

大阪市情報システム開発ガイドライン

(別冊) 機種更新編

令和 5 年 4 月
デジタル統括室

改定履歴

改定日付	改定概要
令和 2 年 4 月 1 日	・ 新規作成
令和 4 年 4 月 1 日	・ 組織名変更に伴う変更
令和 5 年 4 月 1 日	・ 体裁の修正

目 次

第1章	(別冊)機種更新編の概要	1 -
1.	本書の適用範囲	2 -
2.	本書の概要	3 -
第2章	機種更新にあたって	5 -
1.	機種更新の特徴	5 -
2.	機種更新プロジェクト期間中のシステム改修対応	5 -
3.	成果物の作成	6 -
第3章	重要度の高い工程	7 -
1.	「(追加)非互換調査」	7 -
2.	「(2)要件定義」	15 -
3.	「(3)基本設計」	19 -
4.	「(4)詳細設計」	26 -
5.	「(20)プログラム設計・製造」	27 -
6.	「(21)単体テスト」	28 -
7.	「(24)総合テスト」	28 -
8.	「(27)運用テスト」	29 -
第4章	その他の工程	30 -
1.	各工程における補足	30 -

第1章 （別冊）機種更新編の概要

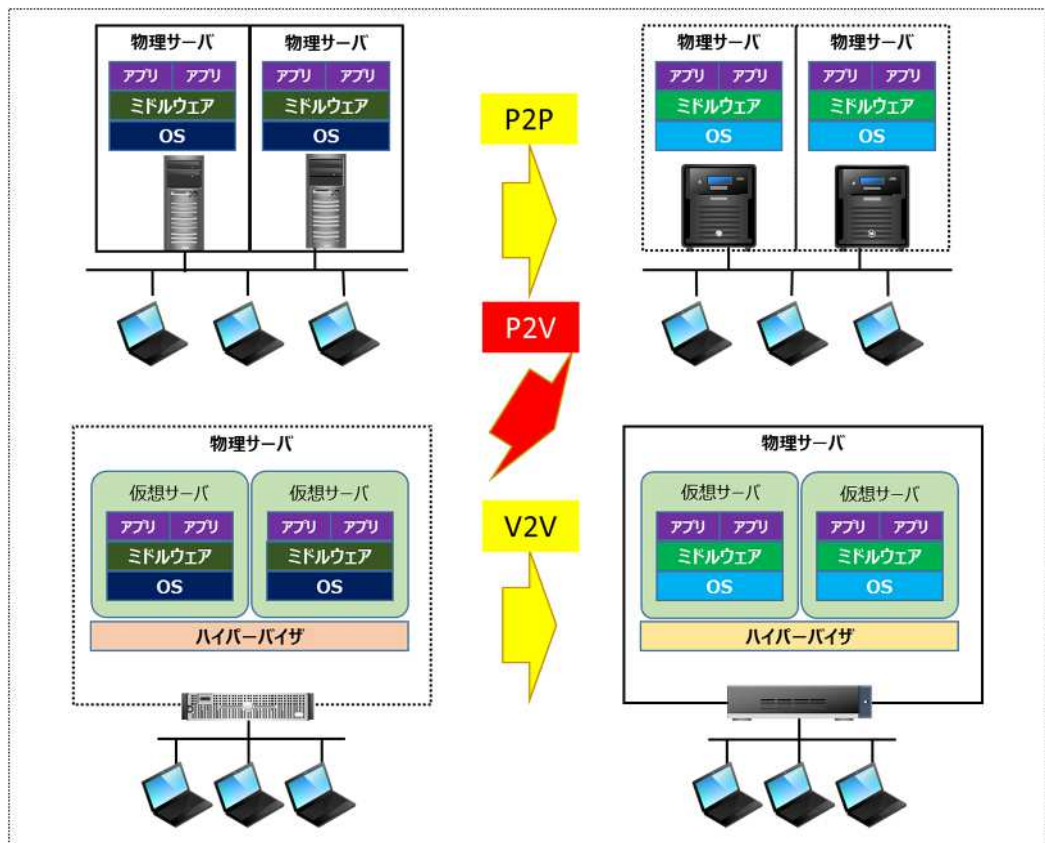
大阪市情報システム開発ガイドライン（以下「開発ガイドライン」という。）の別冊となる機種更新編には、情報システム（以下「システム」という。）の機器のリース期間満了に伴い、新しい機器に更新する際の留意点を記載しています。

機種更新の各工程は、開発ガイドラインに準拠して作業を進めますが、機種更新の特徴として OS やソフトウェア製品等の最新化に伴うシステム改修が必要となる場合が大半を占めることから、新規構築・再構築のプロジェクトと比較して、追加・強化・簡略化される工程があります。これらを踏まえて信頼性・安全性の高いシステム構築を実施するために重要となるポイントを本書で記載し、QCD（Quality=品質、Cost=費用、Delivery=納期）を確保することを目的とします。

1. 本書の適用範囲

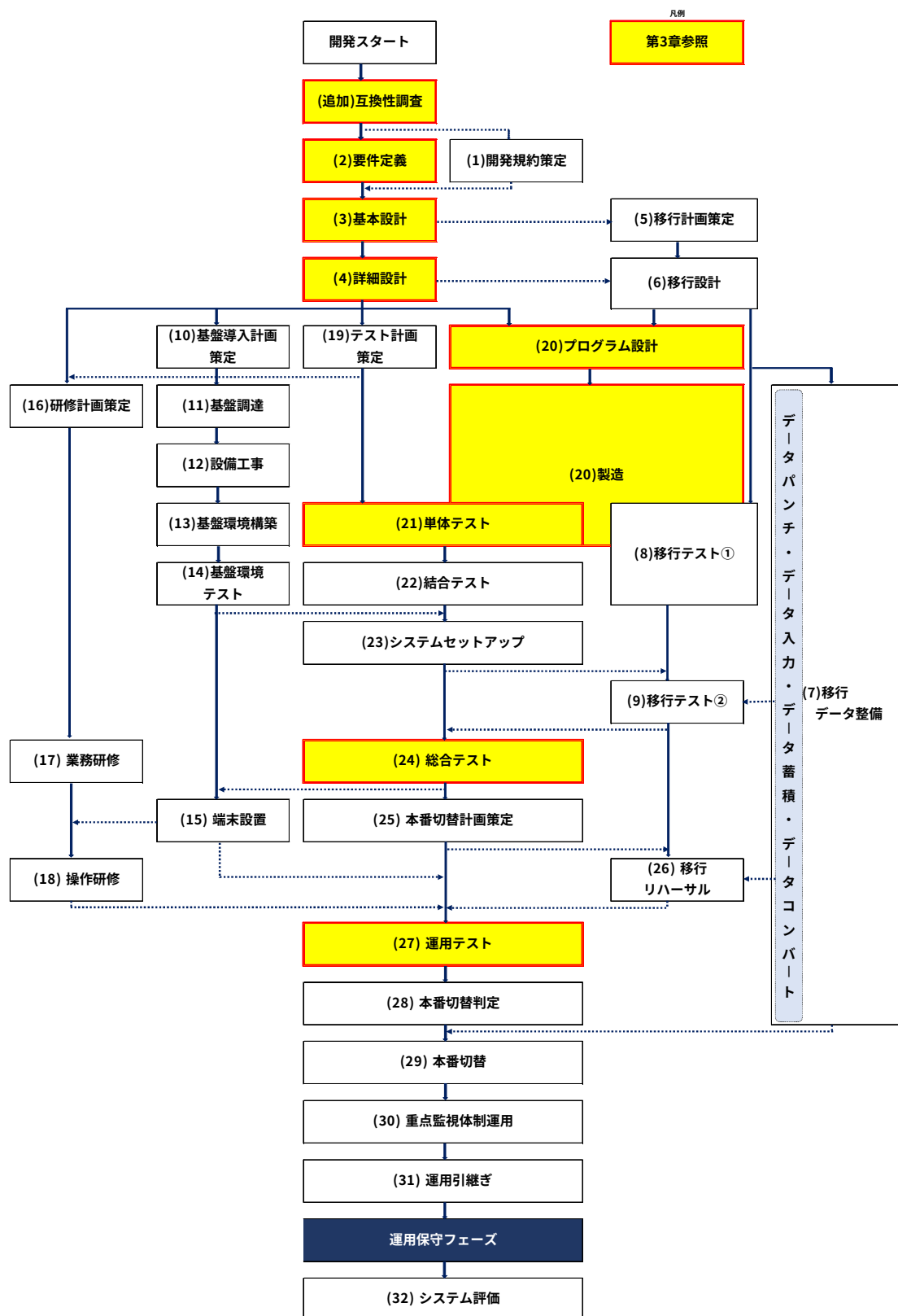
本書は、業務改善やシステム改善を伴わない機器の更新（以下「ストレート移行」という。）を対象とします。業務改善やシステム改善を伴う場合や大幅な処理方式変更を伴う機種更新の場合は、開発ガイドラインに従って実施してください。

なお、本書では物理サーバの仮想化環境へのストレート移行（P2V）を想定した記載となっているため、これから実施する機種更新のシステム構成に応じて適宜読み替えてください。



2. 本書の概要

機種更新における標準プロセスとして、開発ガイドラインで定めるシステム開発の流れに「非互換調査」工程を追加し「機種更新標準プロセス」と定義したうえで、本書の第2章で機種更新の特徴や考慮すべきポイント、第3章では新たに追加した非互換調査をはじめ各工程の中で特に重要な工程について留意すべき点を具体的に説明します。最後に、第4章でその他の各工程について補足事項を一覧にしています。



第2章 機種更新にあたって

本章では、機種更新のプロジェクトを進める上で、最初に理解しておくべき点や計画を立てるうえで考慮すべき点について説明します。これは、プロジェクトを成功に導くために必ず押さえておくべき内容です。

1. 機種更新の特徴

機種更新の最大の特徴は、現行システムの運用実績や現行プログラムの処理結果を有効に活用して各工程を実施できる点です。具体的な内容は3章に記載しますが、主な観点は次のとおりです。

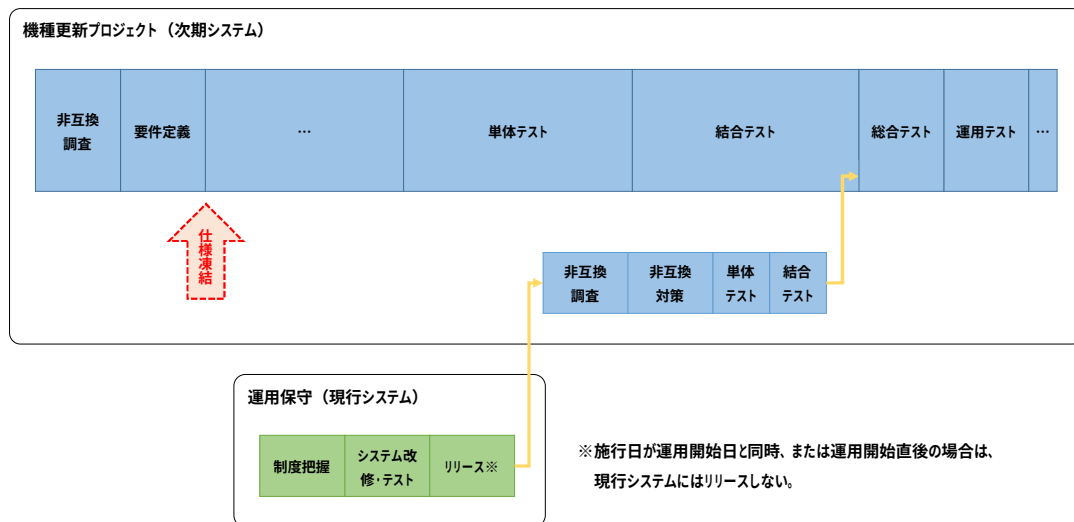
- ・ 運用実績をベースに、次期ハードウェアに求める非機能要件（システムの応答時間、データ保存容量など）を数値化し、ハードウェア選定を行う。
- ・ ハードウェアやソフトウェアの最新化に伴う非互換箇所を確認したうえで非互換対策のためのシステム改修を行う。
- ・ テスト工程では、現行プログラムの処理結果と機種更新対応後のプログラムの処理結果を比較するテスト（以下「現新比較テスト」という。）を効果的に実施する。
- ・ 既存のドキュメント類やテストシナリオを有効に活用して効率的かつ確実に各工程を実施する。

2. 機種更新プロジェクト期間中のシステム改修対応

機種更新のプロジェクト期間中に、現行システムに対して機能追加や制度改正等のシステム改修作業が発生した場合は、その改修内容を次期システムにも漏れなく反映する必要があります。このため、業務担当からの改修要望はできる限り早期に締め切り、発注者と受注者の間で次期システムの仕様を確定（以下「仕様凍結」という。）する必要があります。仕様凍結が徹底されずに新たな要件が発生すると、設計の見直しが必要になり、大幅なプロジェクトの遅延に繋がる可能性があります。

しかし、法改正や制度改正など（以下「法改正等」という。）の外的要因によって仕様凍結以降の対応が求められる場合もあるため、法改正等の内容や施行時期に関する情報を迅速に把握し、法改正等の施行日を基準に機種更新プロジェクトへの影響を最小限に抑える改修計画を検討します。

- ・ 施行日が次期システムの運用開始前や運用開始同日の場合、現行システムの運用保守として法改正等に伴うシステム改修を完了させます。次に、現行システムの改修したプログラムに対して非互換対策を講じて、法改正等に伴うシステム改修対応の結合テストを終え、機種更新プロジェクトの総合テストに臨みます。



- ・ 施行日が次期システムの運用開始後の場合、法改正等対応の規模と時期を踏まえ、当該機種更新プロジェクトにおいて対応を行うか、機種更新終了後に対応を行うかを判断します。

また、重要度の高いパッチが緊急リリースされた場合は、パッチを適用しない場合のリスク、パッチ適用による影響（再テストが必要な範囲やプログラムへの影響など）、本番切替予定日の遵守要否等を総合的に勘案し、パッチの適用要否及び適用タイミングを判断します。この状況を回避するために、パッチのリリース予定を事前に確認し、パッチ適用計画を作成することが重要です。

これらを踏まえ、機種更新プロジェクトへの影響を最小限に抑えたスケジュールを作成する必要があります。

3. 成果物の作成

機種更新プロジェクトの標準成果物は、非互換調査において作成される「非互換・非推奨項目一覧表」を除き、開発ガイドラインと同一のものを作成します。

なお、現行システムの成果物のうち有効に活用できるものについては最新化することを基本とし、成果物作成におけるコストと時間の削減を図ります。

第3章 重要度の高い工程

本章では、新たに追加した非互換調査工程をはじめ、機種更新の中で特に重要な工程について留意すべき事項を具体的に説明しています。

1. 「(追加)非互換調査」

(1) 概要

本工程は、新しいハードウェアやソフトウェア上で、現行システムのプログラムやジョブ制御文等が想定どおり動作しない要因がないか、ソフトウェア製品（ミドルウェア、業務パッケージソフトウェア、汎用ソフトウェア、開発言語等をいう。以下同じ。）の仕様変更に伴う画面表示や操作方法の変更等の非互換箇所がないかを調査する機種更新特有の工程です。

同一ソフトウェア製品の後継版であっても想定どおり動作しないことがあるため、現行システムとの非互換箇所や非推奨手法の利用箇所（以下「非互換箇所等」という。）を特定して、非互換箇所や非推奨手法の回避のための対策（以下「非互換対策」という。）を検討します。その結果に基づきプログラム等の改修範囲及びシステム運用に影響する部分を特定し、対応に必要な工数を明らかにするとともに、次期システムで利用するソフトウェア製品のバージョンを決定します。

(2) 機種更新における留意事項

① ハードウェア環境の見直し

現行システムの運用実績（データ量、通信量、レスポンス時間、同時アクセス数並びに、CPU、ディスクなどのシステム状況）をベースに、次期ハードウェアに求める非機能要件を数値化したうえで、業務環境や情報セキュリティを取り巻く情勢の変化、仮想化への移行などを踏まえてハードウェア選定を行います。

なお、仮想化移行にあたっては、現行機器からの効率化（設置面積、電力量、荷重、機器費用、運用負荷などの縮減）に関する目標を定めます。

② 非互換調査の実施

〔導入バージョンの決定〕

次期システムに導入する OS 等やソフトウェア製品について、各製品が公開している動作保証条件を踏まえ、導入実績、ライフサイクル、サポート提供期間の観点から次期バージョンを選定します。

◆ ハイパーバイザー・OS

現行 OS の後継製品を基本に次期 OS やハイパーバイザーを選定します。複数のバージョンが流通している場合は、ハイパーバイザーと次期 OS の親和性も考慮し、次期ハードウェアのリース期間中はサポートが提供されることを前提に、新機能の必要性を踏まえて選択します。

なお、次期 OS の選択にあたっては、ダウングレード権の利用も検討します。

◆ ミドルウェア

次期 OS で稼働する各ミドルウェアの後継バージョンを選定します。複数のバージョンが流通している場合は、次期ハードウェアのリース期間中にサポートが提供されることを前提に、新機能の必要性を踏まえて選択します。

ミドルウェアに後継バージョンが存在しない場合は、必要な機能が網羅され、次期システムとの親和性が高い代替製品を選択します。

なお、ミドルウェアを代替製品に乗り換える場合は、大幅な処理方式変更となる場合があります。

◆ パッケージソフトウェア（業務）

次期 OS で稼働するパッケージソフトウェア（業務）のバージョンを現行バージョンの利用継続も含め選定します。バージョンアップが必要で、かつカスタマイズを行っている場合は、大幅な対応工数が必要となる可能性があるため、パッケージベンダーとカスタマイズに関する対策を検討します。

◆ パッケージソフトウェア（汎用）

バックアップソフトやウイルス対策ソフトなどの汎用ソフトウェアについて、次期 OS で稼働する各ソフトウェアの後継バージョンを選択します。汎用ソフトウェアの後継バージョンが存在しない場合は、必要な機能が網羅され、次期システムとの親和性が高い代替製品を選択します。

[非互換箇所等の調査]

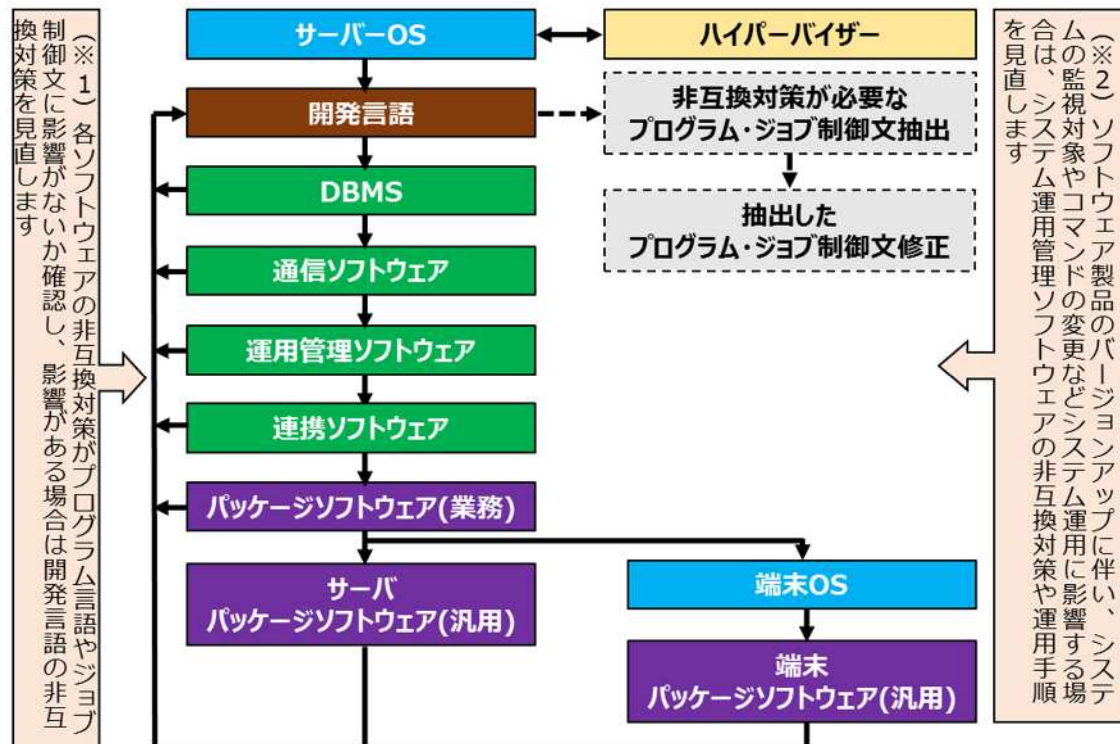
受注業者は、次図【非互換箇所等の調査手順】に沿って、次期システムの OS 等及びソフトウェア製品を次表に示す【非互換確認箇所】を参考に調査し、非互換対策を検討して結果を非互換・非推奨項目一覧表にまとめます。DBMS や通信ソフトウェア、運用管理ソフトウェア、連携ソフトウェア等のミドルウェアの非互換箇所の調査にあたっては、バージョンアップに伴う非互換対策がプログラム言語やジョブ制御文に影響を与えないか確認し、影響を与える場合は開発言語の非互換箇所として非互換・非推奨項目一覧表に追加します。次図※1

また、各ソフトウェア製品のバージョンアップに伴い、システムの監視対象（エラーメッセージ・警告ログ等）の表記や使用しているコマンドの変更などシステム運用に影響

響がないか確認し、影響を与える場合はシステム運用ソフトウェアや運用手順の非互換箇所として非互換・非推奨項目一覧表に追加します。次図※2。

なお、OS等及びソフトウェア製品によっては、バージョン変更時にライセンス形態が変更されることがあり、適切なライセンス形態の選択が必要です。

[非互換箇所等の調査手順]



非互換確認箇所

対象	主な非互換確認箇所	確認観点
OS (基本ソフトウェア)	デフォルト値	設定項目および初期設定値の比較により、次期 OS でデフォルト値（ページングファイルサイズ、システムフォントなど）が変更となる箇所を把握し、主に OS 自身の動作における差異を確認します。
	提供機能	変更・廃止・非推奨となる機能を把握し、代替機能への移行や代替機能の作成要否など、主にシステム運用への影響範囲を確認します。
	API・関数	変更・廃止・非推奨となる API・関数を把握し、代替機能を有する API・関数への移行や代替 API・関数の作成要否など、主に開発言語・システム運用への影響範囲を確認します。
	コマンド (オプション含む)	変更・廃止・非推奨となるコマンドを把握し、代替機能を有するコマンドへの移行や代替手法の作成要否など、主に開発言語・システム運用への影響範囲を確認します。
	ファイル形式	変更・廃止・非推奨となるファイル形式を把握し、代替ファイル形式への移行など、主に開発言語・システム運用への影響範囲を確認します。
	コード体系	変更される文字コードや周辺機器の制御コードを把握し、コード体系の変更要否など、主に開発言語・システム運用への影響範囲を確認します。
	ログ、メッセージ	操作ログやエラーログ等の追加や削除、ログのコードやメッセージ表記の変更を把握し、主にシステム運用への影響範囲を確認します。
	画面表示	システムフォントやメッセージの変更などによる画面表示の変更箇所を把握し、主にシステム操作・システム運用への影響範囲を確認します。
ハイパーバイザー	P2V のため非互換調査なし	本市が求める要件を満たすよう、 <u>設定値を一から検討し</u> 、ハードウェアの物理環境の構築と併せて仮想環境を構築します。
開発言語 (プログラム 言語・ジョブ 制御文)	構文	変更・廃止・非推奨となる構文を把握し、主に開発言語自身への影響範囲を確認します。
	API・関数	変更・廃止・非推奨となる API・関数を把握し、主に開発言語自身への影響範囲を確認します。
DBMS	デフォルト値	設定項目および初期設定値の比較により、次期 DBMS でデフォルト値（メモリ利用方式など）が変更となる箇所を把握し、主に DBMS 自身の動作における差異を確認します。
	提供機能	変更・廃止・非推奨となる機能を把握し、代替機能への移行や代替機能の作成要否など、主にシステム運用への影響範囲を確認します。

対象	主な非互換確認箇所	確認観点
		確認します。
	構文	SQL 文で変更・廃止・非推奨となる構文を把握し、該当部分の SQL 文を変更するなど、主に開発言語への影響範囲を確認します。
	関数	変更・廃止・非推奨となる関数を把握し、代替関数への移行や代替する関数の作成要否など、主に開発言語への影響範囲を確認します。
	コマンド	変更・廃止・非推奨となるコマンドを把握し、代替機能を有するコマンドへの移行や代替手法の作成要否など、主に開発言語・システム運用への影響範囲を確認します。
	ログ、メッセージ	ログのコードやエラーメッセージ表記の変更や、エラーや監査等のログファイルの変更を把握し、主にシステム運用への影響範囲を確認します。
	SQL 統計情報と実行計画	統計情報の内容や取得タイミング・実行計画策定方式・実行計画活用方法の変更内容を把握し、主にシステム運用への影響範囲を確認します。
	格納方式	データの格納方式（並び）に変更がないかを把握し、ソート文追加など、主に開発言語への影響範囲を確認します。
通信ソフトウェア	デフォルト値	設定項目および初期設定値の比較により、次期通信ソフトウェアでデフォルト値（通信プロトコルなど）が変更となる箇所を把握し、影響範囲を確認します。
	提供機能	変更・廃止・非推奨となる機能を把握し、代替機能への移行や代替機能の作成要否など、主にシステム運用への影響範囲を確認します。
	API・関数	変更・廃止・非推奨となる API・関数を把握し、代替 API・関数への移行や代替 API・関数の作成要否など、主に開発言語への影響範囲を確認します。
	コマンド	変更・廃止・非推奨となるコマンドを把握し、代替機能を有するコマンドへの移行や代替手法の作成要否など、主に開発言語・システム運用への影響範囲を確認します。
	ログ、メッセージ	ログのコードやエラーメッセージ表記の変更や、エラーや監査等のログファイルの変更を把握し、主にシステム運用への影響範囲を確認します。
運用管理ソフトウェア	デフォルト値	設定項目および初期設定値の比較により運用管理ソフトウェアでデフォルト値（監視回数など）が変更となる箇所を把握し、主に運用管理ソフトウェア自身の動作における差異を確認します。
	提供機能	変更・廃止・非推奨となる機能を把握し、代替別機能への移行

対象	主な非互換確認箇所	確認観点
		や代替機能の作成要否など、主にシステム運用への影響範囲を確認します。
	画面表示	画面構成や画面表示などが変更される箇所を把握し、運用手順の変更など、主にシステム運用への影響範囲を確認します。
	API・関数	変更・廃止・非推奨となる API・関数を把握し、代替 API・関数への移行や API・関数を代替する機能の作成要否など、主に開発言語・システム運用への影響範囲を確認します。
	コマンド	管理対象機器の OS やソフトウェアのバージョンアップに伴い、ジョブ制御文の実行コマンド変更が必要となる箇所を把握し、主に開発言語・システム運用への影響範囲を確認します。
	ログ、メッセージ	ログのコードやエラーメッセージ表記の変更や、エラーや監査等のログファイルの変更を把握し、主にシステム運用への影響範囲を確認します。
	[端末管理を行っている場合] エージェント	端末にインストールするソフトウェア（エージェント）のバージョンも次期ソフトウェアに対応したものに変更が必要となる場合があるため、端末における対応バージョンを確認します。
連携ソフトウェア	デフォルト値	設定項目および初期設定値の比較により連携ソフトウェアでデフォルト値（フォルダパスなど）が変更となる箇所を把握し、主に連携ソフトウェア自身の動作における差異を確認します。
	コマンド	変更・廃止・非推奨となるコマンドを把握し、ジョブ制御文や運用手順の変更など、主に開発言語・システム運用への影響範囲を確認します。
	ログ、メッセージ	ログのコードやエラーメッセージ表記の変更やエラーや監査等のログファイルの変更を把握し、主にシステム運用への影響範囲を確認します。
	API・関数	変更・廃止・非推奨となる API・関数を把握し、代替 API・関数への移行や代替 API・関数の作成要否など、主に開発言語への影響範囲を確認します。
パッケージソフトウェア（業務、カスタマイズなし）	提供機能	変更・廃止・非推奨となる機能を把握し、代替ソフトウェアへの変更や代替機能への移行など、主にシステム運用への影響範囲を確認します。
	画面	現行ソフトウェアと次期ソフトウェア（Web 化によるブラウザ方式移行への変更含む）の画面遷移・色・画面レイアウト・表示フォント・ボタン類の初期選択値・タブ設定等の比較により変更箇所を把握し、主にシステム操作・開発言語・システム運用への影響範囲を確認します。
	帳票	帳票レイアウト・印字フォント等の比較により変更箇所を把握し、

対象	主な非互換確認箇所	確認観点
		主に業務への影響範囲を確認します。
	入出力ファイルレイアウト	入出力ファイルレイアウトの比較により変更箇所を把握し、主に他システムとの連携への影響範囲を確認します。
	コード体系	コード体系の比較により変更箇所を把握し、主に他システムとの連携への影響範囲を確認します。
	API・関数	変更・廃止・非推奨となる API・関数を把握し、代替 API・関数への移行や代替 API・関数の作成要否など、主に開発言語への影響範囲を確認します。
	ログ、メッセージ	ログのコードやエラーメッセージ表記の変更や、エラーや監査等のログファイルの変更を把握し、主にシステム運用への影響範囲を確認します。
パッケージソフトウェア（汎用）	提供機能	変更・廃止・非推奨となる機能を把握し、代替ソフトウェアへの変更や代替機能への移行など、主にシステム運用への影響範囲を確認します。
	コード体系	コード体系の比較により変更箇所を把握し、主に他システムとの連携への影響範囲を確認します。
	API・関数	変更・廃止・非推奨となる API・関数を把握し、代替 API・関数への移行や代替 API・関数の作成要否など、主に開発言語への影響範囲を確認します。
	ログ、メッセージ	ログのコードやエラーメッセージ表記の変更や、エラーや監査等のログファイルの変更を把握し、主にシステム運用への影響範囲を確認します。

③ システム資産整理

現行システムの全プログラムとジョブ制御文を把握し、未使用となった機能や不要な機能に該当する未稼働・不要なプログラムとジョブ制御文等を抽出して、プログラム資産等の廃棄を行います。

④ プロトタイプ検証

調査で判明した非互換対策について、次期バージョンの OS やソフトウェア製品等で構成した検証用ハードウェア上で、調査結果の対応方法が想定どおりの動作となるか確認を行います。

【プロトタイプ計画書の策定】

開発責任者は、標準化リーダー、基盤リーダー、受注業者と協力してプロトタイプ検証の対象、方法、環境等について立案し、受注業者はプロトタイプ計画書としてまとめ

ます。

【環境整備】

受注業者は、次期システムで採用する OS・ソフトウェア製品で構成したプロトタイプ検証のための環境を用意します。

【検証実施】

受注業者はプロトタイプ検証環境下で非互換対策が想定どおり動作するか、対応工数が想定内に収まるかを確認します。

【コンバージョンツール作成判定】

非互換対策において、コンバージョンツールを利用して一律に現行のプログラムやジョブ制御文などの構文を変換することで、効率的に機種更新が行えないか検討します。コンバージョンツールの作成工数と手作業による工数を比較してツール作成有無を判定します。

2. 「(2)要件定義」

① 概要

本工程は、開発プロジェクトと異なり、次期ハードウェアに求める要件を中心に定義する工程です。

機種更新においては、現行システムの稼働状況を踏まえて次期システムに要求する効果を明確にし、次期ハードウェアに必要な性能・耐障害性やセキュリティ対策などの非機能要件を検討します。

② 機種更新における留意事項

非互換調査のハードウェア環境見直しで調査した結果（データ増量、レスポンス時間、同時アクセス数並びに、CPU、ディスクなどのシステム状況）や業務環境、情報セキュリティを取り巻く情勢を踏まえ、開発ガイドラインに定める要件定義項目を見直して要件定義書を改訂します。特にハードウェアを更新することに伴い、次表：要件定義項目の留意事項を踏まえて重点的に見直します。

要件定義項目

記載目次	記載事項	説明	留意事項
はじめに	システム化の目的	システム化を行う目的や期待する効果、効果の測定方法と指標	—
	システム化の方針	システム化に至った背景・経過を踏まえたシステム化の方針	—
前提条件・制約事項	プロジェクト計画書で記載した事項	プロジェクト計画書で定義した前提条件・制約事項	—
	本工程で発生した事項	プロジェクト計画書確定後、本工程実施までに新たに発生した前提条件・制約事項	—
システム概要	システム化の範囲	事務・業務のうちシステム化する範囲	—
	システム概念図	構築するシステム（ハードウェア構成、ネットワーク構成、ソフトウェア構成）及びそのシステムと連携するシステムとの関連を記載した概念図	—
業務要件	業務概要	業務実施における各担当の役割や実施作業の概要	—
	業務管理体制	システム化後の業務・システム管理体制	—

記載目次	記載事項	説明	留意事項
	業務一覧	機能や事務単位に細分化した業務の一覧（システム化対象業務、システム化対象外業務の別）	—
	業務フロー	システム化後の業務の流れ図（システム化範囲が判断可能なもの）	—
	業務権限	各業務処理について、役職や担当毎に与える権限（閲覧権限/更新権限）を整理したもの	—
	業務時間（日中/窓口終了後/翌開庁日）	各業務を実施する時間帯	—
	業務周期（日次/週次/月次/四半期次/半期次/年次/随時）	各業務を実施する周期	—
	業務量（日/週/月/四半期/半期/年）	一定期間における入力対象伝票数や一括処理における対象人数等の業務量	—
	通常期/繁忙期/閑散期	1年の業務量の偏り具合や1日のうち、最も入力作業等が集中する時間帯とその業務量等	—
	利用部署、利用拠点、利用人数	利用者や端末を設置する部署に関する情報	—
	納期（マイルストーン）	法・制度改正等や業務の計画による納期。その他の確定している予定	—
	システム切替方針	一括リリース、拠点単位のさみだれリリースなど、システムのリリース方法に関する方針	現行システムのサービス終了タイミングとシステム切替方針を踏まえて、次期システムへのデータ移行作業、現行・次期ハードウェアの並行設置期間や撤去作業を見直します。
	移行方針	紙資料等からデータ化の方法、一括移行、差分移行等、移行作業の方針や考え方	
	移行対象	移行対象となる機器やデータ等	
	災害対応方針	災害発生時における業務継続の該当システムの復旧優先度、手作業での代替等、業務継続の方針や考え方 ※上記を踏まえ開発過程でBCP対応を行う	現行システムにおいて発生した災害とその対策状況を踏まえて、方針や考え方を見直します。

記載目次	記載事項	説明	留意事項
機能要件	機能一覧	業務要件を元に、必要な機能を細分化し洗い出したもの。再構築の場合は、現行システムに比べて、新規/変更/廃止が判断できるようにする	—
	入出力一覧	各機能の入出力情報（紙情報、電子情報）の一覧	—
	画面設計方針	開発規約に則って作成する画面の設計方針。本設計方針をベースに全ての画面を作成していく。（サンプルイメージ含む）	—
	帳票設計方針	開発規約に則って作成する帳票の設計方針。本設計方針をベースに全ての帳票を作成していく。（サンプルイメージ含む）	—
	外部連携システム データ一覧	外部連携システムで保有しているデータ一覧とその利用目的、連携先システムに導入されているデータ連携用機能（連携先指定ソフトウェアの利用可否）もまとめる	—
非機能要件	システム構成方針	以下の要件から、システム構成の方針を定義する	—
	性能目標値	オンラインレスポンス（ボタン押下から結果表示まで平均 3 秒など）/ 日次、月次、年次等のバッチ処理時間（2 時間以内など）/ 印刷ボタン押下から帳票出力までの時間等の目標値	現行システムの実績値をベースに、次期システムで必要となるデータ容量や CPU 性能、メモリ容量等を計算し直します。
	拡張性	業務量増加に伴い、サーバリソース増強が見込まれる場合のシステムの拡張方針 ※スケールアウト/スケールアップなど	—
	サービスレベル	IT サービスを提供する事業者と本市の間で締結するサービスの保証項目と提供レベルの設定、定義	現行システムにおいて発生した障害とその業務影響を踏まえて、サービスレベルや復旧水準を見直します。
	目標復旧水準 （業務停止）	業務停止時の復旧水準として RPO（目標復旧地点）、RTO（目標復旧時間）、RLO（目標復旧レベル）を定める	

記載目次	記載事項	説明	留意事項
	目標復旧水準 (災害復旧)	大規模災害時のシステム再開の期間を定める	現行システムにおいて発生した災害とその業務影響を踏まえて、優先して復旧すべき機能や再開までの期間を見直します。
	追跡方針	データ参照・更新に関する監査証跡等のログについて、取得方法や制度・業務要件として保存すべき期間を定める	エラーや監査等のログファイルの変更を踏まえて、取得方法や保存すべき期間を見直します。
	セキュリティ方針	セキュリティを確保するための方針。大阪市セキュリティ対策基準やシステム化する業務で義務付けられている事項等を元にした以下の方針を定める ・脆弱性への対応方針 ・利用者認証の方針 ・暗号化への対応方針 ・侵入検知・監視の方針 ・ウイルス対策の方針	※セキュリティ脅威は年々新たな手法で発生しています。情報セキュリティを取り巻く情勢の変化を踏まえて、セキュリティを確保するための方針について見直します。
	運用保守方針	日次、週次、月次、年次等の定例作業、随時作業、問い合わせ対応等の運用方針。 ※現状の運用体制や運用していくための前提条件・制約事項を記載	—
	研修方針	次期システム利用に関する研修の方針。受講人数や回数、集合/分散等実施方法、実施範囲等	—

なお、開発ガイドラインでは、業務責任者が主体的に要件定義を担うとしていますが、機種更新では開発責任者が要件定義を担うこととします。

3. 「(3)基本設計」

① 概要

本工程は、非互換調査で判明した内容と要件定義で決定した内容に基づき、非互換対策や非機能要件の強化・充実などの基本的な方針を決定する工程です。

機種更新においては、物理サーバの仮想化に伴うハードウェア構成の変更を重点的に検討します。

② 機種更新における留意事項

要件定義において見直しを図った非機能要件を踏まえ、開発ガイドラインに定める基本設計項目を見直して基本設計書を改訂します。特にハードウェア更新に関連する設計内容については、次表：基本設計項目の留意事項を踏まえて重点的に見直します。

基本設計項目

記載目次	記載事項	説明	留意事項
業務設計	オンライン化業務一覧	システム化する業務のうち、オンラインで行う業務の一覧	—
	バッチ化業務一覧	システム化する業務のうち、バッチで行う業務の一覧	—
	入出力一覧	新業務で利用する入力データ、出力データ、帳票等の一覧	—
	新業務フロー	業務要件と機能要件を踏まえ、システム化した業務を反映した業務フロー	—
オンライン機能設計	オンライン機能一覧	システムを構成するオンライン機能の一覧	非互換調査で判明した非互換対策を踏まえて、次期ソフトウェアの設計内容を見直します。
	オンライン機能概要	各オンライン機能の概要。	
	オンライン画面遷移図	各画面の遷移図	
	画面仕様（オンライン）	画面レイアウト、画面項目名、フォント等のユーザインターフェース設計	
	帳票仕様（オンライン）	帳票レイアウト、帳票項目名、フォント等のユーザインターフェース設計	
バッチ機能設計	バッチ機能一覧	システムを構成するバッチ機能の一覧	
	バッチ機能概要	各バッチ機能の概要	
	ジョブネット設計	バッチ処理のフロー ※処理周期や保守性を考慮	

記載目次	記載事項	説明	留意事項
	帳票仕様 (バッチ)	帳票レイアウト、帳票項目名、フォント等のユーザインターフェース設計	
外部連携設計	外部連携データ一覧	外部連携システムとの連携データ一覧レイアウト、データ項目名、処理タイミング、想定件数を整理	—
テーブル設計	マスタテーブル一覧	業務を行うための基礎情報 (マスタ) となるデータを格納するテーブルを定義レイアウト、データ項目の決定や当初件数、増加率、最大件数、保持期限を考慮して全体の格納量を決定	現行システムの実格納量をベースに、次期システムで必要となる業務量増加による格納量の増加見込みなどを計算し直します。
	業務テーブル一覧	業務に伴って発生する業務データを格納するテーブルの定義レイアウト、データ項目の決定や当初件数、増加率、最大件数、保持期限を考慮して全体の格納量を決定	
	ER 図	各テーブルの関係性 (リレーション) を設計した図	—
	CRUD 図	テーブルごとに、どの機能でデータを作成 (Create)、読み込み (Read)、更新 (Update)、削除 (Delete) するかを定義した図 (マトリクス)	—
ファイル設計	ファイル一覧	システムで利用するファイルの一覧	—
	ファイル仕様	ファイル形式 (csv、固定長など)、文字コード、レイアウト (桁数)	—
コード設計	コード一覧	システムで利用するコードと対応する値	変更箇所がある場合は、一覧表を見直します。
システム構成	システムアーキテクチャ	ハードウェア、ソフトウェアのアーキテクチャ	—
	サーバー一覧	サーバの一覧	—
	ハードウェア構成 (論理)	ハードウェアの論理構成図	—
	ハードウェア仕様	ハードウェアの仕様 サーバ (CPU、メモリ、ディスク I/O) やディスク容量、各種端末、ネットワーク機器等	想定している仮想サーバを稼働させるために必要な物理サーバや各種端末、ネットワーク機器等のハードウェア仕様を設計します。
	ソフトウェア一覧	ソフトウェア製品・バージョンの一覧と選定理由、各サーバへの導入有無が判	非互換調査で選定したソフトウェア製品・バージョンを反映した一覧に

記載目次	記載事項	説明	留意事項
		断できるようにする	見直します。
	ソフトウェア構成	ソフトウェアの構成図 ※システムの拡張に備え、ライセンス形態も整理	—
	ネットワーク構成 (論理)	ネットワークの論理構成図	—
	環境定義	本番、研修、保守、開発、それぞれの環境の有無と役割・構成を定義	現行システムの研修、保守、開発、それぞれの環境の利用実績を踏まえて必要性や役割について見直します。
システム 方式設計	DB 方式設計	DB の論理構成、冗長化構成の採用有無、インスタンスの構成、想定 DBMS 製品・バージョン	非互換調査で判明した非互換対策を踏まえて、システム方式の設計内容を見直します。 ただし、大幅な処理方式変更を伴う場合は、再構築と同様に開発ガイドラインに従ってプロジェクトを実施する必要があります。
	オンライン処理方式設計	Web、C/S 等のオンライン機能の処理方式	
	バッチ処理方式設計	バッチ処理の自動化方針、採用するスケジューラ等の処理方式	
	帳票方式設計	オンライン帳票/配信帳票等を印刷する際の処理方式	
	非定型方式設計	汎用的なデータ抽出機能等、非定型な処理の方式	
	外部連携方式設計	想定ソフトウェア、連携方法等の処理方式を記載	
	(追加) コンバージョンツール方式設計	コンバージョンツールを採用することが効果的かを検討する。	コンバージョンツールを作成することになった場合は、コンバージョンツールの設計を行います。
性能 方式設計	性能設計	リソースの選定理由や性能見積。 ※性能の検証方法も定義	想定している仮想サーバが稼働可能な物理サーバに求める性能に設計を見直します。
拡張性 方式設計	拡張性設計	サーバ本体のスケールアップ/スケールアウト、DB 拡張、通信量拡張を設計 ※ライセンスに注意	—

記載目次	記載事項	説明	留意事項
可用性 方式設計	冗長化方式設計	ハードウェア/ソフトウェアの冗長化(代替用の設備、機能を用意しておき、故障や障害が発生した場合にサービスを継続的に提供できるようにすること)の設計	現行システムにおいて発生した障害とその業務影響を踏まえて、業務環境に見合った設計に見直します。
耐障害性 方式設計	リカバリ方式設計	リカバリポイントまでの復旧保障に向けた方式。リストア/オンライン入力/バッチ処理におけるリカバリポイントの考え方	
	バックアップ方式設計	バックアップに関する方式。対象(DB、ファイル)、バックアップ先(自ディスク内、バックアップ用ディスク)、取得タイミング(オンライン終了後やバッチ処理の前後など)、保存期間	
災害時復旧 方式設計	災害時復旧設計	大規模災害発生時のBCPを考慮したシステム復旧方式(サーバ設置拠点から離れた場所に保管した、複製システム・データからの復旧など)	現行システムにおいて発生した災害とその業務影響を踏まえて、優先して復旧すべき機能や再開までの期間などの設計を見直します。
	バックアップ方式設計	バックアップに関する方式。対象(システム、DB、ファイル、ログ)、保有世代数、取得周期、保管先	
追跡・監視 方式設計	ログ出力項目	業務ログ、システムログ、監査証跡への出力項目・内容、出力時期の設計	取得方法や保存すべき期間などの設計を見直します。
	監視設計	監視する項目と通知有無、及び検知するハードウェア/ソフトウェアの設計	—
セキュリティ 方式設計	認証方式	不正利用を防ぐための認証方式	機種更新に合わせて、個別認証方式からシングルサインオンへの変更を行う場合は、認証方式の決定、ハードウェア構成やアプリケーションの変更等による影響などを検討して設計を行います。
	暗号化方式	データの暗号化方式:暗号化アルゴリズムや暗号化ツール等	※セキュリティにおける脅威は年々新たな手法で発生しています。情報セキュリティを取り巻く情勢の変化を踏まえて、セキュリティを確保するための設計に見直します。
	脆弱性対策方式	システムの脆弱性対策:セキュリティパッチの適用方針等	
	ウイルス対策方式	ウイルスへの対策方式:利用する対策ソフトウェアと選定理由	
	不正侵入対策	外部・内部からの不正侵入防止方式:	

記載目次	記載事項	説明	留意事項
	方式	FW (ファイアーウォール)、ポート制御、通信方向の制御等	
	特権アカウント一覧	OS、ミドルウェア等の特権アカウント	—
	権限マトリクス	業務アプリケーションの管理者・利用者アカウントと権限の一覧	—
運用保守設計	運用・保守作業概要	システムの運用・保守時に発生する作業概要	—
	運用体制	独自ヘルプデスクを構築する、システム所管で対応するといった運用体制方針	—
	ドキュメント管理方式	運用・保守時のドキュメントを管理するツールやルール	—
	ライブラリ管理方式	運用・保守時のライブラリ (プログラムのバージョン) を管理するツールやルール	—
	業務アカウント管理方式	業務アカウントの登録/削除の管理ルール	—
	特権アカウント管理方式	基盤アカウントの登録/削除の管理ルール	—
	構成管理方式	システムの構成管理するツールやルール	—
	障害管理方式	運用・保守時に障害が発生した際の対応ルール	現行システムにおいて発生した障害とその業務影響を踏まえて、対応ルールを見直します。
	稼働実績管理方式	システムの稼働実績の取得ツールやルール	—
	スケジュール変更方式	提供サービスのスケジュールを延長等する際のルール	—
	媒体管理方式	データ保管媒体の管理ルール	—
	基盤運用スケジュール	基盤の運用スケジュール	—

③ 仮想化移行における留意事項

仮想環境への移行では、次図【P2V の例】に示すように構成が大きく変更されます。これまで物理サーバーで処理していたアプリケーションを仮想サーバ上で実行するように基盤面の設計を大幅に変更しますが、次の点に留意して受注業者と設計内容を確認してください。

◆ 物理サーバ（仮想基盤サーバ）

- ・ ハイパーバイザーの動作が保証されていること。
- ・ 物理サーバ障害に備え、冗長構成にすること。
- ・ 物理サーバのリソースは、同一サーバ上で稼働する次の内容を考慮すること。
 - (a) 仮想サーバ（クラスタ含む）分のリソース
 - (b) 仮想サーバ障害時のバックアップサーバ分のリソース
 - (c) ライブマイグレーションによるサーバ移転分のリソース
 - (d) 仮想環境を実現するハイパーバイザー等のオーバーヘッド分のリソース

◆ 仮想サーバ

- ・ 物理サーバの障害時を考慮した仮想サーバ配置を行うこと（冗長化したバックアップサーバは、同一物理サーバ上に配置しないなど）。
- ・ システム保守性を考慮し、物理サーバ上には運用条件の近い仮想サーバを配置すること。

◆ 共有ディスク

- ・ 冗長化対策を行うために、仮想サーバ等のデータを格納するディスクは、仮想環境を構成する物理サーバ間で共有できるディスクにすること。
- ・ 共有ディスク障害に備え、冗長構成にすること。

◆ バックアップ

- ・ 仮想サーバにおいては、現行システム（物理サーバ）のバックアップ取得方法では能力面での課題が発生するため、仮想サーバに適したバックアップ方式（専用ツールの利用など）を検討すること。
- ・ バックアップ処理時間を集中させると予定時間内に完了しない可能性が高くなるため、処理を適度に分散させるなど、適切な設計を行うこと。

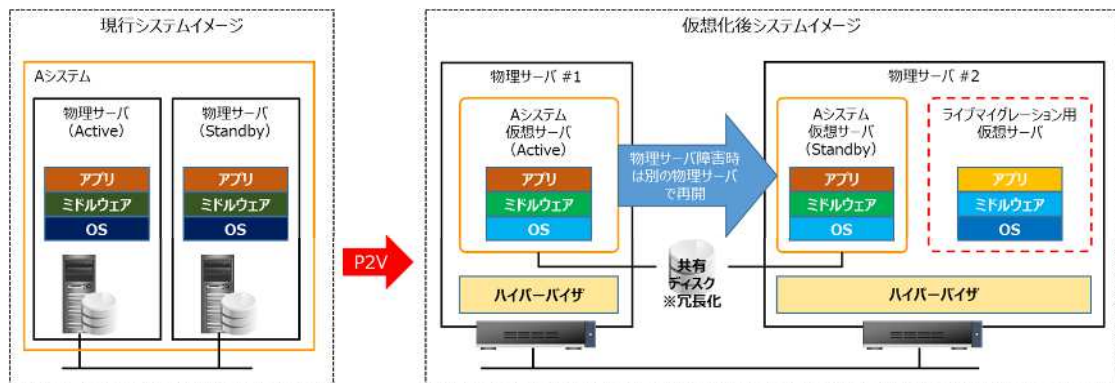
◆ 運用管理・監視

- ・ 物理サーバと仮想サーバの関係性が容易に識別・管理・監視できる運用管理ソフトウェアの導入を検討すること。

◆ ライセンス

- ・ CPU コア数に応じたライセンス（物理サーバ単位・仮想サーバ単位）など必要数に留意すること。

[P2V の例]



※本事例は、Aシステムを構成する複数の物理サーバをP2Vにより仮想化する例です。実際には、仮想化後の物理サーバ上にはAシステムを構成していた他のサーバを同様に仮想化して配置します。

なお、基本設計書のレビューは基本設計書の改訂を行った記載目次（基本設計項目参照）及び関連する記載事項単位で行います。

4. 「(4)詳細設計」

① 概要

本工程は、基本設計で決定した内容に基づき、次期ハードウェア上で非互換対策を講じたシステムを実現するための具体的な方法を決定する工程です。

機種更新においては、物理サーバの仮想化に伴うハードウェア構成の変更や非互換対策に伴って変更になる運用部分について重点的に検討します。

② 機種更新における留意事項

基本設計において見直した内容を踏まえ、開発ガイドラインに定める詳細設計項目を見直して詳細設計書を改訂します。特にハードウェア更新に関連する設計内容については、次の留意事項を踏まえて重点的に見直します。

【オンライン機能詳細設計、バッチ機能詳細設計】

非互換対策対象のプログラムやジョブ制御文を抽出します。非互換対策が命令やジョブ制御文の単純置換で対応できない場合、オンライン・バッチ機能の詳細設計を見直します。

【システム運用管理詳細設計】

DBMS 導入バージョン変更に伴い、統計情報に基づく SQL 実行計画が変更になる場合があるため、新たな統計情報を基にした最適な SQL 実行計画が適用されるよう、特に重点的に設計を見直します。

また、非互換対策及び仮想化に伴う画面表示・操作方法などの変更や 1 台の物理サーバ上に複数のサーバが集約されることによるバックアップ処理の輻輳を回避するためのバックアップスケジュールの変更など、運用管理設計を見直し、運用マニュアルに反映します。

【ハードウェア設計、ネットワーク設計】

仮想サーバを利用するシステムでは、仮想サーバ（クラスタ含む）分のリソースに加え、仮想サーバ障害時のバックアップサーバ分のリソース、ライブマイグレーションによるサーバ移転分のリソース、仮想環境を実現するハイパーバイザー等のオーバーヘッド分のリソースを考慮して設計を行います。

【(追加) コンバージョンツール設計】

コンバージョンツールを作成する場合は、完全な自動化ツールとするのか、一部分は手作業として行うのか、コンバージョン対象の規模や非互換対策内容、ツール作成に必要な工数を見極めて設計します。

【認証設計】

シングルサインオンに切り替える場合は、アプリケーション内で完結していた認証処理を、Active Directory を利用した認証方式やエージェントを利用した認証方式など採用する認証方式に合わせて、必要な情報を連携して認証するよう設計し直します。

※認可（権限）設定の引継ぎについては、4 章の表「(6)移行設計」を参照。

なお、詳細設計書レビューは詳細設計書の改訂を行った記載目次（詳細設計項目参照、ただし、システム運用管理詳細設計については、記載事項単位）及び関連する記載事項単位で行います。

5. 「(20)プログラム設計・製造」

① 概要

本工程は、詳細設計で決定した内容に基づき、プログラムやジョブ制御文を設計・製造する工程です。

機種更新においては、特に非互換対策を具体的にプログラムやジョブ制御文に反映するための設計・製造を行います。

② 機種更新における留意事項

詳細設計において見直した内容を踏まえ、非互換調査工程で作成した非互換・非推奨項目一覧表を基に、コンバージョンツールを利用する場合は有効に活用し、構文の書き換えや代替 API・関数への移行、廃止された API・関数を代替する機能作成などを行います。

6. 「(21)単体テスト」

① 概要

本工程は、機種更新においても全てのプログラムとジョブ制御文が仕様どおりに正しく動作するか、プログラム、ジョブ制御文単位に動作確認、検証を行う工程です。

② 機種更新における留意事項

機種更新においても全てのプログラムとジョブ制御文を対象にテストを行い、網羅性を確保したうえで、主に正常系の品質を確認するとともに、非互換対策として修正したプログラムを重点的にプロジェクト管理ガイドラインに定める品質基準を満たすよう単体テストを行います。

なお、機種更新では、現新比較テストをすることができれば、高い品質確保につながるため、受注業者と現新比較テストの実施について調整を図ります。

7. 「(24)総合テスト」

① 概要

本工程は、次期システムが現行システムとの互換性を確保できているか、非機能要件を満たしているかを確認するための工程です。

機種更新においては、特にハードウェア性能等の非機能について重点的に確認する必要があります。また、機能（業務）面のテストとしては現行システムの処理結果と同一の結果を得られるかを確認するとともに、権限が正しく設定されているかを確認します。

② 機種更新における留意事項

【非機能面のテスト】

非機能要件で定義した内容を中心に、移行テストで投入したマスタ/データを使用して、ハードウェアを更新することに伴い次表：総合テスト観点の性能・耐障害性を重点的にテストを行います。

総合テスト観点

分類	テスト観点
性能	画面機能、帳票出力機能について、要件どおりの性能値（レスポンス 3 秒など）となっているか。
	画面機能、帳票出力機能について、高負荷（ピーク時）の状態においても、要件どおりの性能（スループット 20 件/秒など）を担保できているか。
	高負荷時のマシン性能の限界値の確認。
	バッチ機能について、要件どおりの性能値（処理時間 30 分以内など）となっているか。

分類	テスト観点
	オンライン高負荷時やバッチ処理実行時のリソース（CPU、メモリなど）使用状況が想定内か。
耐障害性	データの重要性分類に応じたバックアップ取得が設計どおりに動作しているか。
	OS、ミドルウェア、業務 AP、マスタ/データ、ログのバックアップの取得、リカバリが可能か。
	障害発生時のリカバリレベルを踏まえて、復旧手順マニュアルどおりに作業を行えるか。また、復旧できることを確認しているか。
	リカバリポイントの DB、差分更新データから復旧できるか。また、復旧にかかる時間は想定どおりの時間か。
	バッチ処理について、リカバリポイントの中間ファイルから再実行可能か。

【機能（業務）面のテスト】

移行テストで投入したマスタ/データを使用してテスト環境を整備し、現新比較テストを行い、次期システムの処理結果が現行システムと同一の処理結果となるかを確認します。また、現行システムと同様に、次期システムにおいても業務権限が正しく設定されているかを確認します。

テストは非互換箇所を変更したプログラムで実現する機能を重点的に正常系主体で実施します。運用管理ソフトウェアや連携ソフトウェアなどのミドルウェア、パッケージソフトウェア（汎用）との連動についても確認します。

8. 「(27)運用テスト」

① 概要

本工程は、次期システムを利用して、日常業務及びシステム運用が滞りなく実施できるかをテストする工程です。

機種更新においては、運用に関わる画面表示や操作方法などの変更箇所について、運用マニュアルに沿って問題なく実施できるかを重点的に確認する必要があります。

② 機種更新における留意事項

運用テストにあたっては、次期システムにおいて変更された運用マニュアルに沿って運用できるかを業務担当（運用担当）・利用部門それぞれが確認します。

第4章 その他の工程

本章では、その他の工程において実施すべきことや考慮すべき点を補足事項として一覧にしました。補足事項を念頭に開発ガイドラインに沿ってプロジェクトを進めてください。

1. 各工程における補足

工程	補足事項
(1)開発規約策定	開発責任者は、標準化リーダー、受注業者と協力して、機種更新プロジェクトに適用する開発規約を策定します。次期システムで採用する OS やソフトウェア製品のバージョンアップに伴う非互換、非推奨となる事項を踏まえて既存の開発規約を改定し、当該機種更新プロジェクトの開発規約とします。
(2)要件定義	第3章参照
(3)基本設計	第3章参照
(4)詳細設計	第3章参照
(5)移行計画策定	非互換調査の結果を踏まえて、現行システムのデータベースやマスターファイル、ログファイル等の移行を前提に手順や方式を移行計画にまとめます。非互換対策のためのデータ形式やデータの追加、コード体系の変換などが必要ないか事前に検証した内容が計画に含まれます。 また、現在の OS やミドルウェアを変更せずに仮想サーバに移行する場合は、物理サーバをイメージファイル化し、仮想サーバ上に再現するツールの活用可否など効率的な移行計画を検討します。
(6)移行設計	非互換対策に伴うデータ形式やデータ追加、コード体系の変換などへの対応が必要な場合は、移行ツールや移行プログラムの仕様を整理し、対応します。 また、現行システムの権限（業務権限・フォルダアクセス権限など）の設定が次期システムに確実に引き継がれるようにします。合わせて、システム管理者等の権限の引き継ぎについても留意します。
(7)移行データ整備	DBMS やパッケージソフト（業務）のバージョンアップなどに伴い、既存データ以外のデータ追加が不可欠な場合は、データパンチや入力作業、他システムからの取り込み等を行い、次期システム移行のための追加データを整備します。 また、総合テストにおいて現新比較テストを行うために移行基準日を定めます。移行基準日以降に現行システムに入力した内容とその処理結果を

工程	補足事項
	データや帳票で保管し、次期システムに反映するなどの対応を図ります。
(8)移行テスト 1	—
(9)移行テスト 2	—
(10)基盤導入計画策定	機種更新では、現行・次期ハードウェアの並行設置が必要となるため、両ハードウェアを設置、稼働するためのファシリティを十分に考慮するとともに、低廉なコストで実現するよう計画を策定します。
(11)基盤調達	非機能要件に基づく設計変更に見合ったハードウェア仕様を検討し、導入することにしたハードウェアやソフトウェアなどのバージョンを調達仕様書にまとめます。また、ハードウェアの並行設置に係る作業内容や作業回数について仕様書に明示します。
(12)設備工事	現行・次期ハードウェアの並行設置が必要となるため、設置場所や電源等に不足がないように検討します。また、設置場所を新たに設ける場合は、セキュリティ対策や災害対策を考慮して設計します。
(13)基盤環境構築	—
(14)基盤環境テスト	—
(15)端末配置	現行・次期端末機の並行設置が必要となるため、複数回に分けた設置手順等を検討し、計画及び手順にまとめます。
(16)研修計画策定	システムの操作方法の変更点について、操作研修の日程や内容等を研修計画に定めます。
(17)業務研修	本書ではストレート移行を対象としており、業務手順の変更等はないものとしているため、業務研修は不要とします。
(18)操作研修	機種更新によりシステムの操作方法や画面遷移等が変更になる場合、改訂した操作マニュアルと研修資料を基に研修を行います。
(19)テスト計画策定	ストレート移行を前提とした機種更新では、現新比較テストをすることができれば、高い品質確保につながります。そのため、受注業者と現新比較テストの実施について調整を図り、テスト計画を策定します。 帳票の確認にあたっては、印字位置の確認等、目視による比較を行うことを想定してテスト計画に含めます。
(20)【- 参考 -】 プログラム設計・製造	第 3 章参照
(21)単体テスト	第 3 章参照
(22)結合テスト	機種更新では、非互換対策として変更したプログラムで実現する機能について重点的にテストを行います。網羅性を確保しつつ、正常系主体に確認します。
(23)システムセットアップ	—
(24)総合テスト	第 3 章参照
(25)本番切替計画策定	—

工程	補足事項
(26)移行リハーサル	移行データは現行システムからの移行基準日を定めて、次期システムへの移行リハーサルを行い、総合テスト工程で現新比較テストを行うためのデータが揃っているかを確認します。また、現行・次期ハードウェア設置場所間のデータ移動の手段・時間が適切かを確認します。
(27)運用テスト	第3章参照
(28)本番切替判定	—
(29)本番切替	—
(30)重点監視体制運用	—
(31)運用引継ぎ	運用設計を変更した点について、現行システムとの差異が分かるよう資料等を作成し、業務担当（運用担当）に引継ぎを行います。
(32)システム評価	要件定義で定めた仮想化の目的や性能向上・耐障害性等の非機能要件が実現されているかなどを評価します。

※「—」は不要な工程ではなく、他の工程と同様に開発ガイドラインに沿って適切に実施することを意味します。