

ガバメントクラウド先行事業一次計画書

令和5年3月27日



目次

① 公募団体名	2
② 公募団体代表者氏名	2
③ 公募団体担当者名と連絡先	2
④ 公募団体のシステムの現況	3
⑤ 公募団体として17の期間業務に付属または密接に連携すると考える業務のシステムとその理由	5
⑥ 公募団体における令和7年度までの標準準拠システムへの移行計画	7
⑦ 先行事業の実施計画	
ア. ガバメントクラウドのクラウド提供事業者に関する希望とその理由	8
イ. ガバメントクラウドにリフトするシステム	9
ウ. ガバメントクラウドにリフト後の業務システム全体の構成図	9
エ. ガバメントクラウドにリフトする手順	10
オ. 協力する見込みのアプリケーション開発事業者名等	16
カ. 検証項目の検証方法	17
標準非機能要件の検証	17
標準準拠システムの移行方法の検証	25
投資対効果の検証	28
R5年度追加検証について	30
1. より効率的な構成検証	31
2. 運用効率化の検証	33
3. 投資対効果の検証	34
4. ネットワーク接続の在り方検討	35
R5年度先行事業実施スケジュール	36
R5年度先行事業実施における環境	37

公募団体情報

公募団体名	盛岡市役所
公募団体代表者氏名	盛岡市長 谷藤 裕明
公募団体担当者名と連絡先	

盛岡市のシステムの現況

盛岡市では基幹系業務システムをハウジングおよび自庁へのサーバ設置により運用しており、複数ベンダの複数のパッケージシステムから構成されています。基幹17業務を中心に、それらに関連する業務システムを類型別、パッケージ別に一覧で示します。

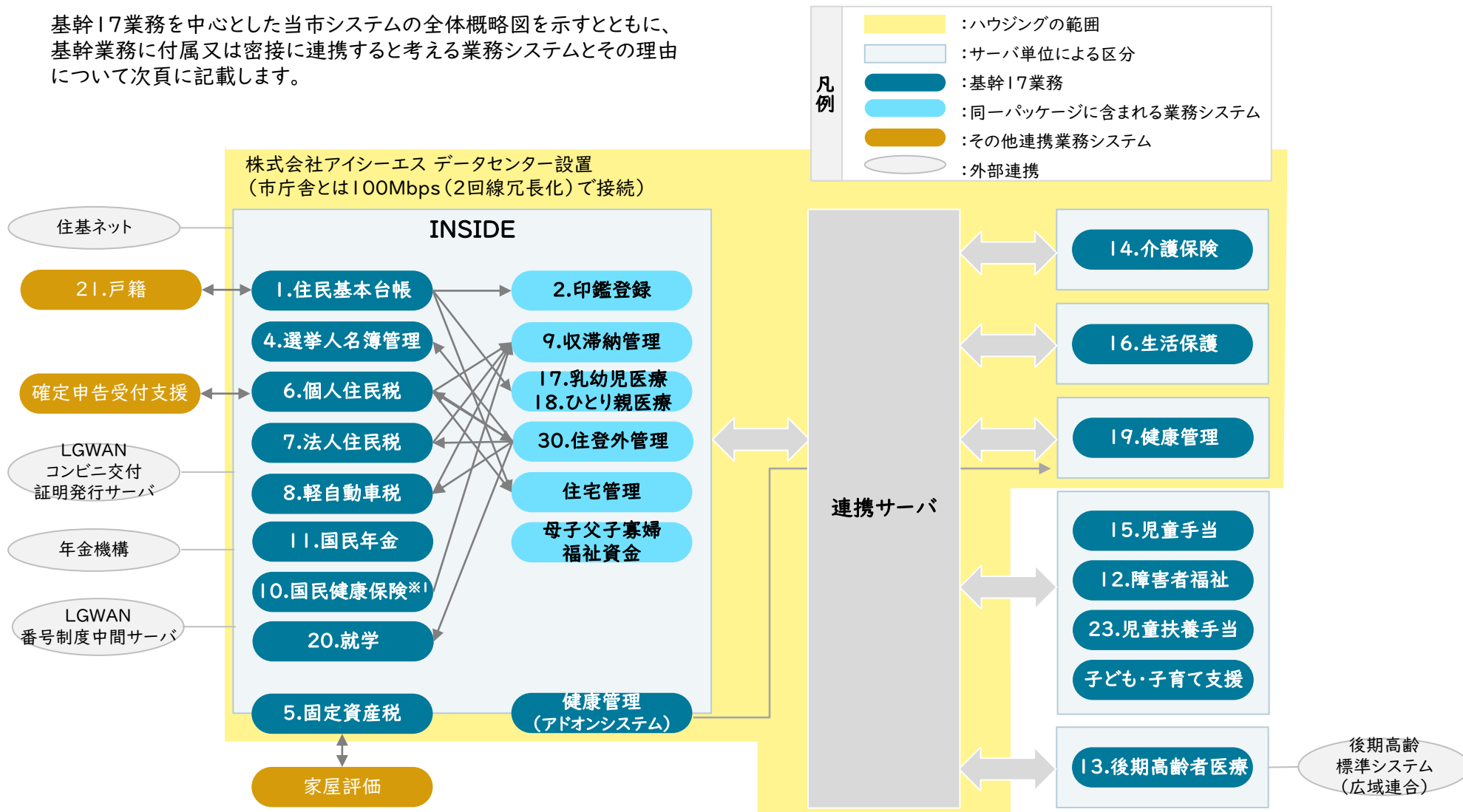
システム 類型	パッケージ名称	システム提供事業者	システム開発元	業務システム名 (基幹17業務に●)	補足事項
【20】オープン系システムハウジング型					
住民情報統合システム INSIDE	株式会社アイシーエス	株式会社アイシーエス	1.住民基本台帳	●	
			4.選挙人名簿管理	●	
			6.個人住民税	●	
			7.法人住民税	●	
			8.軽自動車税	●	
			10.国民健康保険	●	令和5年2月より国保標準システム県域共同利用へ移行予定
			11.国民年金	●	
			20.就学	●	
			2.印鑑登録		
			9.収滞納管理		
			17.乳幼児医療 18.ひとり親医療		システムとしては妊産婦等各種医療費助成をサポート
			30.住登外管理		
			住宅管理		
			母子父子寡婦福祉資金		
【11】オープン系システムハウジング型					
	株式会社アイシーエス	株式会社アイシーエス	5.固定資産税	●	照会機能のみINSIDE利用

盛岡市のシステムの現況

システム 類型	パッケージ名称	システム提供事業者	システム開発元	業務システム名 (基幹17業務に●)	補足事項
【20】オープン系システムハウジング型					
ADWORLD介護保険システム	あゆむくん 生活保護システム 地域健康支援システム 健康かるて	株式会社アイシーエス		14.介護保険	● システム構築および運用保守作業は、システム提供事業者である株式会社アイシーエスが実施 ※1 INSIDE内にアドオンシステム(独自)あり
あゆむくん 生活保護システム		株式会社IJC		16.生活保護	
地域健康支援システム 健康かるて		株式会社両備システムズ		19.健康管理※1	
【19】オープン系システム自庁型					
ADWORLD後期高齢者医療事務支援システム	株式会社アイシーエス	株式会社日立システムズ		13.後期高齢者医療	● システム構築および運用保守作業は、システム提供事業者である株式会社アイシーエスが実施
ADWORLD戸籍総合システム				21.戸籍	
ADWORLD家屋評価システム		株式会社日立ソリューションズ 東日本	家屋評価		
住民税課税支援システム 税務LAN	株式会社リードコナン	株式会社リードコナン		確定申告受付支援	
【10】オープン系システム自庁型					
	株式会社ワーク	株式会社ワーク		12.障害者福祉	●
				15.児童手当	●
				23.児童扶養手当	●
				子ども・子育て支援	●
				地域包括支援	
				飼犬管理	

基幹17業務を中心とした盛岡市システム全体概略図

基幹17業務を中心とした本市システムの全体概略図を示すとともに、基幹業務に付属又は密接に連携すると考える業務システムとその理由について次頁に記載します。



17の基幹業務に付属又は密接に連携する業務システムとその理由



【付属又は密接に連携する業務システム】

⇒ 印鑑登録、収滞納管理、乳幼児医療、ひとり親医療、住登外管理、住宅管理、母子父子寡婦福祉資金

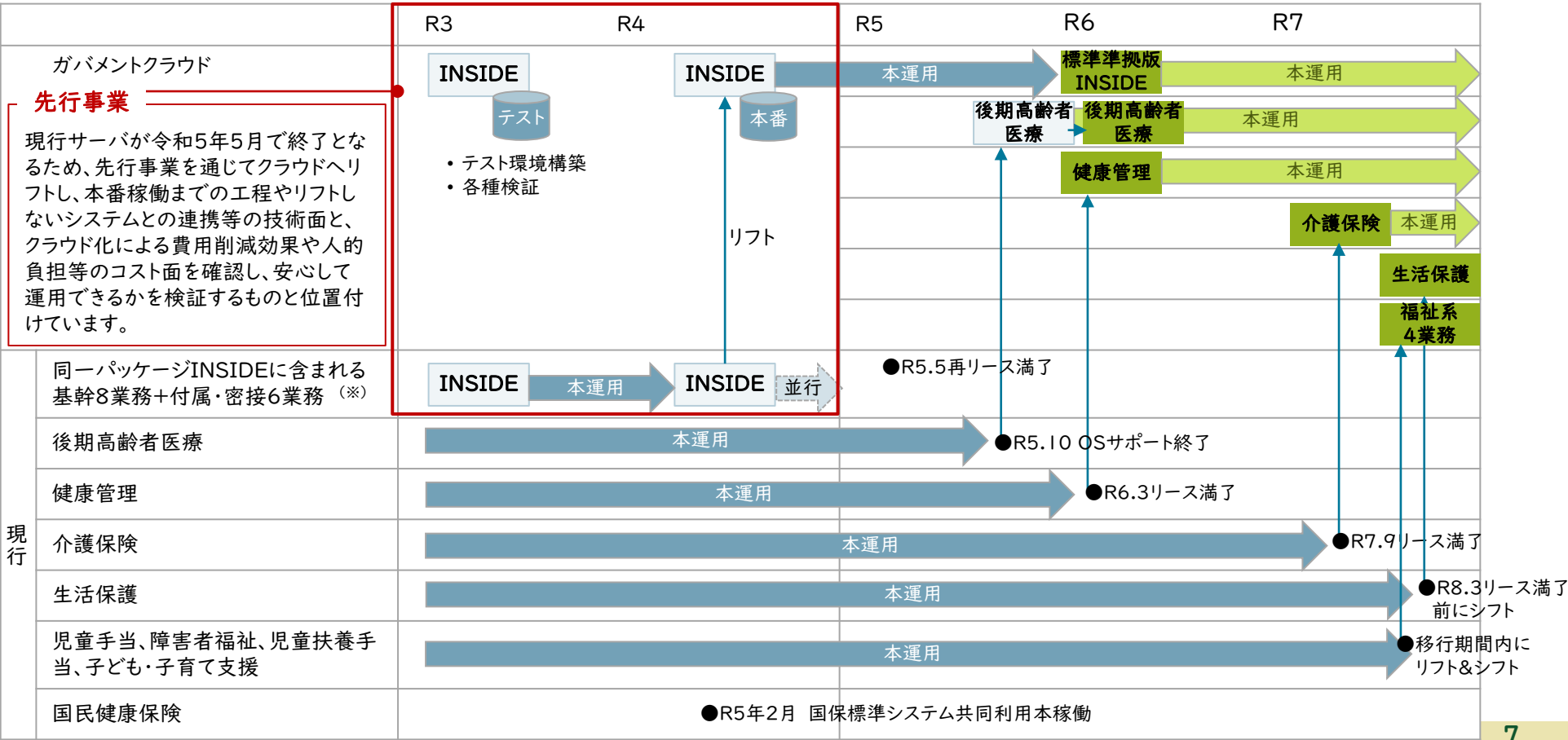
<理由>

- 同一パッケージ(オールインワン)として導入している業務システムである。
- 特に**印鑑登録、収滞納管理、住登外管理**については、下記の通り密接な連携を行っている。
 - 印鑑登録は住民基本台帳と連動し、印鑑登録者が非住民(転出、死亡等)になった場合には自動的に廃止処理を行うことが可能となっている。
 - 収滞納管理(還付、口座管理を含む)は賦課システムと連動し、賦課調定データに基づく納付書消込業務や未納者に対する督促業務、口座振替業務をサポートしている。
 - 住登外管理は住民基本台帳に登録されている住民以外の宛名情報(住登外者)を一元的に管理しており、賦課システムをはじめ各業務から参照されている。

盛岡市における令和7年度までの標準準拠システムへの移行計画

【基本的な考え方】

- ・ 現行サーバの(再)リース満了もしくはOSサポート終了のタイミングでガバメントクラウドへリフトします。
- ・ 標準準拠システムのリリースに応じてシフトを行う。タイミングが合えばリフト・シフトを同時に行うことを検討します。
- ・ 現行システムが標準仕様に準拠しない場合には、標準準拠システムの調達を実施した上で移行します。



※ P2「盛岡市のシステムの現況」に示した業務システムのうち国民健康保険を除いた業務

ガバメントクラウドのクラウド提供事業者に関する希望とその理由

1 ガバメントクラウド上のIaaS基盤で構築されるシステム（クラウドネイティブアプリケーションではない）であっても、従来のシステムと遜色ないパフォーマンスを確保できるサービスを希望

（理由） 一般的に、クラウドネイティブアプリケーションの方がIaaS基盤に構築される従来型のアプリケーションよりもパフォーマンスを発揮できるものとされています。したがって、ガバメントクラウドに構築する基幹システムはクラウドネイティブアプリケーションにするのが最善であると考えられます。

一方、クラウドネイティブアプリケーションとする場合、クラウド提供事業者が提供するサービスに依存することから、クラウド提供事業者が変更となった場合にはアプリケーションの動作保証ができなくなるなどの懸念が生じます。また、ガバメントクラウドは先行事業においては単一クラウドを想定していますが、令和5年度以降は複数のクラウド事業者により構成される可能性も示唆されており、今後クラウド提供事業者が変更になる可能性もあるものと推察します。

これらのことから、現状においてはIaaS基盤での構築が現実的であると考えられるため、その場合であっても従来のシステムと遜色ないパフォーマンスを得られるようなサービスを提供いただくことを希望します。

2 将来的にクラウド提供事業者が提供するマネージド型RDBサービスを利用する場合、文字情報基盤の文字（異体字等）を正しく扱うための設定が可能であるサービスを希望

（理由） 行政（自治体）が扱う文字は非常に複雑であり、複数の異体字がある場合でも正しい文字をデータベースに格納し表示できることが必須要件となります。IaaS基盤で構築する場合には問題になりませんが、将来的にPaaS基盤を利用することを見据え、文字を正しく扱えるサービスを希望します。

3 障害復旧に迅速に対応できるよう、自動化ツールを用いてガバメントクラウド上に構築済みの環境と同じ環境（サーバやネットワーク）を簡単に構築できるサービスを希望

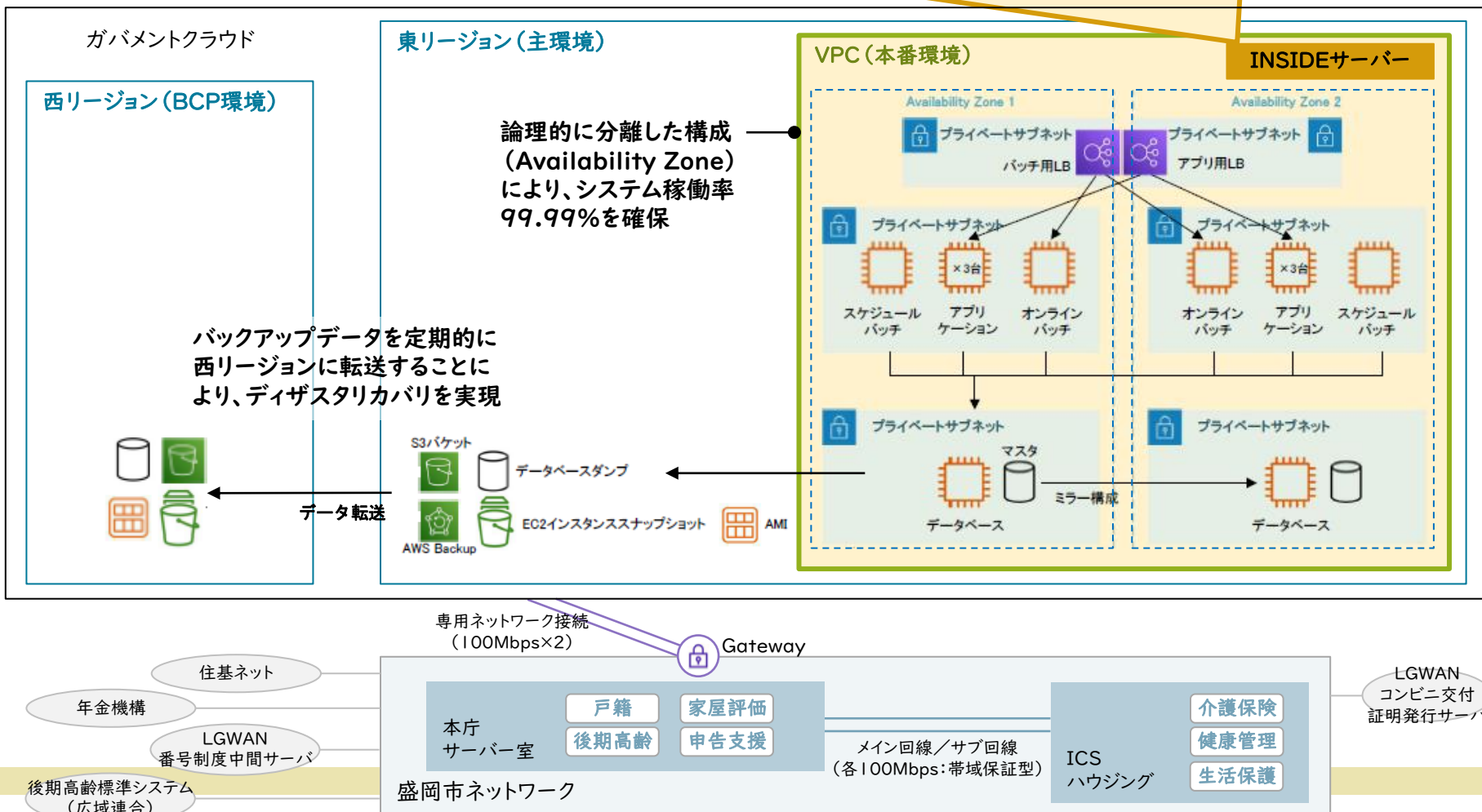
（理由） 「クラウドサービスで利用するデータセンターがある地域」（リージョン）で障害が発生した場合、別のリージョンを用いてサーバやネットワークの復旧を容易に実現できるようにするため、このようなサービスの提供を希望します。

リフト対象システム、およびリフト後の業務システム全体構成図（概略図）

盛岡市が現在導入している住民情報統合システム「INSIDE」をリフト対象とします。（具体的な業務システムは以下の通り）

【基幹8業務システム】 住民基本台帳、選挙人名簿管理、固定資産税、個人住民税、法人住民税、軽自動車税、国民年金、就学

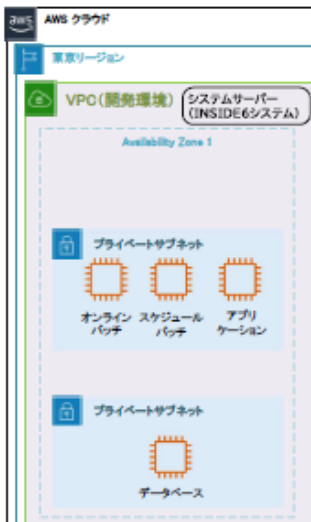
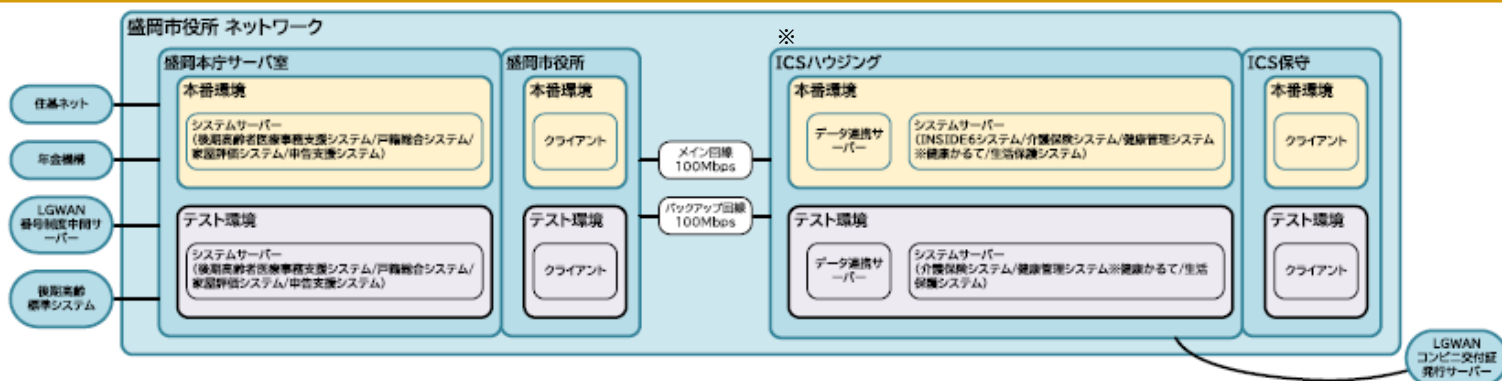
【付属または密接に連携する業務システム】 印鑑登録、収滞納管理、乳幼児医療、ひとり親医療、住登外管理、住宅管理、母子父子寡婦福祉資金



ガバメントクラウドにリフトする手順①：構築

【前提とする考え方】

- 提示されている先行事業のスケジュールに従い、ガバメントクラウド開発環境への業務システム構築から本番環境への移行までの手順を示します。
- 具体的な手順を計画する都合上、クラウド提供事業者がAWS (Amazon Web Service) の場合を想定した記載とし、事業者がAWS以外の場合は同等のサービスを利用または補完する機能を構築することを想定しています。(以降、同様) ※株式会社アイシーエスをICSと表記しています。



AWSクラウドへは、Management Consoleから接続します。(この時点ではインターネット経由での接続を想定)

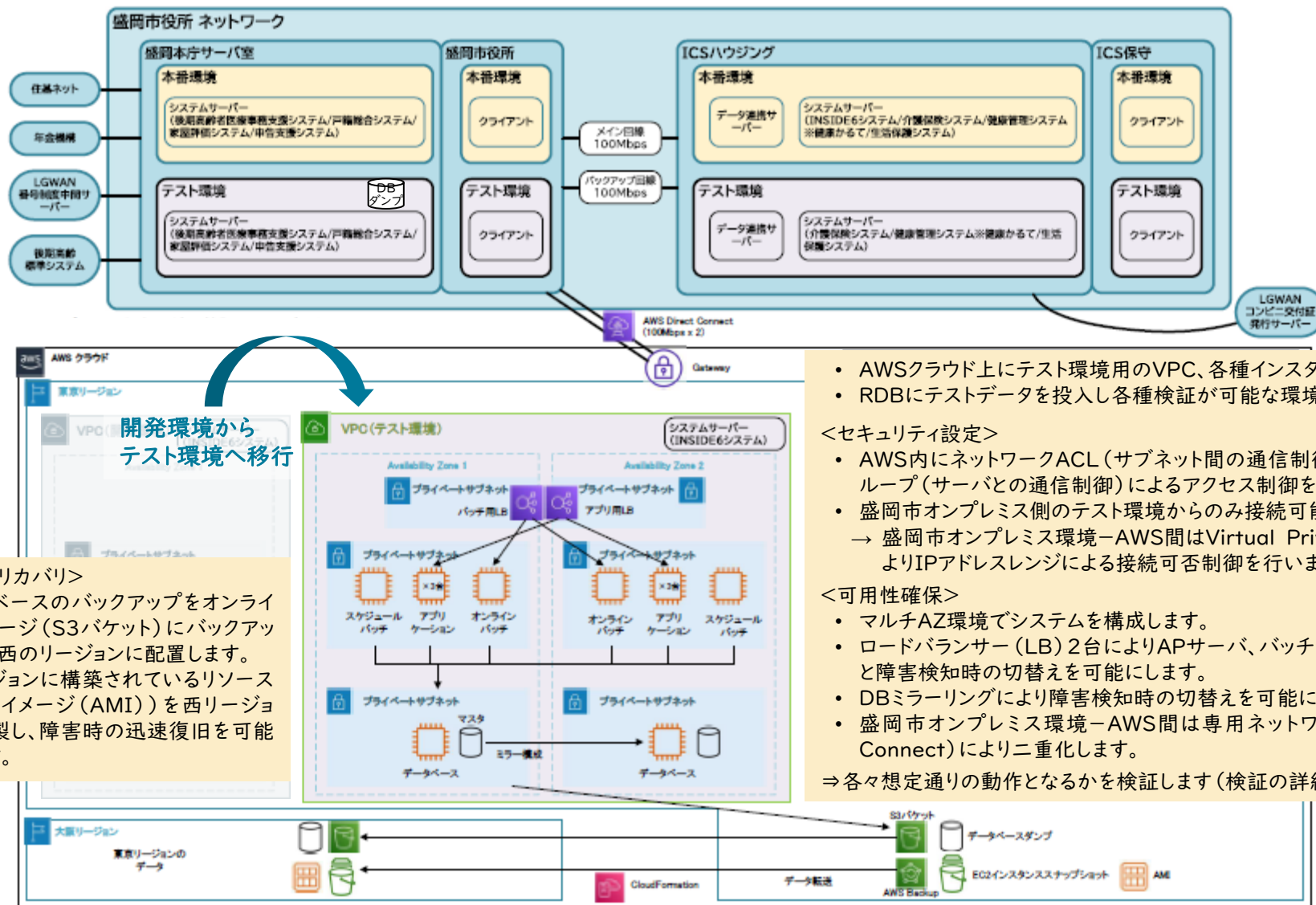
<ガバメントクラウド環境>

- AWSクラウド上に開発環境用のVPC (Virtual Private Cloud)、各種インスタンス(仮想サーバー)を作成します。
- 現行パッケージシステムのサーバー設定に従い、VPC上のサーバに設定を行います。
- RDB (SQLServer) を構築します。
- 現行パッケージシステムの実行環境を構築します。

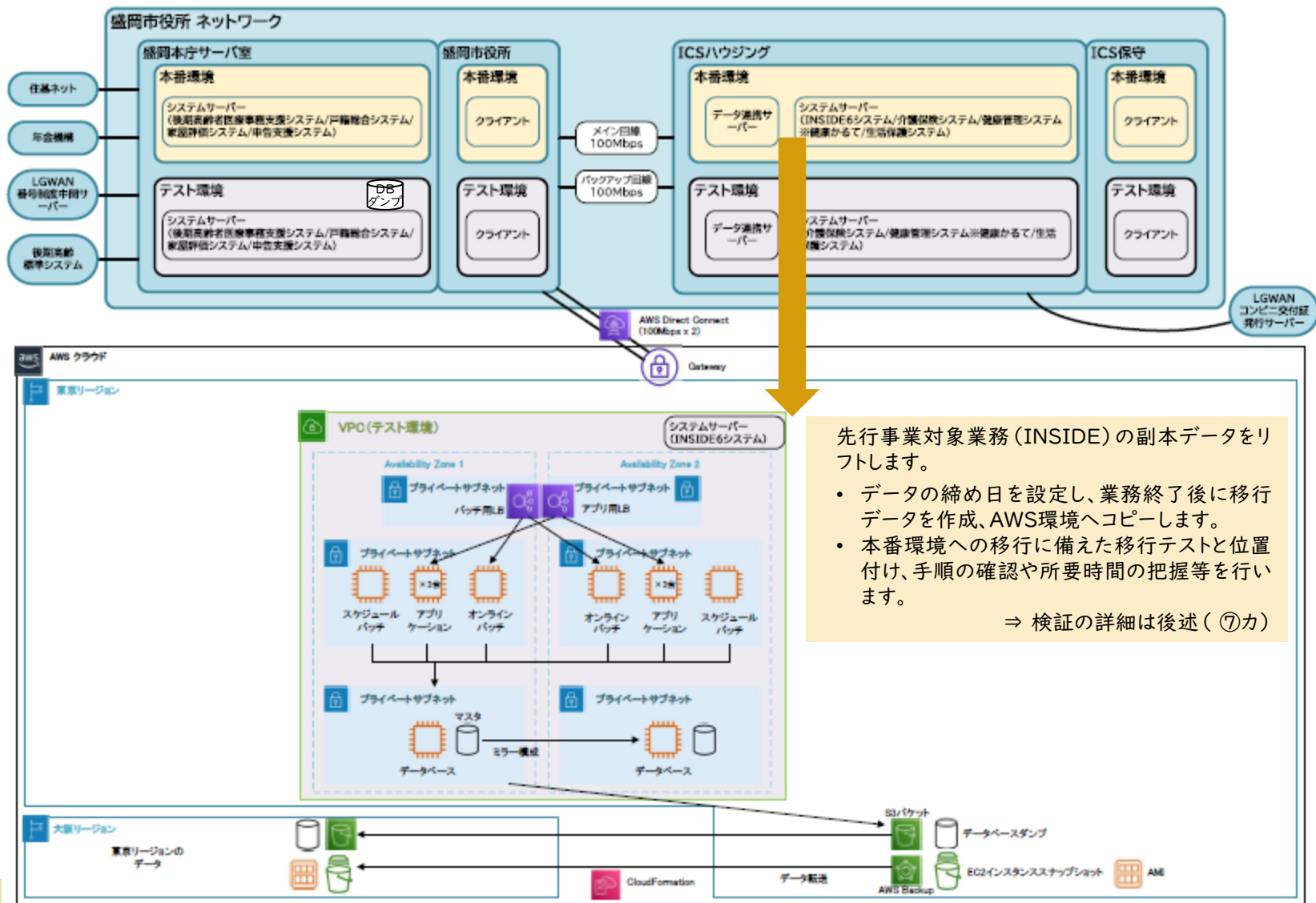
<盛岡市環境>

- 盛岡市およびICSにテスト用クライアントを設置します。
- 盛岡市オンプレミス環境にテスト用サーバ環境を構築します。

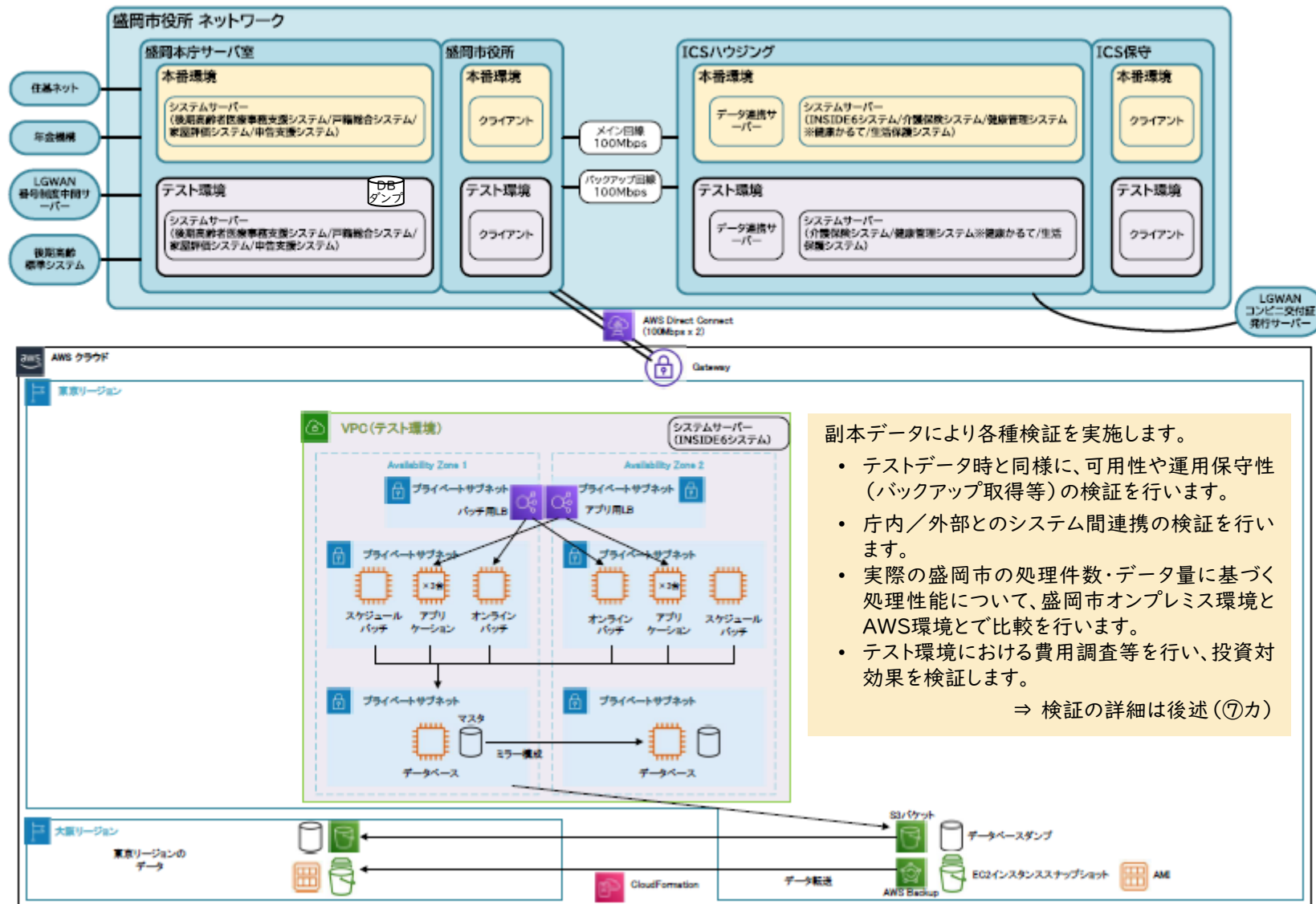
ガバメントクラウドにリフトする手順②：クラウド・回線の検証（テストデータ）



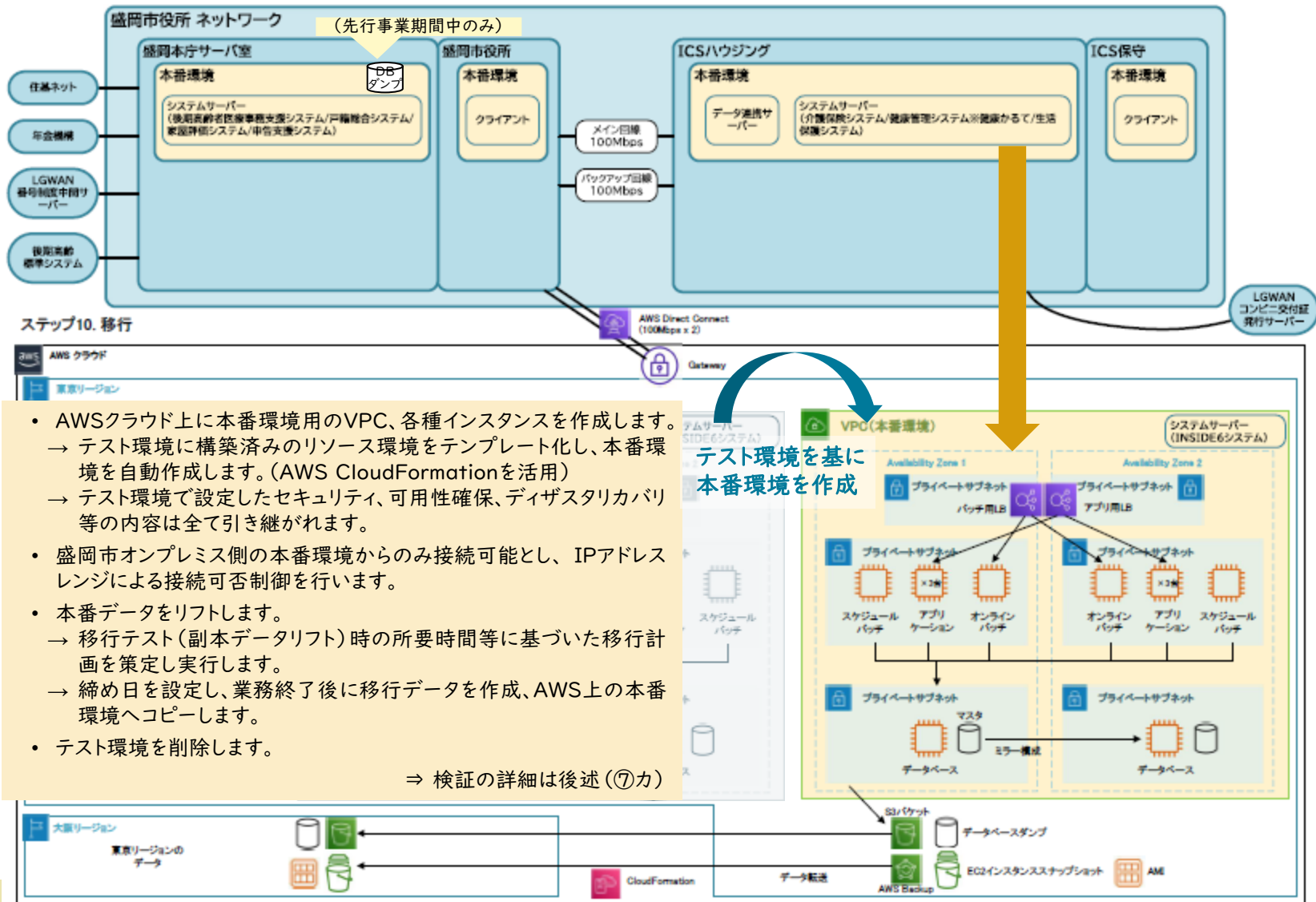
ガバメントクラウドにリフトする手順③：データのリフト



ガバメントクラウドにリフトする手順④：クラウド・回線の検証（移行前検証）



ガバメントクラウドにリフトする手順⑤：移行

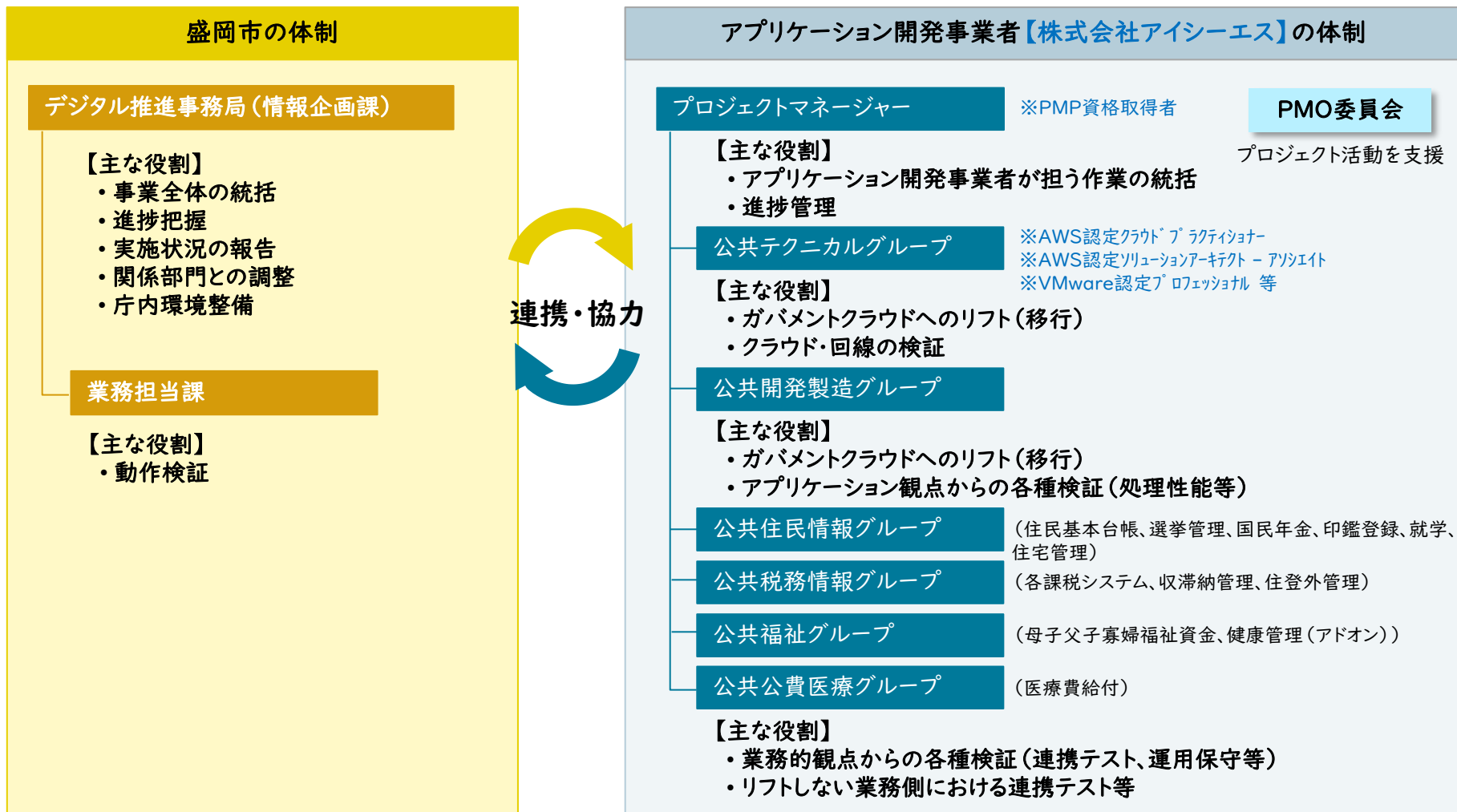


R3～R4先行事業実施スケジュール

No	工程	主な作業項目	2021年度(令和3年度)												2022年度(令和4年度)											
			4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
1	構築											■	■	■												
		ガバメントクラウド上に開発環境構築 アプリケーションとDBを配置																								
2	クラウド・回線の環境構築													■	■	■	■									
		ガバメントクラウド上にテスト環境構築 アプリケーションとDBを配置																								
		回線設定																								
		回線の切替検証																								
		DBにテストデータインポート																								
		テストデータを用いた各種検証(セキュリティ、可用性、性能・拡張性)																								
3	データのリフト(副本データ)																	■								
		DBに副本データをリフト																								
4	クラウド・回線の検証																		■	■	■	■				
		副本データを用いた各種検証(セキュリティ、可用性、性能・拡張性)																								
		回線の検証																								
		投資対効果の検証																								
5	移行																					■	■	■		
		ガバメントクラウド上に本番環境構築 アプリケーションとDBを配置																								
		DBに本番データをリフト																								
		各種連携関連設定を変更																								
		ガバメントクラウド環境総合テスト																								
6	ガバメントクラウド本稼働																							■	■	■

協力するアプリケーション開発事業者名、及び先行事業の実施体制

盛岡市導入パッケージシステムの開発・保守・運用支援を実施している「公共システム事業本部」の各グループとの連携・協力体制を整えています。



標準非機能要件の検証

「地方自治体の業務プロセス・情報システムの非機能要件の標準（標準非機能要件）」が求める非機能要件のうち、「ガバメントクラウド上に構築する環境での検証が必要な項目」について、検証の内容・方法・タイミングを示します。

● セキュリティ

項番	項目	メトリクス(指標)	検証内容	検証方法	検証タイミング
E.5.2.1	アクセス・利用制限	システム上の対策における操作制限	アプリケーションが動作するのに必要な最小限のポートのみ解放されていることを確認する	<ul style="list-style-type: none"> 仮想マシンのセキュリティグループ（ファイアーウォール）で、インバウンド側のトラフィックに対し、サービス提供に必要なポートのみ解放を行うよう設定する。 APサーバにおいては、送信元としてロードバランサのオブジェクトを許可対象とし、ウェブサービスのポートを指定した許可とする。 DBサーバにおいては、送信元の許可は、APサーバとなる。APサーバがオートスケーリングで自動拡張された場合においても、追加となったAPサーバが自動で許可となるよう、送信元をAPサーバのセキュリティグループを指定する事で、許可設定を行う。 連携処理においては、連携先サーバとの通信のみ許可されるよう設定を行う。 <p>⇒ 上記について許可されていない通信が行われなかったことを確認する。</p>	テストデータ時
E.7.1.1	不正追跡・監視	ログの取得	サーバ、ストレージ等へのアクセスに対するログが取得できていることを確認する	<ul style="list-style-type: none"> クラウドのサービス(AWS Config、AWS CloudTrail)によりログを取得する。 <p>⇒ AWS Configでは、AWSリソースの設定が「いつ、どの値が、どのように変更が加えられたか」を追跡できることを確認する。</p> <p>⇒ AWS CloudTrailでは、AWSマネジメントコンソールへのログイン履歴、アクセス権の変更、ネットワークセキュリティ設定の変更、インスタンスへのアクセス・起動/停止操作といったAWSアカウントの一連の操作の追跡ができることを確認する。</p>	テストデータ時

※ E.6.1.1、E.6.1.2「データの秘匿」におけるログイン認証およびパスワード保管（認証情報のみ暗号化）については、要件対応済みの現行アプリケーションをそのままリフトするため検証不要と判断しています。

標準非機能要件の検証

● 可用性 (1/3)

項番	項目	メトリクス(指標)	検証内容	検証方法	検証タイミング
A.1.3.1	継続性	RPO(目標復旧地点)(業務停止時)	前日時点のDBバックアップから復旧を行い、前日業務終了時点からシステムが復旧することを確認する	<ul style="list-style-type: none"> RPO(目標復旧地点)のタイミング(前日業務終了時点)でDBバックアップを取得する。 耐障害性の観点から、DBバックアップデータの保管先は、Amazon S3を利用する。 ⇒ 取得したDBバックアップを利用し、IaaS構成のWindowsサーバで、SQLサーバの管理ツールであるSQL Server Management System(SSMS)から、DBリストアが動作することを確認する。	テストデータ時 副本データ時
A.1.3.2	継続性	RTO(目標復旧時間)(業務停止時)	クラウドのリカバリサービスを利用し、バックアップデータの復旧にかかる時間を計測し、実運用に耐えられるかを検証する	<ul style="list-style-type: none"> 東リージョン上のマルチAZ構成のいずれかが障害となった場合を想定し、クラウドのリカバリサービス(AWS Backup)を用いてバックアップの取得・保管・復旧を検証する。 東リージョンにおいて、以下の3つの構成要素をバックアップする。バックアップの保管先は東リージョン、西リージョンを想定する。 <ul style="list-style-type: none"> ✓ クラウド基盤に作成する「仮想マシン構成情報、ネットワーク設定情報、ロードバランサ、各種リソースへのアクセス権設定」等の構成情報を、Cloud Formationでエクスポートを行い、西リージョンに複製を行う ✓ 仮想マシンのバックアップイメージ、テンプレートイメージについて、AWS Backupでデータを取得し、西リージョンに複製を行う ✓ データベースのダンプイメージについて、Amazon S3バケットのクロスリージョン複製によって、西リージョンに複製を行う ⇒ 上記に要する時間を計測し、実運用に耐えられるかを検証する(RTOに対する国の指針は12時間以内だがさらに短時間とする考え) ⇒ 取得されたデータを用いて全システムが利用できることを確認する。	テストデータ時 副本データ時
A.1.3.3	継続性	RLO(目標復旧レベル)(業務停止時)	クラウドのリカバリサービスを利用し、全システムが復旧できることを確認する		テストデータ時 副本データ時

標準非機能要件の検証

● 可用性 (2/3)

項番	項目	メトリクス(指標)	検証内容	検証方法	検証タイミング
独自	継続性	縮退運転	片系AZのシステムを止め、クラウド上に構築したもう片方のAZのみでシステム継続が可能なことを確認する	<ul style="list-style-type: none"> マルチAZ環境でシステムを構成する。具体的には、APサーバ、DBサーバをそれぞれのAZに配置し、片系AZ障害時にシステムが継続稼働する事を確認する。 ⇒ 片方のAZでAPサーバ、DBサーバを停止した際、以下の動作となることを確認する。 <ul style="list-style-type: none"> ✓ ロードバランサが、APサーバへのヘルスチェック失敗を検知し、トラフィック分散処理の対象から自動で外れること ✓ ミラー構成のDBサーバで、プリンシパルサーバが自動で切り替わり、APサーバからのDB接続要求が継続動作可能なこと 	テストデータ時
A.1.4.1	継続性	システム再開目標(大規模災害時)	東リージョンで稼働しているサーバのスナップショットを西リージョンにコピーし、そのスナップショットで西リージョンでサーバが再現・復旧できることを確認する	<ul style="list-style-type: none"> クラウド基盤のバックアップ保管の考え方は上記A.1.3.3の通り。 東リージョン障害時は、西リージョンに複製されたバックアップデータを利用し、西リージョン上でCloud Formationによりクラウド基盤を復旧する。 AWS Backupで仮想マシンイメージをリストアし、データベースはS3バケットに保管されたデータを利用して復旧を行う。 ⇒ 上記の通り復旧できること、システムが利用できることを確認する。 	テストデータ時
独自	継続性	回線切替え	2重の専用線(稼働系、待機系)の稼働系を停止させ、待機系に切り替わることを確認する	<ul style="list-style-type: none"> 盛岡市オンプレミス側ネットワーク機器にて通信経路の切り替わりが制御できることを確認する。BGPプロトコルを用いて実現する。 稼働系の回線切断時に自動的に待機系に切り替わることを、また回線復旧時にも自動的に稼働系へ切り戻ることを確認する。 回線の切り替わり/切り戻し時の通信停止時間を計測し、停止時間が発生しない、または最小となるよう検証を行う。 	テストデータ時

標準非機能要件の検証

● 可用性 (3/3)

項番	項目	メトリクス(指標)	検証内容	検証方法	検証タイミング
A.1.5.1	継続性	稼働率	システムの稼働率が99.5%以上であることをシステム構成で説明できるようにする	<ul style="list-style-type: none">1つのAZでサーバを構成した場合、稼働率は90%とされている。サーバを2つのAZに配置する事で、システム全体の稼働率を99.99%にする。2つのAZに配置するために、APサーバへの接続はロードバランサ経由でトラフィックを分散し、DBサーバについては、AZをまたいだミラー構成とする。 ⇒ 前ページの「縮退運転」に記載の通り、片系AZがダウンしても、もう片方のAZでシステムが動作することを確認する。	テストデータ時

標準非機能要件の検証

● 性能・拡張性 (1/2)

項番	項目	メトリクス(指標)	検証内容	検証方法	検証タイミング
B.2.1.4	性能目標値	通常時オンラインレスポンスタイム	通常時を想定し、住民情報検索及び照会機能が概ね3秒以内に結果が返ることを確認する	<ul style="list-style-type: none"> 設計上のサーバー性能状況下で検証を行う。 性能が不足している場合には、CloudWatchのパフォーマンス監視によりボトルネックとなる箇所を特定を行う。 その箇所により、自動スケーリングによるスケールアウトで対応できるか、インスタンスタイプ・ディスクタイプなどの変更により性能向上を図る必要があるか、ネットワーク回線の増速が必要かといった対応策を検証する。 <p>⇒ 副本データを対象としたオンラインレスポンスが3秒以内になるまで上記検証を繰り返す。</p>	副本データ時
B.2.1.5	性能目標値	アクセス集中時のオンラインレスポンスタイム	アクセス集中時を想定し、住民情報検索及び照会機能が概ね5秒以内に結果が返ることを確認する	<ul style="list-style-type: none"> DBに負荷のかかる処理(広範囲を対象としたデータ抽出等)を実行するなど処理集中時を再現した上で通常時(上記B.2.1.4)と同様の検証を行う。 	副本データ時
B.2.2.1	性能目標値	通常時バッチレスポンス順守度合い	通常時を想定した定例的なバッチ処理を実行し、再実行の余裕が確保できることを確認する	<ul style="list-style-type: none"> 設計上のサーバー性能状況下において、現行環境で実施している夜間バッチ処理を実行する。 実行途中でエラーを発生させ、ロールバックでデータが処理前に戻ることを確認後、再びバッチ処理を実行する。 現行環境下と同様の運用スケジュールの範囲で終了できることを確認する。 性能が不足している場合には、上記B.2.1.4と同様の方法で対応策を検証する。 	副本データ時
B.2.2.2	性能目標値	アクセス集中時のバッチレスポンス順守度合い	アクセス集中時及びピーク時のバッチ処理を実行し、再実行の余裕が確保できることを確認する	<ul style="list-style-type: none"> DBに負荷のかかる処理(広範囲を対象としたデータ抽出等)を実行するなど処理集中時を再現した上で通常時(上記B.2.2.1)と同様の検証を行う。 業務開始に影響を及ぼさない時間までに終了できるか、再実行の余裕が確保できるかを確認する。 性能が不足している場合には、上記B.2.1.4と同様の方法で対応策を検証する。 	副本データ時

標準非機能要件の検証

● 性能・拡張性 (2/2)

項番	項目	メトリクス(指標)	検証内容	検証方法	検証タイミング
独自	性能目標値	初回起動に要する時間	バージョンアップ直後の初回起動時に要する時間が現行環境下と遜色ないことを確認する	<p>(前提)</p> <ul style="list-style-type: none"> リフトする現行システムは画面制御にかかる実行環境をクライアント側に持つ。そのため、初回起動時のみClickOnce方式でプログラム配布が実行される。 AWSとの接続に際し、現行のサーバー(ハウジング)ークライアント(庁舎)間を結ぶWAN回線と同等の帯域(100Mbps×2回線)及びサービスを採用する。 <p>(検証)</p> <ul style="list-style-type: none"> APサーバにクライアントが利用しているアプリケーションとは異なるバージョンのアプリケーションを配置し、クライアント50台の起動に要する時間(配布時間)を計測する。 回線のスループットを計測する。 計測した数値および配布データ量と、盛岡市の運用状況(クライアント数及び同時起動数)を基に推測し、現行環境下と遜色ない運用が可能であることを確認する。 	テストデータ時

標準非機能要件の検証

● 移行性

項番	項目	メトリクス(指標)	検証内容	検証方法	検証タイミング
D.1.1.2	移行時期	システム停止可能日時	副本データの本番環境へのコピー、及び連携関連の設定変更が6時間以内(夜間)に終わることを確認する	<ul style="list-style-type: none"> 現状のDBダンプサイズ(約140GB)から想定される移行時間(下記)と実測値を比較検証する。 <ul style="list-style-type: none"> ✓ DBエクスポート ✓ 100Mbps回線でデータ転送 ✓ DBインポート データ転送中に連携関連の設定変更を行う。 ⇒ 上記作業にかかる時間を計測する。	副本データ時
D.1.1.3	移行時期	並行稼働の有無	ガバメントクラウドを稼働系、現行オンプレミス環境を待機系とし、前日業務終了時点のデータで待機系システムが稼働することを確認する	<ul style="list-style-type: none"> ガバメントクラウド上のシステムで夜間のバッチ処理完了後に、DBダンプを出力する。 その後、現行のオンプレミス環境にDBダンプを転送し、オンプレ側DBサーバでダンプをインポートする。 ⇒ 上記作業にかかる時間を計測する。 ⇒ インポートしたデータ(前日業務終了時点のデータ)でシステムが稼働することを確認する。	テストデータ時 副本データ時

標準非機能要件の検証

● 運用・保守性

項番	項目	メトリクス(指標)	検証内容	検証方法	検証タイミング
C.1.2.5	通常運用	バックアップ取得 間隔	日次でDBバックアップが取得されていることを確認する	<ul style="list-style-type: none"> IaaS構成でSQLサーバを構成するため、WindowsサーバにインストールしたSQLサーバの管理ツールであるSSMSで、DBバックアップを取得する。 DBダンプの取得タイミングは、RPO(目標復旧時点)のタイミング(前日業務終了時点)と合わせる。 DBダンプの保存先は、Amazon S3とする。 ⇒ 上記について、設定通り取得されていることを確認する。	テストデータ時 副本データ時
C.1.3.1	通常運用	監視情報	死活監視、エラー監視、リソース監視、パフォーマンス監視ができることを確認する	<ul style="list-style-type: none"> クラウド基盤で利用可能なツール(Amazon CloudWatch)を利用し、各種監視を行う。 ⇒ 死活監視では、ステータスチェックによりEC2インスタンスの稼働状態を監視できることを確認する。 ⇒ エラー監視では、システムの稼働に必要なプロセスが起動していること、システムが出力するログから異常を検知できることを確認する。 ⇒ リソース/パフォーマンス監視においては、EC2インスタンスのCPU/メモリ/ディスク使用率が監視できることを確認する。 監視により異常が検知された場合は、インスタンスの自動回復(別ホスト)や自動スケーリングを行うとともに、Amazon SNSにより通知できることを確認する。 	テストデータ時 副本データ時

※ C.4.5.1「運用環境」における外部システムとの接続確認については、次ページの連携検証に記載します。

標準準拠システムの移行方法の検証

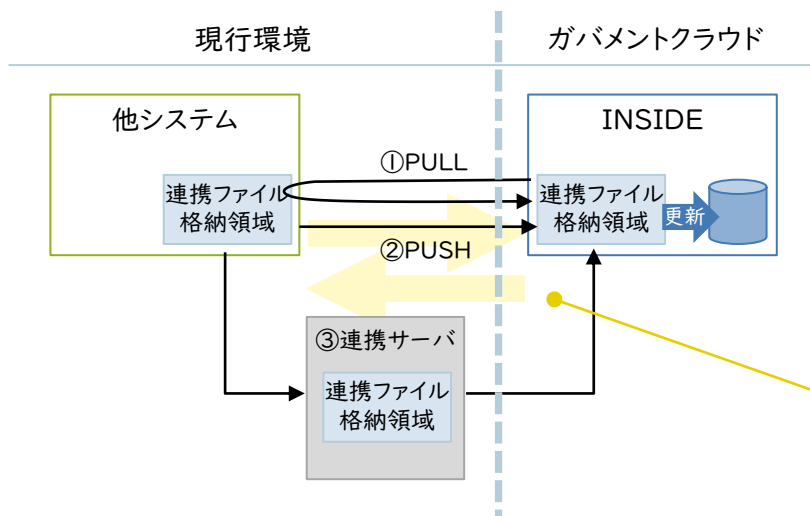
● ガバメントクラウドにリフトしたシステムとリフトしないシステムとの連携検証（1/2）

【基本的な考え方】

- ・ リフトしないシステム（ハウジングまたは盛岡市オンプレミス環境として残るシステム）の環境は変更なし
- ・ リフトする／しないに関わらず、業務システムアプリケーションの業務仕様には変更なし

以上のことから、業務機能、及びガバメントクラウドにリフトしたシステム（INSIDE）としないシステム（他システム）との間に存在する連携プログラムは正しく動くことを前提として下記の連携パターンを検証し、連携先（或いは連携元）の業務システムがガバメントクラウド上にあっても、これまで通りの連携が実現できることを確認します。

<パターン1> ファイル連携



連携パターンとしてはファイル連携が最も多く、以下の3つのケースが存在します。

- ① 他システムが作成し格納領域に保管された連携ファイルをINSIDEサーバが取りに行くケース
- ② 他システムが作成しINSIDEの格納領域に送信するケース
- ③ 他システムが作成した連携ファイルを連携サーバに格納し、INSIDEサーバが取りに行くケース

※他システム⇄INSIDEが逆のケースもあり

いずれの場合も、以下の2点を確認することで連携検証を行います。

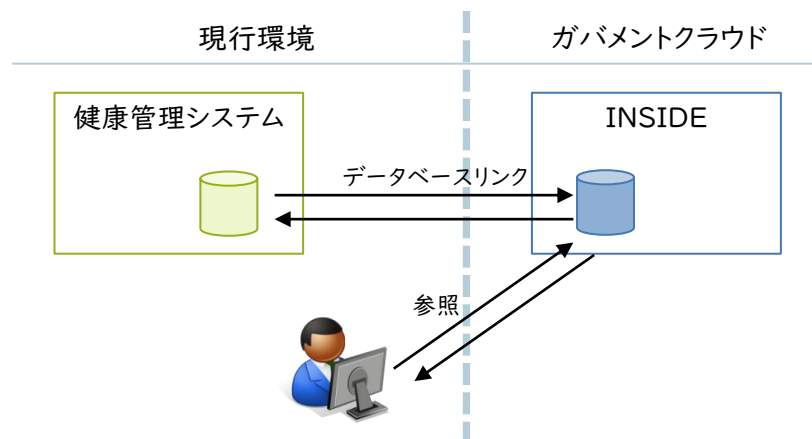
- ・ 連携ファイル格納領域（指定フォルダ）への疎通確認
- ・ 指定フォルダのアクセス権とファイルが格納できることの確認

※連携ファイル作成及び格納後の処理には変更がないため検証不要

標準準拠システムの移行方法の検証

● ガバメントクラウドにリフトしたシステムとリフトしないシステムとの連携検証（2/2）

<パターン2> データベースリンク



INSIDEから健康管理システムのDBをリンクし参照しているケースでは、以下の検証を行います。

- 健康管理システムサーバ(DB)への疎通確認、アクセス権確認
- INSIDEシステム画面に参照データが表示されることの確認

<パターン3> 外部接続

● 自治体中間サーバ

J-LISへテスト環境利用の申請を行い、従来通りINSIDEマイナンバー管理機能を通じて中間サーバを介した情報連携ができることを確認します。

- 住基ネット
- コンビニ交付（証明発行サーバ）

本運用中のため疎通確認のみ実施します。本番処理時に立会いを行い、不具合が発生した場合には迅速なリカバリを行うことで対応します。

- eLTAX
- 年金機構

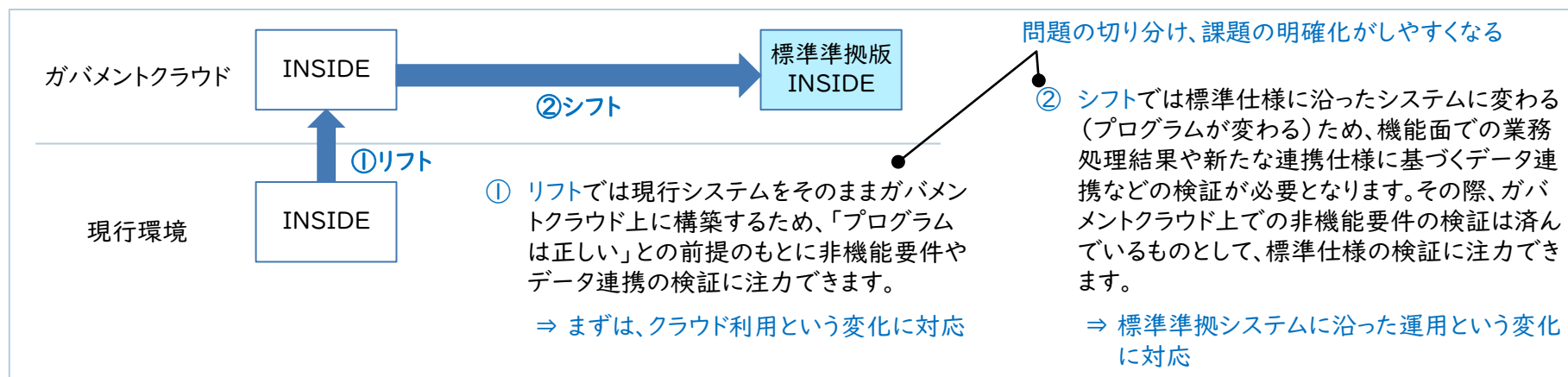
担当職員による手動連携となっているため、検証は不要と判断しています。

標準準拠システムの移行方法の検証

● ガバメントクラウドにリフトしてから標準準拠システムにシフトする方法の有用性の検証

盛岡市では基幹系業務システムをハウジングおよび自庁へのサーバ設置により運用しており、クラウド利用の実績はありません。また、複数ベンダの複数のパッケージシステムから構成されているため複雑なデータ連携を行っています。

自治体システム標準化で求められる「システムが稼働するプラットフォームの変更（ガバメントクラウド）」「システムそのものの変更（標準準拠システム）」の2点を盛岡市として確実に実施するために、まずは先行事業においてリフトを行い、その後シフトを行う方法が下記の観点において有用であると考えています。



盛岡市では⑥の移行計画に記載の通り、先行事業対象業務以外の業務についても順次リフト、シフトを実施する予定です。リフト・シフトを同時に行うことが可能な場合であっても、上記の有用性を検証することで同様の手順で進めることができると考えます。

本事業計画に記載のリフト手順や、ネットワーク接続、セキュリティ、システム間のデータ連携等にかかる各種検証等について、当初計画（想定）通りに行ったこと／行かなかったことについて体系立てて整理を行い、ポストモテムを実施します。

- ・ リフト手順やスケジュールの妥当性
- ・ 検証項目や範囲が適切だったか
- ・ 検証方法の妥当性
- ・ 想定外の課題事項とその解決手法 等々

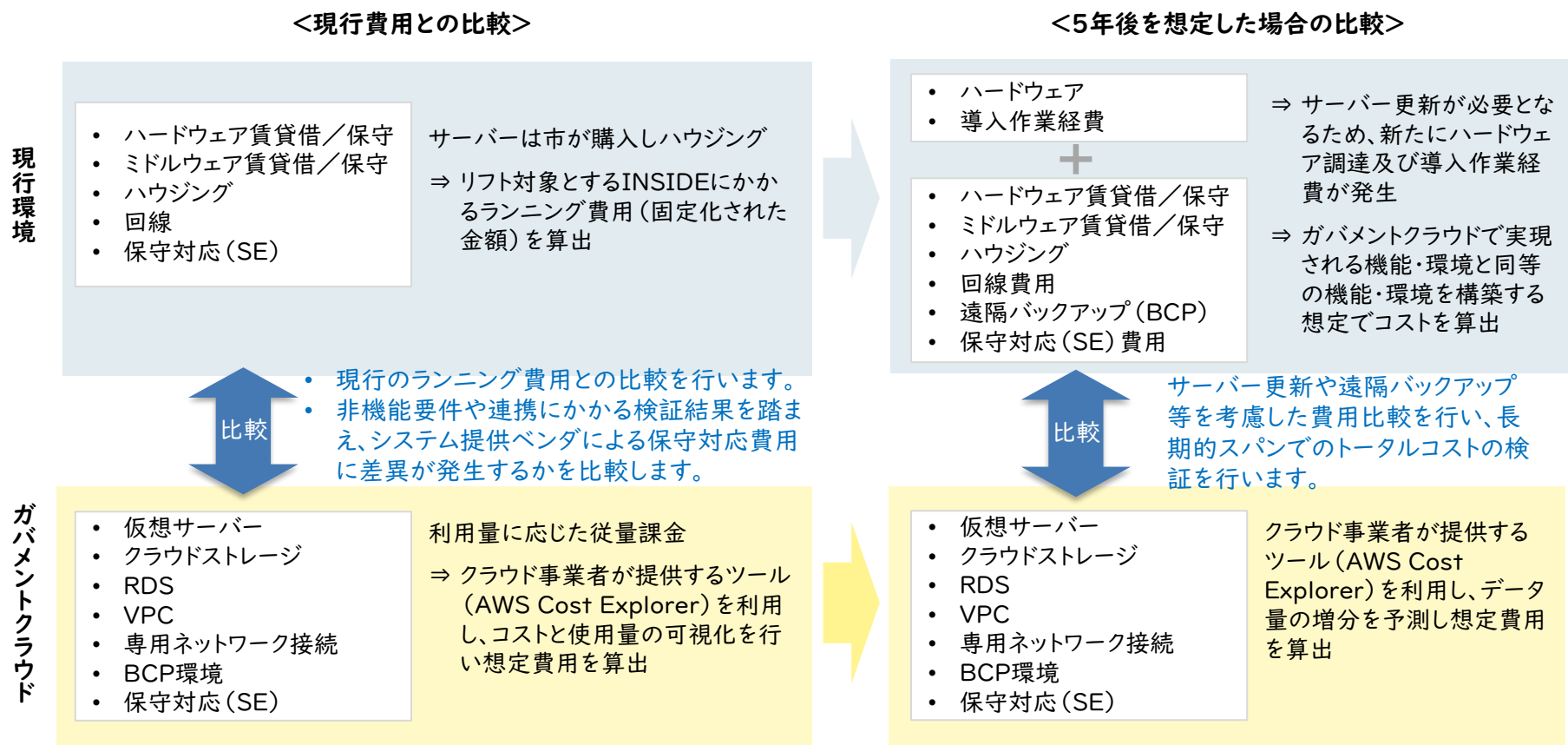
活かすべき事項、改善すべき事項などを体系立てて整理し、先行事業で得られた知見をシフトや次の業務システムのリフト・シフトに活かす

投資対効果の検証

投資対効果について、「費用」「KPI」の両面から検証を行います。

● 費用の検証

現行費用との比較、5年後を想定した場合との比較を実施します。



投資対効果の検証

● KPIの検証

ガバメントクラウド上に構築する環境において新たに実現される項目についてKPIを定義し、その効果を検証します。

<KPI (評価指標)>

- ① サーバー環境の障害復旧に要する作業と時間
- ② APサーバ増強に要する作業と時間、費用
- ③ 遠隔バックアップの取得方法と費用

<現行環境で実現する場合>

- ① 障害切り分けから保守要員、保守部品の手配、再構築等が発生
- ② APサーバの費用、ロードバランサーの再構築が発生
- ③ 遠隔バックアップ先の選定、利用料、回線費用が発生



<ガバメントクラウド上に構築する環境>

- ① クラウド基盤に作成する「仮想マシン構成情報、ネットワーク設定情報、ロードバランサ、各種リソースへのアクセス権設定」等の構成情報を、Cloud Formationでエクスポート
- ② Auto Scalingにより増強可能。使用状況に応じて削減も可能。
- ③ AWS Backupにより西リージョンにバックアップデータを複製

R5年度追加検証について

● 追加検証INDEX

1. より効率的な構成検証

1-1. マネージドサービスの活用など、よりコストメリットや運用効率性が享受できる構成検証

2. 運用効率化の検証

2-1. リソース管理指標についての指標値の検討

2-2. 指標値達成状況の可視化の検証

2-3. R4年度の構築～運用で発生した課題整理と解決策の検討

3. 投資対効果の検証

3-1. より効率的な構成への移行及び運用効率化により得られる効果の検証

3-2. 共同利用方式を想定した按分効果の検証

4. ネットワーク接続の在り方検討

4-1. 主に共同利用方式で想定される自治体毎の利用パターンに応じた接続構成の在り方の検討

1.より効率的な構成検証

1-1.マネージドサービスの活用など、よりコストメリットや運用効率性が享受できる構成検証

◆サーバーレス化の模索によるコストメリット等の検証

- サーバーレス化に向けて、対応サービスの選定、サーバーレス化の検討、プロトタイプ作成、検証環境での検証
- サーバーレス化できたものは、コストメリット等の検証。できなかったものは、サーバーレス化への課題検証。

【現時点で想定する対応サービス】

- ・アプリケーション→コンテナ化
- ・データベース→マネージドサービス化
- ・運用関連サードパーティーツール→マネージドサービス化

◆マネージドサービスを利用したセキュアかつ運用効率性向上への模索と検証

- IaCの改善による運用効率性の向上
- マネージドサービスを利用した、通信トラフィックの暗号化、認証情報の見直し、日常運用自動化の模索

【現時点で想定する案】

- ・IaCのコードをyamlからCDK化
- ・マネージドサービスで認証局(CA)を建て、通信をHTTPS化
- ・マネージドサービスを利用したアプリケーション配信の自動化

2.運用効率化の検証

2-1.リソース管理指標についての指標値の検討

- デジタル庁が示す以下の管理指標について指標値を検討する

ガバメントクラウドは「**迅速、柔軟、かつセキュアでコスト効率の高いシステム**」の構築を目的とし

4つのキーワードを包含・拡張した11項目の管理指標を設定

1	Cost (コスト効率、最適化)	7	Resiliency (レジリエンシー)
2	Elasticity (弾力性)	8	Observability (可観測性)
3	Performance (パフォーマンス)	9	Transparency (透明性)
4	Agility (俊敏性)	10	Improvability (改善性)
5	Velocity (ベロシティ)	11	Automation (自動化)
6	Security (セキュリティ)		

2-2.指標値達成状況の可視化の検証

- 2-1で検討した指標値の達成状況が可視化できる手段の模索と検証
- Cost、Elasticity、Performance、Velocity指標に照らし合わせ、盛岡市本番稼働環境のスペックを見直す
⇒EC2インスタンスタイプの見直し、APサーバーのオートスケール台数の見直しなど

2.運用効率化の検証

2-3. R4年度の構築～運用で発生した課題整理と解決策の検討

◆ 構築時の課題整理と解決策の検討

- ポストモーテムを実施し、課題を整理する
- ポストモーテムで出た課題の解決策を検討する

◆ 運用時の課題整理と解決策の検討

- ポストモーテムを実施し、課題を整理する
- オンプレミス時代とのパフォーマンス差などシステム利用者へのヒアリングを実施する
- 本稼働後の障害一覧や問い合わせ一覧を整理する
- 以上の整理で出た課題の解決策を検討する

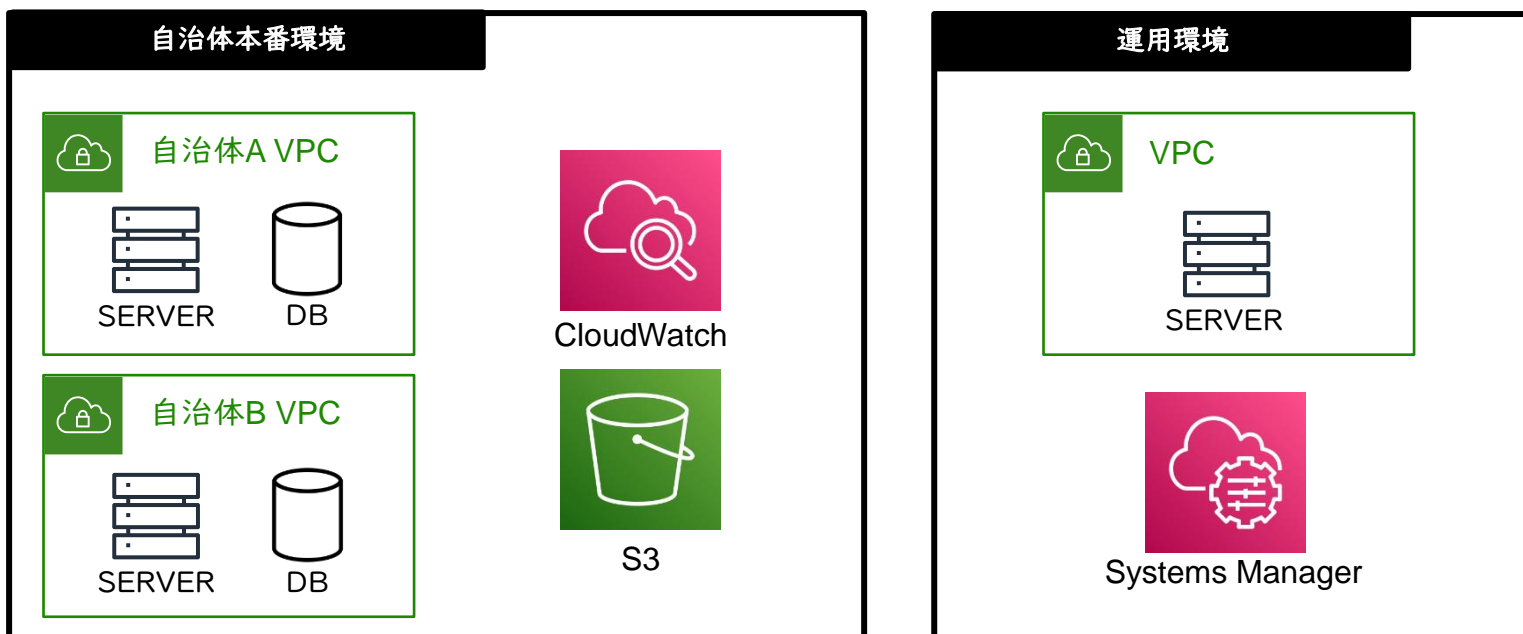
3.投資対効果の検証

3-1.より効率的な構成への移行及び運用効率化により得られる効果の検証

- 「1.より効率的な構成検証」「2.運用効率性の検証」で得られた結果をまとめ効果を検証する

3-2.共同利用方式を想定した按分効果の検証

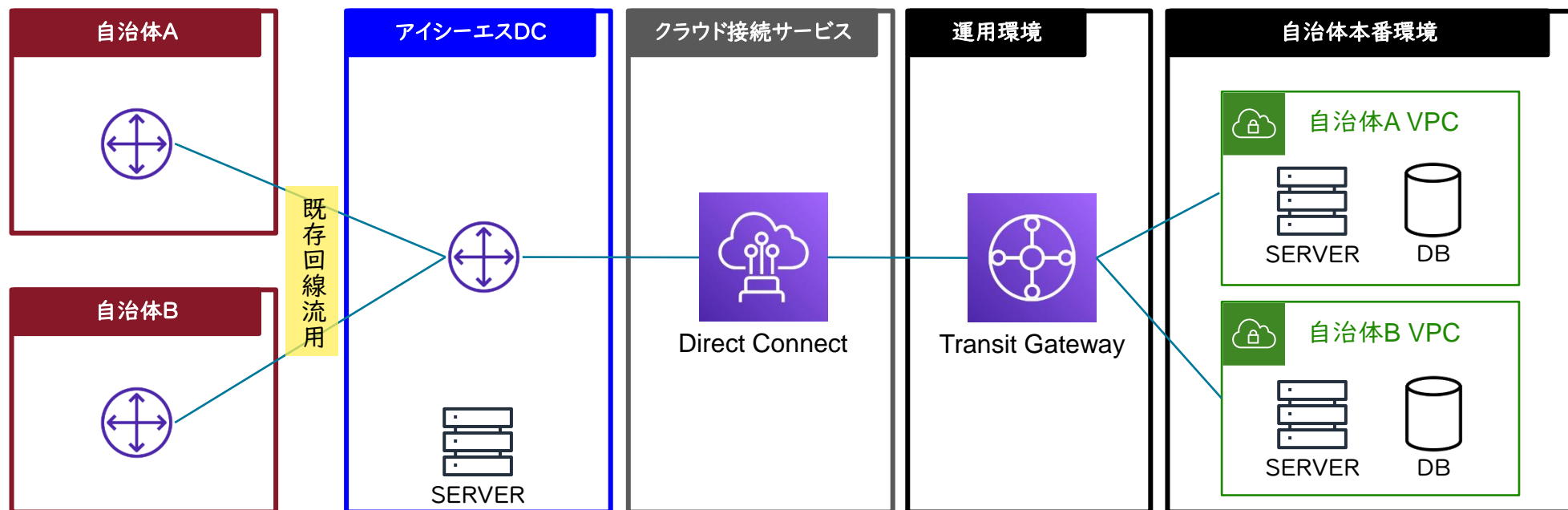
- アプリケーション開発事業者である株式会社アイシーエスでは「**ネットワーク分離**」方式による共同利用方式を想定
「ネットワーク分離」方式による按分効果の机上検証を実施する



4.ネットワーク接続の在り方検討

4-1.主に共同利用方式で想定される自治体毎の利用パターンに応じた接続構成の在り方の検討

- アプリケーション開発事業者である株式会社アイシーエスでは「**データセンタ共同利用**」方式による接続を想定
「データセンタ共同利用」方式による接続構成の在り方を検討する



R5年度先行事業実施スケジュール

No	工程	主な作業項目	2023年度(令和5年度)												
			4	5	6	7	8	9	10	11	12	1	2	3	
1	より効率的な構成検証			■	■	■	■	■	■	■	■	■	■	■	
	1-1	マネージドサービスの活用など、よりコストメリットや運用効率性が享受できる構成検証													
2	運用効率化の検証		■	■	■	■					■	■	■	■	
	2-1	リソース管理指標についての指標値の検討													
	2-2	指標値達成状況の可視化の検証													
	2-3	R4年度の構築～運用で発生した課題整理と解決策の検討													
3	投資対効果の検証					■	■	■				■	■	■	
	3-1	より効率的な構成への移行及び運用効率化により得られる効果の検証													
	3-2	共同利用方式を想定した按分効果の検証													
4	ネットワーク接続の在り方検討							■	■	■					
	4-1	主に共同利用方式で想定される自治体毎の利用パターンに応じた接続構成の在り方の検討													

R5年度先行事業実施における環境

- 令和4年1月から盛岡市はガバメントクラウドで本番稼働を迎えており、令和5年度先行事業の検証においては本番に影響を与えないよう、R5年度先行事業検証用VPCを新設し検証を実施する

