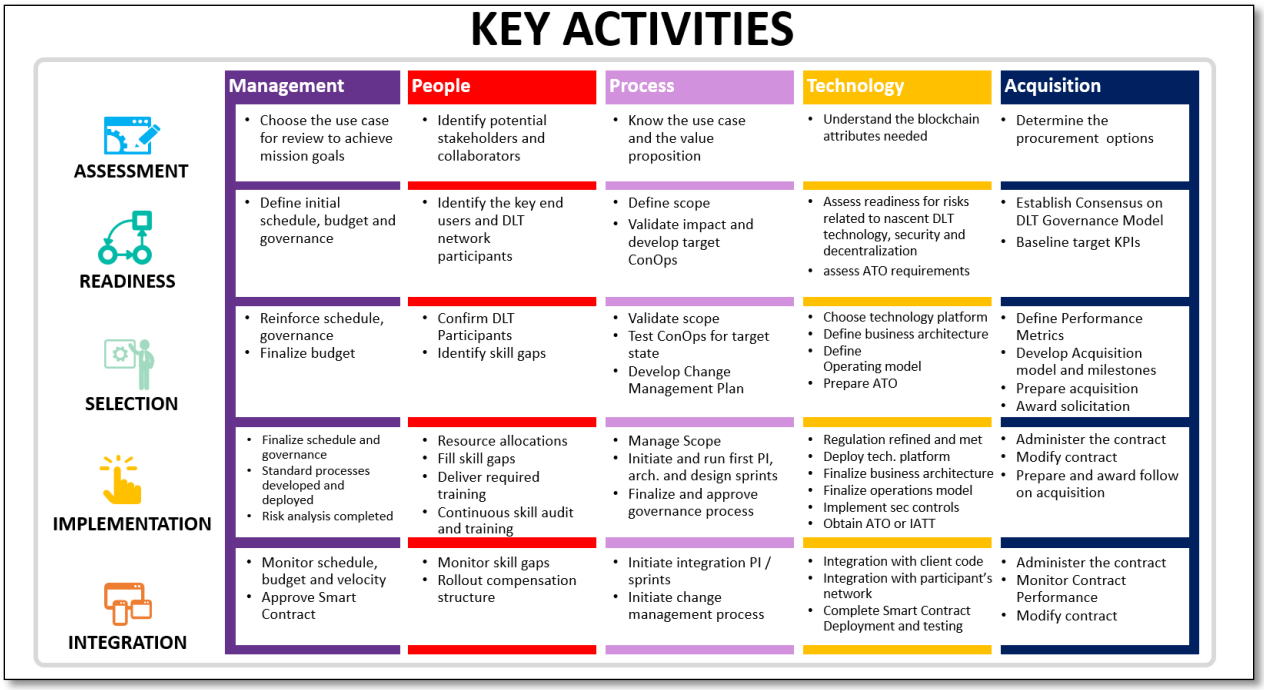

Figure 2: Blockchain Playbook phases and key activities matrix

Also included at the end of this playbook is a glossary to assist with in-depth terminology associated with advanced blockchain concepts, and appendices to elaborate on each selection topic at a more granular level.

Phase 1 – Problem Assessment

This section helps decision makers create the most value through their blockchain initiative. It includes tools to ensure the initiative is designed for addressing a specific use case and advancing toward mission goals, even if that solution is not blockchain.

Phase Inputs

When it comes to emerging technologies, business decision makers are faced with making a choice: keep the status quo or evolve the organization's culture to be able to leverage these new solutions. It is a challenge that goes beyond the definition of these technologies; it includes analysis of "*How does this technology change and improve my business*?"

Today, most businesses operate in a hierarchal manner. Models have hardly changed – adapted or evolved – for the last two decades. Digitalization has created the opportunity to morph business models from linear to dynamic, asymmetric and three-dimensional. However, people and culture have struggled to keep pace with these advances.

Blockchain is the catalyst for tectonic shift in the accepted business model: decentralized organizations. Successful organizations and leaders are those who are able to transcend the status quo and master this reality, combining a focus on the organization's core competency with optimizing operational expenses (OPEX). Executives need to prepare their leaders, associates, and culture for such adoption.

To fully assess opportunities in blockchain, engage with stakeholders to refine the use case the blockchain initiative will address. Identify and document the context of the use case. This includes the business challenge; business process issues, gaps, and/or frictions that contribute to that challenge; and stakeholder needs. Detail the stakeholders' functional requirements. Also, determine their perceived risks related to implementing a solution along with their expected outcomes and the associated metrics.

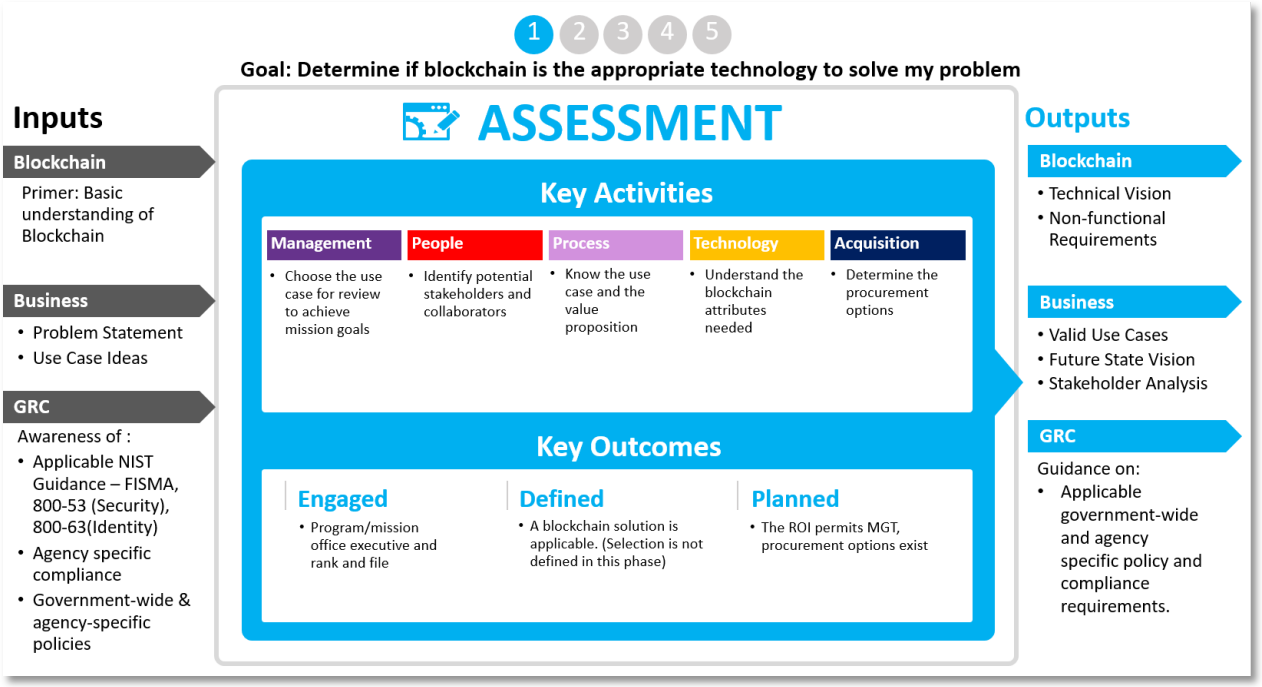

Figure 3: assessment phase (1) summary

"Do I need a blockchain?"

When considering a blockchain solution, evaluate both the associated potential benefits and limitations, such as whether a blockchain solution will remove some of the existing business frictions associated with the current business process. Determine if characteristics, such as trust, immutability, and finality, provide significant value for the use case being evaluated. Assess whether those benefits result in a reduction in cost and/or risk, or if they will achieve process efficiencies. Analyze if there are additional benefits that can be realized by growing the business network associated with the solution.

As the industry considers the new model of decentralized organizations, consider whether executives are ready for the massive shift to the automatic execution of activities without human intervention. Also determine the qualifying factors of decentralized organizations, such as the rules and regulations that need to be adhered to and how the blockchain solution may integrate with existing systems.

The following are key questions to consider when evaluating a blockchain solution (note that discussion regarding public versus permissioned blockchain is provided in Phase Three: Technology Selection of this playbook.) Review and answer each question. Add or subtract the points associated with each question to determine your total score. Point totals will provide insight into the possibility you may get a substantial Return on Investment (ROI) from a blockchain approach.

**Note: This is a notional table and the level of importance associated with each question may be tied to the use case being assessed. Assign Points based on the Attribute Importance Rank (with suggested weighting). You may adjust the weight of questions as they apply to your use case. (5 – critical, 4 – very high, 3 – high, 2 – moderate, 1 – slightly, 0 – not at all)

1. **Will the use case involve a business network which spans multiple organizations/agencies?** While a single organization may be used for an initial proof of concept or pilot, the varying degrees of trust among multiple organizations/agencies lead to a stronger long-term case for blockchain. These organizations will look to sharing a common asset or requiring consensus.

0 (Not at all)	1 (Slightly)	2 (Moderate)	3 (High)	4 (Very High)	5 (Critical)

2. **Is there a current lack of trust among the business network participants and/or sources of data?** Sources may include Internet of Things (IoT) devices, legacy systems, service providers, and users from multiple agencies. Note: Blockchain supports data integrity but does not have inherent capabilities to validate data quality.

0 (Not at all)	1 (Slightly)	2 (Moderate)	3 (High)	4 (Very High)	5 (Critical)

3. **Would the organizations in your use case benefit from a shared governance and data standards approach?** This is specific to the potential need to create a custodian [owner] of membership for the actors and organizations within your process.

0 (Not at all)	1 (Slightly)	2 (Moderate)	3 (High)	4 (Very High)	5 (Critical)

4. **Is this a use case that can be more efficiently solved with other technologies (e.g. distributed database)?** Do those technologies provide the same benefits that a blockchain solution will provide for this use case and would this solution be applicable for all parties?

0 (Not at all)	-1 (Slightly)	-2 (Moderate)	-3 (High)	-4 (Very High)	-5 (Critical)

5. **Does the use case require or can it benefit from strict transaction immutability?** Ledger transaction entries are append-only. Transaction records cannot be altered (even by the administrators).

0 (Not at all)	1 (Slightly)	2 (Moderate)	3 (High)	4 (Very High)	5 (Critical)

6. **Does the use case require or can it benefit from using distributed ledgers and a decentralized authority approach?**

0 (Not at all)	1 (Slightly)	2 (Moderate)	3 (High)	4 (Very High)	5 (Critical)

7. **Can your organization benefit from transforming respective business capabilities into a decentralized organization?**

0 (Not at all)	1 (Slightly)	2 (Moderate)	3 (High)	4 (Very High)	5 (Critical)

8. **If you are able to repurpose business logic to a distributed organization – have you identified how your Core Competencies will benefit?** Consider the full business considerations of a blockchain approach that go beyond the technologies involved and may impact organization, business processes, workflow, customer interaction, and other factors.

0 (Not at all)	1 (Slightly)	2 (Moderate)	3 (High)	4 (Very High)	5 (Critical)

9. **Are there existing inter-organization business process inefficiencies** (e.g. an excessive amount of time being spent on reconciliation)?

0 (Not at all)	1 (Slightly)	2 (Moderate)	3 (High)	4 (Very High)	5 (Critical)

10. **Are you looking for a vehicle to securely share reference data among members of the business network?**

0 (Not at all)	1 (Slightly)	2 (Moderate)	3 (High)	4 (Very High)	5 (Critical)

11. **Does provenance of a digitized asset – a record tracking the lifetime history of an asset – as it is controlled/owned by different members of the business network apply** (e.g. moving from factory to distribution center to final destination across the life cycle)?

0 (Not at all)	1 (Slightly)	2 (Moderate)	3 (High)	4 (Very High)	5 (Critical)

12. **For this use case, is there an existing system that could serve as a trusted source of the truth for all parties?** Would that system be accessible by all parties?

0 (Not at all)	-1 (Slightly)	-2 (Moderate)	-3 (High)	-4 (Very High)	-5 (Critical)

13. **Does the use case have high performance requirements? Transactions per second (TPS) > 3K/sec?** Note: high performance/near real-time requirements are typically not yet met by blockchain solutions, but platforms and performance numbers will continue to evolve.

0 (Not at all)	-1 (Slightly)	-2 (Moderate)	-3 (High)	-4 (Very High)	-5 (Critical)

Results:

Provided here are common questions and relative weights for answers to serve as a preliminary guide for those considering a blockchain approach. While useful, this is still only a guide for consideration and further investigation. Sound engineering analysis and practices should still prevail.

Score Groupings:
In order to assess the applicability of a blockchain approach, scores are grouped to guide the reader to where taking a blockchain approach would be most beneficial (highest score) and where it is less likely (may still be applicable but needs additional scrutiny).

If your score is 20 or below: A score of 20 or below typically represents a small ROI and limited applicability from a blockchain approach. Consider that while the score may be low, your situation may still warrant deeper analysis as there can be a compelling reason to continue with a blockchain approach that did not fall into the standard categorization.

If your score is between 21 and 40: A score of between 21 and 40 could typically be supported with a blockchain approach but is not an overwhelming natural candidate. These situations can have powerful reasons that can still drive a blockchain approach, yet they might also have mitigating factors that make a traditional approach a better alternative. In these situations, a more thorough analysis is typically needed.

If your score is 41 or higher: A score above 41 typically represents a compelling ROI and strong applicability that would benefit significantly from a blockchain approach. It is strongly recommended to consider the costs and benefits of a blockchain approach in these instances while still considering other additive and mitigating factors in the organization, strategic direction, interdependencies, and related items.

Practical Advice and Best Practices for Blockchain Assessment

With blockchain near the top of the hype cycle in 2017[2], it is important to separate reality from hype when it comes to which uses cases can actually benefit from a blockchain solution. Consider the following advice and best practices when evaluating blockchain for any use case.

An essential resource to any organization or individual considering blockchain as a possibility, the GSA Emerging Citizen Technology Atlas[3] provides a clear snapshot into potential use cases and programs

Start Small - Minimal Viable Product(MVP)/Prototype

Before addressing how to introduce this new technology into your ecosystem, define the scope of an MVP/proof of concept that demonstrates blockchain as a viable solution for your use case. Do this while still considering the future state and stakeholder incentives. The MVP should help prove the solution's expected outcomes (e.g. decreased reconciliation costs) associated with the MVP hypothesis.

Business Capabilities and Blockchain Capabilities

Consider mapping your business capabilities to your blockchain capabilities. Below is an example of a General Services Administration (GSA) vehicle fleet model. The stakeholders need to track mileage of vehicles usage across various agencies, users, and states. This alignment serves two purposes: 1) Stakeholders are aware of which business capabilities they plan on transforming, and 2) Blockchain [capabilities] are validated as the correct emerging technology for the business solution.

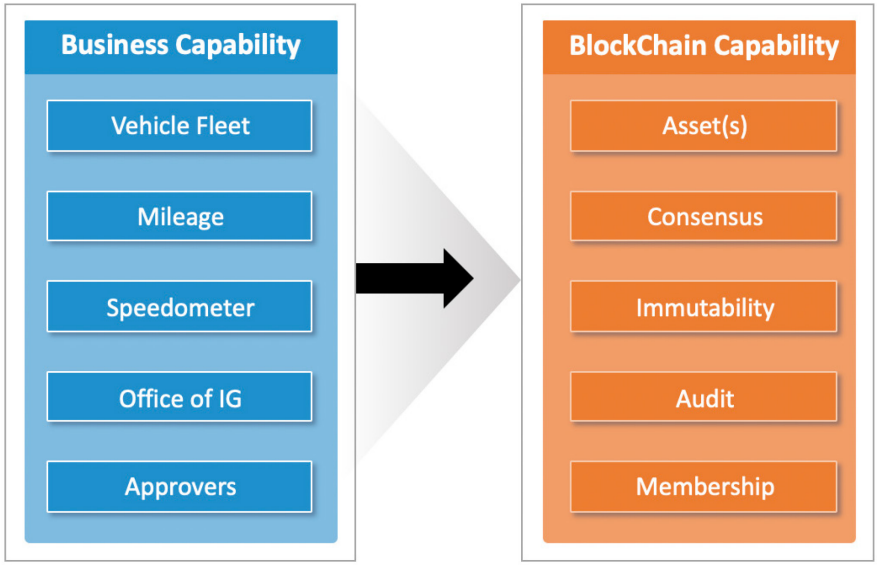

Figure 4: alignment - business-to-business blockchain

Build Blockchain Architectural Blueprint for Future Phases

Develop a vision and a plan for the additional requirements and challenges that will need to be addressed if your solution moves into a pilot phase and subsequent operational phases. This should encompass modernization and integration with legacy systems. The project will also include major change management components both from an operational and cultural perspective.

The organization should examine the desired technologies and subsequent capabilities that can be enabled by the future state blockchain solution. For example, an operational blockchain logging security events may be combined with artificial intelligence as part of a new capability that is a predictive cybersecurity force multiplier. Or a blockchain for supply chain may become the entry point to a property management workflow solution.

Building a working blueprint of the technical architecture will provide a powerful tool for defining the scope and phases of the comprehensive blockchain implementation. Strategic scaling will enable you to optimally address pain points and align stakeholders while tackling one priority area at a time to ultimately accomplish transformational objectives and advance mission goals.

Emphasize ROI and Benefits to the Entire Network

Emphasize ROI while making an assessment. Examine the solution's common costs, benefits, and efficiencies for both the network as a whole and for individual members. Include design thinking based on personas and a prioritization matrix around value versus complexity. An MVP should prove the viability of a blockchain solution, with ROI measured by gains in savings and in efficiencies, reduction of risk, and the meeting of mission goals. ROI considerations should include:

- **Gains in efficiency and cost savings** – Learn how much effort and cost are currently spent on reconciliation to determine how the trust provided by a blockchain solution, when exchanging data or assets, impacts ROI.
- **Incremental gains** – Implementation should be done in small increments, keeping to a true agile methodology. This is not a lift and replace but a gradual shift to a strategically-assured, positive ROI.
- **Cloud first and shared services** – Blockchain relies on distributed computing concepts where nodes may reside in the cloud. Blockchain smart contracts can enable vendor provided shared services to demonstrate adherence to current regulations and policies while providing an immutable audit trail.
- **Reducing risk** – Understand the ROI that blockchain can provide by reducing risk. A blockchain solution can help reduce the risk associated with tampering and Denial of Service (DoS) attacks due to its decentralized, tamper-resistant attributes.
- **Other** – How does the trust provided by a blockchain solution when exchanging data or assets impact ROI? How much effort and cost are currently spent on reconciliation?

Impact of Modernizing Government Technology (MGT) Act

The MGT Act[4] (incorporated as Subtitle G to Title X of the National Defense Authorization Act for Fiscal Year 2018) encourages agencies to take a refined strategic and incremental approach to IT with the goals of driving gains of efficiency and cost savings, as well as lowering risk. If an agency has targeted an IT process to modernize, but does not have the budget to do so, adoption of new technology, such as blockchain, can be used to enhance the existing IT solution with new capabilities and functionality.

This could be applied as part of a strategy enabled by the act to use working capital funds to spur IT modernization. By choosing a smaller project or projects that will realize savings in year one, the agency can apply those savings (which can be banked for up to three years) to the larger, more critical project in year two.

Since it is centered on multiagency business networks and processes, adoption of blockchain technology is consistent with the MGT Act's approach to modernization. A strategic blockchain solution is extensible – able to grow over time, adding new members and processes to the business network. It also facilitates data sharing and the reduction of existing frictions and inefficiencies. Therefore, the assessment that an agency must make through a standard analysis of functions and gaps is to determine if the use of a blockchain will enhance an existing solution through gains of efficiency, cost savings, and lowered risk.

Incorporate Regulations/Mandates

When assessing blockchain or any blockchain technology, be sure to evaluate it within the context your particular use case. For example, an identity use case would require a technology that adheres to particular regulations, such as the technical requirements in the NIST 800-63, Digital Identity Suite[5], Digital Identity Guidelines. Be sure to evaluate the entire technology solution—including smart contracts, oracles, side chains, micro services, any associated cryptocurrencies, etc.—with regard to industry standards, technical standards, regulations, acts, and common law.

Determine Throughput and Latency Requirements of Business Processes

Understanding data requirements and the speed by which data transactions must occur within the blockchain ecosystem are key inputs in determining your optimal architecture, accelerators, etc. When assessing blockchain as a solution, consider the number of TPS required, the size of the data/digital assets being transferred, if the data is highly transactional, and whether the data should be stored on or off the blockchain. Determine where the nodes will be located. Assess if the network bandwidth is constricted and the resulting implications. Large blockchain ecosystems with many systems and players such as public permissionless chains would be best served by minimizing data-load, whereas smaller private/consortium chains may handle a larger number of TPS and can theoretically handle the exchange of larger "blocks" without interruption to business processes.

Assess the Current State of the Customer Experience

As with most transformative efforts, the blockchain initiative should also be leveraged as an opportunity to enhance user experience. When assessing your use case, different users with different roles may interact with the proposed solution. Users have little tolerance for solutions or systems that lack user-friendliness, intuitiveness, and/or accessibility. Involve them early and apply their feedback while assessing how blockchain could elevate those user experiences by automating tasks and improving visibility. Robust usability testing will be key to maintaining the momentum of successful user adoption as the blockchain solution is scaled outward.

Key Outcomes

After the assessment is completed, one or more deliverables/outcomes should be produced:

- One or more use cases identified as candidates for a blockchain solution. Each use case should be evaluated against assessment criteria to determine if a blockchain can be used to realize gains such as cost saving, efficiency, or reduced risk. Which of these use cases would be sensible to implement as a minimum viable product?
- Documentation on the key stakeholders, network participants and their roles for your candidate use case. How will a blockchain solution impact your stakeholders? What are the incentives for these stakeholders to participate in the solution?
- For each candidate use case, identify what business frictions you are looking to address by implementing the solution. Do blockchain attributes help to address those frictions? This list should correlate with the list of anticipated benefits provided by the solution. Both the frictions and benefits may differ for each stakeholder.
- List of federal regulations, government-wide and agency policies, Federal Information Security Management Act (FISMA) and other required compliance regarding adoption and implementation of the solution, these may include existing rules and regulations, data ownership, data location, process ownership, and a stakeholder management plan to account for resistance from individual stakeholders.
- A list of performance and other non-functional requirements such as security and scalability. They may serve as barriers to the implementation.
- Documentation of the existing systems and data sources which may interact with the proposed solution. Examine the integration challenges when interacting with those systems. Could implementation of the solution lead to retirement of a legacy system?
- A model of what the operational end state would be after implementing a blockchain solution. This should be done initially for the MVP/PoC and again for a solution once the scope has been expanded. What should the outcomes be after implementing this solution?
- With the operational end state identified, a list of the anticipated benefits.
- Documentation regarding alternative solutions including a list that identifies the pros and cons associated with alternative solutions.
- Documentation regarding existing assets and/or licenses which may be required for the use case. Are there entitlements associated with those assets?

Phase Outputs

The artifacts generated during the assessment phase, such as the documented use case, the stakeholder analysis, the vision of the operational end state, and other deliverables, directly support and should be leveraged during the Readiness Phase.

Decision Gate

If your score is 41 and above, it is highly recommended to commence the Playbook readiness review.

If your score is between 21 and 40, it is recommended to commence the Playbook readiness review.

If your score is between 5 and 20, it is recommended to further review your inputs and the assigned weights before determining if the proof of concept is applicable for blockchain and continuing with the readiness review.

If your score is between 0 to 5, it is recommended that the proof of concept is not appropriate for blockchain.

Phase 2 – Organizational Readiness

The purpose of the Readiness Phase is to prepare enterprises and organizations for blockchain efforts and define key supporting activities to ensure organizational readiness. The structure and activities of the blockchain Readiness Phase are similar to other emerging technology readiness guidelines or strategy frameworks, such as M3. However, there are nuances specific to blockchain that should be understood and considered before an organization undertakes a blockchain initiative. The purpose of this section is to highlight these blockchain-specific nuances.

Phase Inputs

The Readiness Phase leverages artifacts generated from the assessment phase. Figure 5 lists the inputs of this phase.

Key Goals

The key goal of the Readiness Phase is to prepare enterprises and organizations for blockchain efforts by defining required organizational capabilities for success. This aims to increase the likelihood of success by providing guidance based on best practices and lessons learned to the following supporting activities:

- Standing up a blockchain governance office.
- Defining the scope of blockchain services and governance processes.
- Assessing risks and establishing risk mitigation strategies.
- Assessing existing systems' integration readiness.
- Assessing selected key performance indicators' (KPI) evaluation readiness.

Key Participants

To help ensure success, key participants must be identified and engaged throughout the blockchain Readiness Phase. They could include:

- Product owners/managers who undertake overall management and governance of the Readiness Phase.
- Blockchain subject matter experts who may or may not be from the agency initiating the program.
- Subject matter experts from lines of business and systems with potential blockchain integration.
- An enterprise architect who is well-versed with DLT to own the creation of the governance framework and capability definition exercises.
- An information systems security officer/engineer or equivalent who ensures compliance and security of the proposed solution.

The participants may contain other stakeholders in part or whole of this phase. For example, end users could serve as the voice of the customer during requirements discovery and definition.

Approach Guidance

In most cases, the Assessment Phase will precede the Readiness Phase to ensure use case selection and business relevance for the effort has been determined. Although rare, some government agencies may have assessment and readiness phases running in parallel. For example, this may occur when an agency has already completed a proof of concept and is planning for a larger project based on the proof of concept or integration with an external agency that has already implemented DLT.

Figure 5: Readiness Phase (2) summary

Key Activities

Activities in this phase vary depending on the type and scope of selected use cases. Below is a notional activity guideline to prepare an organization for blockchain implementation:

A. Stand up blockchain Program Management Office (PMO) and governance office.

- Establish PMO processes.
- Establish Enterprise Architecture (EA) guidelines.
- Conduct procurement planning.
- Estimate initial cost for the selected business case.

- Create expected benefits chart.

B. Define the scope of blockchain services and processes.

- Analyze as-is view of people, processes and technology.
- Assess selected use case impact.
- Assess change management approach.
- Assess training needs.

C. Establish risk processes.

1. Determine the components of the risk management processes.
2. Conduct initial risk identification and mitigation planning.
3. Prioritize risks based on criticality and area affected (e.g. data security, change management, etc.)

D. Assess existing systems' integration readiness.

1. Define as-is system context.
2. Identify subject matter experts and points of contact for affected systems and interfaces.
3. Define the system integration management plan.

E. Assess selected KPIs' evaluation readiness.

1. Reprioritize KPIs with the new understanding of risks and scope.
2. Define baseline metrics for selected KPIs for the legacy processes and systems.
3. Conduct new value discovery.

Key Considerations

DLT is yet to be proven at production scale for public sector enterprises, which means best practices from production implementations are yet to emerge. However, the following are some key considerations that blockchain evangelists, chief information officers, and enterprise architects should consider as they conduct the Readiness Phase activities:

No.	Key Consideration	Description	Analysis	Takeaway
1	Readiness assessment	For the selected MVP, assess the people, process and technology readiness for the use case(s) that disrupt least number of business touch points but yet have highest scope of improvement	Starting small allows for demonstrating an emerging technology like blockchain to be refined rapidly. It enables the stakeholders to get the first-hand experience and allows the high level concepts to become tangible. Choosing and assessing the readiness of a process that primary stakeholders have complete control of can allow for better governance,	Assess the readiness for a process that is controlled end to end for demonstrating the highest value.

No.	Key Consideration	Description	Analysis	Takeaway
		because of DLT.	faster change management, and ROI assessment. In the Readiness Phase, assess the readiness for the smallest scope by building the constructs to get prepared for technology expansion.	
2	Change management approach	Have a user-centric change management approach, starting right from project initiation.	Change management for emerging technology areas should not be considered as only a post-rollout activity. It should be a key factor considered for all the phases from assessment to production. The way impacted users understand, learn, and adopt the solution becomes the most important factor for demonstrating the key benefits of the platform over time.	Change management should be the top priority to maximize learning and adoption of the proposed solution.
3	Project management approach	Decide on a project management approach that allows management of all the network participants and their activities.	While setting up the blockchain PMO and governance processes, it is critical to decide the project management approach for the initiative. Given that blockchain technology is fluid, agile development would be favored over traditional waterfall or iterative approaches. Agile allows cross-functional, cross-partner teams to remain continuously involved in the product development. This aspect is also critical for success of any DLT initiative, given the number of participants and responsibilities.	Agile product management is best suited to ensure continuous stakeholder involvement and response to continuously changing landscape.
4	Consortiums	Join or, in rare cases, create consortiums of members that have common goals.	Blockchain ecosystems typically involve multiple parties in an industry working together in a consortium to support and leverage a blockchain platform. It is often better to choose the consortium and become a participant once an organization has assessed its use cases and scope.	It is best to be part of an industry consortium to get maximum benefits from a given blockchain ecosystem. These consortiums will be responsible for standardizing the blockchains in the future.
5	Enterprise integration	Determine the context of the blockchain system.	For most enterprise use cases, blockchain technology will be part of the core infrastructure and should be able to integrate seamlessly with other legacy systems.	Create a concept of operations (CONOPS) for the proposed solution to explain the system context in the vision.

No.	Key Consideration	Description	Analysis	Takeaway
6	Value transfer risks	Identify and manage value transfer risks for the value transfer use cases.	A blockchain needs to manage the risks that were being handled by the central intermediaries whom they aim to eliminate. These include fraud detection, key management, asset security, and other risks associated with the value transfer network.	Risk management will need to go beyond traditional people, process, and technology risks to create the management process for risks associated with security, fraud, and new costs of the proposed solution.
7	Consensus mechanism	Define the consensus mechanism.	Readiness Phase activities should include rethinking the conceptual model for Interagency Agreements and/or Memorandums of Understanding/Agreement to shift away from a centralized security approach. This may need education for information security and procurement teams to understand the complexities and evolving needs of blockchain security.	Create common understanding on consensus and security mechanisms, as well as corresponding participant liabilities and responsibilities.
8	Performance expectations	Establish pragmatic performance expectations in terms of metrics, such as transaction speed.	Blockchains are not a replacement of traditional high-performing, super-tuned databases. They are a complementary technology meant to solve different problem domains/use cases.	Create realistic non-functional requirements due to current blockchain capability tradeoffs.
9	Framework-based design	Establish guidelines for a blockchain technology framework that is modular, reusable, and extendible.	The technological landscape is fluid. Projects based on today's solutions will have to be reworked or re-implemented onto the eventual leading platforms in the future. Consider government wide initiatives using a shared services/platform approach and open source software.	Blockchain is still an emerging technology. Aim for reusability but, more importantly, modularity and extensibility.
10	Cross-functional team	Establish a cross-functional government team.	In addition to enterprise IT and business and functional teams, blockchain initiatives must engage with customers in this phase. The governance team must ensure to engage risk management, regulatory compliance, IT operations, finance, accounting and tax teams, etc. to ensure that the requirements of these stakeholders are recorded appropriately.	Create commitment. Draft charters to ensure ongoing support from multiple organizations. Identify tools that can support inter- and intra-organizational development and communication activities (e.g. using cloud-based public-sector tools, etc.)

No.	Key Consideration	Description	Analysis	Takeaway
11	Talent management	Define the skillset and training needed to implement and maintain blockchain initiatives.	Organizations will need experienced IT talent who can implement and maintain blockchains, as well as support network participants. Government agencies may have to rely on technology partners and third-party vendors who have a working knowledge of different blockchain ecosystems.	Consider training and developing internal talent for continuity while leveraging external talent on an as needed basis for immediate initiatives.
12	User experience	Establish user-centric design guidelines.	Blockchain is generally considered a backend technology which end-user facing system rarely see directly. That may or may not be true for all the use cases. Other than the underlying code and algorithm, every user touch point must be designed with user-centricity focus. All users – such as backend, administrators and enterprise users – should get the same quality of experience as the end users. Laying the ground rules for design right from Readiness Phase helps in enterprise-wide adoption in the long run and in covering all the non-functional requirements, such as privacy, confidentiality, security and personalization.	User experience is critical for enterprise-wide adoption and should be looked from middle/backend users' perspective across every user touchpoint that is in the purview of the selected use case(s). Iterative agile approach with product ownership and Lean UX techniques can be utilized to ensure the best user experience.
13	Emerging tech specific risk management	Understand the agency's risk appetite and plan, communicate, mitigate and discover risks continuously.	Agencies that do not accept risk may not be willing to be involved in blockchain, as this is an evolving technology. Risks related to emerging technology must be managed as the top-most governance activity for such agencies.	Manage risks with the focus on change management, technology immaturity, availability and sustainability of skills, lack of standards, acceptability of disintermediation, switching and sunk costs, network effects, securing agreements and stakeholder adoption.
14	Expansion strategy	Create an implementation strategy that allows the program to expand in a risk-controlled manner.	Starting small in a controlled business process with high impact on the day to day transactions of end users is the best strategy. But that should not mean postponing the strategy planning and design for expansion, its associated risks, and	Start with the expansion in mind. Acknowledge and communicate the gaps in the initial implementation to finally allow filling them up during expansion.

No.	Key Consideration	Description	Analysis	Takeaway
			mitigation strategies.	

Key Outcomes

To ensure a government agency's readiness for an emerging technology like blockchain, several internal and external factors have to be assessed, and in some cases, new areas need to be defined and established. The list below highlights definitions and high-level plans, which are further refined in subsequent phases and throughout the lifecycle of the initiative, resulting from the Readiness Phase:

- Key network participants engaged with formal agreements.
- Security strategy for participants defined and agreed upon by the different parties.
- Onboarding/separation strategy defined for DLT participants.
- In the case of a new consortium, responsibilities and governance model defined.
- Mitigation plans in place for the following risk categories:
    - Technology
    - Business
    - Security
    - Performance
    - User experience
    - Governance
    - Adoption
    - Regulatory compliance
    - Enterprise Integration
- Change management strategy defined for all the impacted parties.
- KPIs defined and baselined for the selected business case.
- Subject matter experts and points of contact from cross-functional teams and integrating systems are on boarded.
- Procurement strategy defined.
- Initial schedule and master plan defined.
- Business capabilities defined.

The business case selected for implementation, external context of the implementation, and stakeholder and regulatory requirements may result in additions or modifications to this list.

Phase Outputs

The following artifacts generated during the Readiness Phase support the Selection Phase and the phases following it. They should be leveraged to ensure the alignment to initial vision, continuous discovery, monitoring and mitigation of risks and continuous feedback to the stakeholders for forthcoming implementations:

- Initial business capabilities
- Scope of services overview
- Target state CONOPS chart
- Change management strategy
- Program governance model
- Prioritized risk management plan with mitigation plans for top risks
- Initial cost and schedule estimates
- Procurement plan
- EA guidelines
- KPI baselines and measurement guidelines
- Validated as-is process maps

Decision Gate

At the end of the Readiness Phase, the agency should be able to answer the following questions for the system (people, process and technology) and its governance:

- Proposed system:
    - What are the key business capabilities of the proposed system?
    - Who are the key participants in the proposed blockchain initiative?
    - Who and what will be impacted? What are their roles?
    - What will be the impact?
    - How will the onboarding/separation happen?

- Strategy and governance:
    - What is the proposed governance and management structure?
    - What are the key technological, business context, security, performance, user experience, program management and governance related risks specific to the proposed DLT solution?
    - How will key risks be managed?
    - Are KPIs defined and baselined?
    - Does the initial schedule and estimated cost allow for agile product development where the time and cost can be recalibrated based on ongoing learning?
    - What is the procurement strategy for the proposed program?
    - How will security be managed?
    - How will change be managed for the impacted people, processes and system?

At the end of a successful Readiness Phase, the stakeholders should have a joint understanding of the responses to the questions highlighted in the Decision Gate section.

Phase 3 – Solution Selection

Blockchain was first introduced as a type of DLT presented as the underlying platform of the Bitcoin cryptocurrency but has now evolved into a platform with much broader uses cases and various technical implementations. When it comes to selecting a blockchain solution for your agency, there are several factors to consider:

- **Business considerations** – Is it a fit for your agency? Consider who manages and controls the platform, whether you need a permissioned blockchain for security reasons, and the type of consensus mechanism used for adding transactions to the blockchain. Ask whether and how you will use smart contracts, in what languages you could code them, and how much involvement will be needed from your legal or acquisitions team. Pay close attention to operational requirements, such as response times and transaction costs, and determine where you plan to host the system and what it would cost. Also unique to government – consider how to obtain the authority to operate and whether options exist within Federal Risk and Authorization Management Program (FedRAMP)-certified environments.

- **Technical requirements** – If it is a fit, which version best matches your requirements? Consider the degree of scalability and volume you need and whether this is best served by public or private infrastructure. This choice will also impact speed and latency, as well as security and immutability. If you will be supporting digital assets, then you must look at how assets are issued, funds are secured, and identity is managed, as well as blockchain specifications, notaries, network security modes, user authentication/authorization, and asset issuance and deployment. The question of open source versus proprietary platforms will be a key consideration, as well as deployment models.

- **How to buy** – Once you have identified your requirements and selected a suitable solution, it is still best to start small and aim for a minimally viable product (MVP). Focusing on the use case allows room for the ongoing evolution of the technology.

This section explores each of these factors in depth to guide you in the selection process.

Phase Inputs

The Selection Phase expands upon the outputs from the Readiness Phase and starts delving into the various concepts and requirement categories that need to be considered and analyzed for successful selection and implementation of a blockchain solution.

Figure 6: Selection Phase (3) summary

Business Considerations

The business problem and requirements must be clearly identified to ensure the appropriate blockchain platform is selected. Blockchain is a cross-cutting and transformative technology, but it is not the solution to every problem. Review and understand the description of the platform to ensure it matches the need. Look past the jargon to understand the specific differences noted below:

- **Governance** – Determine the nature of who controls and governs the software platform. For example, open source platforms, such as Ethereum and Hyperledger Fabric, are governed by their developer communities via nonprofit foundations, whereas Corda is managed by a corporate consortium called R3. The governance model could affect the support available.

- **Mode of operation** – Blockchain infrastructure can be operated as permissioned or permissionless. Government typically requires the permissioned mode for security reasons, but there could be use cases for connecting to the public permissionless chains, especially as techniques like zero knowledge proof protocols evolve.

- **Transaction costs** – Apps deployed on the Ethereum public blockchain incur transaction costs based on computational resources consumed. Private blockchain implementations do not have this requirement, but require their own supporting infrastructure, such as cloud servers.

- **Consensus** – Blockchains must reconcile transactions to maintain a single version of truth. At the time of writing this document, Ethereum uses a proof of work (PoW) algorithm (soon to switch to a hybrid proof of work/proof of stake algorithm called Casper). PoW ensures a high level of immutability and transparency. With the variety of consensus mechanisms available today, some algorithms may have more fine-grained approaches that offer better performance and privacy.

- **Smart contract** – The smart contract refers to the blockchain's ability to store and automatically execute computer programs when specified conditions are met. Each solution supports a different set of programming languages, so consideration should be given to in-house programming expertise. For example, Hyperledger supports Java and offers a composer tool that allows organizations to develop smart contracts without writing much code, while Ethereum uses its own Solidity language. Corda also expands on smart contract by supporting the incorporation of legal prose along with the code.

- **Currency** – Although initial government use cases are likely to be focused on non-currency digital assets, such as contracts or land deeds, multiple forms of cryptocurrencies may need to be supported in the future. Support for this varies by blockchain. For example, Ethereum has Ether built-in. Ethereum and Hyperledger provide the ability to create other cryptocurrencies, and Corda provides little support for currency functionality overall.

- **Chain data requirements** – There are multiple ways that data can be stored on the blockchain, each with implications for security and performance. For example, you could store an entire contract package in a block or just a hash value of the location of those documents stored off that blockchain. One approach may be more secure, but the other could offer higher performance.

- **Operational requirements** – Does the current system satisfy the requirements? Is it a build within the private cloud or a buy (software as a service (SaaS))? Where are all the participating nodes going to reside, and do they trust each other? What interface requirements have to be met? What is the response time requirements for the blockchain? Other considerations include reliability and uptime in addition to the standard operational requirements for any software solution.

- **Security requirements** – A key factor in choosing any blockchain solution is to ensure its ability to obtain an authority to operate (ATO). A blockchain platform operating in an environment that is already FedRAMP authorized might be attractive as it makes getting an ATO much simpler. However, government must consider the potential of the peer nodes – the connection points to the blockchain – residing outside the standard system boundary.

- **Operating cost** – The cost of operating and maintaining a blockchain solution is very difficult to estimate. There are no models to give a rough estimate and some unique factors, such as

increasing storage requirements with the increase of blocks, increasing computational requirements for the participating / consensus node, and no clear 'ownership' of the ledger. The agency should be aware of these factors and the reasons for a non-existent operating cost model when embarking on a blockchain project.

- **Change management** – Any blockchain solution comes with unique change management requirements due to the nature of the technology. Are changes to smart contracts allowed? If so, how does one manage that among all participating nodes? Changes to the asset structure in the block is not allowed. The agency should consider how these fit into their change management procedures, along with any changes to the procedures themselves.

- **Private cloud versus SaaS (Build versus Buy)** – The choice of deploying blockchain on a private cloud versus using blockchain as a service is governed by the following factors (refer Appendix D for more details.)
    - **Sensitivity of the data** – Data sensitivity and security is paramount for government. The blockchain vendor deploying their platform over a private cloud could provide private distributed cloud storage of data for maintaining data sensitivity. Another option would be an Ethereum- or Bitcoin-like data storage where anyone can get access to the data if the credentials are presented to the controlling node. Data sharing among entities would be controlled by the blockchain provider based on permission rules defined in smart contracts.
    - **Deployment model** – Implementation of consortium, semi-private or private blockchains requires a greater level of permissioned control over blockchain resources by providers. With blockchains deployed in their own private clouds, the blockchain providers can better control the resource configurations by implementing their own custom protocols, e.g. permissions, type of consensus protocol, size of Consensus Notary blocks, and number of blocks required to verify transactions etc.
    - **System maintenance** – Will you use your own DevOps resources and manage the service level agreement governance or leave that to the blockchain service provider?
    - **Initial costs** – Setting up an environment to test and research blockchain is not a trivial undertaking. A private cloud is more expensive upfront but allows the organization to better control the resource usage across users. A service offering is simpler, but less configurable.

Technology Considerations

Permissioned versus Public Permissionless Blockchain

A blockchain is a data structure that makes it possible to create a digital ledger of data and share it among a network of independent parties. There are mainly three different types of blockchains.

- Public blockchain networks
- Permissioned blockchain networks
- Private blockchain networks

All three types of blockchains use cryptography to allow each participant on any given network to manage the ledger in a secure way without the need for a central authority to enforce the rules. In addition to an introduction to the types of blockchains in the ACT-IAC Blockchain Primer, Appendix A provides further details about these blockchain networks.

Determining Which Blockchain is Right For You

The following criteria should be considered while evaluating a blockchain for any project:

- **Fitness for purpose** – Consider and research to ensure the business requirements are within the scope of the mission of the product. For example, solutions like SkuChain or R3 Corda are purpose-built for financial transactions, while tools like Linux Foundation's Hyperledger Fabric or Microsoft's CoCo are generic solutions, allowing you to build purpose fit products to meet your requirements.

- **Operational considerations**
  - **Blockchain scalability and volume** – The scalability of a particular blockchain network type determines the maximum transaction throughput (number of transactions processed per second) and, as your blockchain grows, the maximum volume of transactions that can be processed within reasonable performance criteria. Adequately develop the lifecycle requirements to ensure the solution being considered meets the scalability the agency needs over time.
  - **Performance of the blockchain, especially speed and latency (review benchmarks)** –Blockchain speed is the transaction throughput (maximum number of TPS), which is determined by the block size and the consensus delay. It is dependent on the combination of the processing power of the network in which the algorithm is placed, how large the block is, and the particular type of encryption protocol. Performance is not determined by whether a blockchain is public or private.
  - **Security and immutability** – The documented level of confidence of security within the blockchain is high. The blockchain itself is inherently resistant to threats while the off-chain applications (e.g. smart contracts, distributed apps, and microservices) are not.
  - **Storage and structural needs** – Since every node in the network maintains a copy of the chain, every node needs to have enough storage. Hence care should be given to reduce the size of each block.

- **Resource requirements** (**Appendix E**) – The choice of blockchain infrastructure will impact the type of resources needed both internally and with the implementation team. For example, if Ethereum is chosen, there is a need for Solidity developers. Architects who understand blockchain operations and their interactions are also needed.

- **Off-chain data management** – Due to security and speed considerations, the operator of a blockchain may choose to store the bulk of the data off the chain, with a pointer to it stored

on the chain. IT operations should consider the procedures and infrastructure needed to manage this data as well.

There may be exceptions depending on the project, and it is possible to use a different type of blockchain to reach a particular project's goal. Please refer to Appendix F for further explanation of the technical criteria.

Digital Assets and DLT Requirements

A digital asset is a floating claim of a certain service or good(s). It is ensured by the asset issuer, which is not linked to a particular account, and it is governed using computer technologies and the internet – including asset issuance, claim of ownership, and transfer. Blockchain-based DLT provides an alternative to centralized digital asset management system by providing:

- **Distributed transaction processing** – Transactions are processed in a decentralized manner by geographically distributed nodes of the network. Moreover, defining the rules for transaction processing (e.g., defining what valid transactions are) could be split from the processing.
- **Asset issuance** – In the most general case, this could be performed by any user of the network.
- **Security of a user's funds** – This could be performed by third parties using custodial or non-custodial wallets.
- **Identities of services** (and optionally customers) – This could be established by building Public Key Infrastructure (PKI) based on a blockchain.
- **Application development** – This does not require cooperation with blockchain maintainers.

Specific Considerations of a Blockchain Supporting Digital Asset Management

There are some additional considerations for using blockchain technology to support digital asset management (additional details are in Appendix F):

- Blockchain specification
    - **Transaction logic** – What are valid transactions with regard to the present system state (e.g. the rules regarding how transactions transform the system state, etc.)
    - **Immutability logic** – What transactions constitute a block and how are the block headers secured.
    - **Consensus logic** – How nodes agree upon the state of the system; how blockchain forks are resolved, etc.
    - **Network logic** – How transactions, blocks and other data are transmitted among network nodes, etc.
- Blockchain notaries
- Blockchain network
- User authentication and authorization
- Asset issuance
- Deployment models

Deployment Models

There are various deployment models for blockchains in the market. Government should consider which model suits them the best. Listed below are the common models. Please refer to Appendix B for details.

- **Separate Blockchains for Assets**: Each digital asset or a set of assets maintained by the same issuer could potentially have its own blockchain, either permissionless or permissioned.
- **Colored Coin Protocols:** Colored coin protocols share the user authentication model with the underlying blockchain. Some examples include Chromaway, metacoins and Open Assets.
- **Multi-Asset Blockchains**: Multiple assets can be natively supported by a blockchain. Multi-asset blockchains have more space-efficient proofs of ownership, as simplified payment verification could be utilized for all natively supported blockchain assets.
- **Smart Contracts**: User-defined assets could be represented with the help of a smart contract on a smart contract blockchain.

Open Source Versus Proprietary Blockchain Platforms

Different open source blockchain platforms are suitable options in implementing different consensus protocol mechanism, blockchain network types or specific use cases. They are a good option when implementing blockchains with more censorship resistant use cases. The use of open source blockchains would reduce the investment cost in building blockchain services. However, organizations may need to manage the security, scalability and throughput considerations in their own custom ways. Interoperability and ease of integration are areas of consideration, as open source blockchain platforms do not traditionally do well in these areas. (Details for open source/proprietary models are in Appendix C.) Blockchain as a Service (BaaS) (details in Appendix D) is an emerging model that combines the benefits of an open source platform with the benefits of proprietary solutions.

How to Buy – Reference to GSA Atlas

While the procurement methods and vehicles vary, GSA's FastLane process improvement contract serves as a good example of how an agency should procure a blockchain solution.

Leverage commercial contracting methods

Leverage the efficiency of commercial contracting methods. Blockchain was created by the commercial industry, so government should buy it as a commercial item and use commercial buying methods. Similar to how private companies structure blockchain development buys, the procurement process should require vendors show, or demonstrate, how they build blockchain products, not merely tell about process via a long proposal. In addition, industry doesn't use cost-type contracts for blockchain and neither should government.

Buy small, build small, test, and iterate

Set up each contract for a quick win, then determine how to scale that success or pivot quickly. Well-intentioned agencies frequently default to creating large scale software development contracts. However, these procurements often take years. In addition, larger contracts often lead to more bureaucracy, slowing down the delivery of a working blockchain. Instead, adopt modular contracting methods. First award a smaller contract for the development of a blockchain MVP. The time to award will be faster, the procurement risk lower, and within 12 months of award, the agency will likely show a quick win with a deployed working product. Working products also convince others about the utility of the technology and, therefore, play a role in the agency's comfort and adoption of blockchain. Once the MVP is developed, the agency can pivot or procure additional services to further develop features for the blockchain product.

Do not lock technical requirements into the contract

To ensure the use of new, more effective technologies and methods is not blocked, do not lock technical requirements into the contract. Overly prescriptive requirements significantly limit a vendor's flexibility to propose innovative approaches to blockchain. They also wrongly assume that the features described in the original requirements document are equally valuable and the users' needs will not change over time. This significantly limits the ability to leverage valuable blockchain product features that are not yet developed. Instead, use a Statement of Objectives to scope the requirement around the use case or product vision. The government will still control the technical requirements in the form of user stories, which will be crafted during the life of the contract and will be prioritized and selected by the product owner before each iteration cycle. The contract will provide the flexibility to build any features within the broader use case or product vision. In addition, this approach allows technological enhancements to never end.

Pay for results, not time

Buy blockchain design and development services as a repeated process for the delivery of a working product. When technical requirements are not locked into the contract and the government and the vendor together formalize the definition of "done," then one of the most advantageous pricing formats is firm-fixed price per iteration. When the vendor completes the user stories at the end of the sprint, and the definition of done is met, the vendor gets paid and the government receives working code that is ready to be deployed.

Acquisition Considerations

The focus in government in engaging in any new blockchain endeavor begins with the contracting community which includes Contract Officers (CO), Contract Specialist (CS), and Contract Officer Representatives (COR/COTR). The acquisition workforce enables the government to access and consume emerging technologies such as blockchain.

Develop Acquisition Models and Milestones

You have assessed and selected the blockchain technology. You have determined the readiness of the agency. The acquisition now turns its attention to the solicitation and production aspects of the program. If the blockchain project is considered a pilot or proof of concept, the acquisition team can utilize specialty mechanisms such as the Procurement Innovation Resource Center's (PIRC) Commercial Solutions Opening (CSO) procedures (e.g., Other Transaction Authority agreements for research, prototypes, or follow-on OT production). This program is designed to expand beyond the current Federal Acquisition Regulation (FAR) procurement methods. CSO provides a streamlined approach to exploring the acquisition of emerging technologies by the federal government.

Alternatively, you may be working with academia, other agencies, a lab, an intermediary, or a company through a technology transfer program—such as a Cooperative Research and Development Agreement—Small Business Innovation Research Program, or other grant, agreement, or contract program designed to accelerate technology development and/or transition of innovative technologies.

If the project is considered a standard modernization activity of the current portfolio, the Contract Office follows standard procurement methods for their agency. The solicitation defines the method of evaluation, the basis of evaluation, the basis of award, and determination of price fair and reasonableness. These criteria assist in the selection of the successful bidder.

Find more information regarding the FAR and other agency acquisition regulations at Acquisition.gov.

Define Performance Metrics

The project team provides performance metrics for both the technology and services to be included in Section C of the solicitation documents. The Statement of Work (SOW) expresses expected Service Level Agreements (SLA) and performance metrics to assess the success of the implementing contractor and overall success of the program. The project team may create the SOW or, in the case of nontraditional contract vehicles/transactions/agreements, might be an iterative and collaborative effort involving the project team, the contract office, general counsel, and others from both the government and the contractor organizations. For a blockchain solution, this might include (but is not limited to) metrics and milestones on which to focus:

- Establishing a finalized governance model.
- Completing the user interface.
- Development of a smart contract and associated web application.
- The creation of a working solution with multiple nodes and test data in a non-production environment.

- Advancing to a production environment.
- Implementing the use of live data.
- Adding multiple government business partners to the network.
- Adding non-government business partners to the network.

Prepare Acquisition

The SOW states the selected technology for the desired blockchain solution, express the component and configuration of the selected platform, describe the required products, supplies or services required by the government for implementation of the target technology, and define anticipated tasks necessary to successfully implement the Blockchain enabled solution.

The source selection team evaluates offerors, identify deficiencies, weaknesses, risks, and strengths. After scoring proposals, the source selection team with produce proposal analysis reports and brief the source selection authority.

Award Solicitation

Upon award of the solicitation, the Acquisition Team approves the contract, send out notifications to all acquisition participates, announce the award and debrief unsuccessful offerors. Notice to proceed is sent to the project office to implement the selected Blockchain solution.

If the introduction of a blockchain-enabled portfolio requires a blockchain-enabled enterprise, provide appropriate infrastructure requirements to the enterprise architect and information systems security officer for concurrence. Enable and configure the General Support System (GSS) to accommodate the newly tooled portfolio. Enabling the enterprise may require contract modifications in future phases across the vendor landscape. Collaboration with the CO stabilizes the project as a whole.

Phase Outputs

After analyzing the various factors that have to be considered in choosing a blockchain platform, a detailed platform model starts to evolve, which then drives a detailed schedule, cost, resource and task estimate. The corresponding acquisition model, along with the operational framework, start to take shape.

Decision Gate

At the end of Selection Phase, the following use case questions should be answerable:

- What type of blockchain technology should be used? Is it private or permissioned or public?
- The blockchain mechanism to be used to maintain ‘world state’ (PoW, proof of stake, smart contract, etc.)
- A Governance Model
    - Who owns the asset?

  - Who can initiate requests to add to the chain?
  - Which organization governs the smart contract?
  - How are changes to the Smart Contract negotiated?
- What is the asset data structure, and corresponding off-chain requirements?
- A decision on the deployment model – private or SaaS model.
- What are the components of an implementation and operating cost model?
- Will the solution use open source or proprietary blockchain technology?
- What are the resource requirements for the development and operations of the solution?
- A procurement approach and model, with all the necessary approvals.

The following factors will determine your moving to the next phase:

- If your development and operating cost exceed your funding level, stop at this phase.
- If you don't have a contract vehicle, stop at this phase.
- If your resources are not up to speed with the technology, stop at this phase.

Phase 4 – Blockchain Implementation

This phase ensures that the inner workings of the solution are completed and tested. It closely examines the technical implementation of the components of the blockchain network, as well as the operational aspects, such as governance and security posture, to ensure the optimal operations of the blockchain solution. Key activities and outcomes for management, technology, people, process, and acquisition are also examined.

If you were manufacturing a cell phone instead of deploying a blockchain solution, you would expect that at the end of the Implementation Phase, the subcomponents of the phone were complete and tested before moving to the Integration Phase of incorporating a particular carrier's service, other applications, accessories, etc. The concept is the same here.

"One cannot integrate what has not been implemented."

The term "Authority to Operate (ATO)" defines the criteria for success for the blockchain solution: the system should perform and operate according to the specified (functional and technical) requirements set forth by the product owner and the other stakeholders, and the solution's inner workings should be tight and secure.

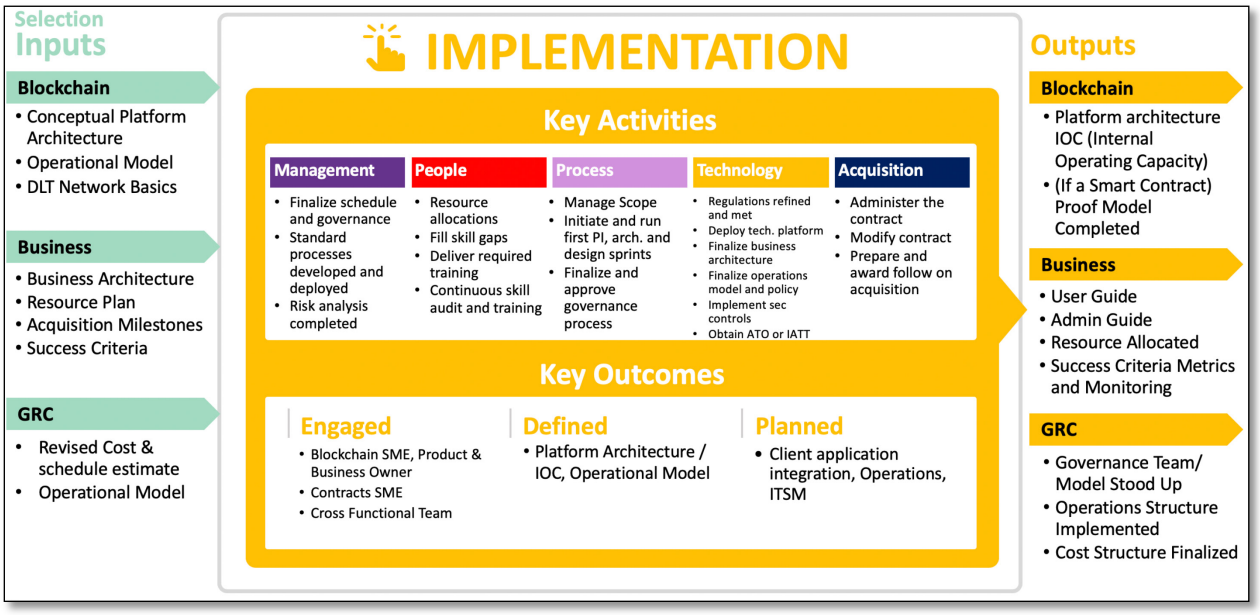

Figure 7: Blockchain Implementation Phase (4) summary

Phase Inputs

To proceed with implementation, the blockchain must leverage the outputs from the Selection Phase. The solution needs to have an established Conceptual Platform Architecture, Operational Model, and DLT Network Basics. The business must have a Business Architecture, Resource Plan, and defined Acquisitions Milestones and Success Criteria. The Governance, Risk

and Compliance (GRC) area must have a revised Cost and Schedule Estimate, Acquisition Plan, and Operational Model.

Key Goals

The key goal of the blockchain implementation phase is to take the models, processes, technologies determined in the previous phases and implement them so they can be integrated within the organization(s) in the next phase like smart contracts, blockchain network, blockchain service, governance, workforce, and metrics.

Key Considerations

Now that the blockchain framework and the protocol you will be implementing is known, it is time to consider some additional factors that have the potential to affect your success:

- The choice of the platform. In the Implementation Phase, the choice you have made in previous phases leads to a series of additional considerations related to programing and architecture.
    - For example, in an Ethereum-based platform, you will most likely program in a language called Solidity and need to choose an integrated development environment for developing applications in your ecosystem.
    - Another example is in a Hyperledger-based platform, you will need to finalize the appropriate number of separate orderers (whose sole responsibility is to ensure the order of creation of blocks) to make certain there are no centralized points negating the benefits of a decentralized system.
- The consensus model, such as Proof of Work or Proof of Authority. This choice was finalized during the Selection Phase, and it affects how governance is approached. During the Implementation Phase, you will need to finalize and implement the related governance rules. Ensure it is implemented as intended as poor implementation may lead to incomplete transaction processing and data integrity issues.
- Participants and their roles. Refinement is needed regarding the identity, credential, and access management decisions made during the Selection Phase. How are participants identified? Are there group accounts? How many and who participates in the consensus mechanism? Do the consensus participants change over time? If changes occur, how might each change impact the governance rules set in place? Who has access to what data? How is physical access addressed? How are personnel changes addressed?
- Operators of the system at different levels. How will end users leverage the functions provided? How will developers use functions included for them? During the Readiness Phase, you defined the skillsets and training needed to implement and maintain blockchain initiatives. Now it is time to write guides, launch supporting resources, and train. It is also time to translate insights from the stakeholder analysis performed in the Assessment Phase and the user experience guidelines you developed during the

Readiness Phase into the user interface designs for items such as auditing and display of the blockchain data.

- Onboarding and offboarding. During the Readiness Phase, you crafted an initial onboarding/separation strategy. After making your technology selections, the strategy should be refined. For example, in a permissioned network, careful thought has to be given to the verification of an organization that wants to participate. Additionally, you must ensure a new entrant does not lead to security vulnerabilities. Examples of security issues include 51% vulnerability on a public blockchain and on a private network, a similarly unfavorable shift or incompatibility with the consensus rules.
- Asset creation and storage. During the Selection Phase, you gave careful consideration to on-chain versus off-chain data (asset) creation, as well as its implication on performance, storage, and security. You also selected a deployment model. Now it is time to implement, ensuring the ability to scale and potentially archive as those assets go through their lifecycles.

Additional considerations and refinement are necessary to take your blockchain solution from the plans and strategies developed in earlier phases to launching and maintaining during the Implementation Phase. It is time to ensure the technical and functional plans of a comprehensive solution.

During the Implementation Phase, be sure not to lose sight of the primary goal. You must ensure the solution solves the business problem discussed in the Assessment Section of this Playbook.

Key Activities

Implementation of an enterprise blockchain platform typically involves:

- The overarching project management plans, which drive the deployment and maintenance of a system, including standard operating procedures and continuous risk management plans. These plans are finalized and begin execution during this phase.
- The stakeholders who will be procuring, providing, operating, and interfacing with the blockchain solution. They engage with the blockchain PMO, receive any necessary training, and begin working in the ecosystem.
- The requirements to be met through implementation. Finalize the list of, priority order for, and iterations in which functional requirements will be developed. Refine the list of operational requirements and governance rules necessary for deployment and maintenance of a system with regard to lifecycle and use case events.
- The physical components of the ecosystem, which include shared systems, infrastructure, and applications that manage the blockchain and its functions, as well as interact with and affect the data of the blockchain. Update and finalize the architectural blueprint and design.
- The blockchain data and transactions, including information stored within the blockchain ecosystem and the data elements/transactions for which network members

have varying authorities. Refine and finalize the logic, security, and other considerations regarding data creation, management, validation, encryption, any archiving, and destruction.

Key Activities: Management

Finalize Schedule and Governance

Ensure that you are managing scope, time, cost, quality, human resources, and risk across the blockchain project. Refine business cases for technology investment, working with enterprise architecture and information security professionals necessary over the solution's lifecycle. Work closely with blockchain technical leads, developers, and testers. Manage any outstanding legal and regulatory compliance issues if associated with digital tokens. Refine the plans for challenges in implementing and integrating blockchain technology solutions, such as testing interoperability of blockchain applications with legacy solutions and data integration challenges. Continue to refine and implement the change management approach developed during the Readiness Phase to address the business paradigm shift made possible through the implementation of blockchain technology.

Standard Processes Developed and Deployed

Standards in policy and processes will be derived from the finalized governance model and integrated technical strategy (both referenced in this playbook). Disciplined operations teams manage enterprise-wide appliances, which govern blockchain appliances, through a rigorous configuration management practice.

Risk Analysis Completed

To achieve the goal of ATO, the risk management team:

- Finalizes its selections for the appropriate security controls.
- Implements the selected security controls.
- Assesses the implementation of the controls to ensure they are operating as intended and producing the desired outcomes.
- Evaluates whether the blockchain fabric is hardened as required.
- Documents actions to ensure continued security and low risk.

During the Implementation Phase, the team monitors and audits the blockchain technology and identifies any weaknesses in its security posture. This includes examination and determination of the likelihood and magnitude of harm from unauthorized access, use, disclosure, disruption, modification, or destruction of the blockchain, ecosystem components, and the data processed, stored, or transmitted. The team then updates its plan of actions and milestones for mitigation and audit defense activities.

Be sure to factor in Federal Information Processing Standard (FIPS) Publication 199, "Standards for Security Categorization of Federal Information and Information Systems," using the

blockchain as a complement to integrity checks to determine how prepared and where the department/agency information system security officer stands.

Additional specific activities incorporated into a rigorous risk management practice, as it relates to the implementation and integration of any blockchain fabric, are discussed throughout this chapter.

Key Activities: People

The implementation of blockchain into the agency/department will be a service and platform procured by the acquisition team. Key to the government being able to implement a blockchain solution is equipping program and project managers with the knowledge, skills, and abilities to manage a technical workforce which includes contractor personnel. The government has recognized the need for this type of blended digital workforce (digital IT acquisition professional)[6], and the Federal Acquisition Institute is working to deliver the training and certification.

It is also critical to have a Contract Officer Representative (COR) capable of understanding and managing the contract terms and conditions as part of the lifecycle sustainment of the procurement approach and model developed during the Selection Phase.

The key participants identified and engaged since the Assessment Phase must continue to be involved. This includes blockchain network members, representative(s) of the line(s) of business, the application portfolio, and supporting IT organizations. Depending on the attributes of the deployed fabric, enterprise governance will transform in support of the new service. Continued engagement of stakeholders impacted by the implementation and integration of the new blockchain service can provide critical feedback and buy-in for continual service improvement.

Effecting Workforce
The ***effecting workforce*** includes the workforce responsible for the acquisition, implementation, and monitoring the blockchain fabric. [6]

Additional key participants include:

- Blockchain Subject Matter Expert has the knowledge and expertise needed to help develop the necessary smart contracts and blockchain network infrastructure, as well as to customize and configure the blockchain service.
- Product and Business Owner need to be iteratively involved to ensure the implementation developed by the blockchain subject matter expert satisfies the requirements.
- (Smart) Contracts Subject Matter Expert is need if the blockchain solution involves smart contracts. This expert will help with the programming of the smart contract, as well as testing and evaluating the logic to uncover potential vulnerabilities. To do this, software

development best practices are employed and game theory is used to test the logic and corner cases of the contract.

- Cross-Functional Teams make connections among the domain-specific knowledge, the blockchain knowledge, and other technical knowledge. This helps the blockchain subject matter experts with the development of the blockchain solution by bringing together the domain logic for the smart contracts development, the organizational infrastructure knowledge for the implementation of the blockchain network, and the application knowledge for future integration of enterprise application.
- Product and Business Owners are also need to be iteratively involved to ensure the implementation developed by the blockchain subject matter expert satisfies the requirements.

Resource Allocations

The resource allocation plan for talent management will be updated, finalized, and implemented. It should pave the way for addressing the skill gap, and it should consider how to best leverage existing resources to successfully implement the blockchain solution.

Fill Skill Gaps

Execute the resource allocation plan to address any outstanding skill gaps according the audit results. Stakeholders will either be trained or complemented with internal and/or external subject matter experts and/or tools that will be added for the duration of the Implementation Phase iterations.

Deliver Required Training

Training will need to be specific to the domain expertise of the resource and according to the skill gap. For instance, acquisition specialists would not need a developer course in Ethereum, but they might need a business understanding of blockchains, token-based economy, smart contracts, and the use of gas if public Ethereum is used for the blockchain solution.

Furthermore, the Office of Federal Procurement Policy (OFPP) launched a new specialization for federal contracting professionals to increase expertise in digital services. The Federal Acquisition Certification in Contracting (FAC-C) Core-plus specialization in Digital Services (FAC-C-DS) builds on the digital IT acquisition professional (DITAP) program and focuses on learning to design innovative and flexible procurements for services such as human-centered design, iterative development methods, cloud, and X-as-a-service.[7]

Continuous Skills Audit and Training

During the Readiness Phase, you defined the skillsets and training needed to implement and maintain blockchain initiatives. In the Implementation Phase, you should monitor and refine the training audit. Any additional surveys and/or evaluation of the current workforce will take place to determine whether the stakeholders have the appropriate skills.

Continuous learning is required to maintain the ecosystem throughout its lifecycle and should be included in the talent management and resource allocation plans being executed during the Implementation Phase. As blockchain evolves, new best practices and lessons learned, as well as updated software, configurations, etc., will result in the need for the effecting workforce to adapt how they are implementing the blockchain ecosystem.

For example, security practitioners should receive incident response training for the blockchain ecosystem upon ATO. But they should also maintain ongoing contact with blockchain security groups, professional associations, academia, and any relevant open source communities to ensure they are current and trained with regard to evolving security practices, techniques, and technologies, as well as to ensure they have awareness of the latest threat and incident information and other relevant updates.

Key Activities: Process

Manage Scope

During the Implementation Phase, you should ensure the project's scope is accurately defined and mapped to a work breakdown structure with sufficient resources allocated. Continued collaboration with stakeholders during the Assessment, Readiness, and Selection phases should have resulted in the identification and refinement of use case ideas, value propositions, risks, and costs. These previous phases produced the blockchain-enabled solution's continuity of operations plan and a validated analysis of alternatives, which translated to several high-level scope decisions.

In the Implementation Phase, it is necessary to clearly refine and finalize what requirements are in scope and what requirements are out of scope without any ambiguity. Any scope creep should be diligently addressed.

Best practices dictate the implementation team should finalize the documented project goals, business objectives to accomplish, success criteria, deliverables, epics, features, user stories, evaluation criteria, tasks, milestones/timelines, assumptions, dependencies, and resource needs. There are specific categories of work scope where the team will need to validate/re-validate assumptions, make decisions, and lock the scope of work with user stories and acceptance criteria in the context of the following:

- Business processes that are being covered.
- End users/blockchain participants and their roles and responsibilities.
- Permissioned versus permissionless network.
- Identity, credential, and access management.
- Consensus mechanism.
- Nature of data, digital assets, and transactions.
- Form factor of nodes.
- User interface/user experience.

- Smart contracts.
- Interfaces to oracles and partner references.
- Third-party intermediaries.
- On-chain versus off-chain information restrictions.
- Security controls.
- Scalability, performance, auditability, and security requirements.

Initiate and Run Series of PI, Architecture, and Design Sprints

Implementation of an end-to-end blockchain solution involves engineering both the infrastructure solution and the application solution that runs on top of the infrastructure solution.

The infrastructure solution involves heavy investments and it cannot easily be changed. When you considered design in the Readiness Phase, you should have refined your architectural blueprint with modular, reusable, and extendible options whenever possible. Given the rapid transformation in the blockchain technology space, what works today may become a significant burden or even obsolete in 3-5 years. In that sense, rapid prototyping with constant iterations is the preferred way of implementing blockchain solutions.

During the Implementation Phase, you should use a systems engineering approach to further refine the detailed design and architecture of the blockchain solution. This may involve evaluating and making or refining decisions regarding at least some of the following:

- Choice of the development platform, technology stack, tools ecosystem.
- Commercial of the shelf versus in-house.
- Open Source versus proprietary solution.
- Cloud, on-premises, or a hybrid deployment architecture.
- As-a-Service solution.
- Network architecture and network registries.
- Business process flows.
- Consensus algorithms.
- Design of blocks, distributed ledger databases, data.
- Design of transactions.
- Design of smart contracts.
- User interface/user experience design.
- Leverage machine learning, artificial intelligence.
- Interoperability with legacy and third-party systems/data.
- Privacy.
- Rules and policy engine.
- Cross-blockchain architecture and interoperability.
- Non-functional requirements.

As described in the Assessment and Readiness phases, the use of Agile and DevOps methodologies are highly recommended for the project management approach. Included in that is the management of the scope for your blockchain solution implementation.

Agile development provides an iterative roadmap where implementation is done in small increments. Achieving incremental gains satisfies stakeholders while enabling you to strategically scale so that you can optimally address pain points, while tackling one priority area at a time, to ultimately accomplish transformational objectives and advance mission goals. This method of continuous integration/continuous delivery begins with building a minimum viable product within a pre-determined timeframe, as referenced in the Assessment Phase.

This Minimal Viable Product MVP should be established during an initial program increment (Increment 0) of 8-12 weeks when the team will finalize the development of the architecture and design of the blockchain solution components. During this time, you should identify the overarching epics and a starter list of product backlog items. These backlog items should include features and architectural enablers. Prioritize the backlog items into 'must have,' 'need to have,' and 'nice to have.'

Next, finalize the schedule for the subsequent program increments and divide the work into sprint iterations. Refine the product backlog frequently to ensure the implementation team gets a clear understanding of the work to be accomplished. As the team completes and deploys increments, ensure continuous stakeholder involvement and team response.

Finalize and Approve Governance Process

As organizations embark on blockchain implementations, it is imperative to take all the network participants along for the journey. In the Assessment Phase, you performed stakeholder analysis to baseline and create a management plan for their level of participation and project support, which you have been executing since.

During the Implementation Phase, it is critical to capture the measurements against your goals for stakeholders. The shift from central ownership of systems to shared ownership takes hold at a new level during this phase. Network partners are now on shared infrastructure actively exchanging information, data, and/or resources. To ensure your project's success and achieve the goals of your stakeholder management plan, make any necessary adjustments to how you are communicating with and engaging stakeholders.

To refine and finalize the governance model and process, you should take into account the changes to the blockchain network affecting each participant's enclave. Also include rules and expectations for each participant to ensure any changes to their enclave do not impair the blockchain implementation.

Security incidents, including information spills that used to just affect one organization, may now impact everyone on the shared infrastructure. Much like before, security practitioners should be continuously monitoring, patching, and security testing—including penetration testing, vulnerability scanning and policy reviews—the systems and applications used by all members. But inside a blockchain network, the results of these efforts should be recorded and published to the other members of the network. Security incidents that occur anywhere in the ecosystem must result in notification and shared documentation and recording of effective investigation, detection and analysis, as well as coordinated action and mitigation regarding containment, eradication, and recovery.

Security is just one area that may have significant impacts to a blockchain implementation. The following areas need to be clearly understood and governance policies must be refined/defined and approved during the Implementation Phase. Further, the change management plan and associated processes might also be updated to address changes in the context of the following:

- Security procedures and approvals.
- Operational and legal risks.
- Governance model/roadmap including the consortium model.
    - Dispute resolution.
- Regulatory compliance.
- Intellectual property.
- Privacy versus transparency.
- Decentralized autonomy.
- Legal enforceability of smart contracts.
- Standards.
- Paradigm shifts.
- Decentralized trust.
- Data sharing.
- Maintenance responsibilities in a decentralized system.
- Accountability and responsibility.
- Onboarding/offboarding procedures.

With the shift to this new decentralized business paradigm in mind and areas examined to update the change management plan, it is now time to refine and finalize the choices associated with the governance model and then implement all the processes and procedures required for operation of the blockchain solution. (A more detailed definition of governance and its components is discussed in Appendix I.)

There are differences between public permissionless versus permissioned blockchain that you should consider regarding governance. The following focus areas exist only for permissioned or can be 'self-regulated' for public permissionless:

- Onboarding / offboarding procedures (non-existent in public).

- Operating cost (non-existent in public).
- Change control mechanism (self-regulating for public).

Key Activities: Technology

Regulations Refined and Met

Blockchain networks are fundamentally different from current technology solutions. Participation includes decentralization and autonomy along with distributed nodes that are not physically or logically in same computing environments.

For ATO of intra-agency and interagency blockchain networks, the blockchain PMO must ensure adherence to and use of FISMA and the Federal Information Security Modernization Act 2014.

In general, you will be required to comply with federal security regulations and guidance covering system architecture best practices, operational processes, infrastructure, reporting, network architecture, configuration management, controls, and continuous monitoring over the entire lifecycle of the technology. However, in case of an agency joining a public permissionless blockchain network, this compliance becomes difficult, as no official governance model exists.

NIST Special publications 800-53 defines access controls, network controls, security controls, risk management, recovery, continuous monitoring, and contingency planning. These security standards cover on-premises datacenters, public cloud, or GovCloud implementations.

In the case of Department of Defense (DoD) systems that include DoD network participants, you must also adhere to DoDI 8500.01, "Cybersecurity," DoDD 8000.01, "Management of the Department of Defense Information Enterprise (DoD IE)," and DoDI 8510.01 and NIST Special Publication 800-37 Rev. 2 regarding the Risk Management Framework (some tailoring is necessary for blockchain). Additional DoD-relevant policies can be found in the Information Assurance Support Environment. The Defense Security Service (DSS) Assessment and Authorization Process Manual (DAAPM) guide is largely applicable, and you may find it extremely helpful in developing your compliant path to ATO.

NIST has also released a draft of Special Publication 800-204, Security Strategies for Microservices based Application Systems, as of March 2019.

Additionally, in nearly all cases, open source software (OSS) is considered commercial software, so the policies regarding commercial software continue to apply to OSS. For DoD, reference DoD CIO memorandum "Clarifying Guidance Regarding Open Source Software (OSS)".

Finalize Business and Solutions Architecture

There are many design options for your network governance model to ensure it meets the business needs of your members. One popular solution has a single agency take the

responsibility of provisioning and managing a complete permissioned blockchain network. Each participant can then request a node and manage their own smart contracts. The participants share the operational costs of the blockchain network.

In another variation, each participant provisions their own nodes, which join a larger blockchain network where the operational cost is funded through the act of transacting. This governance process requires that blockchain participants agree on a voting process and processes about joining or withdrawing from blockchain network.

You should leverage a permissioned blockchain network when the participants have a greater degree of trust, these networks need less computational and network resources to run. Unless these networks have sufficient nodes, these blockchain networks may lack the resilience resulting from distributed and redundant nodes.

A permissionless blockchain network has lower overall management cost, but has a high individual operational cost resulting from each participant provisioning one or many nodes. There are no security governance models in a permissionless public network. Each participant is responsible for securing and managing their node(s). The management costs associated with adding new participant cost is lower.

When looking to solution architecture (infrastructure) and application design, emphasis must be placed on node architecture. This is because the underlying governance framework drives many of the functional requirements for nodes, which can vary greatly.

For example, in one blockchain instance, nodes may be responsible for validating transactions and storing the transactions to the ledger. There might be a mix of light nodes that only make requests and full nodes that validate, store, and act as miners constantly looking for a new block. This computationally intensive activity of full nodes and the decreased capability requirements for light nodes affects the overall development of the node architecture and physical architecture, which are designed specifically to be able to address the functional requirements of the two types of nodes.

Another solution framework uses a different approach to synchronize all ledgers and analyze the validity of transaction. Unlike the case where all nodes replicate the validation activity, only endorsing nodes within a specific channel need to validate the transaction. The rest only read and write blocks once the orderer has satisfied itself with the number of responses that are needed for the quorum. Each node verifies the authenticity of data before committing to the ledger. This significantly reduces the network traffic generated from validating the transaction and also limits traffic to only when a transaction needs to be committed to the ledger. Thus, a node needs lower computational power and network bandwidth.

Blockchain infrastructure should be inherently highly available, a downed node does not impact the transaction process, unless the network does not have enough endorsing nodes on each channel to meet the smart contract's quorum requirements. Once a previously unavailable node becomes available, the blockchain framework under the cover updates the node's distributed ledgers with other nodes. Nodes in a Distributed Ledger Framework are responsible for off-chain data storage and hosting client application. Both these functionalities require a well-designed architecture. The processing order of transactions is important and cannot be modified. This requirement constraints the node architecture's ability to scale horizontally as performance degrades or transactions are queued waiting to validated and written to the ledger.

When considering multi-platform interoperability, blockchain framework implementations and internode communication protocols are substantially different. There are few projects that are trying to create an overarching protocol that hide the underlying blockchain communication protocols. These projects are in infancy stage and are not expected to be available soon. As this is area grows, more insight into this area will be discussed.

There are on-chain versus off-chain considerations throughout the process. Supporting documents to ledger transactions can be large and, for performance and space reasons, these documents are kept off-chain with a hash stored in the on-chain block. Though this is a trivial technical issue, it has major implications to the validity of the business contract, data governance, auditing, and security. The rules around archiving, life term of data, have yet to be worked out. However, the off-chain data is as important as ledger data, and from an infrastructure, governance, and security point of view, need to be handled with same diligence.

Deploy Technology Platform

By building out the governance model and associated policies, you have enabled the automation of many operational functions of the blockchain network once it is deployed.

You should ensure mechanisms are in place for continuous monitoring, analysis, and patching, including for any applicable archived components/data. Among other tactics, this should include vulnerability scanning for patch levels; functions, ports, protocols, and services that should not be accessible to users or devices; improperly configured or incorrectly operating information flow control mechanisms. It should:

- Determine what information about the blockchain ecosystem is discoverable by adversaries (to take corrective action).
- Document results to compare over time.
- Review audit logs to see if any detected vulnerabilities have been exploited and if there are any multi-vulnerability/multi-hop attack vectors.

While the distributed nature of blockchain reduces the complexity associated with disaster recovery and data backups, you should regularly review, test, and refine the contingency plans developed in the Readiness Phase. This should include plans for system failures, disruptions, and compromises.

Ledgers and off-chain data can quickly grow for large networks. Large ledgers require large bandwidth and time to synchronize within nodes. The operations team need good continuous monitoring and alerting framework to detect problems or potential issues well in time. A majority node failure within a small blockchain network is a security concern due to potential lack of quorum or a hijack by a small set of rogue voting nodes.

Additionally, in blockchain, a Denial of Service attack involves submitting more transactions to the blockchain than it can handle. Since many blockchains have fixed size blocks created at a fixed rate and are stored in a distributed fashion, they have a maximum capacity that a determined attacker can exceed, rendering the blockchain unusable.

You should test for alternate storage and processing, recovery, and reconstitution procedures, and more for system-level, node-level, and any external systems that interact with the blockchain.

Also use your detailed documentation of the design for the business functions of the blockchain to do functional testing. This includes creating a transaction, approving a transaction, looking up data, etc. Include security activities and component interactions such as authentication, validating authority to approve transactions, encryption and decryption, and API security. Ensure unauthorized activities across the ecosystem result the prevention of execution, as well as any other desired actions/notifications.

Finalize Operations Model and Policy

Blockchain frameworks are still in evolving with emerging standards slowly showing up in the marketplace. There are frequent announcements of new blockchain frameworks from academia, as well as commercial entities. There will likely be changes to blockchain frameworks and possibly major consolidation in next few years. A new framework(s) may replace current leaders. Your selection of blockchain network and its operating model may affect your implementation. There are three types of operational models:

- **White Label Operating Model** – This popular consumer model allows private brands to off-load all manufacturing logistics and focuses exclusively on design and marketing. This model is equally appropriate for consortiums, enterprises, and exchanges to jumpstart offerings of blockchain-based services. This approach is vulnerable when the supporting vendor is unprepared and not forward thinking. The vendor may downplay future deprecation of underlying blockchain framework, security vulnerabilities, and continuing to use an older unsupported framework that has been supplanted by a faster

and well-supported framework. This model allows the blockchain owner to mitigate risk by transferring the ledger to another vendor. Transferring the ledger is not a simple activity; it will take time and planning.

- **Founder Manager Model** – This model leverages both name recognition and new frameworks quickly. It brings an impartial structure to facilitate collaboration between members of the blockchain platform. The founder has a personal stake in quickly resolving disputes and making the platform succeed. But there are other traits of founder-centric models that may jeopardize success. These usually are issues around weak internal processes, weak security controls, too many ad-hoc decisions, and overconfidence in their own immortality.

- **Third-Party Managed Model** – This is similar to the white label model with a few key differences. There is a need for strong in-house blockchain expertise to enforce management structure and planning. The vendor managing the platform is more focused on day-to-day operations, security, and performance optimization. It is important that contracts include periodic analysis of the long-term viability of frameworks and a roadmap to mitigate market and technology disruptions. From a financial planning, budgeting, and accounting viewpoint, this model can easily split and assign costs to proper buckets.

Implement Security Controls

Though a blockchain ledger is secure from tampering, the data within the blockchain is visible to all nodes. The associated ledger data that is stored off-chain is implemented using traditional means such as databases, filesystems, or content management systems. The off-chain data is sensitive and may also contain Personally Identifiable Information (PII).

As part of FISMA, all agencies are expected to implement security controls based on NIST 800-53 specifications. The document categorizes controls based on the application and data sensitivity. The NIST document looks at security from a larger holistic view which goes beyond the traditional infrastructure and data encryption. Though the NIST document was written for traditional IT systems, the security controls listed in it are generic enough to be applied for blockchain-based systems with some exceptions. Additionally, blockchain-based solutions that are distributed in nature and host smart contracts, require that nodes freely communicate to systems and other nodes, outside the organizational boundary for data feeds or validating real-world facts.

As a best practice, infrastructure-specific security controls need to be baked into Infrastructure As Code (IAC) scripts. Typical data center specific activities, such as applying security patches, encryption of network, creating subnet, setting protocol based firewalls, log maintenance, and code scanning, need to be built into the DevSecOps pipeline. The second group of security

controls are process oriented that cover change management, disaster recovery, and data governance. The blockchain framework and smart contracts are treated like any COTS product.

Blockchain frameworks do not discuss application monitoring, as the blockchain in principle is distributed, and a loss of a node does not bring down the blockchain network. Irrespective of blockchain implementation, the application needs to be available and performing for business continuity. Monitoring instrumentation will have to be built at application level, network level, and operating system level to ensure the blockchain network is functioning and sufficient nodes are available for consensus.

Obtain ATO or IATT

All Federal applications need to go through Authority To Operate (ATO) or Interim Authority To Test (IATT) before they can host any real organizational data. The blockchain core resilience and security may give an impression they need a less stringent regulations. Instead because of the distributed nature of blockchain, extensive use of certificates, legally binding contracts, and presence of sensitive data makes it more important to implement proper security and build over existing FISMA security requirements.

The process of getting of ATO is complex, drawn out, and tailored to every agency. Therefore, it is important to include agency chief security officer at the starting of project. The security officer and project sponsor can help lay out the process, project milestones, list supporting documents, and marshalling all the entities that are needed for ATO.

Key Activities: Acquisition

As you begin to execute in this phase, terms and conditions may evolve via modification as the project proceeds through multiple phases of differing degrees of technological maturity. There may also come a decision point where the expected value of the technology no longer justifies the investment from the government or the awardee, resulting in termination of the contract. Perhaps the technology development yields such success that you may begin planning for a follow-on acquisition.

Administer Contract

At this point, the Contracting Office has completed the procurement of the blockchain technology identified in the Selection Phase. Upon award, the Contract and Program Office establishes the service area readiness and completion criteria for the blockchain-enabled portfolio.

Modify Contract

If enabling the portfolio with the new blockchain technology requires modifications to the scope of the portfolio contractor, the Contracting Office executes the appropriate modifications enabling the portfolio with the new blockchain technology.

Prepare and Award Follow on Contract

A blockchain-enabled portfolio requires a blockchain-enabled enterprise which were identified in the Selection Phase. If the introduction of blockchain technology impacts the scope of additional industry partners within the enterprise, plan and execute all appropriate follow-on acquisitions.

Key Outcomes

Defined

Platform Architecture/IOC, Operational Model, Contract

At the end of this phase, the blockchain infrastructure has been implemented, configured, and customized. Smart contracts and the token system, if needed, have been developed and tested. The documentation and operational model have been defined, recorded, and shared among stakeholders.

Planned

Enterprise Integration, Operations, Information Technology Service Management

At the end of this phase, you should have a finalized plan to integrate the blockchain solution within the enterprise. Also, the operations and maintenance of the blockchain solution and its smart contracts have been initiated and a now finalized plan will guide activities for those components in the Integration Phase.

Phase Outputs

Once implementation is complete, the blockchain will be a full Platform Architecture IOC (Internal Operating Capacity) and if a Smart Contract, a Proof Model will be completed. The business will have a User Guide, Admin Guide, Resources Allocated, and Success Criteria Metrics and Monitoring defined and in place. Specific to the Governance, Risk and Compliance (GRC), the Governance Team/Model will be stood up, the Operations Structure will be implemented and the Cost Structure will be finalized.

Decision Gate

At the end of this phase, you should be able to answer the following questions and decide whether to stop work, iterate, or go to the next phase:

- Is the blockchain infrastructure implemented and operating as required for this iteration?
- Has the blockchain infrastructure been tested and are all security controls in place and performing as expected?
- For smart contracts, is the contract logic implemented, are security controls in place, and have they been thoroughly tested for this iteration?
- Is the documentation and information sharing adequate for this iteration?
- Is the logical access management structure in place, including effective key and digital certificate management infrastructure and access management policies (including expiration and revocation of credentials)?

Glossary

Blockchain (further defined in the ACT-IAC Blockchain Primer)

A blockchain underpins a digital ledger in a peer-to-peer network which timestamps records by cryptographically hashing them into an ongoing chain of blocks, recorded chronologically and publically, forming an immutable record that is distributed across peer nodes.

Colored Coin Protocol / Deployment Model

Colored coin protocols share the user authentication model to associate real assets with addresses on the underlying blockchain.

Custodian

A person or a system that is responsible or looks after something, in the banking environment, custodians are specialized financial institutions responsible for safeguarding an individual's financial assets.

Decentralized Autonomous Organizations (further defined in Appendix H)

Decentralized autonomous organizations (DAOs) are software-based entities whose decisions are generally made electronically by computer code.

Distributed Ledger Technology (DLT - further defined in the ACT-IAC Blockchain Primer)

A peer-to-peer network, which uses a defined consensus mechanism to prevent modification of an ordered series of time-stamped records.

Immutability

Not subject to or susceptible to be changed.

Membership

The ability or permissioned to participate in a blockchain network, usually controlled by a set of rules such as 'PoW' and validated by other members.

Metacoin Protocol / Deployment Model

A metacoin system is a colored coin protocol coupled with a middleware layer in the form of dedicated servers, which verify colored coin transactions.

MGT Act

The Modernizing Government Technology Act is a piece of legislature passed by the United States Congress in 2017 that provides a policy and guidance to modernize technology that supports the Federal Government. https://www.govtrack.us/congress/bills/115/hr2227.

Multi-Asset Blockchain Protocol

A protocol allowing multiple assets to be natively supported by a blockchain.

Nodes
A system that is part of a decentralized peer-to-peer (P2P) blockchain network.

Open-Source
Denotes software for which the original source code is made freely available to the public and may be redistributed and modified.

Permissioned Blockchain (further defined in the ACT-IAC Blockchain Primer)
A private network where permission to write to the blockchain is controlled by a single or a group of organization.

Phase Input
Material or artifacts needed at the beginning of a phase.

Phase Outcome
Derived effect of activities within a phase.

Phase Output
Specific products or deliverables of a phase that can be used in the subsequent phase.

Proprietary
Software owned by an individual or a company (usually the one that developed it), typically with restrictions on its use or its source code (contrast with open-source).

Public Blockchain (further defined in the ACT-IAC Blockchain Primer)
Generally open-source, public blockchains support multiple readers and writers with open read/write capability. Any participant is able to validate to the integrity of the block.

Acknowledgements

The Blockchain Working Group thanks the authors and contributors who provided a tremendous amount of time and good humor to bring Blockchain Playbook for the U.S. Federal Government to completion. The Blockchain Working Group would like to also thank Nancy Delanoche (ACT-IAC), Dylan Yaga (NIST), Wendy Beck (Net Impact Strategies), Michael Soucy (US Air Force), Rick Holgate (Gartner), and Tim Young (Deloitte) who provided invaluable feedback as reviewers.

Authors and Affiliations

This paper was written by a consortium of government and industry. The organizational affiliations of the authors and contributors are included for information purposes only. The views expressed in this document do not necessarily represent the official views of the individuals and organizations that participated in its development.

Jose Arrieta	HHS
Sandy Barsky	GSA
Frederic de Vaulx	Prometheus Computing LLC
Todd Hager	Macro Solutions
Sean Hetherington	Adobe
Bruno Kelpsas	GSA
Jim Keys	Applied Computer Engineering
Venkat Kodumudi	CGI Federal
Brent Maravilla	OMB
Sonia Mundra	Chenega ABS
Kelly Pippin	GSA
Sanjeev Raman	CyberBahn Federal Solutions
Alexander Rebo	IRS
Mike Rice	CornerStone IT
Sherri Sokol	DISA
Arushi Srivastava	NTT DATA
Jon Tame	Deloitte
Jeff Tennenbaum	IBM
Star Vanamali	Publicis Sapient
Andrew Vanjani	GSA
Sudha Venkateswaran	Pyramid Systems
Aleks Zelenovic	Sapient Consulting | Public Sector

Contributors and Affiliations

Lateef Abro	Dun & Bradstreet
Paul Bajinder	Infozen
Padam Damanjit	Macro Solutions
Bill Engel	Rapid Cycle Solutions
Amy Fadida	A.M. Fadida Consulting
Mark Fisk	IBM
Jean Lewis	Sapient Consulting | Public Sector
David Nguyen	United Solutions
Alexander Permison	U.S. Department of the Treasury
Silvana Rodriguez	U.S. Department of State
Liam Speden	RG
John Sprague	NASA
Jamuna Sundararajan	CyberBahn Federal Solutions
Cesar Tavares	Octo Consulting
Michelle White	GSA
Robert Wuhrman	GSA

References

[1] ACT-IAC, "Blockchain Primer: Enabling Blockchain Innovation in the U.S. Federal Government", https://www.actiac.org/system/files/ACT-IAC%20ENABLING%20BLOCKCHAIN%20INNOVATION_3.pdf

[2] Gartner, "Top Trends in the Gartner Hype Cycle for Emerging Technologies, 2017," Gartner, last updated August 15, 2017, https://www.gartner.com/smarterwithgartner/top-trends-in-the-gartner-hype-cycle-for-emerging-technologies-2017/

[3] GSA U.S. Emerging Citizen Technology Atlas, https://emerging.digital.gov/

[4] MGT ACT, https://www.congress.gov/115/bills/hr2810/BILLS-115hr2810enr.pdf

[5] Digital Identity Guidelines, https://pages.nist.gov/800-63-3/

[6] Digital IT Acquisition Professional Training (DITAP) https://techfarhub.cio.gov/initiatives/ditap/

[7] New FAC Specialization Focuses on Digital Services https://www.fai.gov/announcements/new-fac-specialization-focuses-digital-services

[8] Appendix A, Blockchain Types, https://blockchainhub.net/blockchains-and-distributed-ledger-technologies-in-general/

http://www.sepaforcorporates.com/thoughts/difference-between-permissioned-permissionless-blockchains/

https://www.fjordnet.com/conversations/the-trust-trade-off-permissioned-vs-permissionless-blockchains/

https://monax.io/explainers/permissioned_blockchains/

[9] Sidechain, https://genius.com/Adam-back-enabling-blockchain-innovations-with-pegged-sidechains-annotated

[10] Appendix C*, Comparing Blockchain Implementations* A Technical Paper prepared for SCTE/ISBE by Zane Hintzman

[11] Appendix D, Blockchain as a service (BaaS), https://www.computerworld.com/article/3237465/enterprise-applications/blockchain-as-a-service-allows-enterprises-test-distributed-ledger-technology.html

[12] Appendix F, Blockchain Technology Criteria, https://www.researchgate.net/publication/313249614_The_Blockchain_A_Comparison_of_Platforms_and_Their_Uses_Beyond_Bitcoin

[13] FAI's Agile Acquisition 101, https://www.fai.gov/media_library/items/show/81

Appendices

Appendix A – Blockchain Types & Best Fit

Blockchain Types[8]

The following tables describe different types of blockchain

Public/Permissionless Blockchain Any participant is able to become a validator for transactions	
Features	Public blockchain supports multiple readers and writers with open read/write. There is no need to be part of a group or consortium to participate in the network.
	High public verifiability.
	Consistent state of blockchain across all users.
	Potential to disrupt current business models through disintermediation. No middle man or intermediary required as the ledger of transactions and set of programs to update the ledger is shared across every node in blockchain.
	Unrestricted and open membership with access to data. Anyone can join the network, access the transactions, and participate in consensus.
	Slower compared to other types because a large number of designated nodes are involved in validating transaction blocks.
Context	Open source and permissionless so anyone can download the code and validate the transactions (validating integrity of blocks by participating in consensus).
	Anyone in the world can send transactions through the network and expect to see them included in the blockchain if they are valid.
	Anyone can read transactions on the public block explorer and transactions are transparent, but pseudonymous (an encrypted unique 64-character key).
	Public blockchains hold the potential to replace most functions of traditional financial institutions with software fundamentally reshaping the way the financial system works.

Public Consortium Blockchain Pre-selected parties are able to validate transactions	
Features	High throughput and scalability as a relative number of validators is low compared to public blockchain.
	Federated or consortium blockchains operate under the governance of a group (government agencies or financial institutions) which decide the criteria for others to participate in the blockchain network.
	High transaction privacy as any write and consensus access to the blockchain is controlled based on permissions configured by consortium peer nodes.
	The consensus process is controlled by a set of nodes meeting certain pre-defined consensus criteria. For example, a consortium of 15 financial institutions, each of which operates a node and 10 nodes must sign every block in order for the block to be valid and added to the blockchain.
	Tailorable consensus algorithms with flexible chain trust model. For example, a chain may contain 15 nodes, but only 10 may be required to provide consensus to write to the chain.
	Membership with identity-based access (including read/write controls) on data. The identity is decided and controlled by the governing body
	Faster and permissioned read/writes.
	Anyone who meets certain pre-defined criteria can download the code and participate in validating the transactions.
Context	Consortium of government and financial institutions would have one or more peer nodes and each node would have a copy of the ledger and participate in validating the transactions. Other institutions or the general public could be granted limited access, e.g. read-only.

	Potential applications across both financial and non-financial use cases in government or multi-organization blockchain networks allowing controlled access based on individual needs.

Private Consortium Blockchain A single group controls validating transactions	
Features	Higher throughput due to lower number of validators (compared to permissionless blockchains).
	Federated or consortium blockchains operate under the governance of a group (e.g. government agencies or financial institutions) which decide the criteria for other clients to participate.
	High transaction privacy as any access to the blockchain is controlled based on permissions configured by the consortium control node(s).
	The consensus process is controlled by the control group node(s) who also determine who will participate in endorsement and ordering of the transaction blocks. This allows for custom-defined consensus algorithms but client nodes cannot participate in consensus.
	Membership with identity-based access (including read/write controls) on data. For example, a chain may contain 15 nodes, but only 10 may be required to provide consensus to write to the chain.
	Consortium peer nodes have the blockchain ledger and state database with client-only access to blockchain network based on access permissions. Clients will not have access to ledger.
	Faster and permissioned read/writes with only pre-selected nodes determined by administrator nodes in consortium participate in consensus.
Context	Consortium of government and financial institutions would have one or more peer nodes and each node would have a copy of the ledger but only control or administrator nodes could participate in consensus. Select participants could be granted limited access, e.g. read-only.
	Potential applications across both financial and non-financial use cases in government or multi-organization blockchain networks allowing controlled access in a private system.

Which Type of Blockchain Is the Best Fit for My Organization?

The following two tables explore different aspects of blockchain to help figure out which blockchain type is the best fit. The tables have been broken down in two pieces for better readability.

	Need to Store State (Ledger + state)?	Multiple writers	Use online Third Trusted Party always?	Always Trusted Writers	Public Verifiability Required	Settlement Finality (Irreversible)
Permissionless	Yes	Yes	No	Not always	Yes	Yes
Public Permissioned	Yes	Yes	No	Not always	Yes	No
Private Permissioned	Yes	Yes	No	Not always	No	No
No Blockchain	No	No	Yes	Yes	No	No
Blockchain Thread	Yes	Yes	No	Not always	No	Yes

	Censorship	Validators	Assets Suitability	Deployment

	Censorship	Validators	Assets Suitability	Deployment
Permissionless	Anyone can join(Membership with anonymity)	Any node can participate in consensus based proof of work	Suitable for on chain assets (Virtual bearer asset) e.g. , Bitcoin/Ether	Decentralized with consensus among peer nodes
Public Permissioned	Members who fulfill certain criteria can download protocol and have access to blockchain ledger and participate in consensus. Other clients may or may not have ledger copy and access network based on Permissions (Membership with identity)	Any node can participate in consensus based proof of stake	Bearer asset becomes registered asset	Decentralized with consensus depending on blockchain implementation
Private Permissioned	Only Consortium nodes have peer nodes with ledger and pre-selected nodes participate in consensus. Other clients only access network but have no ledger. (Membership with identity)	Pre-selected nodes in consortium participate in endorsement and ordering of transaction blocks in blockchain	Suitable for off-chain assets (securities, fiat, titles)	Decentralized or Centralized with consensus depending on blockchain implementation
No Blockchain	Participants or clients have no ledger or participate in consensus (Membership with identity)	Only trusted validators	Suitable for online/offline asset	Centralized or Distributed databases with no consensus
Blockchain Thread	Membership with identity	Any node which fulfills certain criteria or pre-selected nodes participate in consensus. (Depends on the type of blockchain implementation and use case)	Suitable for on chain assets (Virtual bearer asset) e.g. Bitcoin/Ether	Utilizing just a thread of blockchain. Emphasizing not having to do a full lift and shift but using a thread with existing solutions to get gains in efficiencies, gains in savings and lower risks

Appendix B – Deployment Models and Common Use Cases

Deployment Models Pros and Cons

Multiple, Separate Blockchains for Assets

Each digital asset or a set of assets maintained by the same issuer could potentially have its own blockchain, either permissionless or permissioned. Merged mining allows securing multiple blockchains with the same computational resources. However merged mining in a permissionless environment could be unsafe, as an attacker with enough hash power could deliberately mine empty blocks or otherwise disrupt transaction processing. A permissioned blockchain could be more resilient to attacks, but it would still have a single point of failure in the form of a single transaction processor. From the auditing and regulating points of view, properties of an issuer-managed blockchain could be similar to existing asset management systems. The cost of operating an issuer-specific blockchain (either on-site or using a Platform as a service (PaaS)) could be comparable to traditional asset management systems because of the need to develop end-user applications (such as wallet services with secure authentication, accounting tools, etc.) Additionally, using multiple blockchains could complicate the development of third-party applications and diminish the network effect by requiring additional tools to interact with other digital assets.

Colored Coin Protocols

Colored coin protocols share the user authentication model with the underlying blockchain. However, because the validity of colored coin transactions is not checked by the blockchain network, colored coin protocols lack efficient payment verification methods. Colored coin protocols using Bitcoin blockchain include ChromaWay, Open Assets, and Colored Coins Protocol.

Metacoins

A metacoin system could provide automated order-matching for trading asset pairs, dividend payments, and so on. Metacoin systems may utilize a dedicated cryptocurrency as a means of payment for provided services. Metacoin systems on top of the Bitcoin blockchain include OmniLayer, Counterparty, and CoinSpark.

Multi-Asset Blockchains

Compared to other deployment models, multi-asset blockchains have more space-efficient proofs of ownership, as simplified payment verification could be utilized for all natively supported blockchain assets. On the other hand, known mechanisms of sharing blockchain security (merged mining and blockchain anchoring) pose security risks in permissionless context. The federated governance model puts the greater responsibility on the blockchain maintainers. As the maintainers can effectively determine the state of the blockchain, they could be legally obliged to be able to reverse transactions, freeze funds, etc. by the regulatory bodies. A multi-asset blockchain could be integrated into existing blockchain infrastructure by using sidechain[9] technology. Smart property represents the ownership of real-world objects

with the help of blockchain data. For example, a blockchain-enabled car would operate only if the driver holds the blockchain-based ownership token.

Smart Contracts

User-defined assets could be represented with the help of a smart contract on a smart contract blockchain (e.g. Ethereum blockchain). The contract could store the mapping of the addresses of current holders of the asset to the corresponding balances. These balances could be updated with the help of messages sent to the contract encoding asset transfer or issuance. The contract could use the conventional authorization scheme of the underlying blockchain in order to check transfer and issuance permissions or could specify new rules for asset transactions.

Use Cases of Blockchain Digital Assets

Complex Financial Assets

Digital assets could represent publicly traded financial assets (e.g. securities). These assets require a high level of security, are heavily regulated, and used in business-to-business contexts, therefore requiring permissioned blockchains, at least in the short term. Permissionless blockchains could be useful for novel financial services, such as crowd funding.

Smart Property

Asset ownership could be transferred using a transaction with an input bearing the token. Smart property assets would have slow transaction velocity and would require security before scalability. Therefore, smart property could plausibly be implemented with the help of dedicated ownership protocols on top of highly secure public blockchains, which do not necessarily support the concept of smart property natively.

Electronic Money

Digital assets could represent e-money, such as alternative currencies (e.g. local currencies or in-game currencies) or claims of fiat money. Electronic money pegged to real-world currencies generally have high transaction velocity. Therefore, they would require scalable, high throughput infrastructure provided by multi-asset blockchains. Currencies with lower transaction velocity (e.g. local currencies) could use multi-asset blockchains, colored coin protocols, or metacoins.

Business to Consumer Assets

Digital assets could be used to represent discounts, coupons, vouchers, gift cards, loyalty points, etc. The assets would be issued by a merchant and transferred to buyers during purchases; the merchant would define a transparent set of rules of how assets can be redeemed for goods. A large retailer could issue multiple types of tokens and track their distribution and ownership, which would be useful for analyzing the customer base. Compared to existing implementations, blockchain infrastructure would provide a built-in secondary market for assets (although asset transfer could be restricted with the help of issuance metadata).

Digital Subscription

Digital assets could be used to monetize access to digital resources, such as stream content. Because of the transparency of blockchains, the content provider could easily check when the user's token was issued and whether it is still valid. The provider could issue multiple types of tokens that correspond to various levels of access (read/write or read-only), or access to specific resources or types of resources. Similar to digital subscription, non-transferable digital assets could be useful for role-based authentication.

Digital Democracy

Digital asset coins can be used to implement voting by sending tokens to one of several designated addresses. While the existing digital asset systems are not secure enough to hold government elections, they can be used for voting among shareholders or in contests; in the latter case, a voting process is easily monetized. Permissionless or loosely regulated permissioned blockchains are expected to play a significant role in emerging IoT and consumer-to-consumer markets. Multi-asset blockchain and smart contract blockchains come as a viable alternative for business to consumer and consumer-to-consumer digital asset issuance.

A permissionless blockchain is suitable for on chain assets (virtual bearer assets) whereas in a permissioned permission less blockchain, a bearer asset becomes a registered asset and blockchain maintainers have a greater transparency and control on assets transfer across users compared to a permission less blockchain. A permissioned blockchain is more suitable for off-chain assets (e.g. fiat, securities, or titles).

Appendix C – Popular Blockchain Platforms

Popular open source/proprietary Blockchain/DLT systems[10]

Below are the most popular blockchain platforms in the market at the time of the writing of this Playbook. The table also provides the highlights of the features of these systems. Use these to determine the system that is best suited for your use case.

When deciding whether to use open source over proprietary software, use this table as a guide to determine which suits your organization better, especially if you want to be able to "extend" the software or not. Open source (defined in Glossary) typically allows organizations that use the software to extend as they see fit.

Open / Proprietary	Blockchain	Features
Open source	BigChainDB	• Open source system that "starts with a big data distributed database and then adds blockchain characteristics — decentralized control, immutability, and the transfer of digital assets." • Each write is recorded on the blockchain database without the need for Merkle Trees or sidechains. • Support for custom assets, transactions, permissions, and transparency. • Federation Consensus Model (federation of voting nodes). • Supports public and private networks. • Has no native currency — any asset, token, or currency can be issued. • Set permissions at transaction level. • It is open source. • Consensus mechanism: Federation of nodes with voting permissions.
Open source	Chain Core	• Blockchain platform for issuing and transferring financial assets on a permissioned blockchain infrastructure. • Chain Core runs on the opensource Chain Protocol. Chain Core Developer Edition is free while the Chain Core Enterprise Edition is a commercial product. • The creation, control, and transfer of assets are decentralized among participants on Chain blockchain networks. The operation of the network is governed by a federation — a designated set of entities. The assets on Chain blockchain networks include currencies, securities, derivatives, gift cards, and loyalty points. • Native digital assets — currencies, securities, etc. • Role-based permissions for operating, accessing, and participating in a network. • Support for multi-signature accounts.

Open / Proprietary	Blockchain	Features
		• Support for smart contracts. • Transaction privacy. • Consensus mechanism: Federated consensus.
Open Source	Corda	• Corda is an open source distributed ledger platform with pluggable consensus — "it supports multiple consensus providers employing different algorithms on the same network." • Corda is probably the only distributed ledger platform with pluggable consensus. • No global broadcasting of data across the network. • Querying with SQL, join to external databases, bulk imports. • Consensus mechanism: Pluggable consensus.
Open Source	Ethereum	• Ethereum is a decentralized platform that runs smart contracts on a custom built blockchain. • Ethereum Wallet facilitates holding crypto-assets as well as writing, deploying, and using smart contracts. • Creation of cryptocurrencies. • Creation of democratic autonomous organizations (DAOs). • Command line tools built-in Go, C++, Python, Java etc. • Consensus mechanism: Ethash, a PoW algorithm.

Open Source	Hyperledger Fabric	• Hyperledger Fabric supports the use of one or more networks, each managing different Assets, Agreements, and Transactions between different sets of Member nodes.
		• Hyperledger Fabric's key features include:
		  ○ Query and update ledger using key-based lookups, range queries, and composite key queries.
		  ○ Read-only history queries.
		  ○ Transactions contain signatures of every endorsing peer and are submitted to ordering service.
		  ○ Peers validate transactions against endorsement policies and enforce the policies.
		  ○ A channel's ledger contains a configuration block defining policies, access control lists, and other pertinent information.
		  ○ Channels allow crypto materials to be derived from different certificate authorities.
		  ○ Consensus mechanism: Consensus is ultimately achieved when the order and results of a block's transactions have met the explicit policy criteria checks.

Open / Proprietary	Blockchain	Features
Open Source	Multichain	• Multichain is an open source blockchain platform, based on Bitcoin's blockchain, for multi-asset financial transactions. • Native multi-currency support. • Atomic two- or multi-way exchanges of assets between participants. • Permission management. • Rapid deployment. • Multiple networks can simultaneously be on a single server. • Per-network custom parameters (permitted transaction types, confirmation times, minimum quantities, transaction rate, and size limits). • Data streams. • Consensus mechanism: Distributed consensus between identified block validators. This is similar to Practical Byzantine Fault Tolerance, with one validator per block, working in a round-robin type of fashion.
Proprietary	Chain Core Enterprise	• Chain Core is an enterprise-grade blockchain infrastructure platform for building financial services. • Enables institutions to issue and transfer financial assets on permissioned blockchain networks. • This can be conceived as a novel type of ledger that is shared across entities and enables electronic records to behave like transferable financial instruments, eliminating many of the complex messaging-based systems that are typically involved in clearing, reconciliation, and settlement. • Designed for currencies, securities, and other issued financial instruments. • Role-based permissions for operating, accessing, and participating in a network. • A perfectly auditable record of transaction activity that cannot be forged or altered. • Native integration with hardware security modules, multi-signature support, best-in-class cryptographic primitives, and an auditable, open source stack • Transaction privacy. • Federated consensus designed for immediate transaction confirmation with absolute finality. • Throughput to meet market-scale applications and server architecture designed for high availability. • Assets definitions, compliance data, and arbitrary annotations are included directly in the transaction structure.

Open / Proprietary	Blockchain	Features

Proprietary	DragonChain	• A turnkey proprietary blockchain and smart contract platform.
		• The use cases of Dragon Chain include Identity systems, ticketing, distributed storage, processing, and computing. Dragonchain smart contracts run in a trusted context such that sensitive business data and business logic are not exposed to the network.
		• There are also multiple types of smart contracts in Dragonchain.
		  ○ Transaction smart contract – captures business logic for transaction approve/deny.
		  ○ Broadcast receipt smart contract – allows user to execute code when transactions reach specific level of consensus.
		  ○ Subscription smart contract – allows user to execute code against subscribed transactions/data feed from another node.
		  ○ Cron/scheduled smart contract – allows user to schedule recurring or timed execution of business logic.
		  ○ Library smart contract – allows a node to expose or use reusable utility smart contracts.
		• Based on serverless architecture to enable simple and powerful scaling and allows development in various coding languages (Python, Java, Node, or C++).
		• Currency agnostic platform; applications can be built with or without currency or even with multiple currencies.
		• There is an incubator or marketplace available on this platform for new applications/projects developed on DragonChain platform.
		• DragonChain tokens (also called Dragons) can be used for access to any part of the DragonChain platform, such as spinning up a node, accessing advanced smart contract libraries, access to incubated projects, and early/discounted access to incubated project tokens.
		• The user can control the level of decentralization of business nodes.
		• Some of the benefits of Dragon proprietary blockchain over other blockchains are:
		  ○ Blockchain expertise not required
		  ○ Ease of integration
		  ○ Currency Agnostic
		  ○ Interoperability
		  ○ Protection of Business Data
		  ○ Short fixed length blocks
		  ○ Simple Architecture

Appendix D – Blockchain as a Service (BaaS)

While the Blockchain Primer introduces the BaaS model[11], this Appendix includes detailed analysis and provides the reader with enough information to help decide whether to choose the BaaS model. Many of the leaders in the cloud space have seen the potential benefits of offering BaaS to their customers and have started providing some level of BaaS capabilities. As enterprises look to deploy distributed ledgers, the industry's largest IT providers have launched BaaS, offering a way to test the nascent technology without the cost or risk of deploying it in-house.

The BaaS offerings could help companies who do not want to build out new infrastructure or try to find in-house developers, which are in hot demand.

Microsoft (Azure) –Microsoft partnered with ConsenSys to provide the Ethereum-Blockchain as a Service (EBaaS) in their Azure environment. Offering the service will allow "customers and partners to play, learn, and fail fast at a low-cost in a ready-made dev/test/production environment." Azure BaaS key features are:

- Cryptograhically secure, shared distributed ledger.
- BaaS by Microsoft Azure claims to provide a rapid, low-cost, low-risk, and fail fast platform for organizations to collaborate together by experimenting with new business processes – backed by a cloud platform with the largest compliance portfolio in the industry.
- As an open, flexible, and scalable platform, Microsoft Azure makes it easy to spin up the blockchain of your choice, including leading platforms such as Ethereum, Quorum (EEA), Hyperledger Fabric, R3 Corda, and Chain Core that address specific business and technical requirements for security, performance, and operational processes.
- They additionally claim that their intelligent services, such as Cortana Intelligence, are able to provide unique data management and analysis capabilities unlike any other platform offering.

IBM (Bluemix) – IBM is offering BaaS using the Hyperledger. The IBM release stated that, "using IBM's new blockchain services available on Bluemix, developers can access fully integrated DevOps tools for creating, deploying, running, and monitoring blockchain applications on the IBM Cloud."

Amazon (AWS) - Amazon, in collaboration with the Digital Currency Group (DCG), one of the largest investors in blockchain firms, is providing BaaS to members of DCG's portfolio so they "can work in a secure environment with clients who include financial institutions, insurance companies, and enterprise technology companies."

The BaaS option provides users a low-cost opportunity to use blockchain services offered by IT vendors. The enterprise user of BaaS need to connect with the peer nodes deployed over cloud provided by the IT vendor of choice and leverage blockchain service offerings by the IT vendor, to form their enterprise P2P chain network.

Appendix E – Platform Resource Requirements

At the time of writing , the following generic platforms are used to develop blockchain solutions and are described in detail below:

- Foundation Blockchain Platform
- Ethereum Development Ecosystem
- Hyperledger Ecosystem (Refer to the Linux Foundation Website for details)

Additional technology frameworks common among the blockchain community are listed below and described in subsequent sections. This list is not entirely inclusive as technologies, languages, framework, and packages are added to the growing list daily.

- Angular
- Webpack
- Swarm
- Inter-Planetary File System (IPFS)

Foundation Blockchain Platform

There are several existing networks such as Bitcoin, Ethereum, or Hyperledger that can be used to build DApps. Ethereum and Bitcoin are both decentralized - public chains that are open source - while Hyperledger is used for private chains and open source.

The foundation Blockchain Platform was the original platform built as envisioned by Satoshi Nakamoto in his paper that described Bitcoin and its underlying technology. Bitcoin may not be a good choice to build DApps, as it was originally designed for peer-to-peer transactions and not for building smart contracts.

Ethereum Development Ecosystem

Like the Linux Foundation's Hyperledger, the Ethereum Ecosystem is a continuously growing open source set of tools and technologies that provides a one-shop stop for blockchain applications. Some current list of tools in the Ethereum Ecosystem are defined below:

Ethereum Ecosystem Tools	Information
Solidity	• An object-oriented language that developers can use for writing smart contracts. The best part of Solidity is its potential use across all platforms – making it a top choice among developers. • Similar to JavaScript and more robust than other languages. • At the moment, Solidity is the language that is getting the most support and has the best documentation.

Ethereum Ecosystem Tools	Information
Serpent	Prior to Solidity, Serpent was the primary language for building DApps. It still retains value in DApps construction as real-time garbage collection.
Geth	• The official client software provided by Ethereum. • Written in the Go programming language. • Components include: **Client Daemon** • Kept up to date by connections and communications to other nodes. • Has the ability to mine blocks and add transactions to the blockchain. • Validates the transactions in the block and also executes the transactions. • Acts as a server by exposing Application Programming Interfaces (APIs) to interact via a Remote Procedure Call (RPC). **Geth console** • Command line tool all running node. • Performs various actions such as: ○ create and manage accounts, ○ query the blockchain, and ○ sign and submit transactions to the blockchain. **Mist Browser** • Mist Browser is a desktop application used to communicate with the blockchain. • It can be considered as the Graphic User Interface (GUI) for the Geth console, since it supports the actions performed through the Geth console.
Parity	• An Ethereum client that integrates directly into a web browser providing access to all the features of the Ethereum network, including DApps. • A full node wallet, which stores blockchain data on a local computer. • Allows the following: ○ Access to basic Ether and token wallet functions. ○ Creation and management of Ethereum accounts. ○ Management of Ether and Ethereum tokens. ○ Creation and registry of personal tokens. ○ Has several features that benefit private or consortium settings. ○ Fast transaction processing. ○ Proof-of-Authority consensus engines. ○ Privacy and control features. ○ Variety of deployment solutions. ○ Ability to augment features.
Remix	• An Integrated Development Environment (IDE) for Solidity, which includes an integrated debugger and testing environment. • Allows the following: ○ Develop smart contracts (Remix integrates with the Solidity editor). ○ Debug smart contract execution. ○ Access the state and properties of a deployed smart contract. ○ Debug a committed transaction.

Ethereum Ecosystem Tools	Information
	○ Solidity code analysis, which reduces code mistakes and to enforces best practices. • With Mist (or any tool which inject web3), it can be used to test and debug DApps
MetaMask	• A chrome plugin used to interact with the Ethereum node. • Allows Ethereum DApps to run in a browser without running the full Ethereum node. • Includes a secure identity vault, providing a user interface to manage identities on different sites and sign blockchain transactions.
Embark	• Is a framework for DApps that handles compiling, deploying, and interfacing with contracts • Integrates with Ethereum Virtual Machines (EVMs), Decentralized Storages, (e.g. IPFS) and decentralized communication platforms (e.g. Whisper and Orbit) • Is supported for deployment • It allows blockchain developers to develop and deploy DApps easily, or even build a serverless HTML5 application that uses decentralized technology. It equips developers with tools to create new smart contracts, which can be made available in JavaScript code • Features of the framework include: ○ Decentralized Storage (e.g. IPFS) ▪ Data storage and retrieval on DApps via EmbarkJS, which includes uploading and retrieving files ▪ Full application deployment to IPFS or Swarm ○ Decentralized Communication platforms (e.g. Whisper and Orbit) ▪ Easily sends and receives messages via Whisper and Orbit P2P network channels ▪ Web Technologies • Integrate with any web technology • Use dynamic build pipelines and tools
Truffle	• Truffle is another Framework for DApps • Features include: ○ Built-in smart contract compilation, linking, deployment and binary management. ○ Automated contract testing and rapid development. ○ Configurable build pipeline with support for custom build processes. ○ Scriptable deployment & migrations framework. ○ Network management for deploying to public & private networks. ○ Interactive console for direct contract communication. ○ Instant rebuilding of assets during development. ○ External script runner that executes scripts within a Truffle environment.

Additional Technology Frameworks	Information
Angular	• A platform that makes it easy to build applications with the web, mobile, and desktop. • Combines declarative templates, dependency injection, end to end tooling, and integrated best practices to solve development challenges. • Can be used for developing the front-end pages for the DApps
WebPack	• An open source JavaScript module bundler. • Takes modules with dependencies and generates static assets representing those modules ○ Bundles ES Modules, CommonJS, and AMD modules, ○ Can create a single bundle or multiple chunks that are asynchronously loaded at runtime reducing the load time. ○ Dependencies are resolved during compilation, reducing the runtime size. ○ Loaders can preprocess files while compiling, e.g. Typescript to JavaScript, Handlebars strings to compiled functions, images to Base64, etc. ○ Highly modular plugin system to do whatever the application requires.
Swarm	• Swarm is a distributed storage platform and content distribution service, a native base layer service of the Ethereum web 3 stack. • Designed to deeply integrate with the devp2p multiprotocol network layer of Ethereum as well as with the Ethereum blockchain for domain name resolution, service payments, and content availability insurance. • Provides a sufficiently decentralized and redundant store of Ethereum's public record, in particular to store and distribute DApps code and data as well as block chain data. • Offers peer-to-peer storage and serving solution.
Inter-Planetary File System (IPFS)	• A protocol designed to create a permanent and decentralized method of storing and sharing files. ○ A distributed file system that seeks to connect all computing devices with the same system of files. ○ A content-addressable, peer-to-peer hypermedia distribution protocol. ○ Defines a content-addressed file system. ○ Coordinates content delivery. ○ Combines Kademlia + BitTorrent + Git. ○ Is a web that can be used to view files accessible via HTTP at http://ipfs.io/<path> like the web. ○ Uses cryptographic-hash content addressing. ○ Is a P2P network. ○ Has a name service: IPNS, an SFS inspired name system. ○ Work in progress to integrate domain naming service with IPFS URL to give meaningful URLs.

Appendix F – Blockchain Technology Criteria[12]

Blockchain Scalability and Volume

The scalability of a particular blockchain network type determines the maximum transaction throughput (number of transactions processed per second) and the maximum volume of transactions that can be processed within a reasonable performance criteria, with a growing blockchain.

The scalability of a blockchain is impacted by the volume of transactions processed on blockchain, size limit of a block, size of entire blockchain, number of verifying nodes to provide consensus, time taken to process a transaction, high processing fee to be paid for transactions processing, etc.

A public permissionless blockchain can have unlimited number of participants to join in network and perform read and write transactions without any censorship resistance and thus the blockchain scalability is majorly constrained due to very large volume of transactions and big blockchain size.

Bitcoin blockchains are much less scaled compared to Ethereum network as there is a built-in hard limit of one megabyte per block (10 minutes to mine a new block) compared to Ethereum which takes maximum 20 seconds. Furthermore, there is a cost to performing certain actions on the public Bitcoin or Ethereum networks. BTC spending per transaction is high in Bitcoin compared to 'gas price' per transaction processing required in Ethereum.

Permissioned blockchains have comparatively much lesser blockchain size as participation is controlled and consensus is done using identified selected blockchain notary nodes. The consensus delay is much lower than that in public blockchains. The scalability of a blockchain grows linearly with addition of more hardware. Thus, permissioned blockchains can better scale up by using more storage and addition of peer nodes in P2P network.

Upgradability

What is the record of accomplishment of the developers for delivering enhancements and upgrades to the blockchain?

Tools needed to verify transactions may change over time and thus the steps and associated cost to upgrade those tools should be a consideration. The ability to keep up with changes will be dependent on the ability to accomplish enhancements and upgrades to the blockchain without disrupting or corrupting the blockchain itself. While blockchain applications appear endless, the software security and manageability procedures are a significant concern for future concept design.

Upgradability of blockchain as a service could be costly. This is especially true if existing platforms cannot keep pace with or are incompatible with emerging blockchain or middleware technology.

Due to the volatile Financial Technology (FINTECH) market, new technologies could be cost prohibitive, and with their own inherent security risks. This could be one of the biggest and most challenging concerns in using a technology like blockchain and the investment which may lose its luster if it costs too much to afford the upgrades.

Speed and Latency

The blockchain speed is the transaction throughput (maximum number of TPS) which is determined by the block size and the consensus delay. It does not matter if a blockchain is public or private, the speed of each transaction will be based on the processing power of the network in which the algorithm is placed, and the particular type of encryption protocol. As public blockchains have larger number of verifying nodes that are involved in verifying the transactions, the consensus delay is much higher compared to permissioned networks where consensus is achieved by lessor number of verifying nodes (blockchain notaries or identified incentivizing nodes) and has much less latency due to consensus delay and thus high speed. Most of the permissioned networks implement Byzantine tolerance consensus protocol which does not require consensus from every participating node and provide high transaction processing speed.

The size of the block is what makes the difference. By decreasing the size of the transaction or packing more transactions into one block, the faster and more processing power that will be behind it. The speed of transaction processing on a particular blockchain is further determined by any hard limit set on maximum block size. Bitcoin has a hard limit of one megabyte on a block set which causes a latency of 10 minutes compared to 10-20 seconds of latency in Ethereum network. The ability to quantify how you validate the speed of any transactions appears to still be subjective on the network, the size of the block, etc.

Security and Immutability

The documented level of confidence of security within the blockchain is high. The blockchain itself is inherently resistant to threats while the off-chain applications are not. The blockchain is a mathematically certain way to protect data in both the public and private applications. This certainty is accomplished with the use of the three basic science and mathematical concepts include hashing, keys, and digital signatures.

A public blockchain network is completely open and anyone can join and participate in the network. The blockchain network typically has an incentivizing mechanism to encourage more participants to join the network. Bitcoin is one of the largest public blockchain networks in production today. In a public-facing blockchain, administrators or those responsible for the upkeep and management of the blockchain, must have the ability to see the transactions “in-action” as they are taking place. A blockchain, by design, allows a database to be shared

between entities who do not fully trust each other, without central administration. All blockchains suffer from the same fundamental issue, the content of every transaction is revealed to every participant. This transparency is necessary in order to verify a transaction by every node associated with the blockchain. Conversely, conventional/centralized database transactions are only visible to creators and administrators.

One of the drawbacks of a public blockchain is the substantial amount of computational power that is necessary to maintain a distributed ledger at a large scale. More specifically, to achieve consensus, each node in a network must solve a complex, resource-intensive cryptographic problem called a PoW to ensure all are in sync and thus immutability is very high. The disadvantage is the openness of public blockchain, which implies little to no privacy for transactions and only supports a weak notion of security. Both of these are important considerations for enterprise use cases of blockchain.

A private blockchain network requires an invitation and must be validated by either the network starter or by a set of rules put in place by the network starter. Businesses who set up a private blockchain, will generally set up a *permissioned* network. This places restrictions on who is allowed to participate in the network, and only in certain transactions. Participants need to obtain an invitation or *permission* to join. The access control mechanism could vary: existing participants could decide future entrants; a regulatory authority could issue licenses for participation; or a consortium could make the decisions instead. Once an entity has joined the network, it will play a role in maintaining the blockchain in a decentralized manner. Thus immutability of DLT in permissioned blockchains is largely controlled by set of access permission rules and the industry level protocols to achieve consensus with known number of verifying nodes and also most of the participants are trusted users only. Data privacy is better managed by defining read and writes level access permissions for each user in permissioned blockchains.

Storage and Structural Needs

The permissionless blockchain has always untrusted participants and to maintain immutability of transactions stored in blockchain, consensus is required by all participant nodes involved in the blockchain thus the distributed ledgers are shared with all complete blockchain blocks downloaded in a decentralized manner at all participant nodes with greater computational power needed to validate the transactions. Thus, blockchain size is too large and more number of transactions are processed. Thus, the scalability of permissionless blockchain is managed by adding more storage and processing servers in permissionless blockchains compared to permissioned blockchains where participation is controlled and the consensus size is less as all participant nodes are not required to validate transactions to ensure immutability of distributed ledger but only selected nodes does the transaction validation.

The following table lists the common use cases that are suited for each type of blockchain.

Primary Purpose	Type of Blockchain

Primary Purpose	Type of Blockchain
Move value between untrusted parties	Public
Move value between trusted parties	Private
Trade value between unlike things	Permissioned
Trade value of the same thing	Public
Create decentralized organization	Public or permissioned
Create decentralized contract	Public or permissioned
Trade securitized assets	Public or permissioned
Build identity for people or things	Public
Publish for public record keeping	Public
Publish for private recordkeeping	Public or permissioned
Perform auditing of records or systems	Public or permissioned
Publish land title data	Public
Trade digital money or assets	Public or permissioned
Create systems for IoT security	Public
Build system security	Public

There may be exceptions depending on project and it is possible to use a different type of blockchain to reach the project goal.

Operational Considerations, Tools, and Monitoring

The blockchain protocol defines three functional roles an entity can play on a blockchain network:

- Asset Issuers – define and issue digital assets.
- Account Managers –Custody and transfer assets.
- Observers – receive blocks and view blockchain data but do not create transactions.

Corporations, brands, merchants, and governments can act as asset issuers. Custodians and banks can transform into account managers on a blockchain network. Meanwhile regulators and risk managers can reinvent their roles with real-time insight and perfectly auditable records.

Any entity running a blockchain network can participate in one or multiple of these roles. The firm that launches a blockchain network in market, is called as operator of that blockchain. Exchanges, brokers, payments networks or government agencies are examples of entities that adopt the responsibility of network operators.

Network operators perform following four functions on a network:

- Determine who can participate in the network.
- Gather signed transactions from participants.
- Generate and sign blocks of these valid transactions.
- Distribute blocks to participants.

A block is valid when it is signed by a quorum of block signers in a process called federated consensus. All members of the network know the identities of block signers and accept blocks only if they have been approved by a threshold number of block signers.
Each network participant can also cryptographically validate the whole chain of transactions. This consensus process ensures that competing transactions are resolved and guarantees that transactions are final. In order to operate or participate in a blockchain network, an entity runs a node in the network.

Implementation can be based on open source blockchain protocol or using proprietary blockchain platform or services. The nodes in the permissioned network are designed to run in enterprise IT environments. In case of public permissionless blockchain implementations, complete blockchain node chain is deployed on participant machines in decentralized manner and each machine acts as node connected with other nodes to form Internet of Value.

Any blockchain network operator manages following layers of implementation:

- Storage Layer – stores global blockchain data as well as local account data.
- Services Layer - services layer is on top of storage layer that allows creation of assets and transactions.
- Communication layer – consists of API that connects applications and link nodes together.
- SDK layer – allows developers to create applications on top of stack.

The major considerations in operating a blockchain network are security, performance throughput and scalability.

- Proper network management and rotation of key material is required to secure digital assets.
- Industry standard hardware security modules (HSM) technology should be integrated with blockchain network to secure the blockchain nodes and transaction signing. Multi-signature accounts using independent HSMs could further increase security.
- The deployment model of blockchain network should be based on vertical scaling of server resources as well as scaling blockchain horizontally over several servers, deployed across many data centers. The linear scale out strategy for increasing scalability by addition of more hardware would provide a very scalability of blockchain network with a high availability.
- The communication and service layer should follow stateless architecture so that high availability could be simply achieved by simple addition of active redundant servers.

- All new features should undergo rigorous performance testing and optimization to ensure optimized resource utilization and high throughput.
- The high availability of storage layer should be achieved with a combination of synchronous and asynchronous replication together with a simple failover scheme.
- The application requests should be load balanced across the communication and services layer and data should be replicated and sharded across storage layer to achieve high performance and high availability.
- Blockchain platforms are not just data management platforms but need to be integrated with enterprise integration adaptors and identity management platforms to provide specialized DApps based functionalities built upon blockchain network. So there is a need to implement EAI patterns based interoperability standards in designing middleware for blockchain applications.
- Tool based monitoring of blockchain network is important to monitor blockchain transaction activities and detect any suspicious or maligning activities.
- Using blockchain explorers (Etherscan.io, Etherchain.org, Digix, Augur, etc.) for quickly checking transaction or a specific smart contract activity is acceptable. But it gets complex when you want to do real monitoring on the long run as:
    - There is no control on what is scanned or what information.
    - The service is not local, so you are at risk any moment the service is not available.
    - Since these explorers take the task of monitoring and reporting activity about the whole blockchain you will end up with some restrictions. Etherscan, for example does not process requests that return more than 10,000 transactions.
    - The solution is to create a local tool that can run on blockchain network or server that will monitor specific addresses you specify and return the whole activity they conduct.
- High availability and disaster recovery planning should be provisioned to ensure the critical service availability during network failure.

The blockchain operator should maintain a robust and up to date and internal knowledgebase repository and resources with right skillset to manager blockchain network and operations effectively.

Performance Considerations – Benchmarks

Most of the blockchain vendors claim performance of blockchain network in terms of TPS. According to the claim of fabric, 100,000 TPS is the aim to achieve if there are about 15 validating nodes running in proximity in Hyperledger fabric blockchain network. However as per the results of past performance stress tests done in fabric environment using the simplest example of running chaincode on 4 peer nodes running on different servers in close proximity, query performance for each peer could reach 7000 QPS per second, while the simple invoke performance for each peer was only 700 TPS (benchmark hardware environment: Intel(R) Xeon(R) CPU E5-2620 v3 @ 2.40GHz 64G DRAM 1T SATA Disk).

Blockchain throughput is linearly scalable by addition of more peer nodes. However even if the throughput could be linear scalable, the peak TPS of current system would be only 10,000 on P2P networking of 15 nodes, which is only 1/10 of the claim made by fabric. This due to the fact that the overhead of PBFT consensus would grow exponentially with the increasing number of nodes offsetting the linear scaling factor of blockchain throughput with addition of more nodes.

The TPS performance of blockchain is largely impacted by following three factors:

- Total Number of nodes in blockchain network as blockchain scalability linearly scales with addition of more nodes in P2P network.
- Total number of nodes involved in consensus to validate a transaction as consensus delay exponentially grows with more number of nodes participating in consensus validation.
- Type of consensus protocol used in validation of transaction by verifier nodes.

The consensus delay is the most impacting factor in determining the performance of a blockchain network. For example, it's evident that scaling the number of nodes in a broadcast network using a probabilistic consensus protocol such as PoW presents an enormous scaling barrier.

This has motivated a number of platform builders, including R3, to consider "performance and scalability" in their platform designs. For example, Corda limits the consensus interaction to only the parties involved in a particular transaction, along with the consensus pool needed to verify uniqueness, and validate the contract if requested. Other platforms such as Hyperledger Fabric V1.0 have also taken a bespoke approach to minimizing transaction sharing. Of course, the primary reason for restricting transaction sharing is "privacy" under the principle "the best way to keep a secret is to not share it." However, this policy does also provide ancillary performance benefits. Some might debate the loss of network resiliency in such a restrictive model.

Some of the blockchain network vendors like Corda delegates the task of validating transactions to pool of selected notes (Consensus Notary pools). These Notary pools provide a uniqueness service by operating consensus over uniqueness by nodes operated by a set of distrusting entities.

A notary consensus pool could differ by the protocol configuration, and by their size (number of notary nodes in the pool), and their location (for a given pool, notary node location could be in any geographic location) which may impact the performance of a blockchain network.

Blockchain's performance is further determined by the number of transactions in a block (block size) and the time between blocks (dwell time). Playing with parameters by increasing block

size, decreasing transaction size, or dwell time can provide a significant one-time boost and optimize blockchain for today's network.

The performance of a blockchain application is also determined by the architecture of storage, services and interoperability layer, and the capacity of the hardware used and the network used to connect the peer nodes.

The mining volume is an additional constraint for Ethereum, as serialising mining as Bitcoin does, limits the number of computations per block. Sharding an Ethereum chain might improve its performance as it would enable smart contracts to be processed in parallel.

Open source Hyperledger "Caliper" project could be used to conduct performance benchmarking on a given blockchain network, before deciding on a particular vendor choice. The tool is designed for Hyperledger but is platform agnostic and can be used with any other blockchain network.

Blockchain Notaries

An asset issuer using blockchain infrastructure is not generally required to process transactions or to write data to the blockchain – this task could be delegated to blockchain notaries. Notaries could be either known entities (in permissioned blockchains), or any users satisfying technical capabilities imposed by a blockchain consensus algorithm (in permissionless blockchains).

Permissioned blockchains could be more beneficial for financial institutions in the short term because of the flexibility of the blockchain specification and increased compliance. On the other hand, permissionless blockchains could prove more attractive for consumer-to-consumer markets and IoT applications because of inherent trustlessness and permissionless entry and exit.

Blockchain notaries get revenue incentives by keeping blockchain safe e.g. by running services in top of it. For Chain protocol, the Consensus Notary pools (e.g. RAFT, BFT-SMaRT, etc.) provide a unique service by operating consensus over uniqueness by nodes operated by a set of distrusting entities.

A notary consensus pool could differ by the protocol configuration, and by their size (number of notary nodes in the pool), and their location (for a given pool, notary node location could be in any geographic location). The size of notary consensus pool determines the performance (TPS) of a blockchain as it directly impact the consensus delay in verifying the transactions.

Blockchain Network

A public blockchain network provides three security modes for constituent nodes:

- **Full verification nodes** that verify and store every transaction circulating in the network. This security mode could be used by blockchain notaries, regulators, auditors, analytical services and dedicated "blockchain as a service" providers
- **Simplified payment verification (SPV) nodes**, which would be used by a vast majority of end users, as this security mode requires little computational resources and memory space
- **Partial verification nodes** made possible with the help of segregated witness and fraud proofs. These nodes could verify a small percentage of transactions (e.g. 1%), while contributing to the overall security of the blockchain network. Partial verification nodes could be operated by service providers on the blockchain

In the case of a blockchain with restricted read access, the architecture of the blockchain network would be determined by transaction processors. For example, transaction processors could operate full nodes, and all other users could be provided to concerning transactions either through SPV network nodes or through equivalent web application interfaces. Thus, blockchains with restricted access could be less scalable or reliable because of uneven distribution of transaction processing.

There is an important distinction between SPV nodes and web API access to blockchain data. While SPV nodes do not increase the security of the blockchain network, their use together with the publicly available chain of block headers could provide uniqueness of blockchains and immutability of data as any change would not be accepted by SPV nodes. Alternatively, same thing could be achieved by PoW consensus. In the case that access to the blockchain is provided via web APIs without disclosing the blockchain structure, reliably proving uniqueness and immutability becomes more difficult. Even if the regulator or an auditor would have complete access to the blockchain (e.g. by operating a full verification node), data provided to the regulator could differ from data served via API as a result of an eclipse attack performed by colluding blockchain notaries.

User Authentication and Authorization

User authorization in blockchain is performed using public key cryptography. In the simplest case, blockchain-based assets are bearer assets (i.e., the ownership of an asset is determined by the knowledge of a private key.) Two-factor authentication or other security measures comparable to those of centralized e-money systems could be implemented by using dedicated wallet services.

Security properties of public key cryptography could be boosted by the use of specialized *hardware wallets* for signing transactions. In order to maintain user privacy, blockchain users could utilize hierarchical deterministic wallets and the pay-to-contract protocol.

In the case of more complex transaction models, e.g. for smart contracts, zero knowledge proofs, and secure multi-party computations could be used in order to execute contracts while not disclosing data to any of computers.

Asset Issuance

As asset issuance is a special type of transactions, the identity of the issuer could be determined according to the general user identification rules (using the blockchain-based PKI or other techniques).

A regulatory body could explicitly acknowledge asset issuance by co-signing the corresponding transaction together with the issuer, or by granting the issuer a special kind of the digital certificate.

An asset could be marked as ***locked***; meaning the assets of the same type cannot be issued in the future by anyone, including the initial issuer. This type of assets is useful (e.g. for creating non-dilutable shares).

- An asset could be marked as ***divisible*** to several decimal places (cf. with Bitcoin, which is divisible to 8 decimal places).
- An asset could be made ***non-transferable*** in order to limit secondary market (e.g. due to regulation requirements).
- Additional metadata could be associated with the asset, either directly or in the form of a hash commitment. In the second case, off-chain data could be retrieved with the help of distributed hash tables, e.g., implemented using IFSC or Bit Torrent protocol. Metadata could be useful in implementing event tickets, for example.

Appendix G – Sample Friendly Contract Vehicles – Refer to GSA Atlas

When procuring blockchain or DLT, agencies should consider agile acquisition methods. Agile acquisition advocates rolling out capabilities in smaller chunks, more frequently. It also deemphasizes extensive upfront capabilities planning. Instead, developers put capabilities into action as soon as possible, then modify and adapt them as needed. Agile acquisition allows programs to be more responsive to changes in operations, technology, and budgets. It also offers more opportunities for collaborating with users and other stakeholders to deliver priority capabilities rapidly. Implementing agile development practices often requires changes in an agency's policies, processes, and culture. But the rewards are ample.

Traditional procurement methods for waterfall software implementations lack the flexibility to take advantage of the benefits of time, schedule, and cost that agile software development methods bring to the Federal Government. For this reason, the acquisition workforce needs to make its processes agile, using innovative and creative solutions to procure IT services while maintaining compliance with the Federal Acquisition Regulation (FAR) and Federal law. For more information on agile acquisitions, please see FAI's Agile Acquisition 101[13].

The table below provides a list of possible acquisition vehicles that can be used to procure blockchain:

Acquisition Type	Information
Stand-Alone Acquisition	Using the procedures identified in FAR Part 12 Acquisition of Commercial Items, FAR Part 13 Simplified Acquisition Procedures, or FAR Part 15 Contracting by Negotiation. Competition can be categorized in three ways, in accordance with FAR Part 6, Competition Requirements: 1. Full and Open Competition: All responsible sources can compete. 2. Full and Open Competition "After Exclusion of Sources": This type of acquisition is reserved for competition among a specific type of business concern. 3. Other Than Full and Open Competition: Sources are limited based on one of the seven reasons listed at FAR 6.302. This generally requires a justification and approval (J&A) in accordance with FAR 6.303 and 6.304.
Use of SBA's 8(a) Program	Allows procuring agencies to make quick, direct awards for procurements up to $4 million in value or $22 million for Alaskan Native Corporations (ANC). Benefits: • May select the awardee through market research or capabilities briefings and award directly to the firm without further competition. • A sole source J&A is not required for contracts under $4 million or $22 million for ANC contracts. • Awards are not protestable. Options: • SBA's TechFAR Hub memo in conjunction with US Digital Service highlights the use of the 8(a) Program for digital services and clarifies that procuring agencies which award digital services requirements through the 8(a) Program need not request release from 8(a) competition when awarding digital service developmental iterations or add-on services. The memo identifies each iteration or add-on is a distinctly new project and should be treated as a new requirement for purposes of 8(a) release requirements and permits agencies to award contracts for additional development utilizing different acquisition strategies (to include non-8(a) strategies) without requesting release from 8(a) competition. • GSA's 8(a) STARS II government wide acquisition contract (GWAC) leverages SBA's 8(a) Program authority 8(a) allowing directed task orders up to $4 Million, including options.

Acquisition Type	Information
	Orders more than $4 Million must be competed among the industry partners in your chosen constellation and functional area.
Leverage of Interagency Vehicles	• GSA FAS Schedule 70 Contracts: Long-term government wide contracts with commercial companies to provide access to commercial products and services at volume discount pricing. GSA Schedules are Indefinite Delivery, Indefinite Quantity contracts that provide for an indefinite quantity of supplies and services during a fixed period of time. The contract has a five-year base period with (3) fiver year options resulting in a potential 20 year contract. IT Schedule 70 is the government's largest IT contract vehicle that delivers federal, state, and local customer agencies the tools and expertise to shorten procurement cycles, ensure compliance, and acquire the best value for innovative technology, products, services, and solutions. Schedule 70 includes the FAStlane and Startup Springboard programs that are part of the Making It Easier initiatives. • 8(a) STARS II Government wide Acquisition Contract (GWAC): offers access to highly qualified, certified 8(a) small disadvantaged businesses. The contract has a $10 Billion program ceiling with a five-year base period and one five-year option. • Alliant and Alliant Small Business Government wide Acquisition Contracts (GWAC): represent the next generation GWAC vehicles for comprehensive information technology solutions through customizable hardware, software, and services solutions purchased as a total package. • VETS 2 Government wide Acquisition Contract (GWAC): the successor to the VETS GWAC, VETS 2 is a service-disabled, veteran-owned small business set-aside that provides access to customized IT solutions from a diverse pool of industry partners. • NIH Chief Information Officer Solutions and Partners 3 (CIO-SP3) and CIO-SP3 Small Business Government wide Acquisition Contract (GWAC): NIH's three GWACs for information technology procurement. CIO-SP3, CIO-SP3 Small Business, and CIO-CS can be used by any federal civilian or DoD agencies to acquire information technology products, services, and solutions.
Other Transactional Authority	Other Transaction Agreements (OTAs): generally do not follow a standard format or include terms and conditions required in traditional mechanisms, such as contracts or grants. Meant to help meet project requirements and mission needs. The statutory authorities for most agencies include some limitations on the use of their agreements, although the extent and type of limitations vary. Benefits: • OTAs make it easier to work with nontraditional partners such as start-up companies) • Flexible and streamlined method for procurement which can reduce the time and cost of delivery of technological advancements while improving capabilities • Ability to address industry concerns regarding intellectual property and cost accounting provisions that would otherwise need to be included when using traditional mechanisms • Opportunity to tailor some terms and conditions of agreements as needed when working through the agile development of a capability • Not considered procurement contracts, grants, or cooperative agreements even though considered legally binding instruments so they are not subject to the Federal Acquisition Regulation, the Competition in Contracting Act, the Truth in Negotiations Act, or many other federal contracting regulations Note: Consult with your agency's legal counsel to determine which laws and regulations are still applicable and which provisions for intellectual property are suitable.
NTIS Joint Venture Program	NTIS' basic ATO a permanent clearinghouse of scientific and technical information is codified as chapter 23 of Title 15 of the United States Code (15 U.S.C. 1151-1157). This chapter also established NTIS' authority to charge fees for its products and services and to recover all costs through such fees "to the extent feasible."

Appendix H – Decentralized Organizations

Today's business model is typically based upon a centralized hiearchy. This conflicts with the value of digital platforms. Team members become burdened with having to assist manual processes.

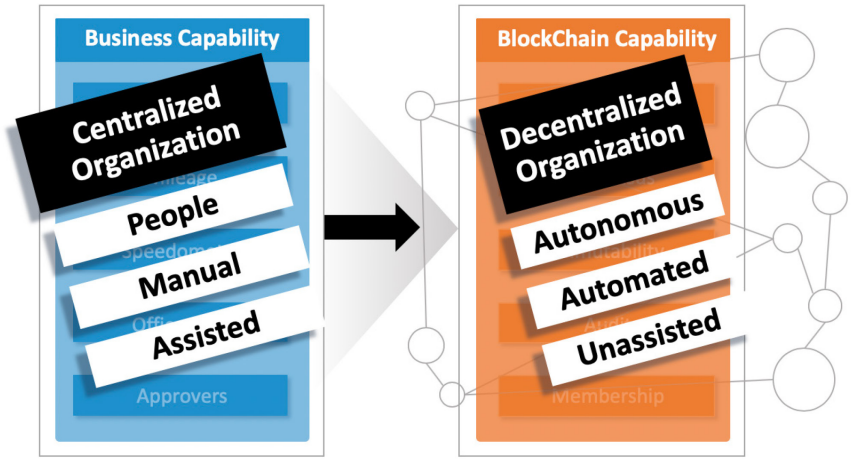

Figure 8: transformation - centralized-to-decentralized

By shifting to a decentralized organization, the most important asset – people – are freed from timely manual operations, and can be re-purposed to value-add activities.

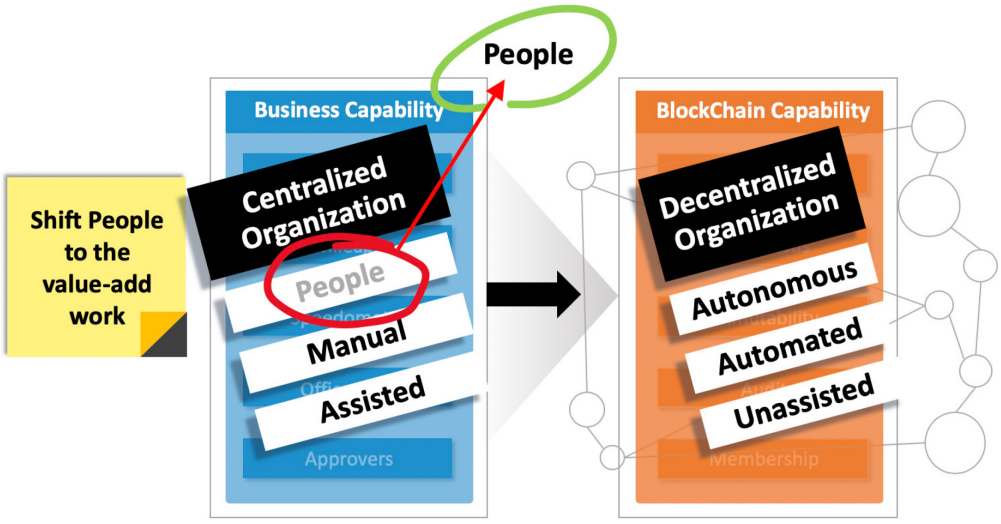

Figure 9: let technology do the routine work

Even when an organization is physically structured in a decentralized format, it can still be operating in the older model. In this example, 'Trust' of enforcing rules automatically does not exist.

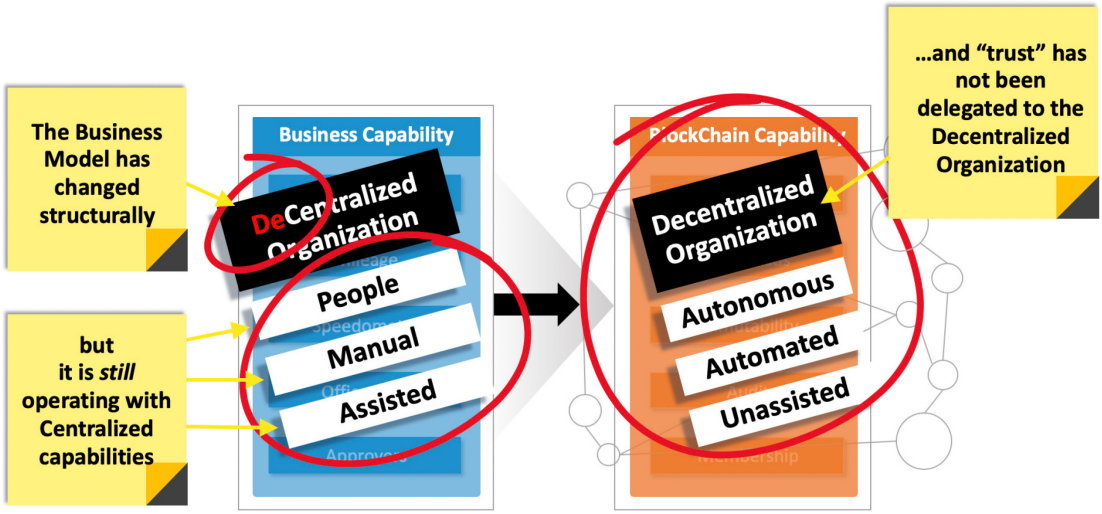

Figure 10: transformation - decentralized-to-decentralized

Appendix I – Implementation Appendices

People

Office of the Chief Information Officer (OCIO) – The responsibilities of the OCIO varies from agency to agency. The OCIO is typically responsible for the portfolio's entire technical stack to include cloud, on-premises server services, LAN/WAN, operating systems, directory services, database services, middleware, and agency and industry-facing application development. The OCIO governance model will be unique to the agency's implementation of the blockchain fabric. A public permissionless fabric will be governed differently than a permissioned fabric. The agency's blockchain may be intra-agency whereby managing the fabric of services follows a more traditional internal governance policy and procedure. Interagency fabrics will demand collaboration with other agencies who are consumers of the fabric's services. Finally, hybrid blockchain implementations that include both private and public services will demand equally unique governance models discussed later in this chapter. It will be the OCIO that will charter the blockchain implementation effort and govern business rhythms within its authoritative reach.

Blockchain Project Management Office – A proposed extension of the OCIO, the Blockchain PMO is strongly aligned with the enterprise architects and is responsible for managing the interface between all stakeholder members. The PMO captures requirements from the business line in support of the portfolio and how the blockchain fabric will best be integrated. These requirements are further refined into requirements for implementation. The PMO coordinates between development and production teams to ensure appropriate configuration management that will not compromise the operational state of the network. In addition, the PMO partners with risk management points of contact for data integrity planning, audit compliance and defense hardening, and plan of action and milestones mitigation. Finally, requirements, evaluation criteria, and funding data is provided to the Contract Office for execution by the appropriate industry partners.

Contract Office – The agency's contract office actively participates in the blockchain implementation and integration effort in their typical role as a liaison between the government and any involved private contractor(s) by providing FAC-C professionals (government employees who have knowledge, skills, and abilities as a Contracting Officer Representative.)

Business Line – The application portfolio supports the agency's line of business and is the consumer of the blockchain services. This member ensures the integration of any blockchain technology never compromises the stability of the portfolio and that the business drives the appropriate blockchain technology, not the other way around. The business line, in partnership with its own application developer, will assist in selecting what blockchain technology best satisfies the business requirement. Pre-implementation of the business will drive the application portfolio roadmap to include integration and use of the blockchain technology. The business line will establish Development, Modernization and Enhancement (DM&E) activities to identify components of application data that will be chained, to identify expected performance

requirements of the application, and to monitor the user experience to ensure alignment with all the above requirements. User stories and user acceptance test criteria will surface during sprint planning to ensure the application takes advantage of all appropriate blockchain attributes.

Data Management versus Data Provenance

Due to the provenance of data that blockchain enables, users of the system can determine the acceptable risk of using the data stored in the system. By being able to track the data back to documented inputs, entities, systems, and the processes that influenced it, the blockchain could provide a historical record of the data to its known origins. The user can then make an informed decision on if, where, when, and how to use the data.

It is important to understand that data management in a blockchain is enhanced by the auditability of the blockchain. This allows for faster read of the metadata and increased flexibility to control permissioning which now can be done at two levels. The first level is permission to read what is on the blockchain and the second is permission to read what is off the blockchain that is pointed to by the metadata on the blockchain. These are often referred to as on-chain and off-chain data.

The rules of the network will need to define the process for final disposition of data owned by the government which is regulated through the National Archives Records Administration (NARA).

Infrastructure Considerations

A web of distributed, decentralized nodes is needed to create DLT. Nodes are the computers participating in distributed networks. It is the type of distributed network that drives the roles and resource requirements of the nodes.

In permissionless blockchains, nodes need resources to perform three distinct tasks:

1. Assess the value of cryptocurrencies.
2. Synchronize the blockchain with the nodes' local database.
3. Maintain the state of the chain.

Any node can contribute computing resources to perform and confirm transactions. The nodes can also take part in a computation-intensive consensus process, such as proof of work, to validate and add new blocks to the chain. The participants in mining efforts to be the first to confirm a new block must have large amounts of computing and processing power.

The nodes can also be full nodes or light modes. Full nodes have a copy of the entire state of the blockchain. Light nodes fetch only subsets of the state while connecting to a full node. These subsets are small enough to fit in devices with small storage and memory capacities, such as mobile devices. This means that download and storage requirements are significantly less

intensive than that of a full node. Light nodes are becoming more common due to the rapid growth of ledgers.

In permissioned blockchains, nodes are known and limited in number. Many take part in more traditional light computation consensus mechanisms, such as Traditional Byzantine Fault Tolerance. The primary goal of nodes is to maintain the state of the blockchain. Each permissioned blockchain protocol takes a different approach with nodes. For example, Hyperledger Fabric uses a model built on pluggable components. In a typical implementation, Hyperledger has three types of nodes:

1. Orderer nodes batch transactions into blocks and create a hash-chained sequence of blocks. Orderer nodes are two types: Solo for development and testing, and Kafka for production.
2. Peer nodes simulate and endorse transactions. They also receive blocks from orderer(s) and commit transactions to the ledger in addition to maintaining the state of the ledger.
3. Client nodes operate similar to light nodes in public permissionless blockchains. They invoke transactions by submitting them to an endorsing peer node and broadcasting the transaction proposal to the orderer.

In this architecture, node configuration can range from allocating a dedicated machine for each component to using dedicated docker containers for each component on a single machine.

Blockchain is an efficient peer-to-peer communication model. But this new paradigm introduces challenges of network performance and security. In a blockchain for example, packets travel the network only when new blocks are added to the chain. Thus, a blockchain implementation needs proper networking and advance edge devices that can provide high networking and bandwidth speeds to ensure fast and reliable transfer of data between network participants. Also, blockchain introduces security challenges, most notably replay and 51 percent attacks.

There are varying forms of storage capacity as it relates to on-chain versus off-chain considerations. There are general storage options to offset the use of centralized off-chain storage platforms like Dropbox, AWS, etc. For public blockchain platforms, such as File Coin or Sia, they may use off-chain data solutions, such as InterPlanetary File System or distributed databases. A different form of storage capacity relates to the specific architecture of a private enterprise blockchain application that requires storage capacity.

The decision regarding what data should be stored on-chain versus off-chain takes careful consideration. Sensitive data such as personally identifiable information should never be posted on-chain. Store it off-chain and use metadata and hash pointers to obfuscate and reference this data's location. Throughput and storage assessments must be made regarding how much data is necessary to store on-chain to accomplish the business needs. Often times, much of the

noncritical data associated with a transaction can be stored in an off-chain data store, similar to where the data may exist today.

Interoperability

Interoperability is the idea that transactions and smart contracts can execute in a frictionless manner across varying blockchain applications and protocols. The community has recognized a need for a greater degree of connectivity between different protocols to achieve mass adoption. Currently, interoperability between blockchain platforms is in its infancy.

As of the end of October 2018, Hyperledger Fabric has integrated with the Ethereum Virtual Machine to be able to execute Ethereum-based smart contracts. For example, if an organization were a member of a supply chain ecosystem operating on a Hyperledger Sawtooth and it wanted to integrate the data into the Quorum-based procurement blockchain network it was also a member of, it would not be able to do so.

From a crypto asset-based blockchain perspective, interoperability could also represent the ability to exchange tokens between users across chains (atomic swaps) without trusted a third-party.

There are many projects working to solve both forms of interoperability (mutualized messaging and atomic swaps) involving public permissionless blockchains. These projects include, but are not limited to: Hyperledger Quilt, Polkadot, Aion, and Cosmos. Each of these projects has its own nuance. Polkadot, for example, is a relay mechanism that acts as a conduit for valid transactions between blockchains. The concept is to parallelize chains that join Polkadot and create parachains which are constituent blockchains that gather and process transactions.

The platforms working to solve interoperability for permissioned blockchains are developing in a more a siloed manner within their respective native platforms (e.g. R3 Corda, Hyperledger, etc.). Based on blockchain evolution thus far, it is most likely that there will not be a single blockchain all networks adopt, and a single organization will be a member of multiple blockchain ecosystems for different use cases (e.g. supply chain, finance, employee records, etc.) Therefore, business applications will need to be able to interact with one another, and the broader blockchain community will need to work together to solve for interoperability.

To accomplish this, standards are needed around blockchain/DLT similar to the work of the International Organization for Standardization, Technical Committee 307: blockchain and distributed ledger technologies. The Sovrin Foundation is leading the development of industry standards/protocols in the identity space. Once these standards are established, businesses will need to ensure they are implementing the right technology with the right privacy model to interoperate across the expanded ecosystem.

Onboarding and Offboarding

While other options are possible, *enterprise blockchains* are likely to be private permissioned blockchains that regulate access via some form of membership control. There are privacy considerations related with onboarding effecting the consensus mechanism, endorsement policy, certificate distribution, and access to various elements of the blockchain. Similarly, off-boarding members requires analysis of impact on quorum rules in smart contracts, legality of smart contracts, data governance, auditing, and archiving of off-chain data. The governance structure must regulate this process and that needs to be translated into supporting enterprise architecture.

Public permissionless blockchains do not regulate the addition of new nodes to the network. As new nodes are added to the blockchain network, there are negative impacts on the performance of the blockchain. It is frictionless for nodes to leave a public permissionless blockchain at any point.

Testing Execution

Execution of the test plan created in the Readiness Phase needs to be done in parallel with developmental activities, also known as shift left testing. By shifting testing to the left, teams can perform various tests early, and reuse those tests continuously. The iterative quality feedback received throughout the development process decreases the number of defects found later in the lifecycle, where the impact on the organization can be severe and expensive.

There are many shift left-focused blockchain testing suites available to assist with conducting the necessary API, functional, node, smart contract, and performance testing.

Testing Requirements:

The concept of the shift left approach is to shift the testing to the earlier stages of blockchain design and development framework. In this approach, testing begins during the requirements gathering process. Additional iterations occur during the design stage and during the development activities. Before each deliverable, testers work in sync with the designers and developers, provide feedback and suggest potential changes to be incorporated in the next iterative deliverable. There are many free and open source automated shift left testing suites available in the market that can be used for implementing this approach.

With performance testing, certain metrics are determined and assessed against the blockchain. This is an essential test approach to make the blockchain instance as fast as possible, keeping other desired properties in mind. Make sure to be able to identify bottlenecks and at which point stability is affected.

Testing activities include, but are not limited to, assessing:

- Throughput.

- Speed of transactions, examining the validation of transactions throughout their life cycle as data flows through apps/vaults/wallets and other components of the system.
- Processing speeds.
- Targeted volume inputs.

API testing is also needed for blockchain integration. With this testing approach, the goal is to discover bugs (security and implementation related) and resolve the errors found. Important activities test the enablers of scalability. There are many automated, online software suites that can be used for performing API testing.

Testing activities include, but are not limited to, assessing:

- Acceptable load/stress the blockchain can handle.
- Correctness, reliability, security, dynamism, cross-platform portability, and conformance to the required user experience.
- Stakeholders, environments, and normal circumstances in which the API will be used. Classify the expected results of the deliverables or the product.
- API calls (requests and responses) to ensure the blockchain transactions are secure and reliable. Be sure to scan any web interfaces for vulnerabilities, include the use of the fuzzing technique to test contract functions.
    - How applications interact with the system's internal components and end points.
    - The format of the request and response.
    - The response and feedback given to the user when a transaction is rejected.
- Are inputs with sensitive information stored out of band?
- The interface with APIs for access control, payments, track and trace, balances, etc.

Functional testing verifies the blockchain instance's conformance to business and user requirements it was designed to meet. In this approach, testers compose and execute test scripts to check if the application (constituting use cases and smart contracts) is performing the desired functions. Two main subdomains of functional testing include black-box testing and white-box testing.

Contract tests, making sure that the automation behind smart contracts works flawlessly is another thing that can make or break your blockchain application.

A blockchain platform includes certain key components in its architecture that are essential in its working, which are smart contracts and nodes. The inputs to smart contracts and the methods to obtain the data and execute the contract need to be validated, boundary conditions must be checked, unit testing needs to occur, and integration testing should be performed once the architecture gets interlinked.

Nodes, on the other hand, are independent units distributed throughout the application, and they need to be unit tested independently as well. The status of each node must be tracked during all phases of testing.

Testing activities include, but are not limited to, assessing:

- Consensus between nodes and transaction consistency, including the impacts of multiple data consensus failures. Also test for different data types and the correct consistency of expected data formats.
- Transactions are correctly sequenced (both when functioning as normal and when nodes are failing).
- Impacts of different locations of servers.
- Proof of distribution.

Finally, the entire platform should be tested to examine how the components work together as a unit.

Testing activities include, but are not limited to, assessing:

- Network resiliency, including testing the synchronization after nodes restart or rejoin the network.
- Fault tolerance for both failing nodes and malicious nodes.
- Onboarding and offloading peers, as well as when new network features are introduced.
    - Performance and speed impacts.
    - Effects on governance and validation.
    - Responsiveness and stability of the system.
- Network security.
    - Access, authentication, and authorization.
        - Have all admin account names and credentials within system components been changed from the default?
        - How authorization controls work within the system?
        - Would certificates be vetted by a compromised CA server?
        - Do databases within the system support authentication?
        - Can records be changed or authentication of assets be altered?
        - Are users of APIs registered?
    - Secure hash algorithm.
    - Transaction message signing.
    - Effects of modified the data stored in a blockchain-based database.
    - Are databases containing sensitive information firewalled from non-local peers?
- Data privacy (based on selective permissions).
- Communication of code between different components (including external systems).
- The ability to audit transactions.

A few examples of the testing suites providing a comprehensive testing methodology for blockchain instances are Ethereum Tester, Populus, Tineola, and Hyperledger Composer. IBM Hyperledger is an open source tool that allows you to model and test your blockchain network with a minimal set of tools: Docker and a browser. It is done through Hyperledger Composer which is a framework for facilitating blockchain app development consisting of a modeling language, a UI (Composer), and a command-line interface. It supports automated system tests, interactive testing, and automated unit tests.

Functional testing is a testing approach where the blockchain application is tested to verify its conformance to business and user requirements that the designers started with. The main motive is to compose and execute test scripts to check if the application (constituting use cases and smart contracts) is performing the desired functions. The functional testing should be holistic, meaning it should cover all aspects of business transaction

- The testing includes the validation of a business transaction between multiple parties associated with the transaction.
- The transaction is tested to see if it is valid, is executed between the right parties.
- The transaction is tested to see the current state of the transaction and whether the transactions has been validated through electronic signatures and through other sufficient endorsements.
- Further the transactions are tested for correct input and output without any conflict with other transactions or its associated past events.
- The input and output are tested independently by the components that use them as well. When there are mismatches, suitable countermeasure such as failure notification is set.
- Data Transmission after the transaction is tested to ensure that there are no losses during the data transmission.
- The rejected transactions are stored in test logs for further examination by the owners of the transaction.
- Test to ensure that inputs into the blockchain match the expected format. Also test the rejected records for the root cause.

API testing is also needed for blockchain integration. With this testing approach, the goal is to discover bugs (security and implementation related) and resolve the errors found. Important activities test the enablers of scalability. There are many automated, online software suites that can be used for performing API testing.

API Testing activities include, but are not limited to, assessing:

- Acceptable load/stress the blockchain can handle.
- Correctness, reliability, security, dynamism, cross-platform portability, and conformance to the required user experience.

- Stakeholders, environments, and normal circumstances in which the API will be used. Classify the expected results of the deliverables or the product.
- API calls (requests and responses) to ensure the blockchain transactions are secure and reliable. Be sure to scan any web interfaces for vulnerabilities, include the use of the fuzzing technique to test contract functions.
    - How applications interact with the system's internal components and end points.
    - The format of the request and response.
    - The response and feedback given to the user when a transaction is rejected.
    - API testing needs to ensure the correctness of interaction between different applications and the blockchain are as expected (requests and responses)
    - API testing should include end to end testing, especially when there are application interfaces.
    - Test the APIs for access control, payments, track and trace, balances etc.

Node testing, using a specific protocol for authentication, checks with the majority of nodes and approves the block for its validity to make a successful transaction. Hence all heterogeneous nodes must be independently tested.

Testing activities include, but are not limited to, assessing:

- Consensus between nodes and transaction consistency, including the impacts of multiple data consensus failures. Also test for different data types and the correct consistency of expected data formats.
- Transactions are correctly sequenced (both when functioning as normal and when nodes are failing).
- Impacts of different locations of servers.
- Proof of distribution.

In performance testing, certain metrics are determined and assessed against the blockchain to calculate the throughput, speed of transactions, processing speeds and so on. One of the key issues is blockchain's inability to scale and process large volumes of transactions. This impact proprietary payment processing systems of financial service organizations.

This is an essential test strategy to make the blockchain application as fast as possible, keeping other desired properties in mind.

Performance testing activities include, but are not limited to, assessing:

- Throughput.
- Speed of transactions, examining the validation of transactions throughout their life cycle as data flows through apps/vaults/wallets and other components of the system.
- Processing speeds.
- Targeted volume inputs.

- Performance testing should integrate client applications, smart contacts and other external system interfaces to meet the service level agreements given by the stake holders.
- Performance testing should be targeted to test scenarios when multiple data consensus failures during node updates across the network
- Performance testing should be further planned for myriad of data-types and server locations
- Performance test cases for end to end testing is key, it should cover multiple end points, the test cases should ensure that the sequence of transactions is the same across all the shared ledger across all nodes.
- Block Size centric performance testing should be done, currently 1 MB is the fixed maximum size of a Block in most Blockchain network. Hence Performance testing should be done to see the behavior of the network when new data into block exceeds this limit.

Smart Contract testing is making sure that the automation behind smart contracts works flawlessly in your blockchain application. A smart contract should be viewed as piece of software that stores the business rules (contract) of a transaction and automatically gets trigged based on events and executes the agreed terms as the business rules.

Smart contract testing is fundamentally the same as a regular code test – essentially to find security flaws and vulnerabilities before the code is publicly deployed. It is very important to ensure that the test environment is refreshed frequently to keep APIs that enable connectivity to other BC nodes and application are the latest versions.

Smart contract is a piece of software that stores the rules for the negotiating terms of the contract automatically verifies the contract and executes the agreed terms. Therefore, the goal of the smart contract testing is to ensure that each smart contract is executed to individually to ensure that the smart contract meet the rules of the business. Further, if when executed along with the sequence or parallel execution of the multiple smart contracts the resulting output is the same.

Smart contract testing can be categorized into two areas

1. Over all smart contract testing goals (Functional and Technical)
2. Smart contract level testing

Functional goals for the smart contract testing:

- Trusted parties can transact directly with each other using smart contracts.
- Smart contracts are stored on the blockchain that can be accessed by authorized parties.
- Smart contracts can execute agreed stored processes when triggered by an authorized / agreed event just like traditional systems.

- All contracts transactions are stored in chorological order of the blockchain for future access along with the completed audit trail of events.
- All parties associated with the transaction need to agree to amend it.
- Any party fails, the system continues without loss of data or the integrity
- Creating what logic makes it function like a secure computer system but without the risks, costs and trust issues of centralized model.

There are five technical goals to test for in a smart contract:

1. Always check for overflows and underflows. If you are making any sort of mathematical calculation, you have to make sure that your code is not overflowing or underflowing.
2. Check that the return values of your functions are always within the range of the expected values.
3. Test the limits of the functions.
4. Make sure that the return values are properly formatted.
5. The contract is ready for all the possible values of the parameters of your functions to avoid security risks.

A Smart contract Testing Framework at a Smart Contract Level:
While planning to test a smart contact, the following items need to be considered. The players will vary but the transactional components will be predominantly the same across industry verticals (healthcare, finance etc.)

- Define the transaction.
- Define the participating members and their roles.
- Define who should approve these outcomes of transaction.
- Define an asset – essentially an outcome of transaction. Where the asset should be.
- Define how the inputs are going to be given, essentially whether the input data is going to be provided via a decentralized application (DApp) or whether the smart contract is going to get the input data from another application or smart contracts. If the input data is encrypted, then the data should be decrypted and consumed. If the outgoing data is sensitive, then the data should be encrypted and sent across to other systems or nodes.
- Define every transaction should have an implementation logic, essentially like who is going to create or initiate the transaction.
- Define who is going to approve and validate the transaction.
- Define what the outcome is when the transaction is completed.
- Transactional charges – Mostly applies for public blockchain applications. When the transactions demand a transactional service charge (gas), transactions should be tested as follows:
    - Positive testing – When the transaction occurs, the agreed upon transactional service charge should be deducted from the owners of the account. Point to note – It should be predetermined in the smart contract with regards who bears the transactional service charge (gas).

- Negative testing technical – When a transaction fails due to technical reasons, such as node unavailability, then the gas should be deducted from the pre-determined accounts.
- Negative testing functional – When a transaction fails due to functional reasons such delayed response by the parties involved, the gas charges should be deducted from pre-determined accounts.

Validation Testing is to ensure that the transactions initiated by authentic sources, the sender does cannot deny that the message was not sent (Non-repudiation), the transaction was not altered during the transmission (integrity)

Validation testing policy should have the following:

- The Data/Document required for the transaction should be hashed.
- Use the private keys to encrypt the hash A.
- The recipient decrypts the document using the public key, the resulting hash A.
- The recipient applies the same has algorithm on the received document hash B.
- The recipient compares both the hash A and hash B to ensure that the data / document was not altered during transit.

Blockchain constitutes certain key components in its architecture that are essential in its functionality, which are smart contracts and nodes. There are methods within the application that make up the smart contracts. These methods need to be validated, boundary conditions are to be checked (as described above), followed by unit testing, and integration testing once the architecture gets interlinked. Nodes, as mentioned before, are independent units that are distributed throughout the application and they need to be unit tested independently. Nodes are an important part of the blockchain for securing a blockchain application. For instance, if a node on the network is attacked by something like DDoS, then the application hosted on blockchain will be affected.

Few testing suites that provide a comprehensive testing methodology for blockchain applications are Ethereum Tester, Populus, and Hyperledger composer.

a. Ethereum Tester is a straightforward testing library that's available as a GitHub repo. Since it is open-source, the setup is relatively easy and it has a decent API support for forks (Homestead, DAO, etc.) mining, as well as other testing-critical functions.

b. Populus has Ethereum testing functionality baked in the form of a specific set of features for test contract deployment. It's built around the py.test framework, which makes it easy to implement.

c. IBM Hyperledger is an open source tool that allows you to model and test your blockchain network with a minimal set of tools: Docker and a browser. It's done through

Hyperledger Composer, which is a framework for facilitating blockchain app development consisting of a modeling language, a UI (Composer), and a command-line interface. It supports automated system tests, interactive testing, and automated unit tests.

Infrastructure Requirements Example:

For example, the requirements for a node in the Ethereum network are at least one CPU Dual-core, 2GB of memory (RAM), and a storage capacity of 256GB. One item to note is nodes need little CPU or memory power to function, but they require fast and unlimited internet connection. In fact, 50 - 100 Mbps is the recommended internet connection speed. Like Ethereum, Bitcoin requires a fast and reliable internet connection above 50 Mbps. Bitcoin nodes also need at least 1GB of memory (RAM) and 145GB of free disk space. This difference in storage requirements between Bitcoin and Ethereum is largely due to Ethereum's rapid growth and rate of change since 2017.

Appendix J – Playbook Navigation

In this section, you will find valuable information to help you understand the framework laid out in this document and a series of questions to guide you in the playbook and help you kick start your blockchain development journey.

Framework Flow

The playbook introduces a framework made out of several phases connected to one another sequentially to guide you through key activities that will help you leverage blockchain technologies to tackle your use case.

Each phase is connected to a decision gate before automatically going to the next phase. This decision gate helps you determine if you should:

- Go to the next phase – using the outputs generated by the phase n, you determine that there is enough value to keep going to the next phase.
- Stop – using the outputs generated by the phase n, you determine that blockchain does not bring enough value at this time.
- Iterate – using the outputs generated by the phase n, you determine that more work is needed or data from previous phases need to be adjusted.

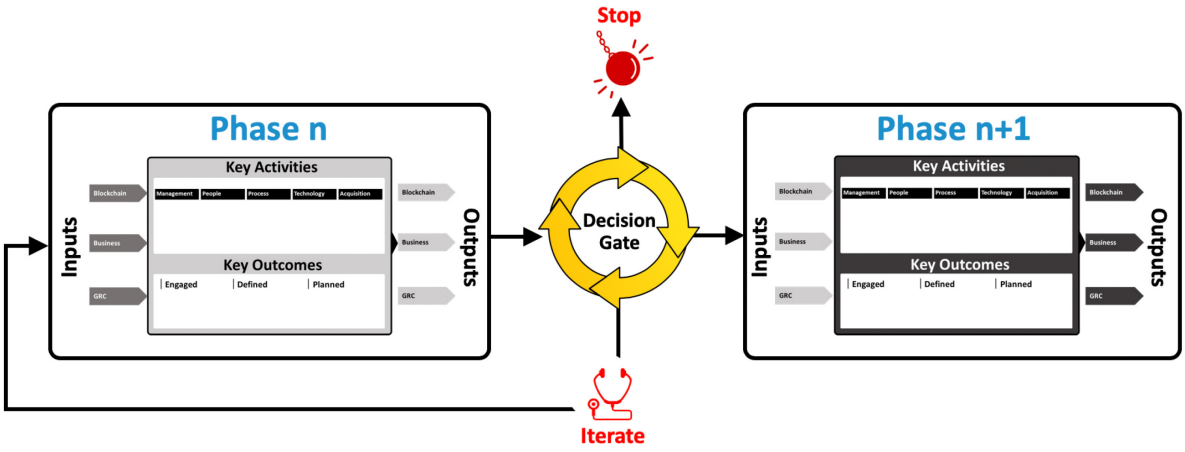

Figure 10: phase to phase flow

In addition, each phase is made up of key activities and outcomes whose purpose is to help you work through aspects needed to achieve your use case objectives leveraging blockchain technologies. The phase has inputs necessary for the key activities and generated outputs that will be needed for the decision gate. These outputs will also be used as inputs for the next phase.

Where do I start?

The introduction section contains graphics that show the process and summarize the key activities for each of the phases. In addition, each phase also contains a more detailed graphic that summarizes the objective of the phase, its key activities and outcomes, inputs, and

outputs. It is a good start to take a look at each of them to get a sense of what will happen in each phase.

I am a senior executive, what should I focus on?

Each playbook phase is composed of phase inputs, phase outputs, key activities and outcomes, and a decision gate. You should make sure you look at the inputs/outputs of each phase as well as the phase decision gate. You can also look at the "management" key activity category of each phase to get an understanding of what the management team will do.

I am on the management team, what should I focus on?

As part of the management team, you need to understand the overall process. The diagrams are a good start. You should also focus on the "management" key activity category of each phase.

I am on the acquisition team, what should I focus on?

You can focus on the "acquisition" key activities category of each phase as well as Appendix G.

I am on the development team, what should I focus on?

You should have a good understanding of the inputs needed to implement the technical solution. Focus on the "technology" key activities category of each phase.

What is Blockchain or DLT?

Please refer to the ACT-IAC Blockchain Primer.

What about my workforce?

Each playbook phase has a set of key activities that are grouped in high level categories. The category labeled "People" highlights key activities regarding the workforce. Also, the Readiness Phase handles organizational readiness.

What phase(s) should I follow for a proof of concept?

This playbook is intended to be used in an iterative way. All the phases should be used to produce a proof of concept with only the most valuable functions of the use case developed.

Can I use the playbook in an agile manner?

This playbook is intended to be used in an iterative way. One can go through the playbook phases multiple times to develop proofs of concept, pilots, and full implementation. Each iteration will dive deeper in the activities of each phase. Iterations can also occur to refine data in each phase and adapt the roadmap.

How do I handle a use case with multiple organizations?

Each phase deals more or less with understanding the organization(s) in play for the implementation of the solution. If you know more than one organization will be involved, when

you look at the organizational aspects of the solution, address from the start the need for a multi-organization solution.