
WebPerformer-NX

開発の流れ解説資料

2023年01月30日

キャノンITソリューションズ株式会社

本資料について

本資料はWebPerformer-NXにおけるシステム開発の流れの一例を解説する資料です。

「推奨する複数名でのシステム開発手順」についての解説と
「現機能において複数名で同一アプリケーションを開発する手順」を解説しています。

※後者は現機能においては運用で解決する手順を紹介しておりますが
今後の計画として、複数名で同一アプリケーションを効率よく開発する機能実装を計画しています。

◆ 前提

- 本資料における登場人物の役割
 - ① 管理者：開発／実行の各環境定義情報、及びDBの管理を担当 ※開発者の兼務でも可
 - ② 開発者：開発を担当
- 開発における標準化を事前に定めることをお勧めします。
 - 命名規則、構成など。特にUIやDBの命名は定義名が被らないようルール化ください。
- DBの定義は設計済で、管理者から定義情報を共有する前提で記載しています。
 - 開発者によるDBの作成／更新については、管理者がまとめて全開発者に共有する前提となります
- 本資料の画面イメージについては、変更となる可能性があります。
- 無償版ではデプロイできるアプリケーション数は1つとなります。

WebPerformer-NXの利用による開発工程の効率化

WebPerformer-NXの利用による開発工程の効率化

1. UI開発による画面デザインのプロトタイピング

- ①設計・開発体制の用意
- ②プロトタイプを簡単作成
- ③プロトタイプの画面設計書へのイメージ利用
- ④プロトタイプのレビュー対象者への展開が容易

3. テストを効率よく進めることが可能

- ①修正が容易
- ②素早い修正版の再デプロイが可能

2. システム開発を効率化

- ①アプリケーション開発を効率化
- ②業務シーンに合った部品群を用意
- ③複数名開発
- ④素早い修正版の再デプロイが可能
- ⑤豊富な稼働環境

要件定義

外部設計

内部設計

開発

総合テスト

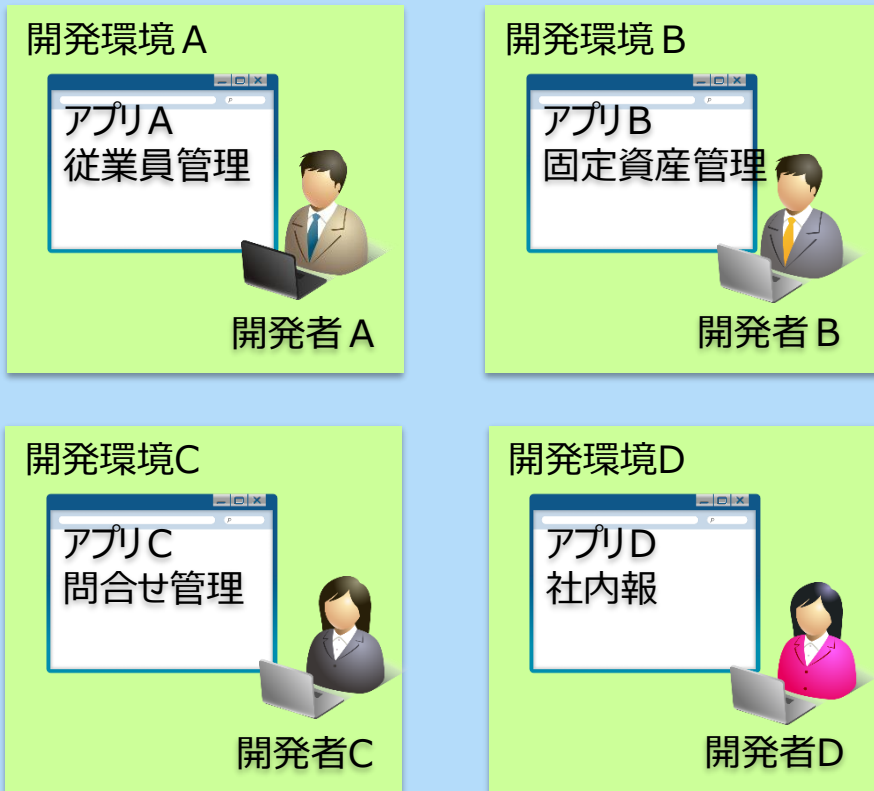
システムテスト

推奨する複数名でのシステム開発手順

1. 推奨：開発体制のイメージ

アプリケーションは開発者ごとに実装単位を分けて、開発することを推奨
それぞれで担当するサブシステムを完成させ、最後に同一実行環境に統合します

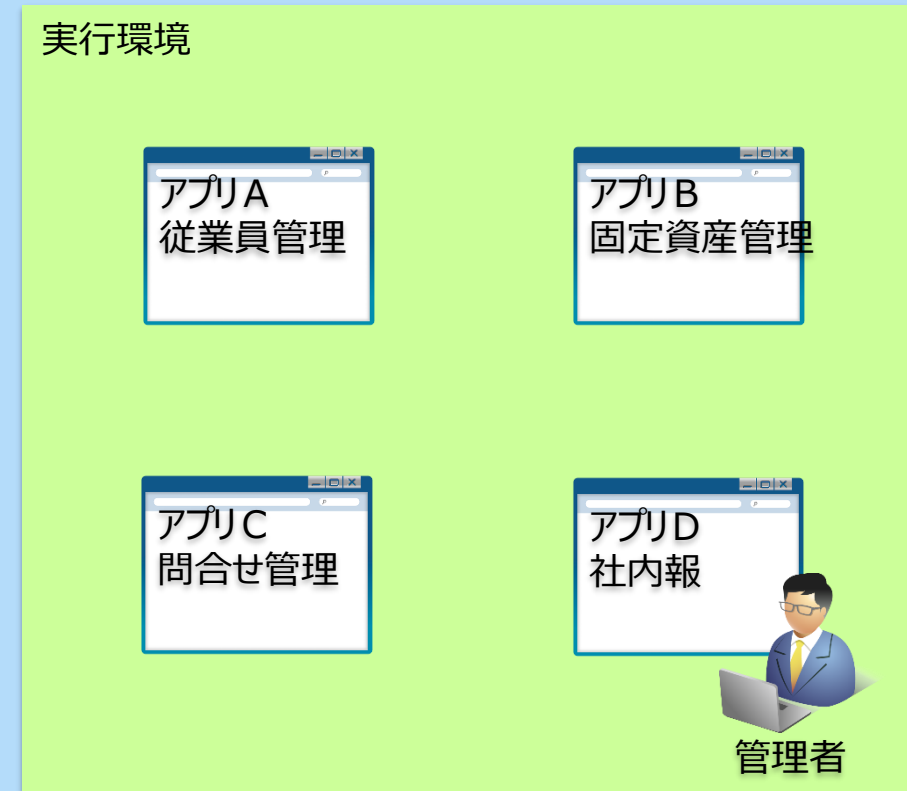
例：統合社内業務システム（開発時）



①デプロイ



例：統合販売システム（実行環境）



2. 推奨：役割毎、運用イメージ

WebPerformer-NXを利用した開発における、定義情報やそのデプロイに関する運用イメージ

管理者：準備フェーズ

1. アカウント登録

2. 有償の申込をする

有償環境へのアプリケーションのデプロイを開発者も実施する場合
「管理者分のアカウント」と「開発者分のアカウント」を添えて申し込む

・管理者による、有償環境へのアプリケーションのデプロイが可能になる



開発者：準備フェーズ

1. アカウント登録

2. 作成されたアカウントの情報を管理者に連絡する

・開発者による、有償環境へのアプリケーションのデプロイが可能になる



管理者：開発フェーズ

3. DBの変更情報などあれば周知と、インポートデータの配布

開発者：開発フェーズ

3. 開発者ごとに、アプリケーションをそれぞれを作成する

4. DBへテーブル構造とテーブルデータをインポートする

5. 各アプリケーションごとに、それぞれ担当者が、開発し定義を更新する

6. 完成したアプリケーションをデプロイする

→ 開発終了時、各アプリケーションを
有償の実行環境にそれぞれ担当者がデプロイする

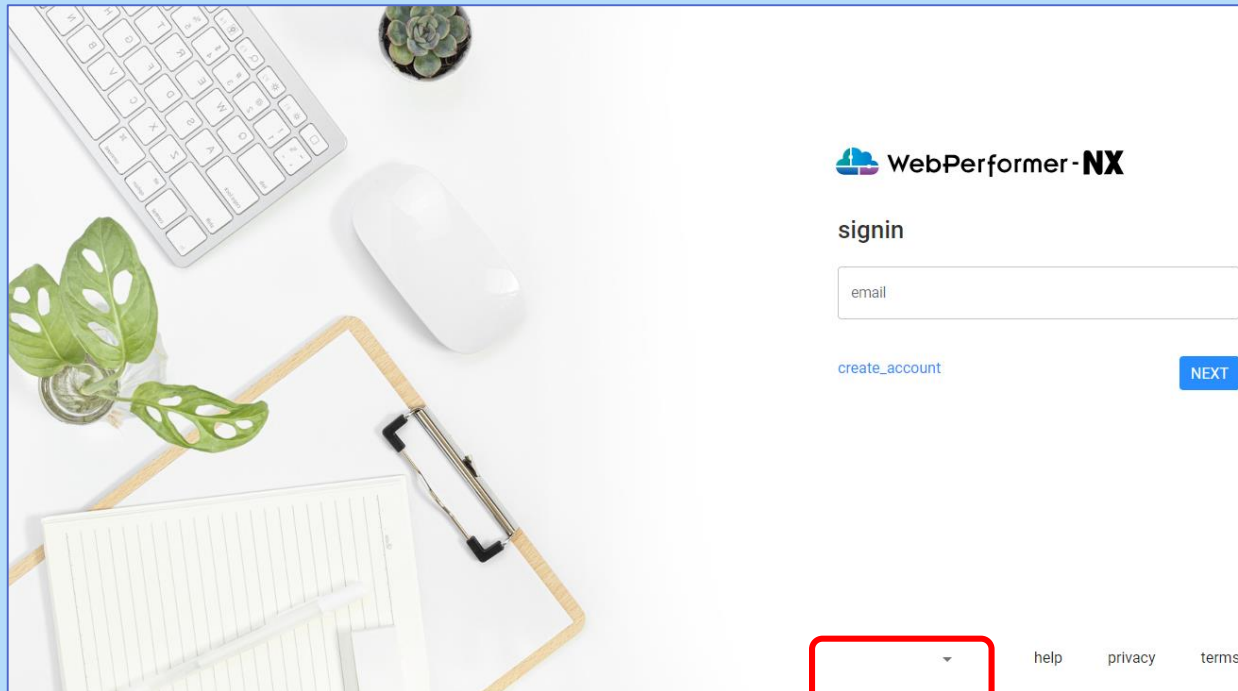
2. 推奨：役割毎、運用イメージ詳細

管理者：準備フェーズ

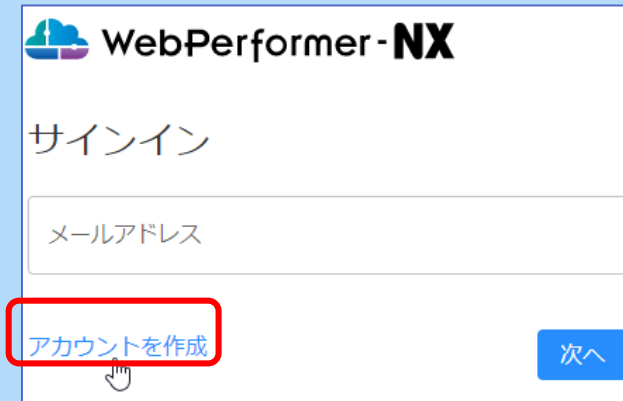
1. アカウント登録

■ URL <https://nx.webperformer.jp/> にアクセス

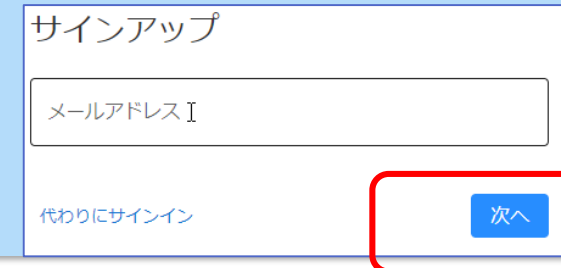
①「言語切り替え」で「日本語」を選択



②「アカウントを作成」を選択



③サインアップでOTPを受け取るメールアドレスを登録



2. 推奨：役割毎、運用イメージ詳細

管理者：準備フェーズ

1.アカウント登録

④アカウント情報を登録し、「利用規約とプライバシー」にチェックを入れると利用規約が表示されます。「閉じる」ボタンを押下します。「アカウントを作成する」を押下すると、登録したメールアドレスにOTP（ワンタイムパスワード）が送信されます。

 WebPerformer-NX

🖱️

登録したメールアドレス

アカウント作成プロセスを完了するために、次の詳細についてサポートしてください

フルネーム

パスワードの作成
必須最小8文字、上1文字、下1文字、数字1文字、記号1

☒ Business ☐ Individual

Company Name

Department (Optional)

☐ 利用規約 とプライバシー ポリシーに同意します。

戻る

アカウントを作成する

[illegible]

⑤メールで送られてきたOTPを入力するとアカウントが作成されます。

送信元：no-reply@verificationemail.com

タイトル : Your verification code



WebPerformer-NX

メールによる確認

登録したメールアドレス
で送信した6桁のOTP
を入力してください

検証コード

[OTPを再送する](#)

確認

2. 推奨：役割毎、運用イメージ詳細

管理者：準備フェーズ

1. アカウント登録

⑥サインインします。

再度URL <https://nx.webperformer.jp/> にアクセスし、
・メールアドレス と 登録したパスワード で「サインイン」します。



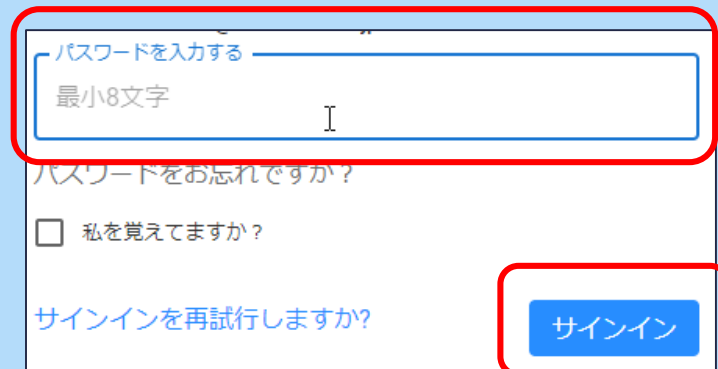
WebPerformer-NX

サインイン

メールアドレス

アカウントを作成

次へ



パスワードを入力する

最小8文字

パスワードをお忘れですか？

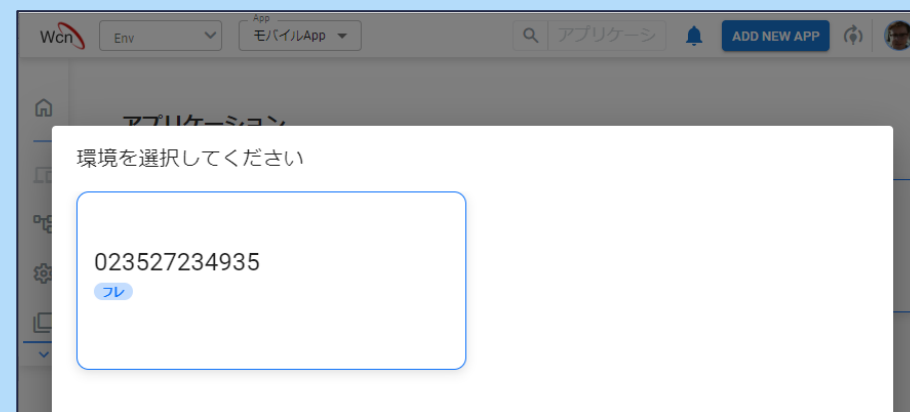
☐ 私を覚えてますか？

サインインを再試行しますか？

サインイン

⑦環境を選択します。

※アカウント毎に番号が異なります

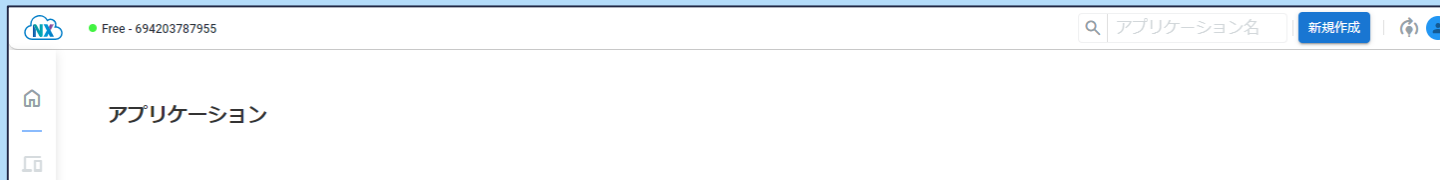


環境を選択してください

023527234935

フレ

⑧アプリケーション選択画面が表示され、準備完了です。



Free - 694203787955

アプリケーション名

新規作成

アプリケーション

2. 推奨：役割毎、運用イメージ詳細

管理者：準備フェーズ

2.有償の申込をする

※実行環境の共有（実行環境へのアプリケーションのデプロイを管理者だけでなく、開発者も実施する場合に利用）は有償版のみ可能です

※無償環境のみの利用の場合、こちらの作業は不要です。それぞれで作成したアカウントを使った開発は既に可能となっています

①有償版の申込書を記入のうえ、申込みます

・申込書の実行環境の欄に、共有する開発者の情報を添えてお申込みください

※有償版を既にご利用中であれば、サービスのメニューからお申込み可能です

お申し込みは弊社にて受付られ、お申し込みの確認をさせて頂いた後
弊社作業後に利用が可能となります

Share Environment

Invite People

Remarks

SEND

Environment Details

- Monitor Environment
Usages, Database, Applications
- Monitor Logs
Access & Event Logs
- Domain, Security & Sharing
IP Filtering & Sharing
- Password & Email Settings
Verify Email & Password Policy

カスタムドメインとセキュリティ

ドメイン 値

デフォルト	fedax.webperformer.jp
-------	-----------------------

[カスタムドメイン](#)

許可するIPリスト

注意) ここで設定するルールは実行環境に含まれるすべてのアプリケーションに影響します。

フィルタ	VALUE
▼	Japan
▼	192.168.0.1/32
▼	684D:1111:222:3333:4444:5555:6:77

[+ルール追加](#)

実行環境の共有先

注意) 設定したユーザはアプリケーションデプロイ先の実行環境にアクセスできるようになりますが、アプリケーションマネージャ、ファイルマネージャ、モニタリングのリソースセットです。

NAME	EMAIL ID
+ Add Users	

2. 推奨：役割毎、運用イメージ詳細

管理者：開発フェーズ

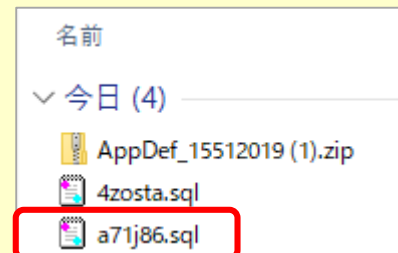
3.DBの変更情報などあれば周知と、インポートデータの配布

①データベース定義の共有方法（エクスポート）

- ・「DB定義とデータ」、「DB定義のみ」、「データのみ」を選択のうえエクスポートを実行し、対象ファイルの出力を確認します。



エクスポートファイルが出力



2. 推奨：役割毎、運用イメージ詳細

開発者：準備フェーズ

1. アカウント登録

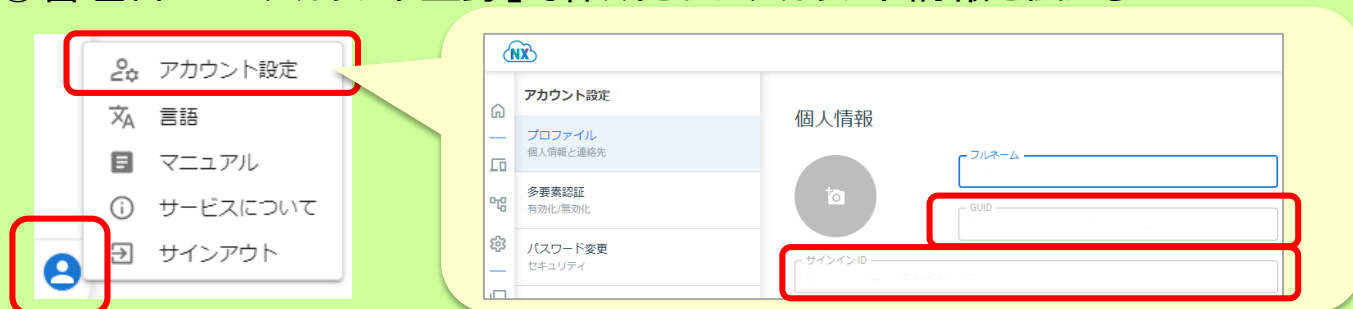
①推奨：「管理者：準備フェーズ」の「1.アカウント登録」と同じ手順となります

開発者：準備フェーズ

2. 作成されたアカウントの情報を管理者に連絡する

※実行環境の共有（実行環境へのアプリケーションのデプロイを管理者だけでなく、開発者も実施する場合に利用）は有償版のみ可能です
※無償環境のみの利用の場合、こちらの作業は不要です。それぞれで作成したアカウントを使った開発は既に可能となっています

①管理者に「1.アカウント登録」で作成されたアカウント情報を伝える

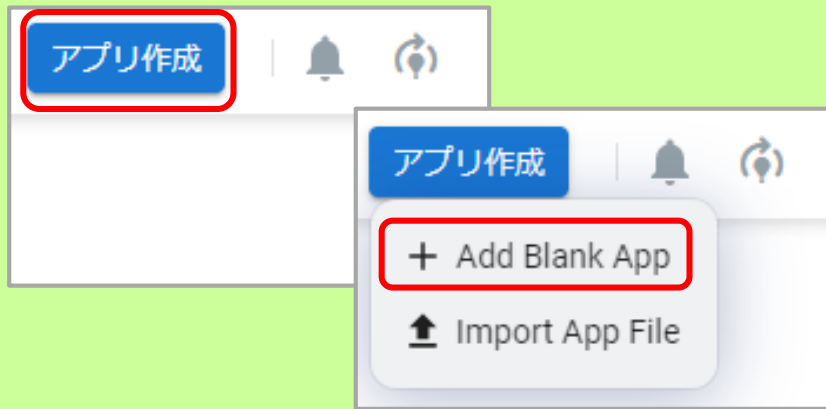


2. 推奨：役割毎、運用イメージ詳細

開発者：開発フェーズ

3. 開発者ごとに、アプリケーションをそれぞれ作成する

①アプリケーションの「新規作成」ボタンを押下し
「Add Blank App」を選択



②タイプを選択し、IDとLabelを入力し、「ADD APPボタン」を押下

A screenshot of a form titled 'アプリ作成' (Create App). It features two tabs: 'DESKTOP' (selected and highlighted with a red box) and 'MOBILE'. Below the tabs, there are two input fields: 'ID' containing 'MASTER_MAINTENANCE' and 'Label' containing '商品メンテナンス' (Product Maintenance). Both input fields are highlighted with a red box. At the bottom right of the form, there are two buttons: 'キャンセル' (Cancel) and 'ADD APP' (highlighted with a red box).

③UIエディタが白紙の状態が開かれ、アプリケーション作成完了



2. 推奨：役割毎、運用イメージ詳細

開発者：開発フェーズ

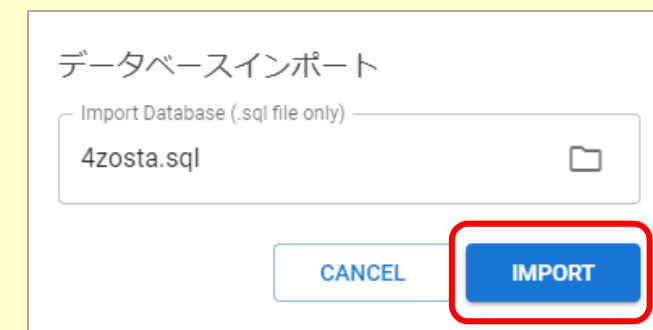
4.DBヘテーブル構造とテーブルデータをインポートする

①データベース定義の共有方法（エクスポート）

- ・「DB定義とデータ」、「DB定義のみ」、「データのみ」を選択のうえエクスポートを実行し、対象ファイルの出力を確認します。



インポート対象ファイルを選択し、インポートを実行



2. 推奨：役割毎、運用イメージ詳細

開発者：開発フェーズ

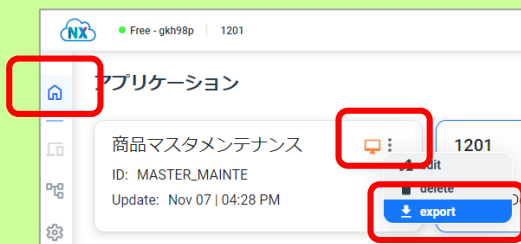
5.各アプリケーションごとに、それぞれ担当者が、開発し定義を更新する

①WebPerformer-NXによるアプリケーション開発を実施

開発者：開発フェーズ

6.完成したアプリケーションをデプロイする

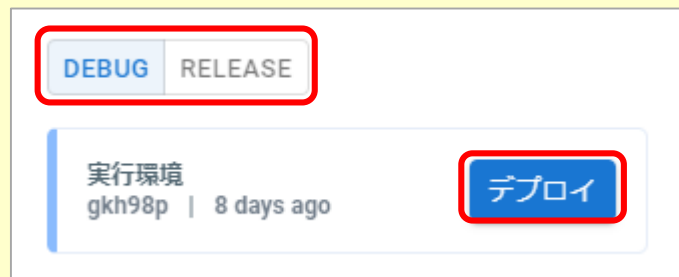
①アプリケーション情報をエクスポートして、マスターの定義情報として管理する



②開発終了時、各アプリケーションを有償の実行環境にそれぞれ担当者がデプロイする



デプロイ方法を選択肢し、デプロイを実行



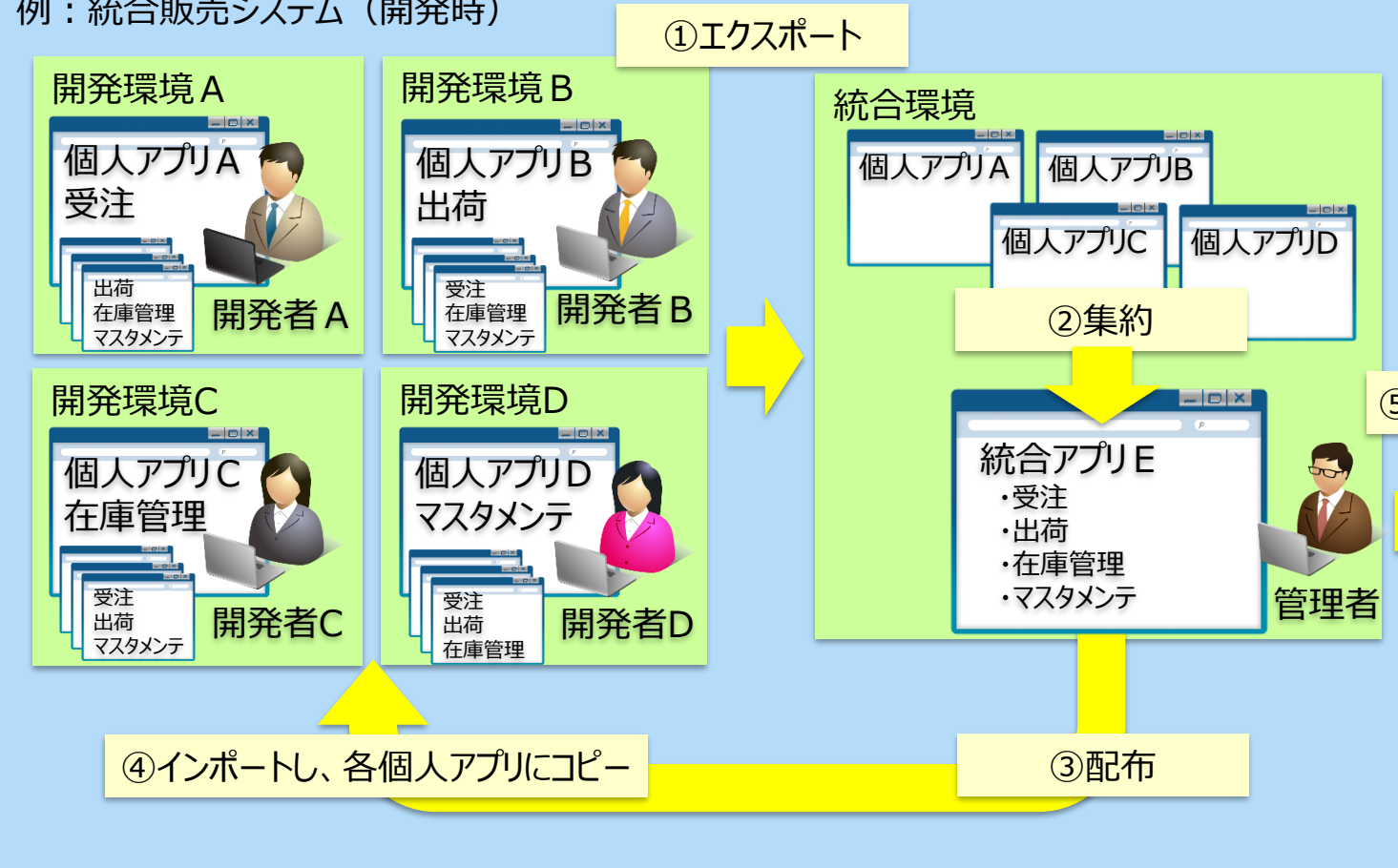
※デプロイ時に、アプリを停止する機能もございます。（デプロイ前にアプリを停止し、アクセスさせないようにしたい場合に利用）

複数名で同一アプリケーションを開発する手順

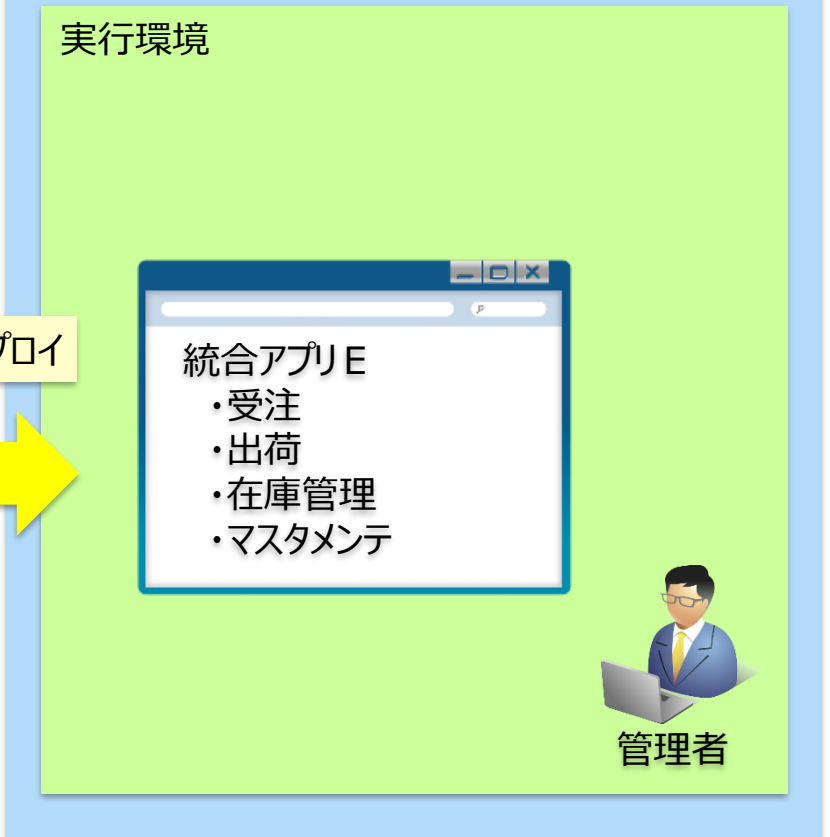
3. 複数名開発：開発体制のイメージ

アプリケーションは開発者ごとに実装単位を分けて、開発することを推奨
それぞれで担当するサブシステムを完成させ、最後に同一実行環境に統合します

例：統合販売システム（開発時）



例：統合販売システム（実行環境）



4. 複数名開発：役割毎、運用イメージ

WebPerformer-NXを利用した開発における、定義情報やそのデプロイに関する運用イメージ

管理者のやること：準備フェーズ

1. アカウント登録

2. 有償の申込をする

有償環境へのアプリケーションのデプロイを開発者も実施する場合
「管理者分のアカウント」と「開発者分のアカウント」を添えて申し込む

・管理者による、有償環境へのアプリケーションのデプロイが可能になる



開発者のやること：準備フェーズ

1. アカウント登録

2. 作成されたアカウントの情報を管理者に連絡する

・開発者による、有償環境へのアプリケーションのデプロイが可能になる



管理者のやること：開発フェーズ

3. 統合アプリケーション「E」を作る

4. DBの変更情報などあれば周知と、インポートデータの配布

5. 各開発者が作成したエクスポートデータを全てインポートし、統合する
取り込んだアプリケーション「B、C・・・」の定義から、UIコピーや
部品単位で統合アプリケーション「E」にコピーし、最新化する

6. 統合アプリケーション「E」の定義情報をエクスポートし、各開発者者に配布

7. 完成した統合アプリケーション「E」をデプロイする

開発者のやること：開発フェーズ

3. 開発者ごとに、アプリケーションをそれぞれを作成する（「B、C・・・」）

4. DBへテーブル構造とテーブルデータをインポートする

5. 各アプリケーションごとに、それぞれ担当者が、開発し定義を更新する

6. 作業終了時に、開発したアプリケーションをエクスポートし管理者に共有する

7. 最新定義情報のインポート

→ 統合アプリケーション「E」の定義情報をインポートし「B、C・・・」環境を
最新化する

4. 複数名開発：役割毎、運用イメージ詳細

管理者：準備フェーズ

1.アカウント登録

※推奨：「管理者：準備フェーズ」の「1.アカウント登録」と同じ手順となります

管理者：準備フェーズ

2.有償の申込をする

※推奨：「管理者：準備フェーズ」の「2.有償の申込をする」と同じ手順となります

4. 複数名開発：役割毎、運用イメージ詳細

管理者：開発フェーズ

3.統合アプリケーション「E」を作る

ここで作成する統合アプリケーションとは、各開発者が開発した各アプリケーションを統合して管理するためのアプリケーションです。

※推奨：「開発者：開発フェーズ」の「3.開発者ごとに、アプリケーションをそれぞれ作成する」と同じ手順となります。

管理者：開発フェーズ

4.DBの変更情報などあれば周知と、インポートデータの配布

※推奨：「管理者：開発フェーズ」の「3.DBの変更情報などあれば周知と、インポートデータの配布」と同じ手順となります。

4. 複数名開発：役割毎、運用イメージ詳細

管理者：開発フェーズ

5.各開発者が作成したエクスポートデータを全てインポートし、統合する

①各開発者が作成した定義（アプリケーション）のエクスポートデータを全てインポートする

例：開発者Bが作成した「アプリケーションB」と、開発者Cが作成した「アプリケーションC」をインポートする

The screenshot illustrates the process of importing application files into the NX system. It is divided into two main parts by a large yellow arrow pointing downwards.

Top Part (Initial State): The main application list shows only "アプリケーションE" (ID: app_e, Update: Nov 07 | 04:28 PM). On the right, a dropdown menu is open, showing "アプリ作成" (Create App) and "Import App File" (highlighted with a red box).

Bottom Part (After Import): The main application list now includes three applications: "アプリケーションE" (ID: app_e, Update: Nov 07 | 04:28 PM), "アプリケーションB" (ID: app_b, Update: Dec 01 | 04:49 PM), and "アプリケーションC" (ID: app_c, Update: Dec 09 | 05:27 PM). The "Import App File" option remains highlighted in the dropdown menu.

Import App File Dialog (Right Side): A yellow callout box highlights the "Import App File" dialog. It contains the text "インポート対象ファイルを選択しインポートを実行" (Select the file to be imported and execute the import). The dialog has a text input field "Select File(zip file only) *" (highlighted with a red box) and two buttons: "CANCEL" and "CONTINUE" (highlighted with a red box). A yellow arrow points from the "CONTINUE" button to the next dialog.

Import App File Dialog (Bottom Right): This dialog shows the details for importing "アプリケーションB". It has a "タイプ" (Type) dropdown set to "DESKTOP". Below it, the "ID" field contains "app_b" and the "Label" field contains "アプリケーションB". At the bottom, there are two buttons: "キャンセル" (Cancel) and "ADD APP" (highlighted with a red box).

4. 複数名開発：役割毎、運用イメージ詳細

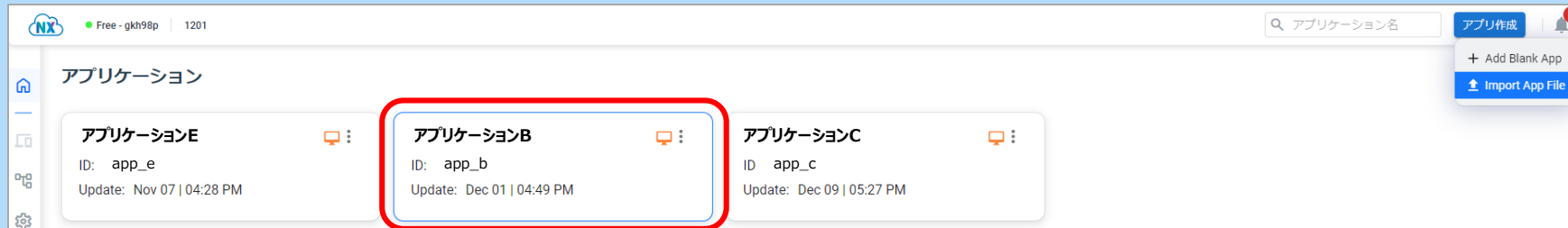
管理者：開発フェーズ

5.各開発者が作成したエクスポートデータを全てインポートし、統合する

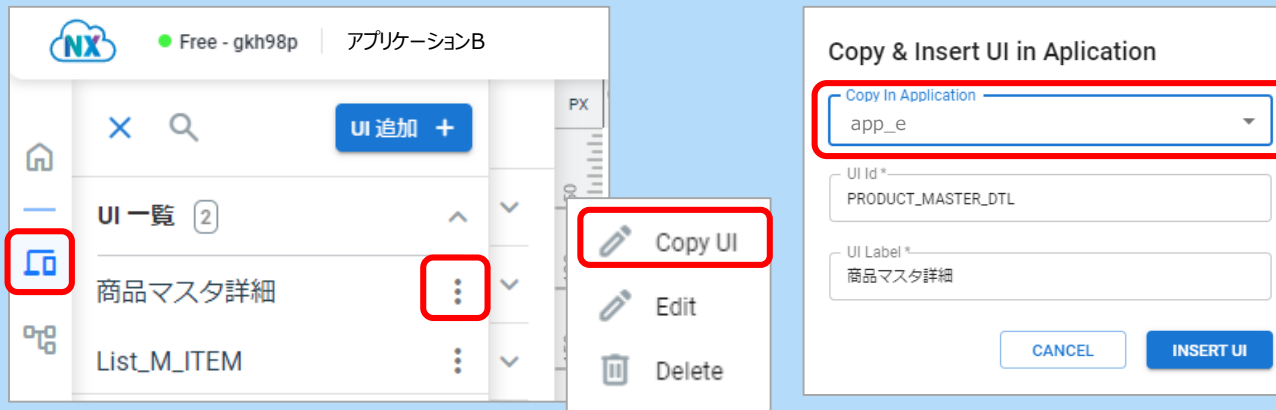
②各開発者が作成した定義（UI）を統合アプリケーション「E」にコピーUIする

例：開発者Bが作成したアプリケーション「B」のUI「商品マスタ詳細」を、統合アプリケーション「E」にコピーする

アプリケーション「B」を選択する



アプリケーション「B」から、開発者Bが開発したUI「商品マスタ詳細」でCopy UIを選択する。コピー先に統合アプリケーション「E」を指定する



※これを各アプリケーション毎
対象のUIのCopy UIを繰り返し
統合アプリケーション「E」のUI全てを最新化する

4. 複数名開発：役割毎、運用イメージ詳細

管理者：開発フェーズ

5.各開発者が作成したエクスポートデータを全てインポートし、統合する

③インポートした定義のDBは、定義を作成した環境のDBが記載されているため、統合したいDB名に更新する
UIから対象のUIを選択し、アクションボードを開き、アクションボードから対象アクションを開き、DB名を修正する

The screenshot displays the NX development environment interface. The top navigation bar shows the project name 'Free - gkh98p' and the current view '商品マスタメンテナンス / 商品マスタ詳細 / save_generated'. The left sidebar contains a list of UI components, with '商品マスタ詳細' highlighted. The main area shows the 'Action_Board' for the 'SQLSave' action. The 'SQL STATEMENT' section contains the following code:

```
1 INSERT INTO a71j86.M_ITEM (ITEM_ID, ITEM_NM, PRICE, STOCK) values(
```

A yellow callout bubble points to the database name 'a71j86.M' in the SQL statement, with the text 'DB名を自環境のDB名に修正する' (Update the DB name to the DB name of the self-environment).

On the right, the 'アクションボード' (Action Board) shows the following JavaScript code:

```
1 const params = {};  
2 params.ITEM_ID = $ui.ITEM_ID.value;  
3 params.ITEM_NM = $ui.ITEM_NM.value;  
4 params.PRICE = $ui.PRICE.value;  
5 params.STOCK = $ui.STOCK.value;  
6 const res= await SQLSave(params);  
7 const affectedRowsInsert=res.data.affectedRows;  
8 if(affectedRowsInsert>0){
```

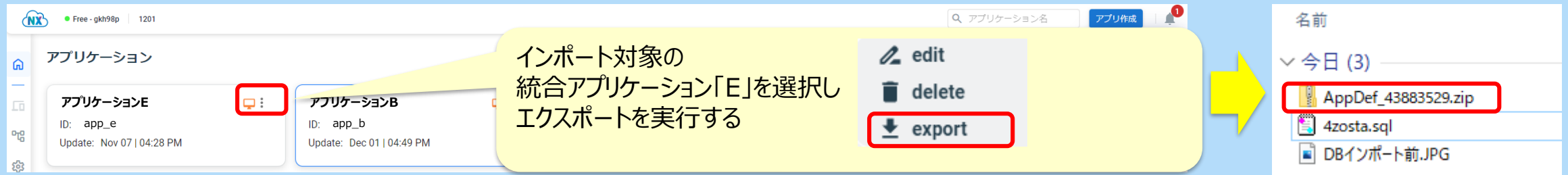
A second yellow callout bubble points to the 'a71j86' database name in the sidebar, with the text 'DB名の確認はデータベースを選択し確認する' (Check the DB name by selecting the database and confirming).

4. 複数名開発：役割毎、運用イメージ詳細

管理者：開発フェーズ

6.統合アプリケーション「E」の定義情報をエクスポートし、各開発者に配布

①統合アプリケーション「E」を選択し、「エクスポート」を選択する



②エクスポートしたファイルを、各開発者者に配布する

管理者：開発フェーズ

7.完成した統合アプリケーション「E」をデプロイする

※推奨：「開発者：開発フェーズ」の「6.完成したアプリケーションをデプロイする」と同じ手順となります。

4. 複数名開発：役割毎、運用イメージ詳細

開発者：準備フェーズ

1. アカウント登録

①推奨：「管理者：準備フェーズ」の「1. アカウント登録」と同じ手順となります

開発者：準備フェーズ

2. 作成されたアカウントの情報を管理者に連絡する

①推奨：「開発者：準備フェーズ」の「2. 作成されたアカウントの情報を管理者に連絡する」と同じ手順となります

4. 複数名開発：役割毎、運用イメージ詳細

開発者：開発フェーズ

3.開発者ごとに、アプリケーションをそれぞれを作成する（「B、C・・・」

※推奨：「開発者：開発フェーズ」の「3.開発者ごとに、アプリケーションをそれぞれ作成する」と同じ手順となります。

開発者：開発フェーズ

4.DBへテーブル構造とテーブルデータをインポートする

※推奨：「開発者：開発フェーズ」の「4.DBへテーブル構造とテーブルデータをインポートする」と同じ手順となります。

4. 複数名開発：役割毎、運用イメージ詳細

開発者：開発フェーズ

5.各アプリケーションごとに、それぞれ担当者が、開発し定義を更新する

①WebPerformer-NXによるアプリケーション開発を実施

開発者：開発フェーズ

6.作業終了時に、開発したアプリケーションをエクスポートし管理者に共有する

※アプリケーションのエクスポート手順は

複数名開発：「管理者：開発フェーズ」の

「6.統合アプリケーション「E」の定義情報をエクスポートし、各開発者に配布」と同じ手順となります。

4. 複数名開発：役割毎、運用イメージ詳細

開発者：開発フェーズ

7. 最新定義情報のインポート

※アプリケーションのインポート手順と、インポートしたアプリケーションからUI定義のコピー手順は
複数名開発：「管理者：開発フェーズ」の
「5. 各開発者が作成したエクスポートデータを全てインポートし、統合する」と同じ手順となります。

例：開発者 B は、最新の統合アプリケーション「E」をインポートのうえ、自己の開発アプリケーション「B」を最新化する

①統合アプリケーション「E」をインポートする

②インポートした統合アプリケーション「E」のUIをCopyUIで、アプリケーション「B」に全てコピーし、アプリケーション「B」を最新化する