

平成 23 年 6 月 30 日  
情報連携基盤技術ワーキンググループ 構成員  
長島 哲也

## ユースケース検討と並行して実施可能な情報連携基盤設計アプローチ

### 1 はじめに

第5回情報連携基盤ワーキンググループでは、複数の構成員から「リンクコード変換の性能にかかる理論値」についての報告がなされましたが、実際に可逆暗号化関数によるコード変換サンプルプログラムを作成し、実測してみましたので、その結果を報告するとともに、当ワーキンググループにおける重要な検討項目である“情報連携基盤システムの全体アーキテクチャー”の検討を、標準的なシステム化検討プロセス（標準的なアーキテクチャー構築手法）を用いて、利用事務（業務ユースケース）の検討と並行して行った結果が参考となるのではないかと考え、報告いたします。

なお、情報連携基盤システムの全体アーキテクチャー検討に当たっては、基本方針および要綱に示された情報連携基盤にかかるシステム要件を整理・検討し、さらに諸外国の行政機関等で実績のある情報連携基盤のアーキテクチャーも参照して、情報連携基盤の機能に関する以下の仮説を立て、基本方針で示されている業務ユースケースを用いて、この機能による連携の実現可能性を検討・評価しました。

#### 仮説 1

情報連携基盤は、論理レベルでは業務ユースケースに縛られない仕組みなので、諸外国の行政機関や国内民間企業で実績のある情報連携基盤のアーキテクチャーがほぼ活用できるのではないかと

#### 仮説 2

連携シナリオには法則性（パターン）があり、そのパターンの組み合わせにより、適用業務領域の各業務シナリオが実現できるのではないかと

## 2 当資料の要点

# Javaを用いた可逆暗号化関数サンプルプログラム・コード

```

try {
    System.out.println("Start");

    String init_idcode = "1234567890abcdefghijklmnopqrstuvwxyz"; //暗号対象文字列
    String
    key = "ABCDEFGHIJKLMNQRSTUUVWXYZabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNQRSTUUVWXYZabcdefghijklmnopqrstuvwxyzABCDEFGHIJK
    LMNOPQRSTUUV"; //キー(1024ビット)
    System.out.println("Initial value");
    System.out.println("idcode = " + init_idcode);
    System.out.println("key = " + key);

    SecretKeyFactory keyFac = SecretKeyFactory.getInstance("AESede");
    AESedeKeySpec keySpec = new AESedeKeySpec(key.getBytes());
    SecretKey secKey = keyFac.generateSecret(keySpec);

    //暗号化
    System.out.println("Start Encryption");
    Cipher encoder = Cipher.getInstance("AESede");
    encoder.init(Cipher.ENCRYPT_MODE, secKey);
    byte[] b = encoder.doFinal(init_idcode.getBytes());
    String linkcode = new String(Hex.encodeHex(b));
    System.out.println("linkcode = " + linkcode);
    System.out.println("End Encryption");

    //復号化
    System.out.println("Start Decryption");
    Cipher decoder = Cipher.getInstance("AESede");
    decoder.init(Cipher.DECRYPT_MODE, secKey);
    b = Hex.decodeHex(linkcode.toCharArray());
    String result_idcode = new String(decoder.doFinal(b));
    System.out.println("idcode = " + result_idcode);
    System.out.println("End Decryption");

    System.out.println("End");
} catch (Exception e) {}

```

**① 所与の値:**  
**init\_idcode: IDコード**  
**key: 暗号化キー**

**② IDコードからリンクコードを演算**

**③ リンクコードからIDコードを演算**

**④ 所与の値の表示**

**⑤ IDコードからリンクコードを演算した結果**

**⑥ リンクコードからIDコードを演算した結果**

**⑦ 可逆であることが確認できる**

**暗号化/復号化にかかる時間は1/1000秒以下で問題にならない**

### Javaを用いたサンプルプログラム・コードの実行結果

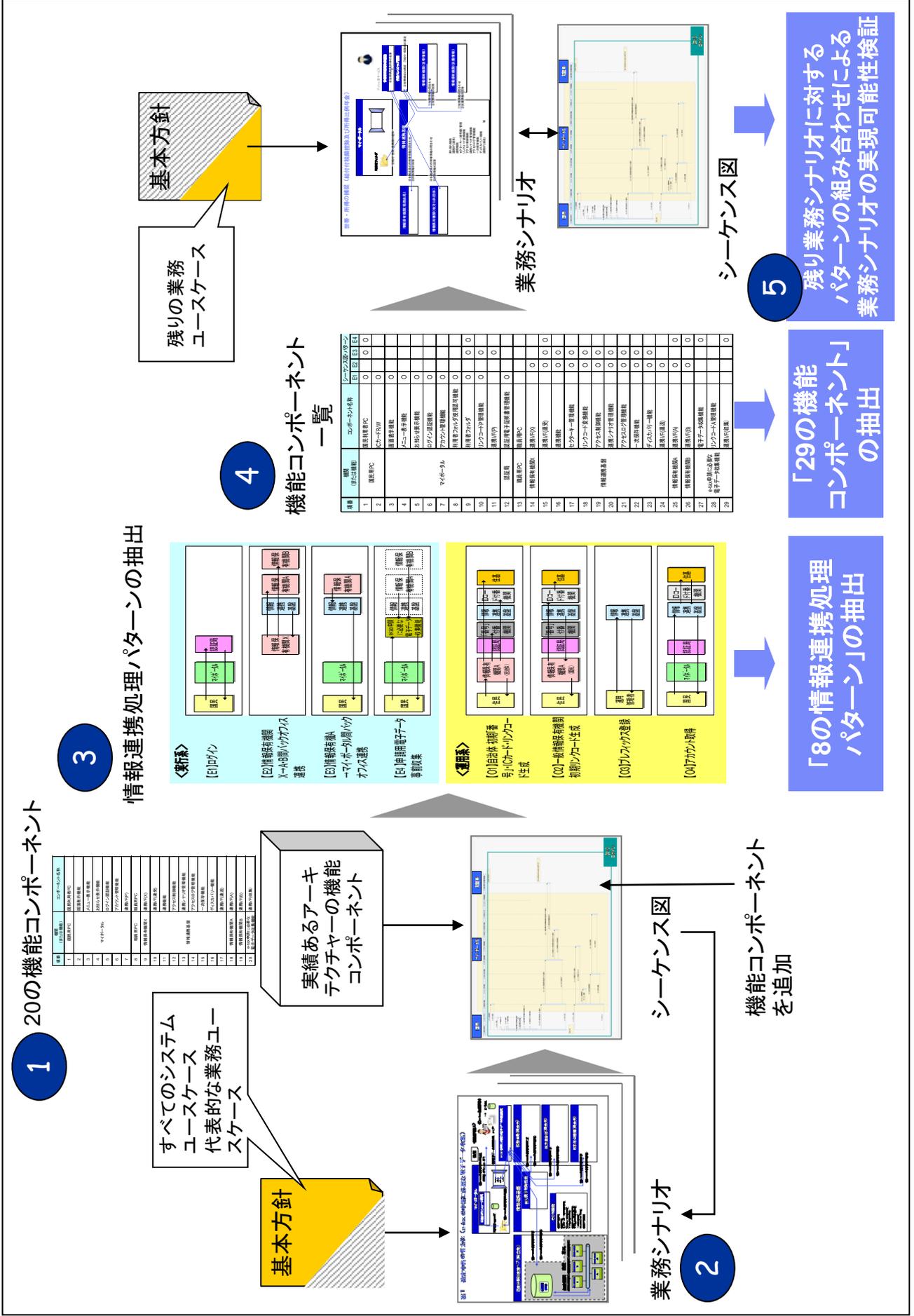
```

[11/04/19 11:06:45:671 JST] +++ Start +++
[11/04/19 11:06:45:671 JST] Initial Value
[11/04/19 11:06:45:671 JST] Initial Value
ABCDEFGHIJKLMNQRSTUUVWXYZabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNQRSTUUVWXYZabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNQRSTUUV
Start Encryption
[11/04/19 11:06:45:671 JST]
[11/04/19 11:06:45:671 JST] linkcode = 038fdb64d17c20dec6bc759ea0d6d4d9542b2766bebf26f6a20744f98efd67cfc04b79980239f120
[11/04/19 11:06:45:671 JST]
[11/04/19 11:06:45:671 JST] idcode = 1234567890abcdefghijklmnopqrstuvwxyz
[11/04/19 11:06:45:671 JST]
[11/04/19 11:06:45:671 JST] End Decryption
[11/04/19 11:06:45:671 JST] +++ End +++

```



# 概念アーキテクチャ策定の詳細



	オーストリア	ベルギー	デンマーク	韓国
番号制度	中央住民登録番号 (CRR 番号)	国民登録番号 (RRN 番号)	国民登録番号 (CPR 番号)	住民登録番号
ID モデル	セクトラルモデル	フラットモデル	フラットモデル	フラットモデル
番号連携	<ul style="list-style-type: none"> <li>「CPR 番号」から不可逆関数にて「ソース PIN」を生成し、ソース PIN から不可逆関数にて「ssPIN」を生成</li> <li>データ保護委員会から発行される情報保有機関の「暗号化された ssPIN」を本人識別子として利用</li> </ul>	「RRN 番号」を用いた連携	「CPR 番号」を用いた連携	「住民登録番号」を用いた連携
連携方式	1:n 型連携	ハブ・スポーク型連携	1:n 型連携 (個別連携)	ハブ・スポーク型連携
情報連携	<ul style="list-style-type: none"> <li>「ポータルグループ」と呼ばれる、アクセス制御の仕組みを、連邦政府・地方自治体あわせて 400 機関が導入</li> </ul>	<ul style="list-style-type: none"> <li>「社会保障クロスロードバンク (CBSS)」と呼ばれる機関が設置され、情報連携基盤を提供</li> <li>約 3,000 機関が CBSS に接続</li> </ul>	<ul style="list-style-type: none"> <li>情報連携基盤といった仕組みは導入されていない</li> <li>個々のアプリケーションが個別に必要な電子情報を交換</li> </ul>	<ul style="list-style-type: none"> <li>「行政情報共同利用センター (PISC)」と呼ばれる機関が設置され、情報連携基盤を提供</li> <li>全ての行政機関 (国と地方) が接続</li> </ul>
認証サービス	<ul style="list-style-type: none"> <li>「市民カード」認証サービスにより、IC カードのみならず携帯電話を活用した認証</li> <li>政府としては、携帯電話利用型を普及させるとしている</li> </ul>	<ul style="list-style-type: none"> <li>「eID カード」による PKI 認証方式とキークード「トークン」を用いた認証方式を併用</li> </ul>	<ul style="list-style-type: none"> <li>キークード「Nem-ID」を用いた認証</li> <li>金融機関といった民間サービスでも利用可能</li> </ul>	<ul style="list-style-type: none"> <li>「公認電子証明書」を用いた PKI 認証方式</li> </ul>
個人情報保護	<ul style="list-style-type: none"> <li>第三者機関としてデータ保護委員会が存在する</li> <li>情報連携における ssPIN の取り扱いや、情報連携の合理性について、検査権限を有する</li> </ul>	<ul style="list-style-type: none"> <li>第三者機関として「プライバシー保護委員会」が存在する</li> <li>プライバシー保護委員会には、6 の部門委員会が設置され、「社会保障と健康に関する部門委員会」が CBSS</li> </ul>	<ul style="list-style-type: none"> <li>第三者機関として「データ保護庁」が存在する</li> <li>行政機関間での連携に関する最終決定権を有する</li> </ul>	<ul style="list-style-type: none"> <li>個人ファイルの情報連携には、「行政安全部」長官による承認が必要</li> </ul>

### 3 検証目的

#### 3.1 標準的なシステム化検討プロセスの適用

一般的に、システム化検討は、「要件と制約の抽出」→「概念アーキテクチャーの策定」→「論理アーキテクチャーの策定」→「物理アーキテクチャーの策定」というプロセスで行います。この過程を通じて、システムの構成要素（コンポーネント）を抽出し、その配置を含めたシステム構造とシステム構成要素の仕様を定義して、その仕様を実現するためのハードウェア製品やソフトウェア製品、および開発対象を特定します。この一連の検討においては、最終的な実装イメージが明確でない初期の検討局面では、実績があるアーキテクチャーを参照することは、高品質で手戻りのない設計・実装を実現する上で有効とされています。

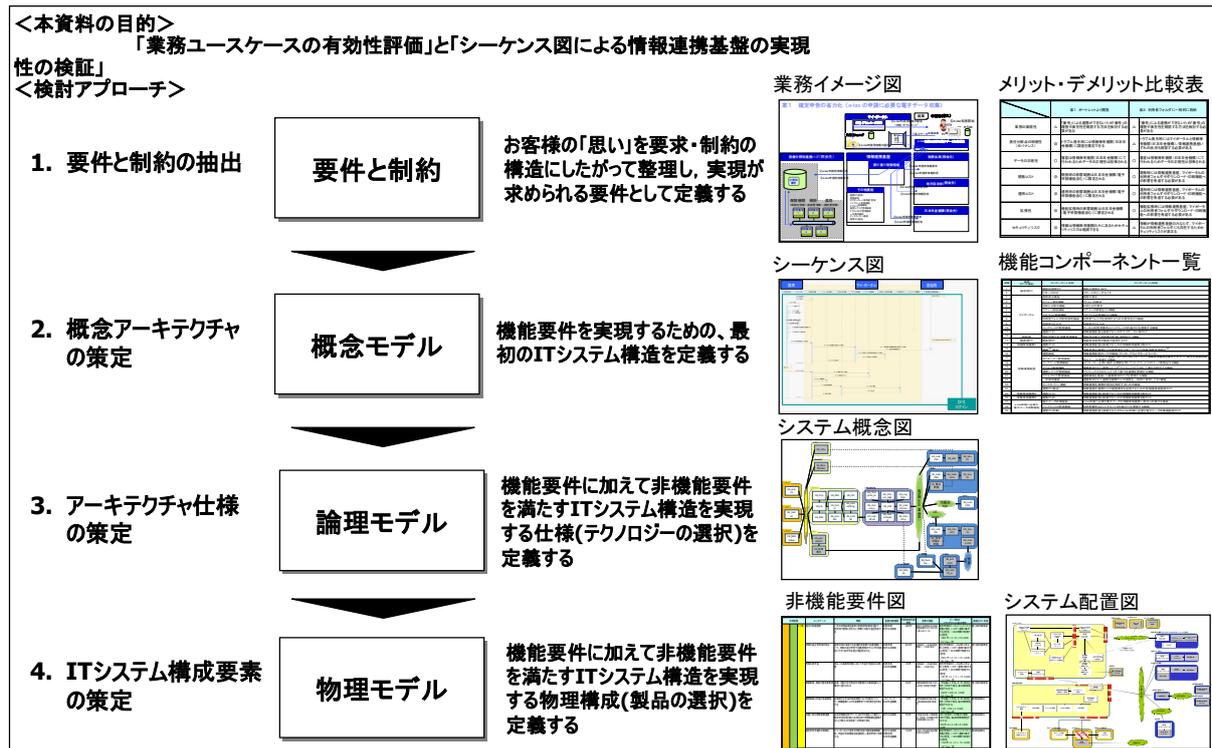


図1 標準的なアーキテクチャー構築手法

今回は、諸外国の行政機関や国内金融機関で実績のある情報連携基盤のアーキテクチャーを参照して、主要なシステム構成要素（コンポーネント）を抽出し、シーケンス図のフェンス上のライフラインとして配置しました。次に、「社会保障・税に関わる番号制度についての基本方針」とワーキンググループで議論されてきたユースケースをもとに業務シナリオを描き、これらのシーケンス図に反映して分析を進めました。各工程の概要を以下に示します。

#### 要件と制約の抽出：

お客様の「思い」を要求・制約の構造にしたがって整理し実現が求められる要件として抽出します

- ① 業務ユースケースから情報連携基盤を介した複数案のシナリオを検討し求められる要件と制約を検討しました
- ② 各業務の処理量・時限性などを考慮して非機能要件と制約を検討しました

#### 概念アーキテクチャーの策定：

機能要件を実現するための最初のITシステム構造を策定します

- ① 諸外国の行政機関や国内金融機関で実績のある情報連携基盤のアーキテクチャーをもとに、主要なシステム構成要素（コンポーネント）を抽出し、シーケンス図のフェンス上のライフラインとして配置しました
- ② このシーケンス図を雛形として、業務ユースケースとシステムユースケースを分析し、

- 情報連携処理パターンを抽出しました
- ③ 上記過程で①で不足する機能コンポーネントを追加し、後続の論理モデル分析の入力としました
  - ④ ②の結果を用いて、業務ユースケースが情報連携処理パターンの組み合わせにより実現できるという仮説の検証を行いました
  - ⑤ 上記の概念モデルの構築過程を通じて、諸外国の行政機関や国内民間企業で実績のある情報連携基盤のアーキテクチャーが再利用可能であることを確認しました

#### 論理アーキテクチャーの策定：

機能要件に加えて非機能要件を満たす IT システム構造を実現する仕様(テクノロジーの選択)を策定します

- ① 機能要件からでは抽出できない技術系コンポーネント（運用管理コンポーネントなど）を確定するために、非機能要件を考慮して必要となる技術系コンポーネントを抽出しました
- ② 上記アーキテクチャー策定結果として、情報連携基盤を中心としたシステムの論理アーキテクチャーを構築しました

#### 物理アーキテクチャーの策定：

機能要件に加えて非機能要件を満たす IT システム構造を実現する物理構成(製品の選択)

- ① 論理アーキテクチャーで策定した論理モデルの仕様を充足するハードウェア製品、ソフトウェア製品を選択します
- ② 開発対象はそれらを特定して表現します
- ③ 上記アーキテクチャー策定結果として、情報連携基盤を中心としたシステムの物理アーキテクチャーを構築しました

## 3.2 今回実施したアーキテクチャー策定の詳細

### 3.2.1 概念アーキテクチャーの詳細

基本方針で想定されている業務ユースケースの実現可能性を以下の要領で検証しました。

基本方針に記載されている業務ユースケースのうち主要なものについて、業務シナリオを作成し、また、上記業務ユースケースを実現するために必要なシステムユースケースを導出しシステムシナリオを作成しました。

検討中の情報連携基盤においてこれらのシナリオが実現可能であることを検証するために、実績のある情報連携基盤アーキテクチャーで採用されている機能コンポーネントを用いて実現可能性を確認しました。これは情報連携という処理が比較的業務に依存しない論理構造を持っており、実績のある情報連携を実現しているシステムの論理構造を当情報連携基盤に適用可能であるとの経験に基づいています（上記仮説①）。

この検証は当初、実績のある情報連携基盤アーキテクチャーで採用されていた 20 の機能コンポーネントで開始しました。また、検証方法としては標準的な UML シーケンス図を記述し主要な業務シナリオ・システムシナリオをウォークスルーすることとしました。この過程で当情報連携基盤に必要な機能コンポーネントが追加され、最終的には 29 の機能コンポーネントが抽出されました。

また、これらの業務シナリオとシーケンス図を作成し、分析していく過程で情報連携処理にはパターン性があるとの仮説を設けました（上記仮説②）。この仮説を実証するために、それまでに分析した業務シナリオ・システムシナリオとそれらを検証するために作成したシーケンス図から共通性を見出してログインやバックオフィス連携などの実行系 4 パターンと IC カード・リンクコード生成や一般情報保有機関初期リンクコード生成などの運用系 4 パターン合わせて 8 種類の情報連携処理パターンを導出しました。



① 20の機能コンポーネント

以下は、諸外国の行政機関や国内金融機関で実績のある情報連携基盤のアーキテクチャーを参照して作成された主要な20の機能コンポーネントを表しています。

表 1 当初抽出した機能コンポーネント

項番	機関 (または機能)	コンポーネント名称	コンポーネント役割
1	国民用PC	国民利用者PC	国民が使用するPC
2	マイポータル	画面表示機能	画面の表示
3		メニュー表示機能	メニューの表示
4		お知らせ表示機能	お知らせの表示
5		ログイン認証機能	ログインの認証を行う機能
6		アカウント管理機能	アカウントの管理を行う機能
7		連携I/F(P)	情報連携基盤と接続するためのマイポータル側のI/F
8	職員用PC	職員用PC	情報保有期間の職員が使用するPC
9	情報保有機関X	連携I/F(X)	情報連携基盤と接続するための情報保有機関X側のI/F
10	情報連携基盤	連携I/F(連受)	情報連携元機関からの連携要求を受けるための情報連携基盤側のI/F
11		連携機能	情報連携基盤のハブの機能(エンタープライズサービスバス)
12		アクセス制御機能	連携要求された連携シナリオ(プリフィックス)に対して実行を認可する機能
13		連携シナリオ管理機能	プリフィックスされたシナリオに基づき連携を管理する機能
14		アクセスログ管理機能	連携機能を通過した連携要求のログを管理する機能
15		一次保存機能	連携要求された連携先機関からの返答を一時的に保存しておく機能
16		ディスカバリ機能	情報連携先機関の宛先を特定するための機能
17		連携I/F(連送)	情報連携先機関からの連携要求を送信するための情報連携基盤側のI/F
18	情報保有機関A	連携I/F(A)	情報連携基盤と接続するための情報保有機関A側のI/F
19	情報保有機関B	連携I/F(B)	情報連携基盤と接続するための情報保有機関B側のI/F
20	e-tax申請に必要な電子データ収集機能	連携I/F(収集)	情報連携基盤と接続するためのe-tax申請に必要な電子データ収集機能側のI/F

② 業務シナリオ

下図は、情報連携に関わる業務シナリオです。必要な機能や、図の配置を統一して記述しました。

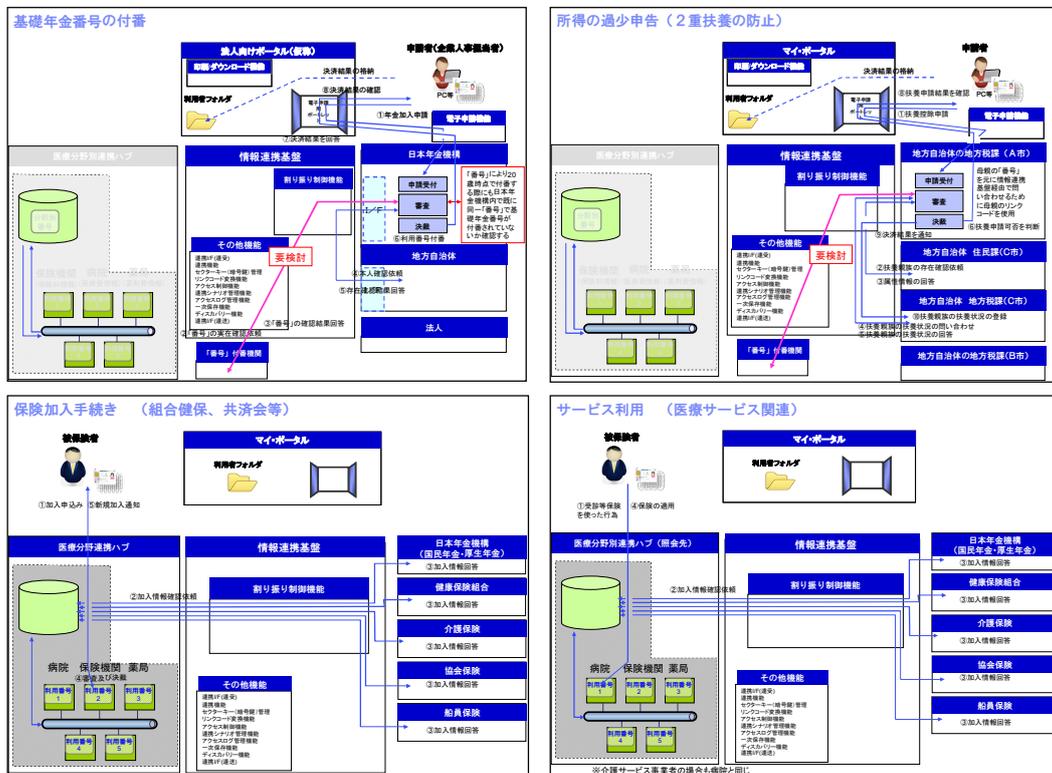


図 3 作成された業務シナリオの例

③ 情報連携処理パターン

以下は、ウォークスルーにより抽出された、情報連携処理パターンを表しています。

例えば、「業務ユースケース：年金番号の付番」のケースでは、業務を支えるパターンとしては「ログイン（1回）」、「報保有機関 X→A・B 間バックオフィス連携（1回）」の2つの組み合わせで実現できることが確認されました。

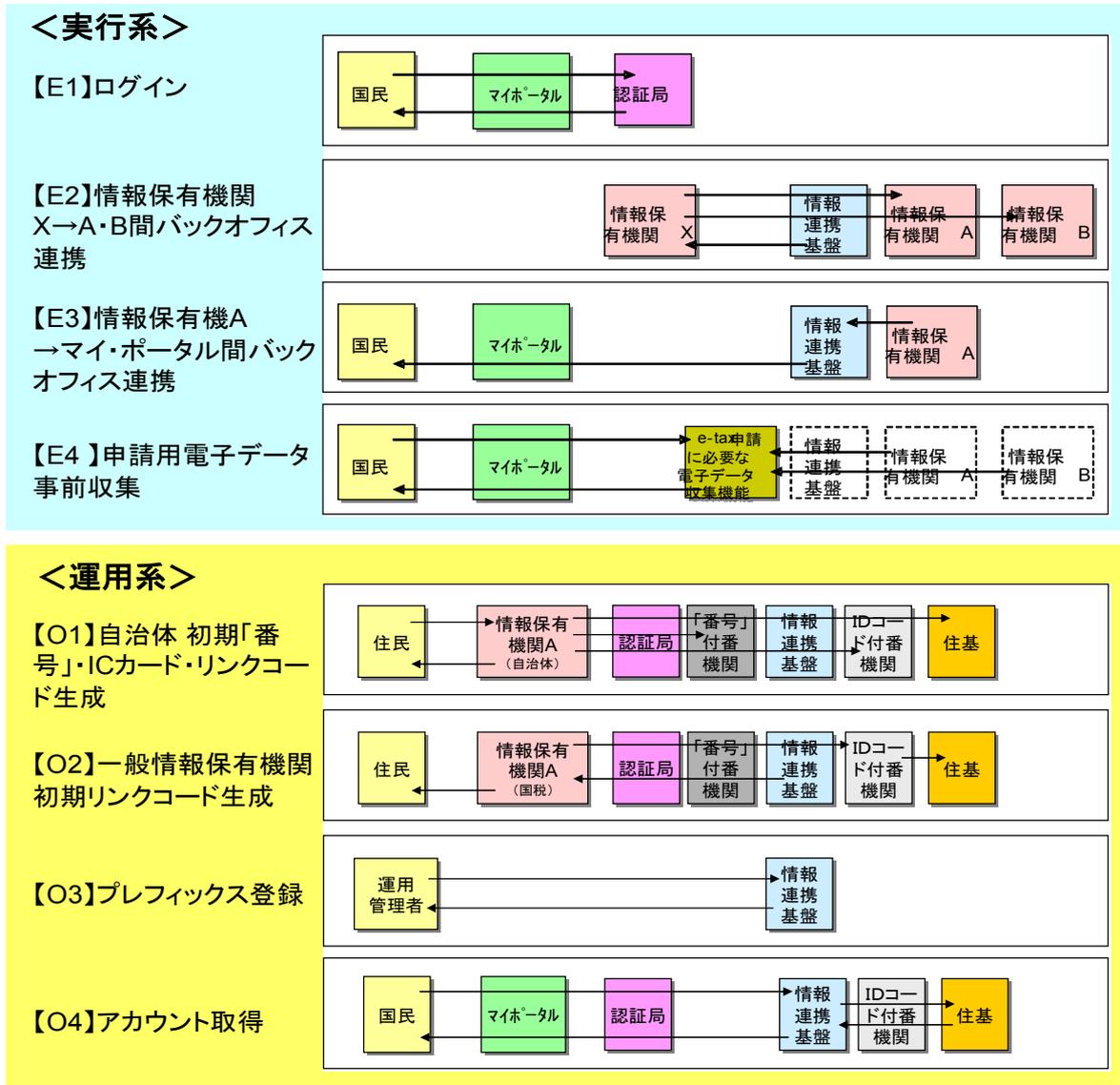


図 4 抽出された 8 つの情報連携処理パターン

④ 29 の機能コンポーネント

以下は、ウォークスルーの結果、追加あるいは見直された機能コンポーネントの一覧を表しています。

表 2 29 の機能コンポーネント一覧

項番	機関 (または機能)	コンポーネント名称	コンポーネント役割
1	国民用PC	国民利用用PC	国民が使用するPC
2		ICカードR/W	ICカードのリーダーライタ
3	マイポータル	画面表示機能	画面の表示
4		メニュー表示機能	メニューの表示
5		お知らせ表示機能	お知らせの表示
6		ログイン認証機能	ログインの認証を行う機能
7		アカウント管理機能	アカウントの管理を行う機能
8		利用者フォルダ使用認可機能	利用者フォルダを使用するための認可を行う機能
9		利用者フォルダ	利用者のフォルダ
10		リンクコードP管理機能	ポータル利用者番号とリンクコードPの紐付けを管理する機能
11		連携I/F(P)	情報連携基盤と接続するためのマイポータル側のI/F
12	認証局	認証用電子証明書管理機能	認証用の電子証明書を管理し認証を行う機能
13	職員用PC	職員用PC	情報保有期間の職員が使用するPC
14	情報保有機関X	連携I/F(X)	情報連携基盤と接続するための情報保有機関X側のI/F
15	情報連携基盤	連携I/F(連受)	情報連携元機関からの連携要求を受けるための情報連携基盤側のI/F
16		連携機能	情報連携基盤のハブの機能(エンタープライズサービスバス)
17		セクターキー管理機能	リンクコード/IDコード変換に用いられるセクターキー(情報保有機関を識別するキー。暗号化関数用暗号化キー)を管理する機能
18		リンクコード変換機能	セクターキーを用い暗号化関数を用いてリンクコード⇔IDコード変換を行う機能
19		アクセス制御機能	連携要求された連携シナリオ(プリフィックス)に対して実行を認可する機能
20		連携シナリオ管理機能	プリフィックスされたシナリオに基づき連携を管理する機能
21		アクセスログ管理機能	連携機能を通過した連携要求のログを管理する機能
22		一次保存機能	連携要求された連携先機関からの返答を一時的に保存しておく機能
23		ディスカバリー機能	情報連携先機関の宛先を特定するための機能
24		連携I/F(連送)	情報連携先機関からの連携要求を送信するための情報連携基盤側のI/F
25	情報保有機関A	連携I/F(A)	情報連携基盤と接続するための情報保有機関A側のI/F
26	情報保有機関B	連携I/F(B)	情報連携基盤と接続するための情報保有機関B側のI/F
27	e-tax申請に必要な 電子データ収集機能	電子データ収集機能	e-tax申請に必要な電子データを情報保有機関へ要求し収集する機能
28		リンクコードA管理機能	利用者番号AとリンクコードAの紐付けを管理する機能
29		連携I/F(収集)	情報連携基盤と接続するためのe-tax申請に必要な電子データ収集機能側のI/F

⑤ 情報連携処理パターンの組み合わせによる業務シナリオの実現可能性検証

以下は、情報連携処理パターンの組み合わせにより、業務シナリオの実現可能性を検証した結果を表しています。例えば、基礎年金番号の付番の業務ユースケースでは、ログイン (E1) と情報保有機関 X-A-B 間バックオフィス連携(E2)の組み合わせで実現可能であることが確認されました。

表 3 業務ユースケースと情報連携処理パターンの検証結果

シーケンス図パターン		E1	E2	E3	E4
		ログイン	情報保有機関X →A・B間 バックオフィス連携	情報保有機関A →マイ・ポータル間 バックオフィス連携	申請用電子データ 事前収集
確定申告の省力化 (e-tax の申請に必要な電子データ収集)	確定申告	○			○
基礎年金番号付番 (年金制度的確な運用)	年金基礎番号付番	○	○		
所得の過少申告 (2重扶養の防止)	所得過少申告	○	○		
保険証機能の一元化	国民年金等 加入手続き	○	○		
	組合健保等 加入手続き		○		
	組合健保等 サービス利用		○		
	臨時保険証の発行	○			○
退職	退職 企業が行う手続き	○		○	
	退職 個人が行う手続き	○	○	○	
世帯所得の捕捉 (給付付税額控除及び所得比例年金)				○	○
高額医療・高額介護合算制度	医療費・介護費情報の自己情報確認	○		○	○
	申請手続きのワンストップ	○		○	○
	医療費の現物給付				○
雇用保険と年金の併給調整			○	○	
引越に係る所得情報連携		○			
自己の検診情報の閲覧		○		○	○
給付可能サービスの行政からの通知、自己情報の閲覧		○	○	○	
引越ワンストップ		○	○	○	○
出生ワンストップ	出生届・出産育児一時金支給	○	○	○	
	健康保険証交付・医療証交付		○	○	
ハローワークワンストップ			○		
災害時の安否確認					○

基礎年金番号の付番

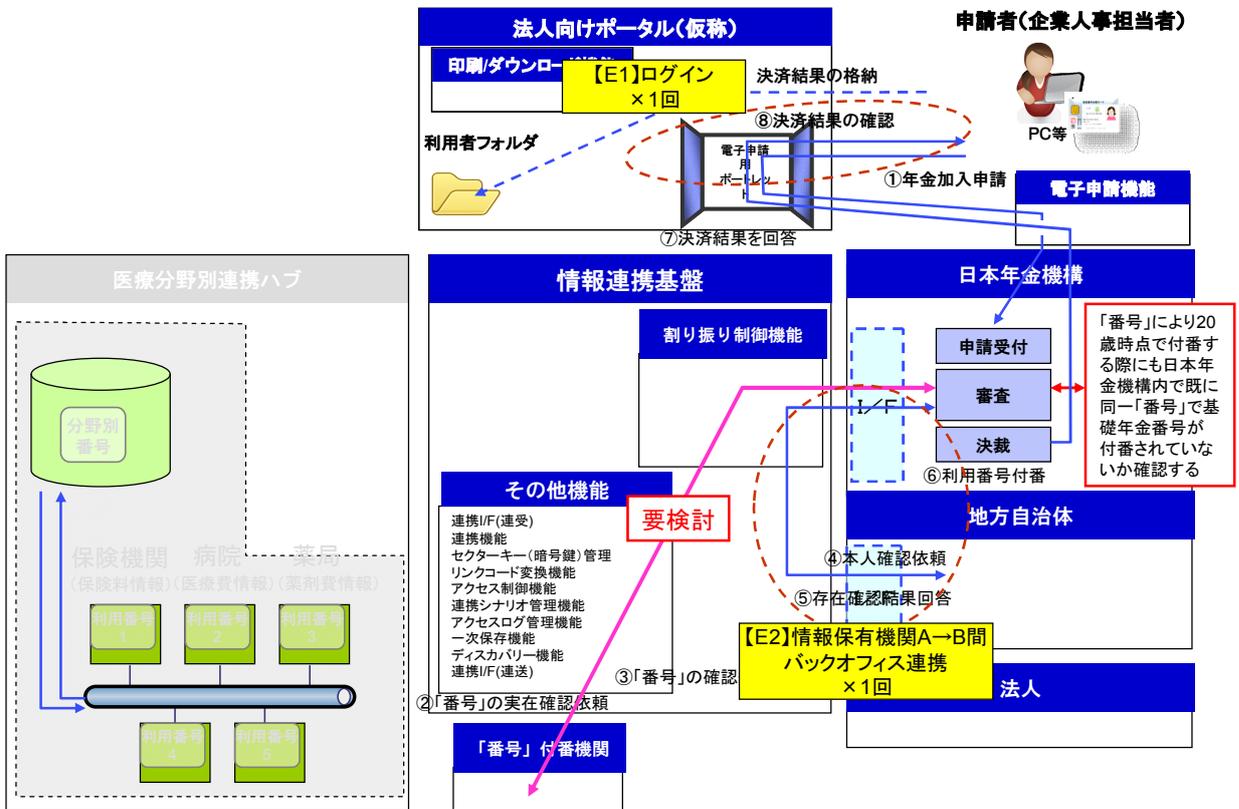


図 5 情報連携処理パターンによる業務シナリオの実現



① 論理アーキテクチャー図

下図は、機能コンポーネントを論理的なノードに配置して設計した論理アーキテクチャーの図を表します。

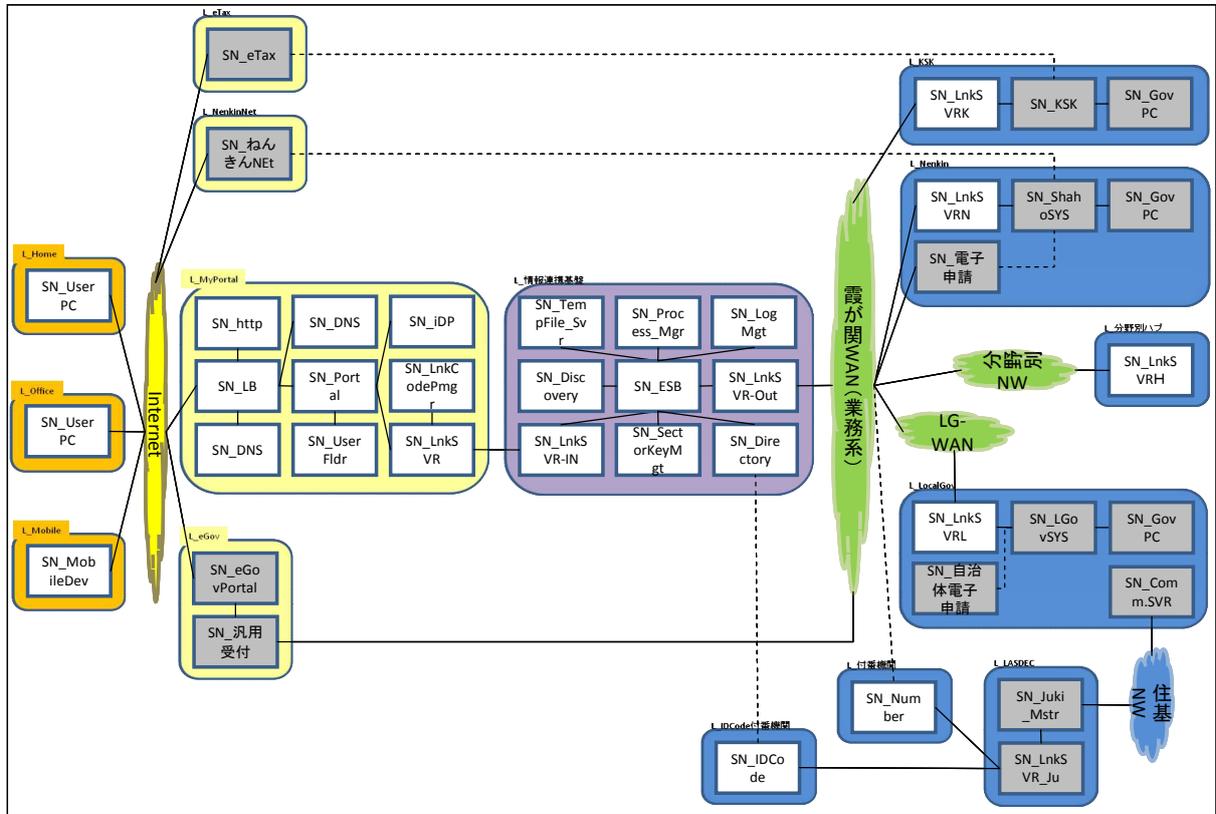


図 7 論理アーキテクチャー図

### 3.2.3 物理アーキテクチャー

これらコンポーネントには、各々、固有の責務が割り当てられており、それら責務を遂行するために必要な機能が定義されています。また、これらの機能の実行においては、性能、可用性、運用管理、セキュリティなどの一般的に非機能要件と呼ばれる要件を満たす必要があります。そこで、非機能要件を表4 非機能要件（抜粋）のように仮定し、これらの品質を満たした上で、上記の機能が提供できるように機能配置の最適化を検討し、不足している技術コンポーネントの抽出を行いました。

表 4 非機能要件（一部抜粋）

利用範囲	ユースケース	概要	連携対象機関	年間処理件数見積	根拠資料	ピーク日の想定
B-1案	給付付税額控除	一定の所得金額を基準に税額控除(減税)を設け、低所得で控除しきれない世帯には給付金を支給する	税務当局 地方公共団体	1800万	国税庁の民間給与の実態調査結果の300万以下の人数 (H23/3/10)	確定申告のピーク(3/11-3/15)と連動と想定。1/3が一週間に集中すると仮定。一日10時間で処理すると仮定。 = 1800万 ÷ 3 ÷ 5 ÷ 10 ÷ 3600 = 33 Trans / 秒
	所得の過少申告等の防止	税務当局が保有する各種所得情報や扶養情報について、所得の過少申告や扶養控除のチェックが効率的にでき、税の不正還付等を防止する。	税務当局 地方公共団体	2253万	税務統計 一申告所得税関係一 (平成21年分)	確定申告後のピーク以降(4月以降)と想定。1/3が一週間に集中すると仮定。一日10時間で処理すると仮定。 = 2253万 ÷ 3 ÷ 5 ÷ 10 ÷ 3600 = 42 Trans / 秒
	所得比例年金	支払った保険料の額に応じて年金が支給される制度	税務当局 日本年金機構	720万	税務統計 一申告所得税関係一 (平成21年分)	確定申告後のピーク以降(4月以降)と想定。1/3が一週間に集中すると仮定。一日10時間で処理すると仮定。 = 720万 ÷ 3 ÷ 5 ÷ 10 ÷ 3600 = 13 Trans / 秒
	高額医療・高額介護合算制度	医療・介護の自己負担の合算額が上限額を超えた場合に還付される		432万	健康保険事業年報 平成20年度 支給数で仮置き	一年を通して平均。土・日・祝日を除き、245日で割る。毎日8時間処理を行うとする。 = 432万 ÷ 245 ÷ 8 ÷ 3600 = 0.8 Trans / 秒
	雇用保険と年金の併給調整	失業給付と年金の併給調整における届出について、情報連携により年金事務所での手続きを不要とする	ハローワーク 日本年金機構	15万	雇用保険事業月報・年報(高齢受給者数で仮置き)	一年を通して平均。土・日・祝日を除き、245日で割る。毎日8時間処理を行うとする。 = 15万 ÷ 245 ÷ 8 ÷ 3600 = 0.02 Trans / 秒
	引越に係る所得情報連携	乳幼児医療を受けていたものが引越した場合に、転出元自治体と転入先自治体で所得情報を連携することで転入先自治体への申請で済む	地方公共団体	556万	住民基本台帳人口移動報告(2008年)の市区町村間の移動者数(556.4万)	3月に全体の1/4が集中と仮定。20日で割る。毎日8時間処理を行うとする。 = 556万 ÷ 4 ÷ 3 ÷ 20 ÷ 8 ÷ 3600 = 0.8件
	確定申告手続きの簡略化	マイ・ポータルで前年の申告内容や国民健康保険料、年金の支給額等を確認し、確定申告に活用する。	地方公共団体 税務当局 日本年金機構	1184万	平成21年の確定申告者数の約半分と仮定	確定申告のピーク(3/11-3/15)と連動と想定。1/3が一週間に集中すると仮定。一日10時間で処理すると仮定。 = 1184万 ÷ 3 ÷ 5 ÷ 10 ÷ 3600 = 22 Trans / 秒
B-2案	保険証機能の一元化	券面に「番号」を記載した1枚のICカードの提示により、年金手帳、医療保健証、介護保険証を提示したものとみなす。	地方公共団体 健康保険組合 日本年金機構	-		
	自己の検診情報の閲覧	医療・介護サービスの現場において、本人が自分の診療情報などを容易に入手し・活用することにより、地域医療連携や医療・介護連携が進む。	病院、診療所 介護事業者	7841万	平成21年(2009)社会医療診療行為別調査結果の概況の内科+歯科の診療総数	一年を通して平均。土・日・祝日を除き、245日で割る。毎日8時間処理を行うとする。 = 7841万 ÷ 245 ÷ 8 ÷ 3600 = 11 Trans / 秒
	給付可能サービスの行政からの	本人の同意に基づき利用可能な様々な施策の情報を提供する。	地方公共団体 日本年金機構 税務当局 医療機関 等	-		
C案	引越ワンストップ	引越に伴い発生する様々な手続きについて一度の手続きで完結させるサービス	地方公共団体 学校 警察 等	556万	住民基本台帳人口移動報告(2008年)の市区町村間の移動者数(556.4万)	3月に全体の1/4が集中と仮定。20日で割る。毎日8時間処理を行うとする。 = 556万 ÷ 3 ÷ 4 ÷ 20 ÷ 8 ÷ 3600 = 0.8件
	退職ワンストップ	退職に伴い、企業や個人が行う手続きを一括して完結させる	地方公共団体 日本年金機構 税務当局 健康保険組合等	752万	一平成22年上半期雇用動向調査結果の概況-の2倍と仮定	一年を通して平均。土・日・祝日を除き、245日で割る。毎日8時間処理を行うとする。 = 752万 ÷ 245 ÷ 8 ÷ 3600 = 1.1 Trans / 秒
	出生ワンストップ	妊娠・出産、出生の手続き、子育て支援における各種手続きのワンストップ化	地方公共団体 日本年金機構 税務当局 医療機関 等	107万	厚労省統計(平成21年)	一年を通して平均。土・日・祝日を除き、245日で割る。毎日8時間処理を行うとする。 = 107万 ÷ 245 ÷ 8 ÷ 3600 = 0.2 Trans / 秒
	ハローワークワンストップ	ハローワークでの活動記録から生活保護の申請内容の確認や、就労可能性のある人のハローワークへの紹介等	地方公共団体 ハローワーク	-		
	災害時の安否確認	地震・津波、火災等の大規模災害時の安否確認や被災者支援、生活再建のための支援	地方公共団体 医療機関 等	-		



そして、各コンポーネントをサーバーなどのように物理的な配置を意識したノードという単位にまとめて整理し配置を行い、システム全体の論理アーキテクチャーを描きました。これを図 9 に示します。ここには実装の元となるテクノロジーが検討され反映されています。

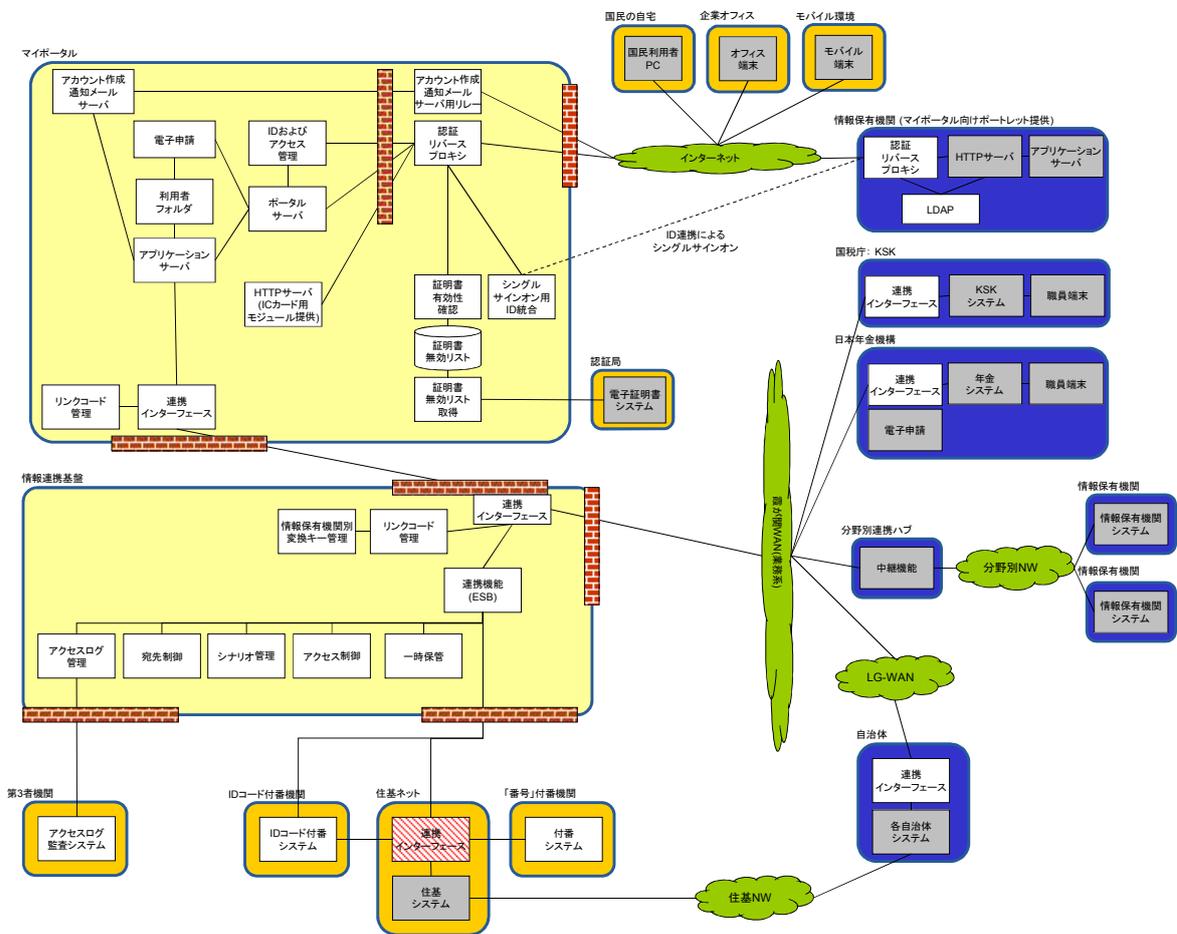


図 9 インフラの全体像イメージ

そして、それぞれのノード毎に対して個々の機能要件に加えて非機能要件を満たすハードウェア製品、ソフトウェア製品を選択しました。また、それら製品で満たせない要件の実現は開発対象として特定し、実現方式を検討しました。

## 4 結論

以上説明した標準的なアーキテクチャー構築手法を用いることで、社会保障・税に関わる番号制度や国民 ID 制度を下支えする情報連携基盤の設計を進める事が可能であることがご理解いただけたと思います。また、今回、概念アーキテクチャー構築中に導入した「情報連携パターンを組み合わせ、様々な情報連携シナリオが構築可能である」と言う仮説は、上述の過程で検証されました。ただし、現在導出されている 8 種類の「情報連携パターン」で、今後も、追加が予定されている様々な情報連携シナリオ全てをカバー出来るとは思いませんが、少数の「情報連携パターン」を追加する事により、社会保障・税に関わる番号制度や国民 ID 制度を下支えする情報連携業務の多くに対応可能と思われます。

また今後、公表が予定されている大綱では「社会保障・税に関わる番号制度」で実現予定の業務名が具体的に記述されることとなっていますが、これらの業務を今回検討している情報連携基盤を用いて実現可能か否かの判断を、今回導出した前述の「情報連携パターンの組み合わせ」手法を用いて具体的に、また、スピーディーに評価出来ると思います。