

特許庁アーキテクチャ標準仕様書 (参考3) アプリケーション基盤編

第1.1版

平成28年6月

特許庁

はじめに

(1) 本書の位置づけ

本書は、アプリケーション基盤(以下、「AP基盤」と称す)を設計・開発する際に定めておくべき方針の具体例を、参考情報として示したものである。

AP基盤は、ToBeアーキテクチャに対応した業務アプリケーションを構築する際に共通的に必要となる機能を提供し、業務アプリケーションのコンポーネント内部の構造定型化を図り、保守性及び移植性を高めるための機能群である。AP基盤は、原則としてサブシステム毎に設計・開発するものであるが、複数のサブシステム間で同一のフレームワーク部品を利用する等して、開発効率を向上させることを禁止するものではない。

また、本書におけるAP基盤は、完全な作り込みではなくOSS等のフレームワーク製品を利用して実現する想定であるが、特許庁固有の要件により、一般的なフレームワーク製品又はその組み合わせてだけでは実現できない場合は、特許庁固有の要件に対応したフレームワークの機能追加を行う。

(2) 本書の利用者及び利用目的

本書は、個別システム刷新に関するステークホルダ(情報技術統括室職員、特許庁PMO、システム利用者、原課、要件整理補助(支援)業者、調達支援業者、設計・開発ベンダ、システムインテグレーションベンダ等)向けに作成されたものであり、当該ステークホルダが本書を利用して個別システム刷新を行うことにより、段階的刷新を通じ特許庁システム全体として統一された保守性と移植性の高いシステムを実現することを目的とする。

(3) 本書の文書構成

本書は、以下の章から構成される。

1章 アプリケーション基盤

AP基盤の設計方針、分類に基づき、AP基盤の適用範囲や機能、サブシステム間の共有に関する方針の具体例を、参考情報として掲載する。

(4) 本書の利用方法

本書の利用者及び利用方法について以下に示す。

表 (4)-1 本書の利用者及び利用方法

(○:利用する, -:利用しない)

利用者 利用方法	情報技術 統括室	特許庁P MO	システム 利用者, 原課	要件整理 補助業 者, 調達 支援業者	設計・開 発ベンダ	システム インテグ レーショ ンベンダ	ハードウ ェアベン ダ	オペレー ションベン ダ
システム構造の 定型化(ルール の理解・遵守)	○	○	○※	○	○	○	-	-
技術的整合性 確保(コントロー ル及びチェック)	○	○	-	○	○	-	-	-

※本書の記載例に従い画面設計等の設計レビューに関与するために必要。

(5) 本書の運用方法

本書の運用方法について以下に示す。

①運用開始時期

平成28年6月から運用を開始する。

②改定時期

平成29年3月末, 平成30年3月末及び平成31年3月末の3回の時期において改定を予定している。

③整備及び管理

『特許庁PMO標準・規約類における整備及び管理方針』に従う。

－ 目 次 －

1. アプリケーション基盤	1
1.1 アプリケーション基盤の適用範囲	2
1.2 アプリケーション基盤の機能	5
1.3 アプリケーション基盤のサブシステム間の共有	17

1. アプリケーション基盤

(1) アプリケーション基盤の設計方針

AP基盤は以下を実現できるように設計・開発する。

- コンポーネント内の個別の作り込みを小さくすること。
- 共通的に利用する機能を提供すること。
- コンポーネント間の依存性を排除すること。
- サブシステム毎にバージョンアップができること。

(2) アプリケーション基盤のフレームワークの分類

AP基盤のフレームワークは、処理方式(詳細は、『本冊』の「3.1.1.3.1.2 処理方式に基づくシステム構成要素」を参照のこと。)に合わせて、以下に分類される。

表 1.1-1 AP基盤のフレームワークの分類

項番	AP基盤のフレームワークの分類	対応する処理方式
1	画面オンライン・フレームワーク	<ul style="list-style-type: none">● オンライン処理方式(画面系) ※サーバで動作する部分が対象● 帳票出力方式
2	サービス・フレームワーク	<ul style="list-style-type: none">● オンライン処理方式(サービス系)● 帳票出力方式● 連携処理方式
3	バッチ・フレームワーク	<ul style="list-style-type: none">● バッチ処理方式● デイレードバッチ処理方式● 帳票出力方式● 連携処理方式
4	リッチクライアント・フレームワーク	<ul style="list-style-type: none">● オンライン処理方式(画面系) ※リッチクライアントで動作する部分が対象

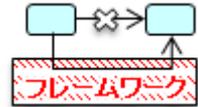
帳票出力方式及び連携処理方式は、オンライン処理方式とバッチ処理方式の組み合わせであることから、画面オンライン・フレームワーク、サービス・フレームワーク又はバッチ・フレームワークに含まれる。

1.1 アプリケーション基盤の適用範囲

(1) コンポーネントにおける適用パターン

業務アプリケーションを構成するコンポーネントにおいて、AP基盤の適用パターンは以下のとおりである。

表 1.1-1 コンポーネントにおけるAP基盤の適用パターン

項番	適用パターン	目的	対象コンポーネント	フレームワーク機能の例
1	<p>コンポーネントそのものをフレームワークが提供するもの</p> 	<p>共通的な機能を提供する。</p>	<ul style="list-style-type: none"> ● ロジック層、インフラストラクチャ層の一部のコンポーネント(共有コンポーネント) 	<ul style="list-style-type: none"> ● 共有DBアクセス(基盤API) ● 共有ビジネスロジック(業務特有機能)
2	<p>コンポーネントの基盤部分をフレームワークが提供するもの</p> 	<p>個別の作り込みを小さくする。</p>	<ul style="list-style-type: none"> ● インタフェース層のコンポーネント(共有画面部品を含む) ● インフラストラクチャ層のコンポーネント 	<ul style="list-style-type: none"> ● RESTサービス ● 個別DBアクセス等
3	<p>基本的に個別に作り込むもの</p> 	<p>個別の作り込みを小さくする(フレームワークへ依存させずに、コンポーネントの独立性を高めるべきもの)。</p>	<ul style="list-style-type: none"> ● ロジック層のコンポーネント(個別に作成するコンポーネント) 	—
4	<p>コンポーネント間の呼び出しを制御するもの</p> 	<p>依存関係を制御する。</p>	<ul style="list-style-type: none"> ● 全てのコンポーネント 	<ul style="list-style-type: none"> ● 依存関係制御等
5	<p>コンポーネント横断的に利用される共通的な機能を提供するもの</p> 	<p>個別の作り込みを小さくする。</p>	<ul style="list-style-type: none"> ● 全てのコンポーネント 	<ul style="list-style-type: none"> ● ログ ● ユーティリティ ● パラメータ設定値提供等

APレイヤ(『本冊』の「3.1.2.1.1 APレイヤの定義及びコンポーネントとの関係」を参照)におけるコンポーネントの適用パターンに基づく各コンポーネントの配置を下図に示す。

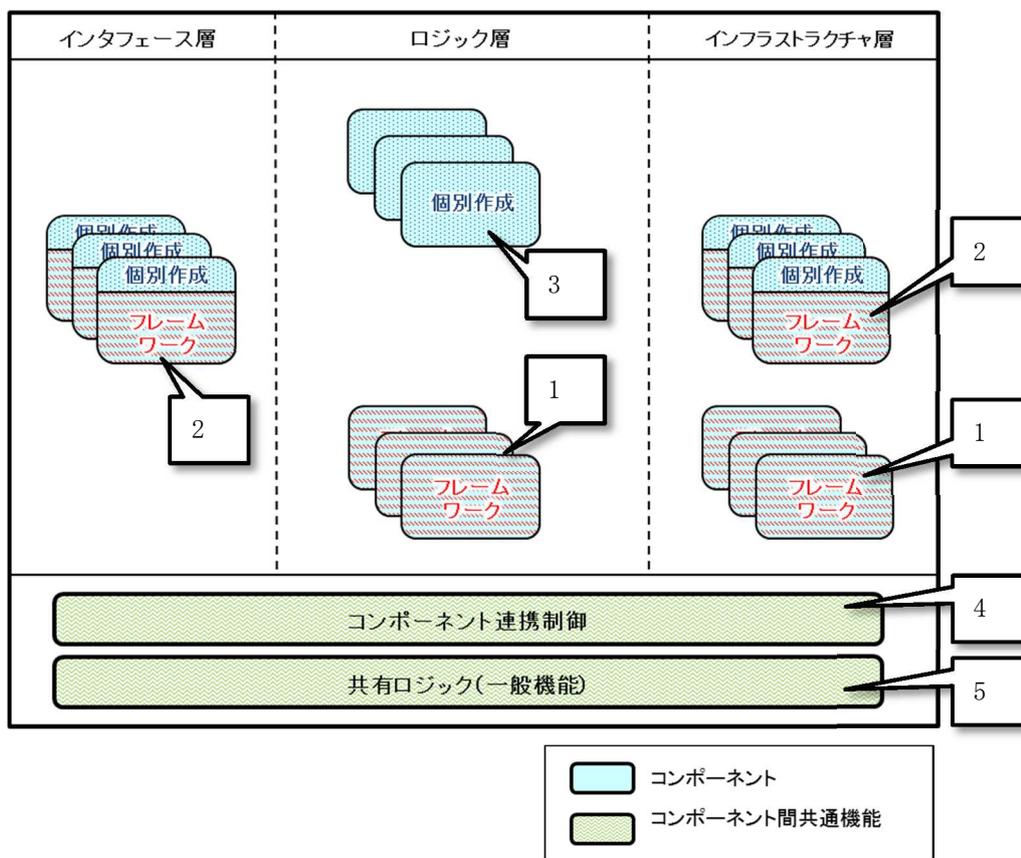


図 1.1-1 AP基盤における各コンポーネントの配置¹

¹ 図中の番号は、「表 1.1-1 コンポーネントにおけるAP基盤の適用パターン」の項番を表す。

- (2) 業務アプリケーションにおけるアプリケーション基盤の適用範囲
 業務アプリケーションは、AP基盤とそれを利用して構築するサブシステム固有の個別作成プログラムで構成する。
 業務アプリケーションにおけるAP基盤の適用範囲を下図に示す。

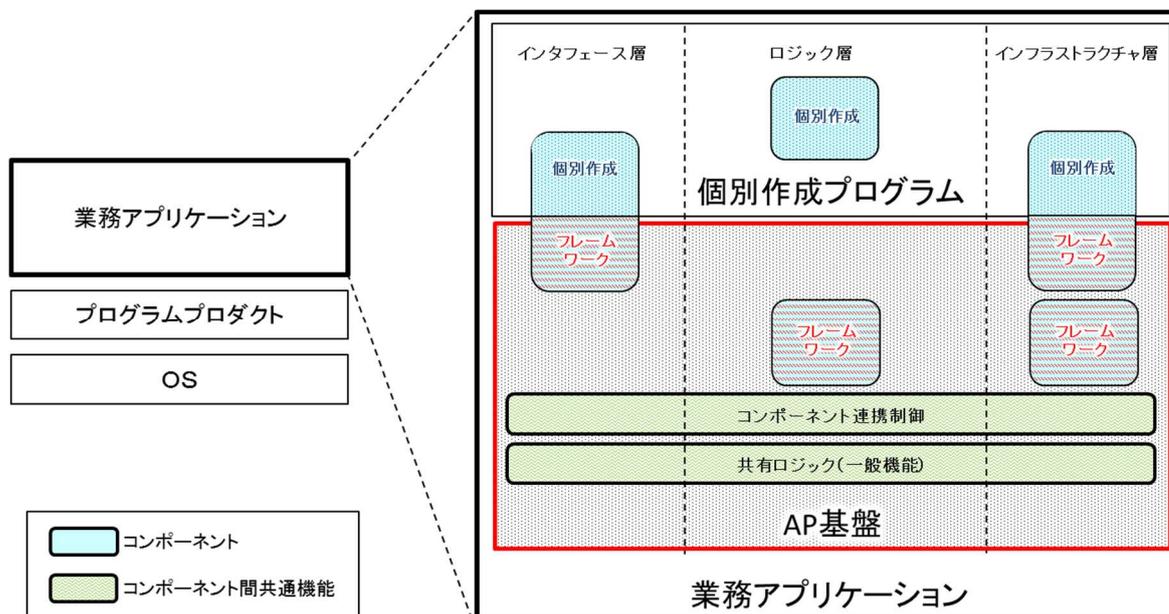


図 1.1-2 業務アプリケーションにおけるAP基盤の適用範囲

- (3) 業務アプリケーションに適用するフレームワーク
 開発対象の業務アプリケーション(『本冊』の「表 3.1-14 ソフトウェア(開発対象のアプリケーション)一覧」を参照のこと。)に対して適用するAP基盤のフレームワークを下表に示す。

表 1.1-2 フレームワークの適用対象

項番	開発対象の業務アプリケーション	適用するAP基盤のフレームワーク
1	UI層アプリケーション(リッチクライアント)	リッチクライアント・フレームワーク
2	プレゼンテーションロジック 及び 業務アプリケーション(ユーザ)及び 動的Webコンテンツ	画面オンライン・フレームワーク
3	静的Webコンテンツ (UI層アプリケーション含む)	画面オンライン・フレームワーク
4	BPMS補完機能	サービス・フレームワーク
5	業務アプリケーション(システム)	サービス・フレームワーク
6	業務アプリケーション(バッチ)	バッチ・フレームワーク
7	DBアクセス基盤サービス	サービス・フレームワーク
8	外部システム互換機能	対象なし ※階層定型化サブシステムではないため、対象としない。

1.2 アプリケーション基盤の機能

(1) 機能の分類

AP基盤を構成する機能には、以下の分類がある。

表 1.2-1 AP基盤を構成する機能の分類

項番	AP基盤のフレームワークの分類	フレームワークを構成する機能の分類
1	画面オンライン・フレームワーク	<ul style="list-style-type: none">● サーバ汎用機能● クライアント汎用機能(ブラウザ)● 共有コンポーネント
2	サービス・フレームワーク	<ul style="list-style-type: none">● サーバ汎用機能● 共有コンポーネント
3	バッチ・フレームワーク	<ul style="list-style-type: none">● サーバ汎用機能● 共有コンポーネント
4	リッチクライアント・フレームワーク	<ul style="list-style-type: none">● クライアント汎用機能(リッチクライアント)

AP基盤を構成する機能分類の詳細を以下に示す。

表 1.2-2 AP基盤の機能分類

項番	分類	説明
A	サーバ汎用機能	ToBeアーキテクチャのシステム的な要件から導出されたフレームワーク機能のうち、サーバで実行される機能。
B	クライアント汎用機能(ブラウザ)	ToBeアーキテクチャのシステム的な要件から導出されたフレームワーク機能のうち、ブラウザで実行される機能。
C	クライアント汎用機能(リッチクライアント)	ToBeアーキテクチャのシステム的な要件から導出されたフレームワーク機能のうち、リッチクライアントで実行される機能。
D	共有コンポーネント	サブシステム間共通機能の共有コンポーネントである共有DBアクセス(基盤API)を指す。

AP基盤への機能分類の配置イメージを下図に示す。

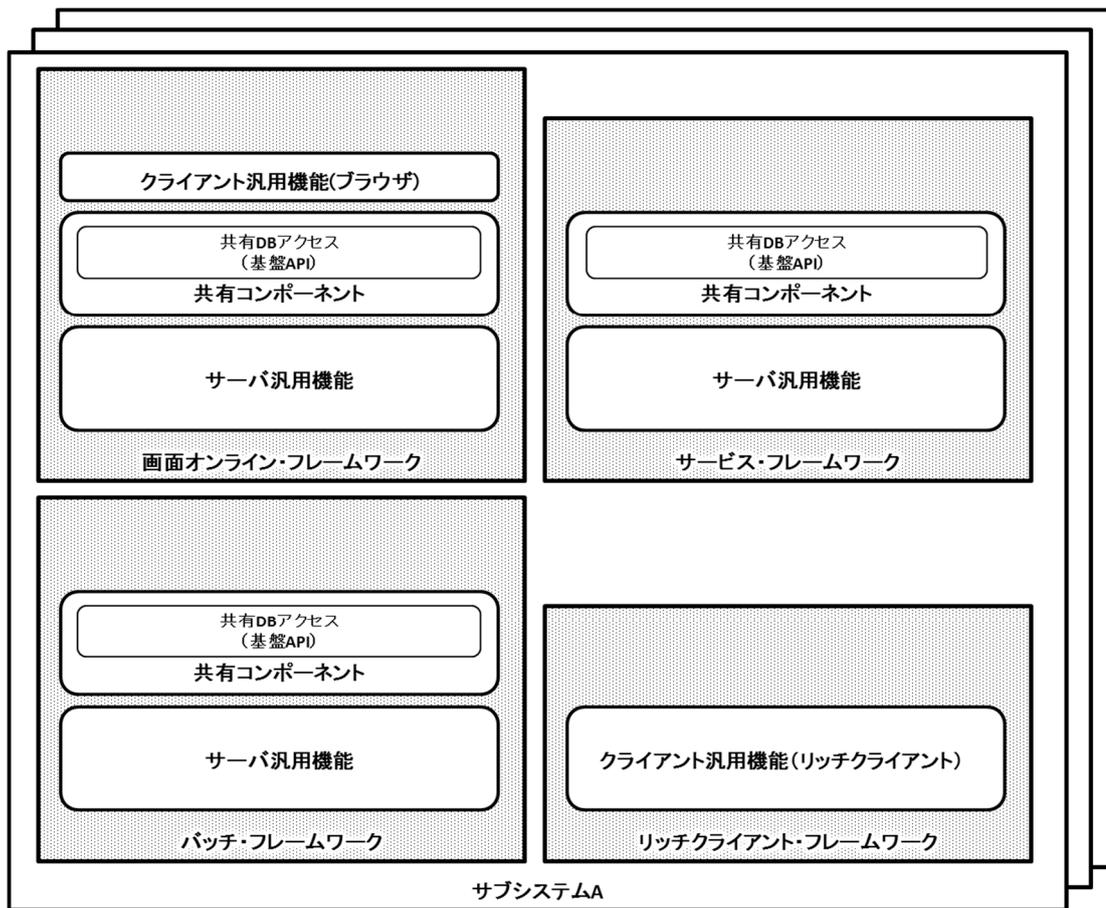


図 1.2-1 フレームワーク機能の配置イメージ

(2) 機能概要

A. サーバ汎用機能

サーバ汎用機能をAPレイヤ別に示す。

(A) インタフェース層

インタフェース層におけるサーバ汎用機能を以下に示す。

表 1.2-3 サーバ汎用機能(インタフェース層)

項番	機能名	機能概要	システム構成要素	コンポーネント名
1	Webサービス	XMLスキーマの形式に従ったWebサービス ² を作成するための機能 ● インタフェース定義 ● 電文の解析 ● 応答電文作成 等	プレゼンテーションロジック	リクエストハンドラ
			業務アプリケーション(システム)	サービスインタフェース
			BPMS補完機能	サービスインタフェース
2	RESTサービス	RESTの形式に従ったサービスを作成するための機能 ● インタフェース定義 ● 電文の解析 ● 応答電文作成 等	プレゼンテーションロジック	リクエストハンドラ
			DBアクセス基盤サービス	基盤サービスインタフェース
3	LDAPアクセス	LDAPサーバへのアクセス機能 LDAPサーバを用いた認証・認可については、『別冊4 システム機能共通編』のセキュリティ方式設計のルール(認証・認可)の章を参照。	プレゼンテーションロジック	リクエストハンドラ
4	認証・認可チェック	ユーザ情報を利用し、認証を行う。取得した認可情報で必要に応じて、認可チェックをする。 詳細は、『参考1 処理シーケンスサンプル集』のオンライン処理方式の章、『別冊4 システム機能共通編』のセキュリティ方式設計のルール(認証・認可)の章を参照。	プレゼンテーションロジック	リクエストハンドラ
			業務アプリケーション(システム)	サービスインタフェース
			BPMS補完機能	サービスインタフェース
			DBアクセス基盤サービス	基盤サービスインタフェース
5	入力チェック	サーバでの入力値のチェック(必須チェック、フォーマットチェック、範囲チェック、関連性チェック、文字種チェック等)を行う機能。 入力チェック機能が利用されるタイミングや利用方法については、『参考1 処理シーケンスサンプル集』のオンライン処理方式、バッチ処理方式の章を参照。文字種チェックについては、『本冊』の「3.4.2 文字コードの扱いのルール」を参照。	プレゼンテーションロジック	リクエストハンドラ
			業務アプリケーション(システム)	サービスインタフェース
			業務アプリケーション(バッチ)	バッチインタフェース
			BPMS補完機能	サービスインタフェース
6	ディレード処理要求制御	定期的に「ディレード処理要求管理情報」を確認し、登録された要求がある場合、要求内容に応じたビジネスロジックを実行する。詳細は、『本冊』の「3.1.2.4 ディレードバッチ処理方式」を参照。	業務アプリケーション(バッチ)	バッチインタフェース

² 「RESTサービス」(リソース指向のインタフェースの提供サービス)として採用するフレームワーク製品が、「Webサービス」(XML電文形式のインタフェースの提供サービス)の機能を包含する場合は、「RESTサービス」のみを設計・開発することとしてよい。

(B) インフラストラクチャ層

インフラストラクチャ層におけるサーバ汎用機能を以下に示す。

表 1.2-4 サーバ汎用機能(インフラストラクチャ層)

項番	機能名	機能概要	システム構成要素	コンポーネント名
1	個別DBアクセス	個別データベースのデータ取得及び更新機能。汎用的なAPIを提供するために製品固有のデータアクセス技術、永続化手段を隠蔽する。	業務アプリケーション(ユーザ)	個別DBアクセス
			業務アプリケーション(システム)	
			業務アプリケーション(バッチ)	
2	HTTPクライアント	HTTPを受け付けるWebサービス呼び出すための機能。HTTPによる通信技術や手順を隠蔽する。	業務アプリケーション(ユーザ)	業務AP(システム)アクセス
			業務アプリケーション(システム)	
			業務アプリケーション(バッチ)	
			BPMS補完機能	
3	ルールエンジンアクセス	BRMSヘルールの実行を要求する機能。ビジネスルール管理サブシステムのサービス呼び出しを行う。ロジック層に対してBRMSの呼び出し方式を隠蔽する。	業務アプリケーション(ユーザ)	ルールエンジンアクセス
			業務アプリケーション(システム)	
			業務アプリケーション(バッチ)	
4	外部システムアクセス	ESBが提供する外部システム連携インタフェースを呼び出す機能。上述のHTTPクライアントをそのまま使用するか、それをラップした形態とする。	業務アプリケーション(システム)	外部システムアクセス
			業務アプリケーション(バッチ)	
			BPMS補完機能	
5	ワークフローエンジンアクセス	BPMSのサービス呼び出す機能。上述のHTTPクライアントやBPMS製品が提供するAPIを利用してアクセスするが、アクセス方式をロジック層に対して隠蔽する。	プレゼンテーションロジック	ワークフローエンジンアクセス
			業務アプリケーション(システム)	
			業務アプリケーション(バッチ)	
			BPMS補完機能	
6	FTPクライアント	他サブシステムの業務アプリケーション(バッチ)に対し、ファイルでデータを連携するためにFTP通信を行う。FTP呼び出しを行う際の通信技術や手順を上位層から隠蔽する。	業務アプリケーション(バッチ)	FTPクライアント

(C) Aレイヤ横断

Aレイヤを横断して利用するサーバ汎用機能を以下に示す。

表 1.2-5 サーバ汎用機能(Aレイヤ横断)

項番	機能名	機能概要	システム構成要素
1	依存関係制御	変化の多いコンポーネントの交換可能性を高めることを目的とし、コンポーネントの依存関係を制御する。 DIコンテナ, MVCフレームワーク, AOPフレームワーク, ファクトリ等	● UI層以外のアプリケーションが配置されるシステム構成要素
2	ログ出力	デバッグやシステム監視等の目的でアプリケーションの動作をファイルに記録する機能。 ログ出力機能を利用したログ管理方式については、『別冊4 システム機能共通編』のログ管理方式設計のルールを参照。	● UI層以外のアプリケーションが配置されるシステム構成要素
3	メッセージ管理	画面に出力するメッセージをアプリケーションから切り離された定義情報を使用して管理する機能。メッセージのKeyやID等を指定してメッセージを取得する。	● UI層以外のアプリケーションが配置されるシステム構成要素
4	パラメータ設定値提供	設定ファイルやDBに設定された値を読み取り、アプリケーションで使用するパラメータ値を応答する機能。 ビジネスルールをアプリケーションから切り離す仕組みとしても利用する。『本冊』の「3.3.6.1.1 ビジネスルールをアプリケーションから切り離すための仕組みとして利用する技術」を参照。	● UI層以外のアプリケーションが配置されるシステム構成要素
5	ユーティリティ	日付・年号操作, 文字列操作, CSV, XMLの解析等といったユーティリティ機能。	● UI層以外のアプリケーションが配置されるシステム構成要素

B. クライアント汎用機能(ブラウザ)

(A) インタフェース層

インタフェース層におけるクライアント汎用機能(ブラウザ)を以下に示す。

表中の「システム構成要素」及び「コンポーネント」は、機能の実行場所を示す。機能の配信場所は、プレゼンテーションロジックのリクエストハンドラ・コンポーネントである。

表 1.2-6 クライアント汎用機能(ブラウザ) (インタフェース層)

項番	機能名	機能概要	システム構成要素	コンポーネント名
1	共通画面部品	テキストボックス、ボタン等の基本部品や、カレンダーや書式付きエディタ等といった応用部品のスタイルや機能。 画面設計ルール画面部品の実現する。画面部品については、『参考2 画面・帳票編』の画面の内容(画面部品の扱い)の章を参照。	ブラウザ	画面
2	イベント制御	画面操作によって発生するクリック等のイベントを制御し、対応する処理を実行する機能。 画面設計ルール画面の動作を実現する。画面の動作については、『参考2 画面・帳票編』の画面の動作の章を参照。	ブラウザ	イベントハンドラ
3	入力規制	禁止文字や入力桁数制限等といった、不正となる文字を入力できないように制御する機能。入力済みの値をチェックする入力チェックとは異なり、文字自体の入力を正当な入力値のみを許可するよう制限する機能である。不正な入力を入力規制で防止するか、クライアント汎用機能の入力チェックで防止するかは、ユーザ利便性や使用する技術などの観点から設計すること。	ブラウザ	イベントハンドラ

(B) ロジック層

ロジック層におけるクライアント汎用機能(ブラウザ)を以下に示す。

表中の「システム構成要素」及び「コンポーネント」は、機能の実行場所を示す。機能の配信場所は、プレゼンテーションロジックのリクエストハンドラ・コンポーネントである。

表 1.2-7 クライアント汎用機能(ブラウザ) (ロジック層)

項番	機能名	機能概要	システム構成要素	コンポーネント名
1	入力チェック	ユーザの入力値チェック(必須チェック, フォーマットチェック, 範囲チェック, 関連性チェック等)を行う機能。 画面設計ルールの入力チェックを実現する。入力チェックについては, 『参考2 画面・帳票編』の画面の動作の章を参照。 入力チェック機能が利用されるタイミングについては, 『参考1 処理シーケンスサンプル集』のオンライン処理方式の章を参照。	ブラウザ	画面ロジック

(C) インフラストラクチャ層

インフラストラクチャ層におけるクライアント汎用機能(ブラウザ)を以下に示す。

表中の「システム構成要素」及び「コンポーネント」は、機能の実行場所を示す。機能の配信場所は、プレゼンテーションロジックのリクエストハンドラ・コンポーネントである。

表 1.2-8 クライアント汎用機能(ブラウザ) (インフラストラクチャ層)

項番	機能名	機能概要	システム構成要素	コンポーネント名
1	サーバ通信	サーバと通信するための機能。通信技術や手順を隠蔽する。	ブラウザ	プレゼンテーションロジックアクセス

(D) Aプレイヤー横断

Aプレイヤーを横断して利用するクライアント汎用機能(ブラウザ)を以下に示す。

表中の「システム構成要素」は、機能の実行場所を示す。機能の配信場所は、プレゼンテーションロジックのリクエストハンドラ・コンポーネントである。

表 1.2-9 クライアント汎用機能(ブラウザ) (Aプレイヤー横断)

項番	機能名	機能概要	システム構成要素
1	ユーティリティ	日付・年号操作, 文字列操作, CSV, XMLの解析等といったユーティリティ機能。	ブラウザ

C. クライアント汎用機能(リッチクライアント)

(A) インタフェース層

インタフェース層におけるクライアント汎用機能(リッチクライアント)を以下に示す。

表 1.2-10 クライアント汎用機能(リッチクライアント)(インタフェース層)

項番	機能名	機能概要	システム構成要素	コンポーネント名
1	共通画面部品	テキストボックス、ボタン等の基本部品や、カレンダーや書式付きエディタ等といった応用部品のスタイルや機能。 画面設計ルール画面部品の実現する。 画面部品については、『参考2 画面・帳票編』の画面の内容(画面部品の扱い)の章を参照。	リッチクライアント	画面
2	イベント制御	画面操作によって発生するクリック等のイベントを制御し、対応する処理を実行する機能。 画面設計ルール画面の動作を実現する。 画面の動作については、『参考2 画面・帳票編』の画面の動作の章を参照。	リッチクライアント	イベントハンドラ
3	入力規制	禁止文字や入力桁数制限等といった、不正となる文字を入力できないように制御する機能。入力済みの値をチェックする入力チェックとは異なり、文字自体の入力を正当な入力値のみを許可するよう制限する機能である。不正な入力を入力規制で防止するか、クライアント汎用機能の入力チェックで防止するかは、ユーザ利便性や使用する技術などの観点から設計すること。	リッチクライアント	イベントハンドラ

(B) ロジック層

ロジック層におけるクライアント汎用機能(リッチクライアント)を以下に示す。

表 1.2-11 クライアント汎用機能(リッチクライアント)(ロジック層)

項番	機能名	機能概要	システム構成要素	コンポーネント名
1	入力チェック	ユーザの入力値チェック(必須チェック、フォーマットチェック、範囲チェック、関連性チェック等)を行う機能。 画面設計ルール画面の入力チェックを実現する。 入力チェックについては、『参考2 画面・帳票編』の画面の動作の章を参照。 入力チェック機能が利用されるタイミングについては、『参考1 処理シーケンスサンプル集』のオンライン処理方式の章を参照。	リッチクライアント	画面ロジック

(C) インフラストラクチャ層

インフラストラクチャ層におけるクライアント汎用機能(リッチクライアント)を以下に示す。

表 1.2-12 クライアント汎用機能(リッチクライアント)(インフラストラクチャ層)

項番	機能名	機能概要	システム構成要素	コンポーネント名
1	サーバ通信	Webサービスを呼び出すための機能。通信技術や手順を隠蔽する。	リッチクライアント	プレゼンテーションロジックアクセス

(D) APLレイヤ横断

APレイヤを横断して利用するクライアント汎用機能(リッチクライアント)を以下に示す。

表 1.2-13 クライアント汎用機能(リッチクライアント)(APレイヤ横断)

項番	機能名	機能概要	システム構成要素
1	依存関係制御	コンポーネントの依存関係を制御する。DIコンテナ, MVCフレームワーク, AOPフレームワーク, ファクトリ等	リッチクライアント
2	ログ出力	デバッグやシステム監視等の目的でアプリケーションの動作をファイルに記録する機能。 ログ出力機能を利用したログ管理方式については、『別冊4 システム機能共通編』のログ管理方式設計のルール of 章を参照。	リッチクライアント
3	メッセージ管理	画面に出力するメッセージをアプリケーションから切り離された定義情報を使用して管理する機能。メッセージのKeyやID等を指定してメッセージを取得する。	リッチクライアント
4	パラメータ設定値提供	設定ファイルやDBに設定された値を読み取り, アプリケーションで使用するパラメータ値を応答する機能。 ビジネスルールをアプリケーションから切り離す仕組みとしても利用する。『本冊』の「3.3.6.1.1 ビジネスルールをアプリケーションから切り離すための仕組みとして利用する技術」を参照。	リッチクライアント
5	ユーティリティ	日付・年号操作, 文字列操作, CSV, XMLの解析等といったユーティリティ機能。	リッチクライアント

D. 共有コンポーネント

共有コンポーネントは、インタフェース層及びロジック層の機能は持たない。

(A) インフラストラクチャ層

インフラストラクチャ層における共有コンポーネントを以下に示す。

表 1.2-14 共有コンポーネント(インフラストラクチャ層)

項番	機能名	機能概要	システム構成要素	コンポーネント名
1	共有DBアクセス (基盤API)	共有データベースのデータ取得及び更新機能。 『DBアクセス基盤 利用者向けガイドライン』の基盤APIを指す。	業務アプリケーション(ユーザ) 業務アプリケーション(システム) 業務アプリケーション(バッチ) DBアクセス基盤サービス	共有DBアクセス(基盤API)

- (3) フレームワークに対する配置
 フレームワーク機能の各フレームワークへの配置を以下に示す。

表 1.2-15 フレームワーク機能の配置

項番	機能分類	Aレイヤ	機能名	特許庁開発フレームワーク			
				画面オンライン	サービス	バッチ	リッチクライアント
1	サーバ汎用機能	インタフェース層	Webサービス	○ ³	○	×	×
2			RESTサービス	○ ³	○	×	×
3			LDAPアクセス	○	×	×	×
4			認証・認可チェック	○	○	×	×
5			入力チェック	○	○	○	×
6			ディレード処理要求制御	×	×	○	×
7		インフラストラクチャ層	個別DBアクセス	○	○	○	×
8			HTTPクライアント	○	○	○	×
9			ルールエンジンアクセス	○	○	○	×
10			外部システムアクセス	×	○	○	×
11			ワークフローエンジンアクセス	○	○	○	×
12			FTPクライアント	×	×	○	×
13		Aレイヤ横断	依存性管理	○	○	○	×
14			ログ出力	○	○	○	×
15			メッセージ管理	○	○	○	×
16			パラメータ設定値提供	○	○	○	×
17			ユーティリティ	○	○	○	×
18	クライアント汎用機能(ブラウザ)	インタフェース層	共通画面部品	○	×	×	×
19			イベント制御	○	×	×	×
20			入力規制	○	×	×	×
21		ロジック層	入力チェック	○	×	×	×
22		インフラストラクチャ層	サーバ通信	○	×	×	×
23		Aレイヤ横断	ユーティリティ	○	×	×	×
24	クライアント汎用機能(リッチクライアント)	インタフェース層	共通画面部品	×	×	×	○
25			イベント制御	×	×	×	○
26			入力規制	×	×	×	○
27		ロジック層	入力チェック	×	×	×	○
28		インフラストラクチャ層	サーバ通信	×	×	×	○
29		Aレイヤ横断	依存関係制御	×	×	×	○
30			ログ出力	×	×	×	○
31			メッセージ管理	×	×	×	○
32			パラメータ設定値提供	×	×	×	○
33			ユーティリティ	×	×	×	○
34	共有コンポーネント	インフラストラクチャ層	共有DBアクセス(基盤API)	○	○	○	×

凡例： ○:配置する(ただし業務要件がない場合は配置しない), ×:配置しない。

³ UI層がリッチクライアントの場合に、Webサービス、RESTサービス的一方又は両方が配置される。詳細は「3.2.1.1.2 内部インタフェースの Protokol」参照。UI層がブラウザの場合には配置されない。

1.3 アプリケーション基盤のサブシステム間の共有

AP基盤は、原則としてサブシステム毎に設計・開発するものであるが、複数のサブシステム間で同一のフレームワーク部品を利用する等して、開発効率を向上させることを禁止するものではない。しかしながら、フレームワーク機能への変更要求の内容やサブシステムの刷新のタイミングはサブシステム毎に異なる可能性があるため、サブシステム毎に独立してAP基盤を更新できなければならない。特に、AP基盤は完全な作り込みではなくフレームワーク製品を利用して実現する想定であるため、製品のバージョンや特定バージョンに依存した作り込み部分の資材管理をサブシステム毎に行う必要がある。

あるサブシステムの刷新がAP基盤の更新を伴うときに、同じフレームワーク部品を利用している別のサブシステムに影響を及ぼすような状況を防ぐため、AP基盤の設計に関与するステークホルダ(設計・開発ベンダ等)は、サブシステム毎に独立してバージョンアップができるよう、資材管理の仕組みを整備することとする。