

Windows 10、モバイルもOK！ 最新OSガイドラインに適合したUI/UX構築

第35回 エンバカデロ・デベロッパーキャンプ

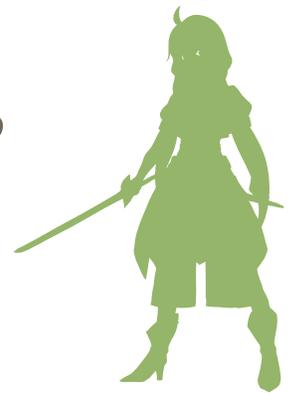
エンバカデロ・テクノロジーズ
セールスコンサルタント 毛利春幸



embarcadero®
DEVELOPER CAMP

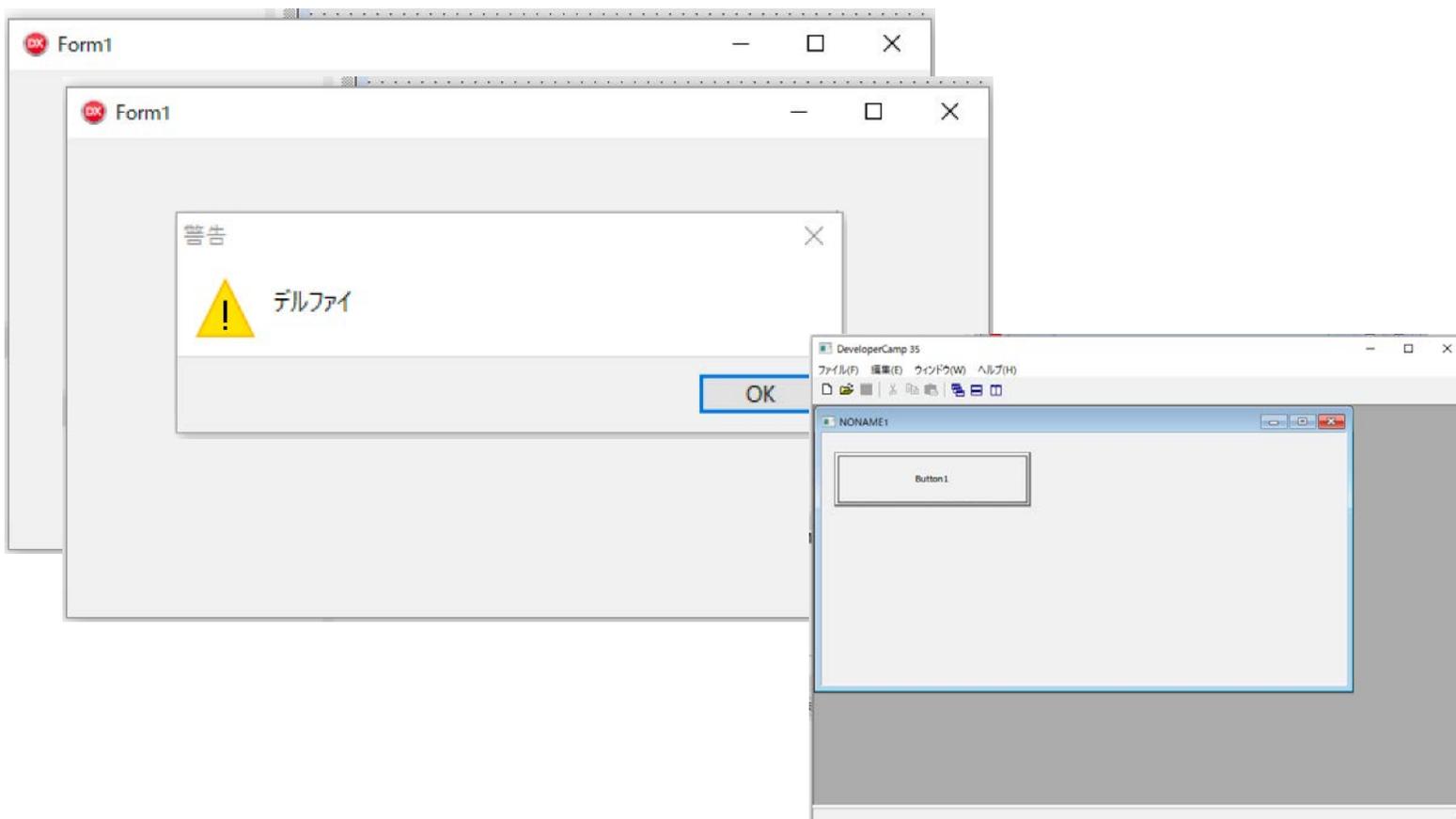
■はじめに

- iOS/Android レイアウトやWindows 10に関し大きくUI/UXのルールが異なります。いったいどう違うのか？
- さまざまなデバイスの要素・対話型などを例に大きく設計思想をシフトさせましょう



過去のさまざまなデバイス デザイン

- 複数ウィンドウやMDIですか？
- AndroidもiOSも共通レイアウトですか？



マルチデバイス デザイン

- Windows 10、iOS、Androidそれぞれデザインに関してのガイドラインが異なります
- すべてのOSで共通画面
 - 他のアプリと動きが違うので利用者がとまどいます

OSそれぞれの
デザインガイドラインがあります。



iOS Human Interface Guidelines

■ Clarity

- テキストはあらゆるサイズでみやすい・アイコンは明瞭・適切な装飾・機能性に鋭い焦点

■ Deference

- 流体の動きと鮮明で美しいインターフェイス、対話形式、多くのヒント
- ベゼル、グラデーション、およびドロップシャドウの使用は最小限

■ Depth

- 理解しやすい階層レイヤー
- コンテンツをナビゲートする際の奥行き

ざっくりでイメージしにくい



Android マテリアルデザイン

- Android は紙とインクからヒントを得たデザイン メタファーを使って、ユーザーが安心できる使用感
- Material is the metaphor
 - 紙とインクと光と影
- Bold, graphic, intentional
 - 利用者に没入した体験させる
- Motion provides meaning
 - 意味を連想させるモーション



Windows 10

- Windows 10になって動作環境が大きく広がりました...

- 高解像度モニタ
- タッチスクリーン
- キーボード着脱式PC
- タブレット

：

- UWP

- UI/UXに関するガイドライン



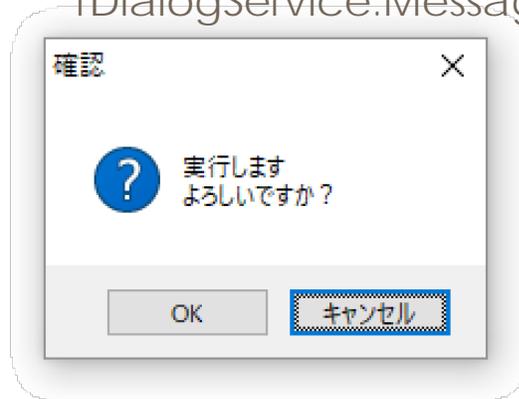
メッセージBOX

- デバイス単位で違いがよくわかる

VCL(Win10 スタイル)



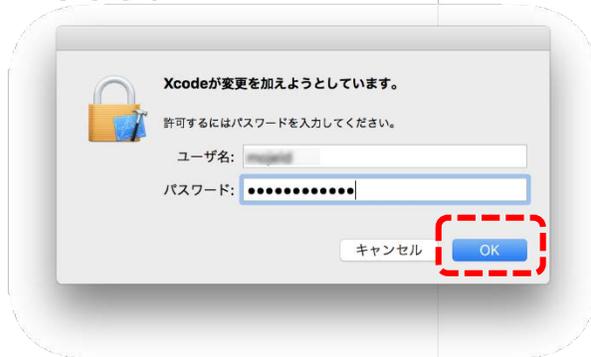
FireMonkey(Win)
TDialogService.MessageDialog()



Ext JS(Win)
Ext.Msg.confirm()



macOS



Ext JS(macOS)
Ext.Msg.confirm()



Android/iOS Alert

- メッセージBOX(Alert)

Android
AlertDialogService.MessageDialog()



iPhone
UIAlertController



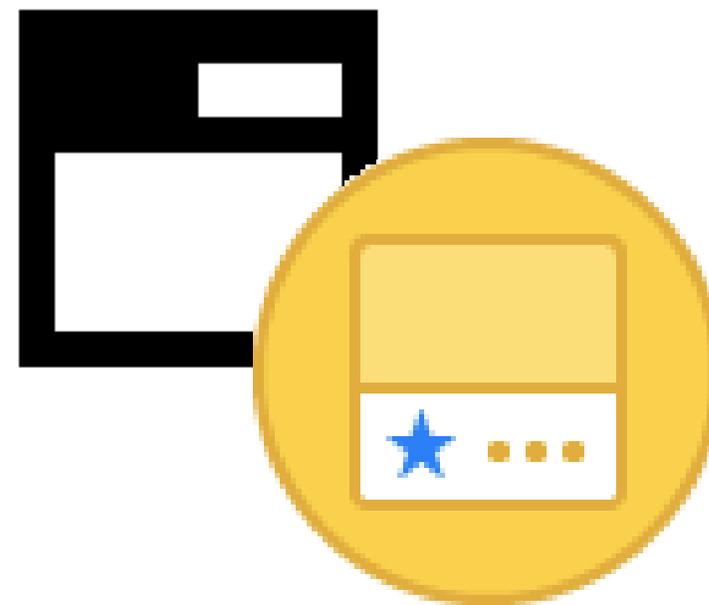
iPhone
Action Sheets



iOSはネガティブな情報は左に配置する

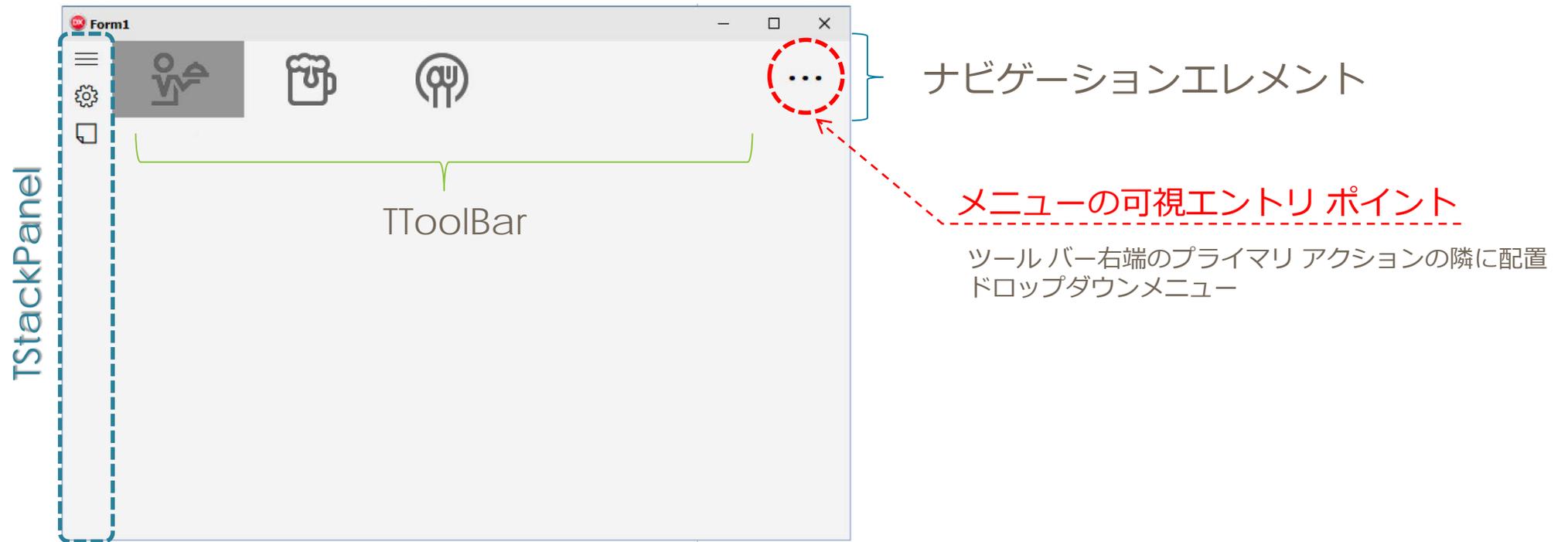
タブレイアウトの違い

- Windowsではナビゲーションエレメント
- Androidではタブレイアウト
- iOSタブバー



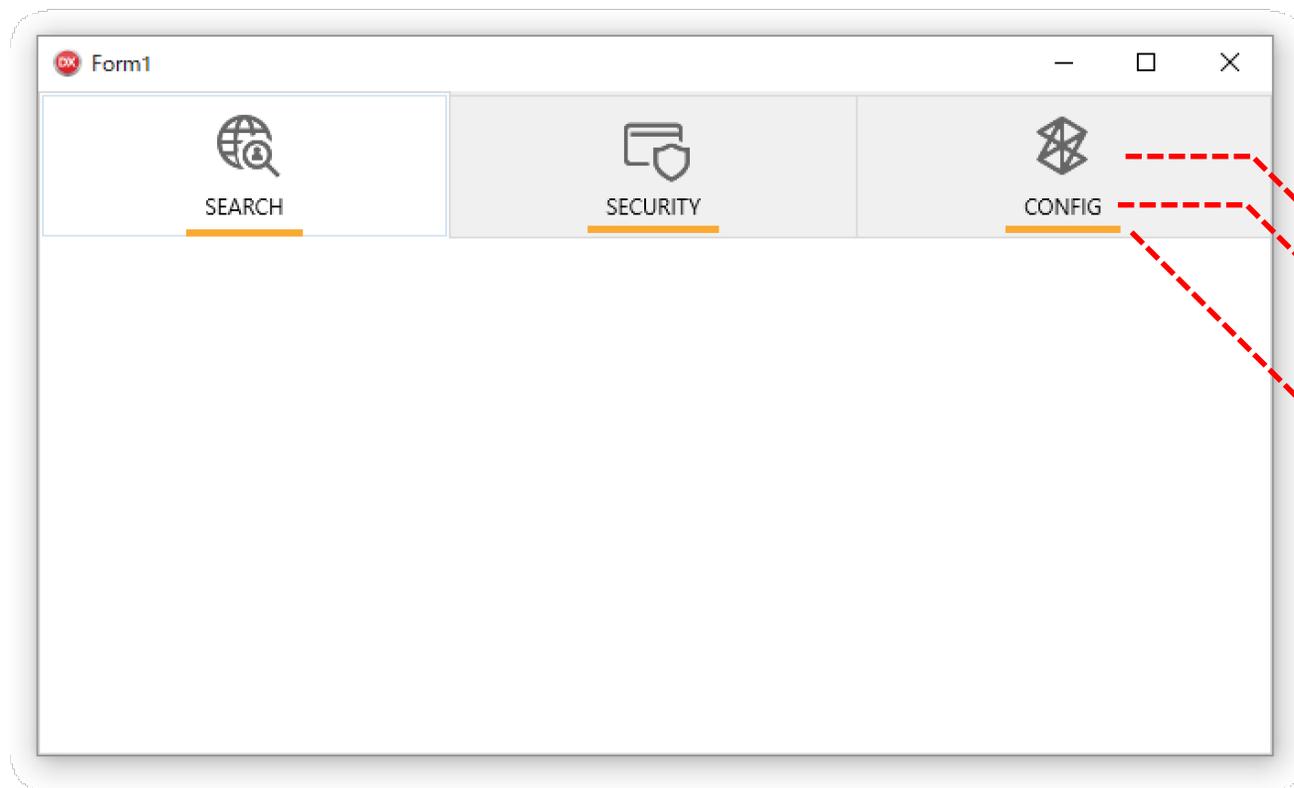
VCL Windows 10 TToolBarを使ったナビゲーションエレメント

- TToolBarを使いナビゲーションエレメントを再現

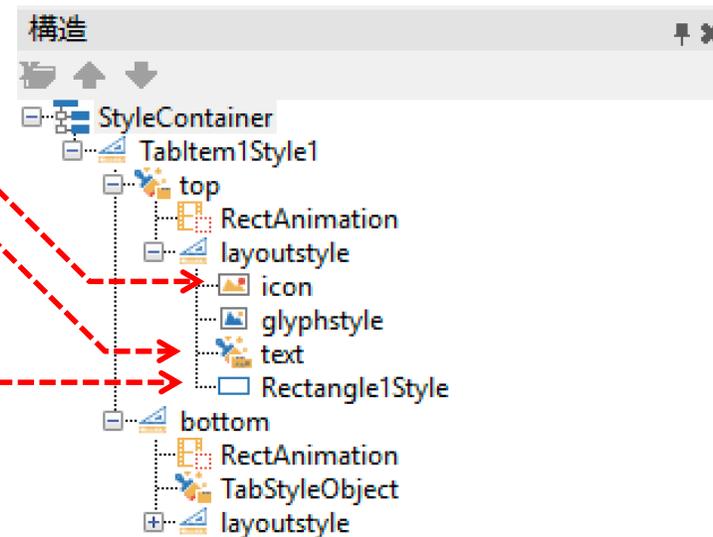


FMX Windows 10 ナビゲーションエレメント

- TTabControl を使ったWindows 10ナビゲーションエレメント

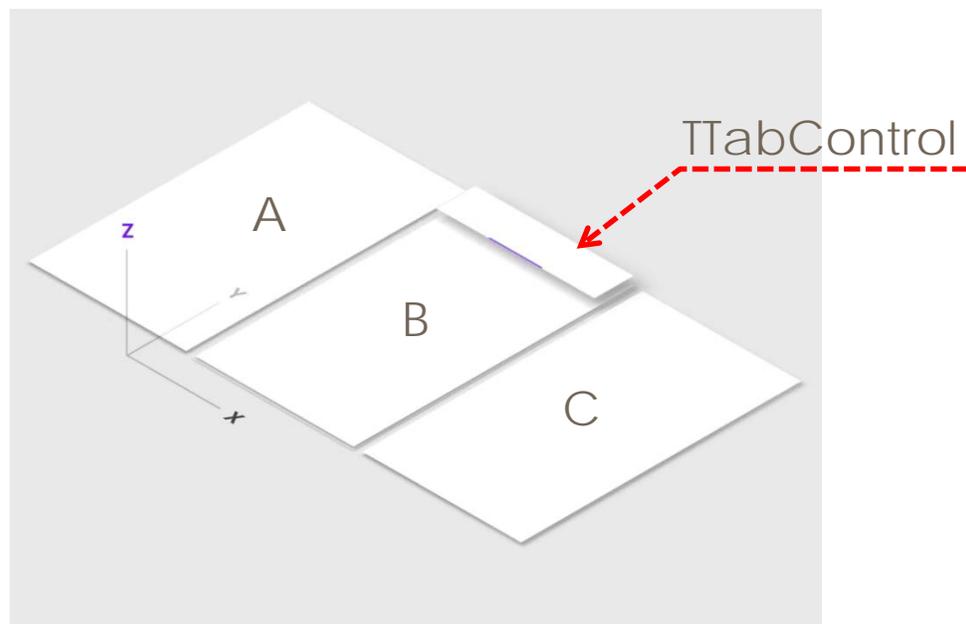


FireMonkeyの場合
スタイルのカスタマイズで対応できます。



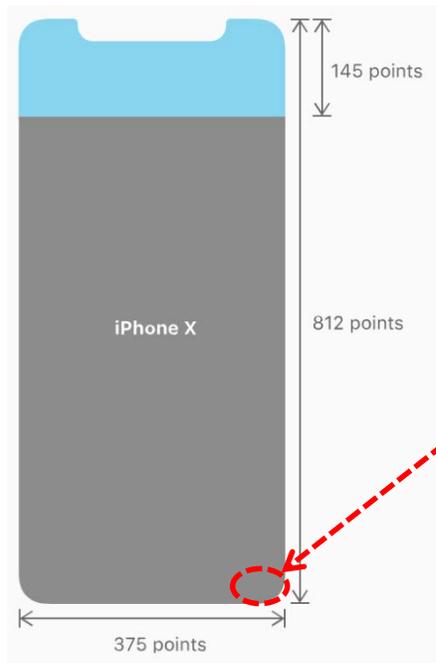
Android タブレイアウト

- タブレイアウト位置は上部



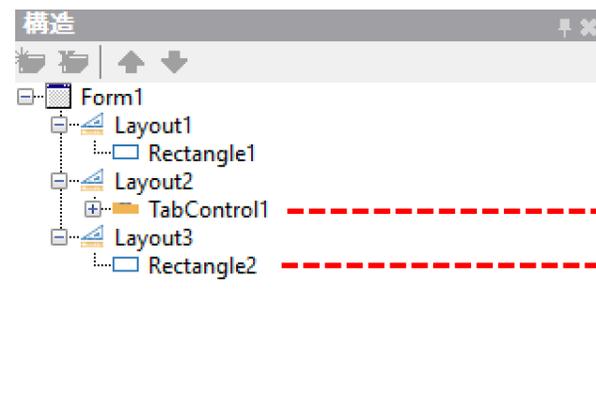
iOS タブバー

■ iOS TTabControlを利用



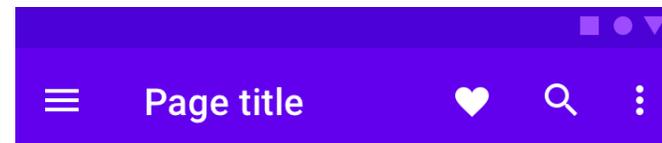
iPhone Xは812pt × 375ptですが、
Delphi/C++Builderで利用できるのは縦768です。
上部44ptはStatusバー

32pt浮かす



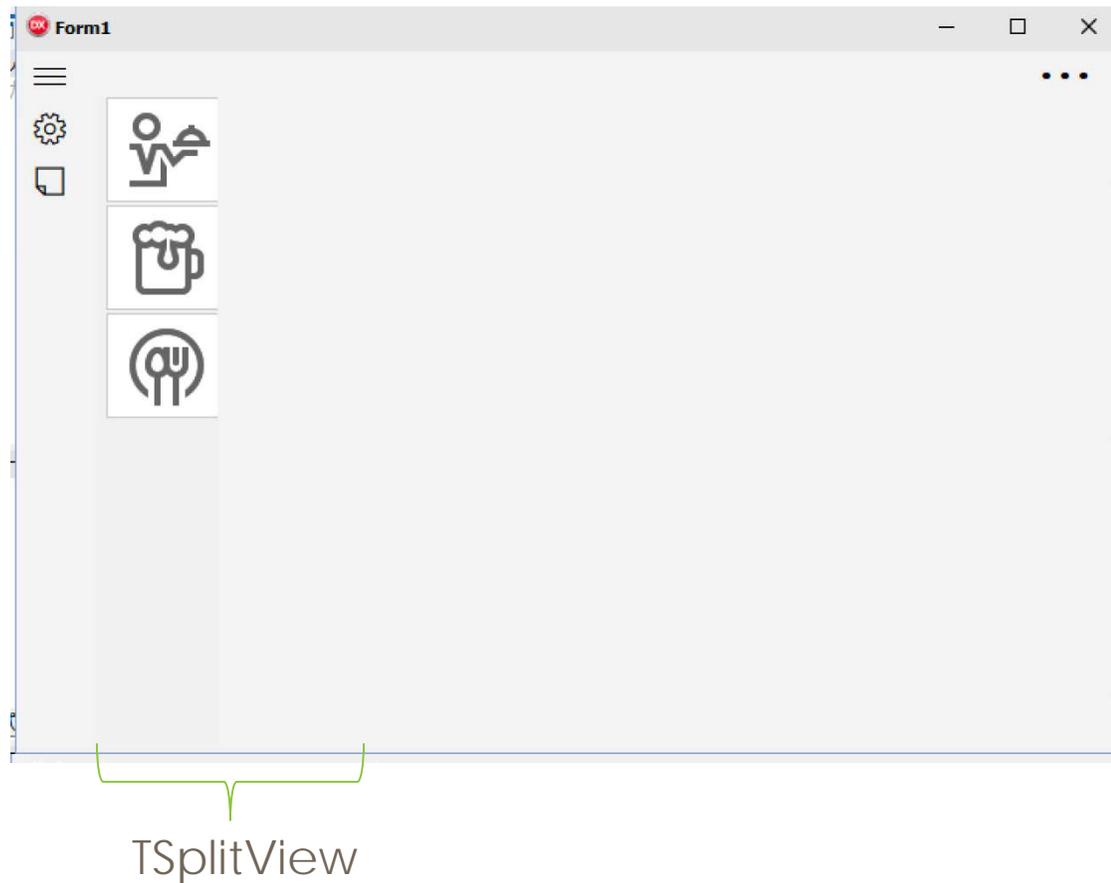
バーガーメニュー

- Windows 10やAndroidのバーガーメニュー
 - ドロアー展開方法の違い
 - ナビゲーションウィンドウ
- iOSでバーガーメニューは無いほうがよい
 - バーガーメニューついてるアプリは多いです。
 - タブバーを利用しiPhoneの場合3-5タブで完結できるデザインがよい



VCL Windows 10 TSplitView ナビゲーションウィンドウ

- バーガーボタンからTSplitViewを開閉

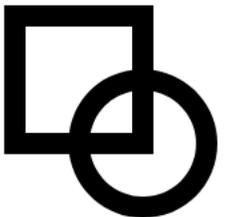
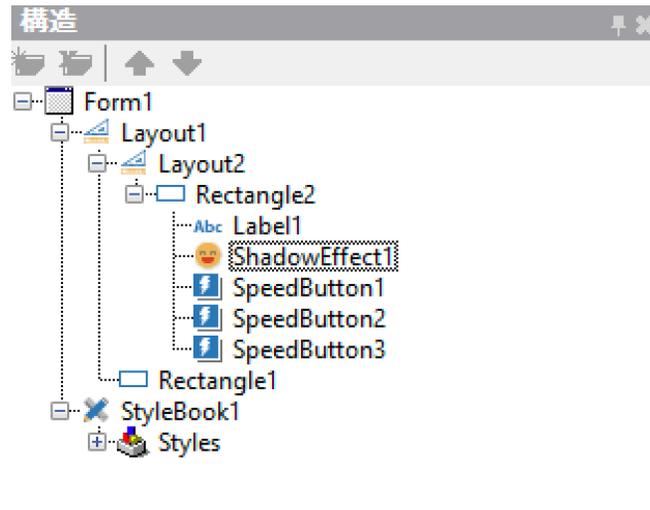


Android App Bar

- バーガーボタンはApp Barに配置します

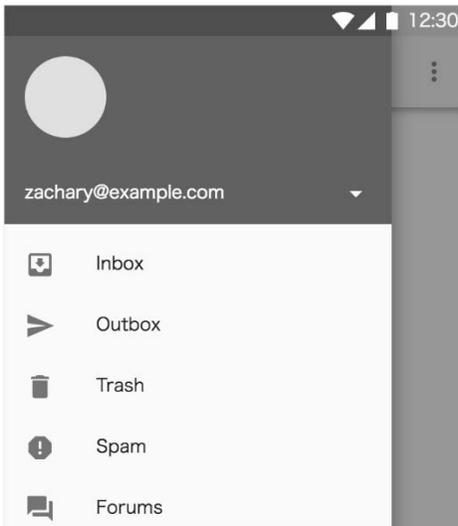


} 56

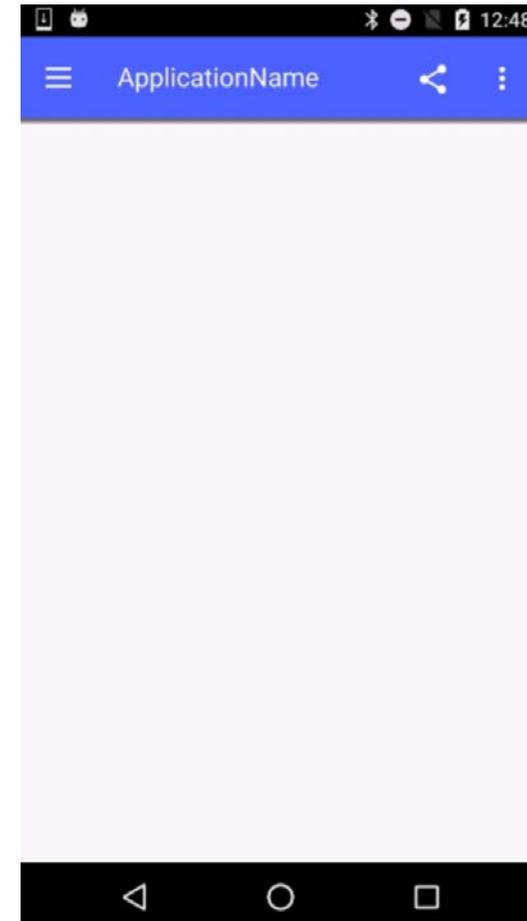
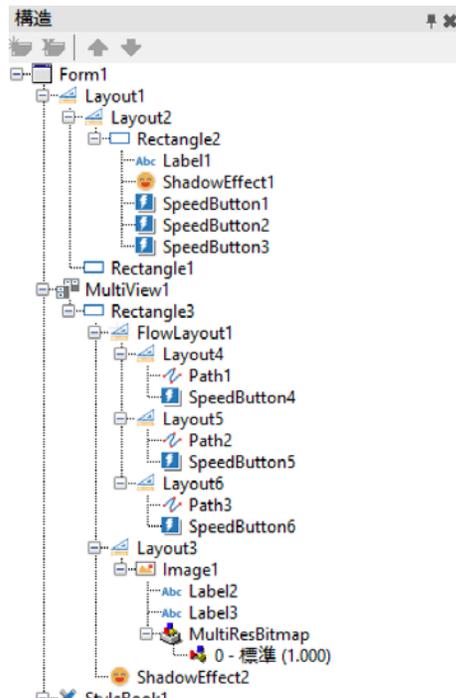


Android Navigation View

- バーガーボタンからナビゲーションビューを表示します

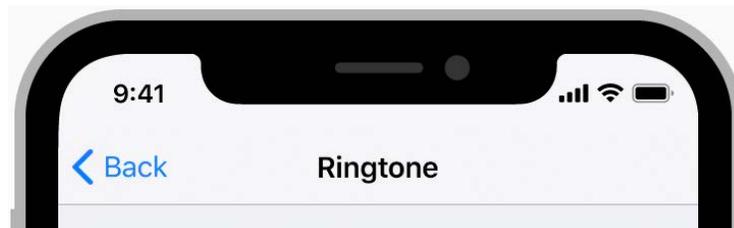


TMultiViewで再現可能です
ドロアーオープン時ステータスバーは
被せれないので注意が必要です



ナビゲーションバー

- iOSナビゲーションバー
 - 左から右へ流れるイメージ



- Android ナビゲーションバー
 - 標準についている黒い戻るボタン



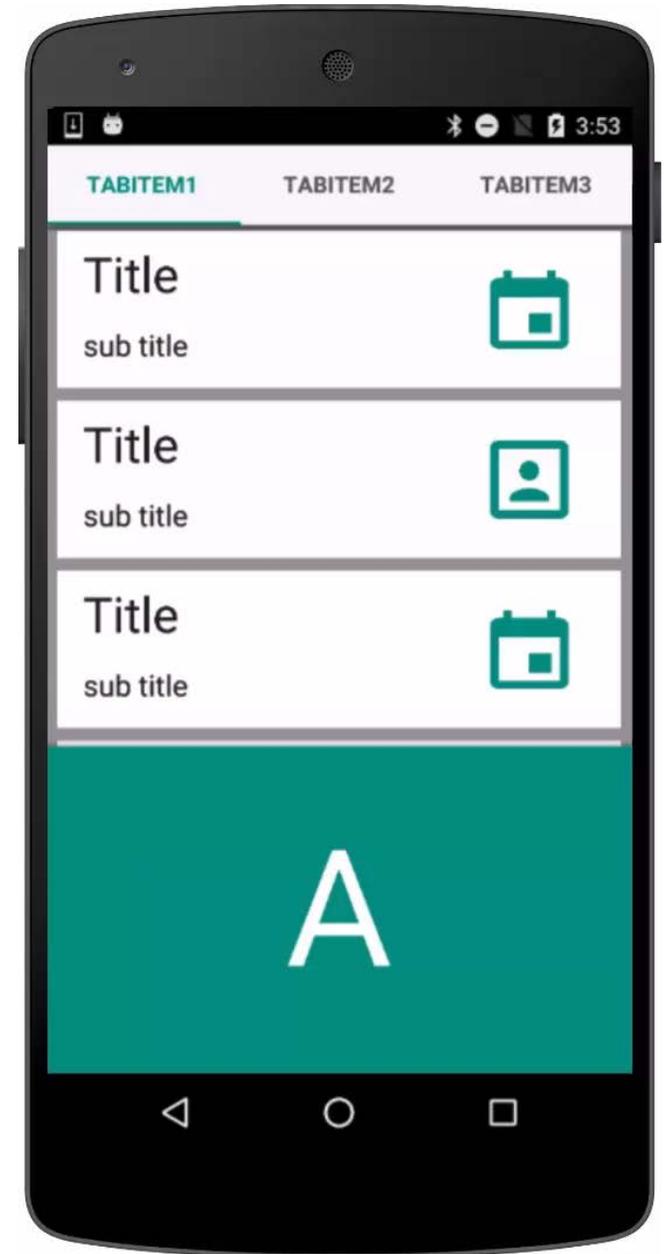
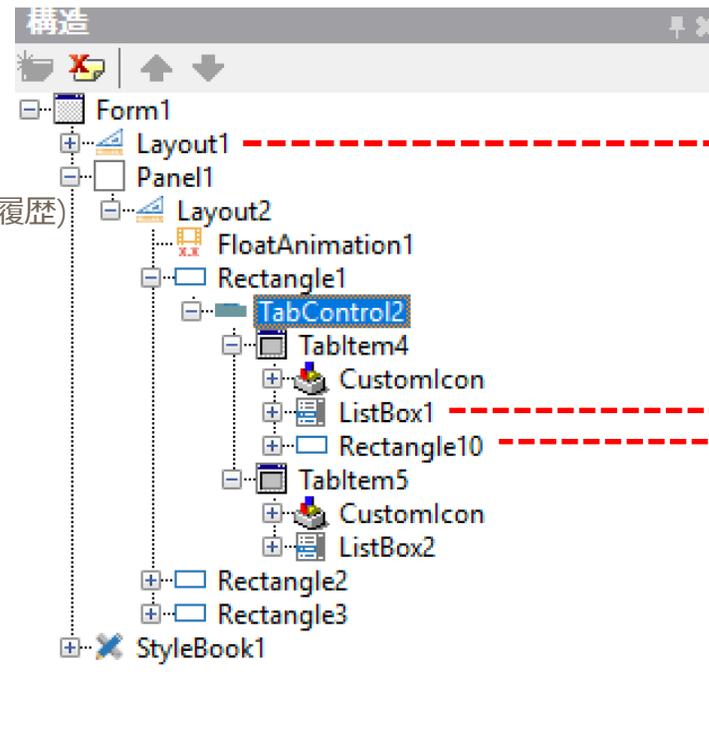
△(戻る) ○(ホーム) □(タスク - アプリ履歴)

Android ナビゲーションバーの[戻る]

- ナビゲーションバーの△[戻る]の動きに注意が必要です
 - Android慣れている人はナビゲーションバーを押す癖があります

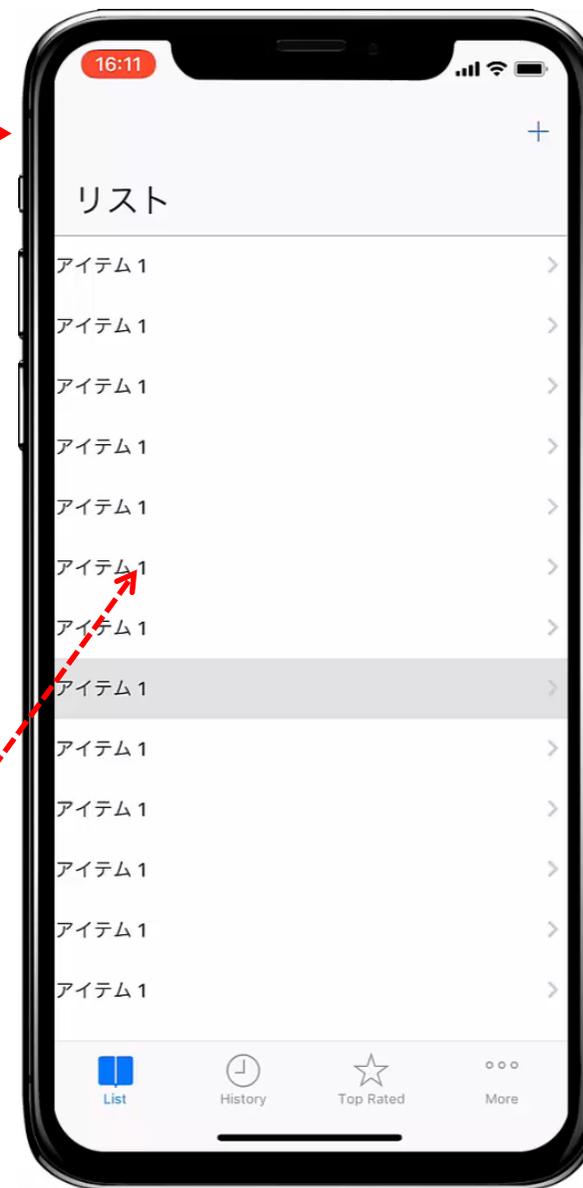
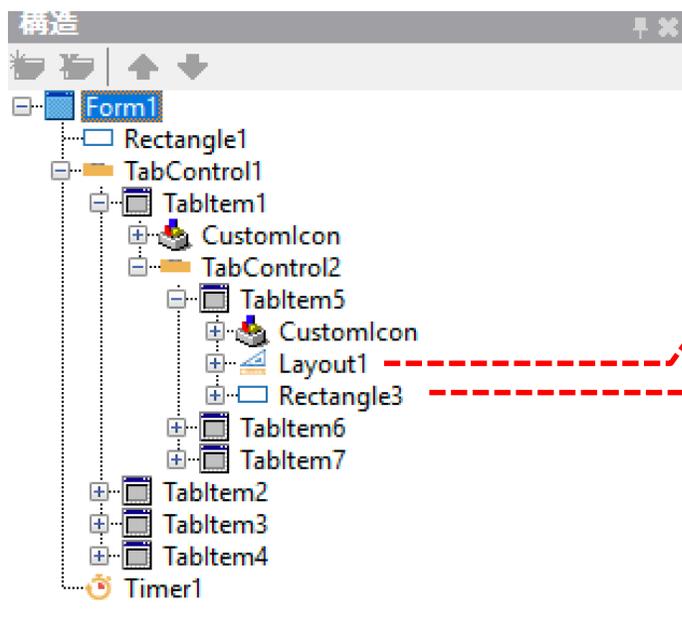


△(戻る) ○(ホーム) □(タスク - アプリ履歴)



iOS ナビゲーション

- 情報は左から右に流れる



ツールバー

- Windows 10 コマンドエレメント
 - ファンクションボタンのような扱い

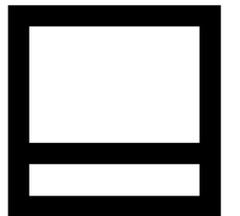


- iOS Safariの下側に表示されているバー



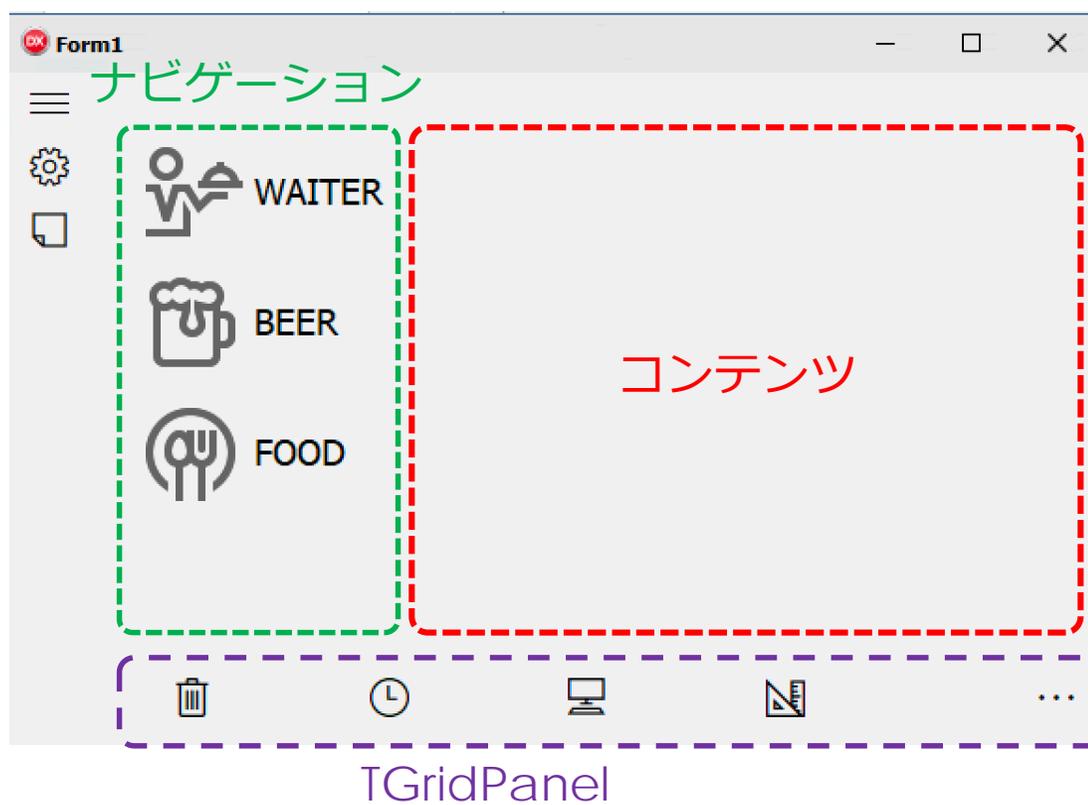
Android Bottom Navigation

- 下部ナビゲーションバーです。
- ワンタッチでトップレベルのコンテンツビューを簡単に切り替えることができます。
- このパターンは、3~5つのトップレベルの目的地がある場合に使用できます。



VCL Windows 10 TGridPanelコマンドエレメント

- TGridPanelで平均的にボタンを配置する

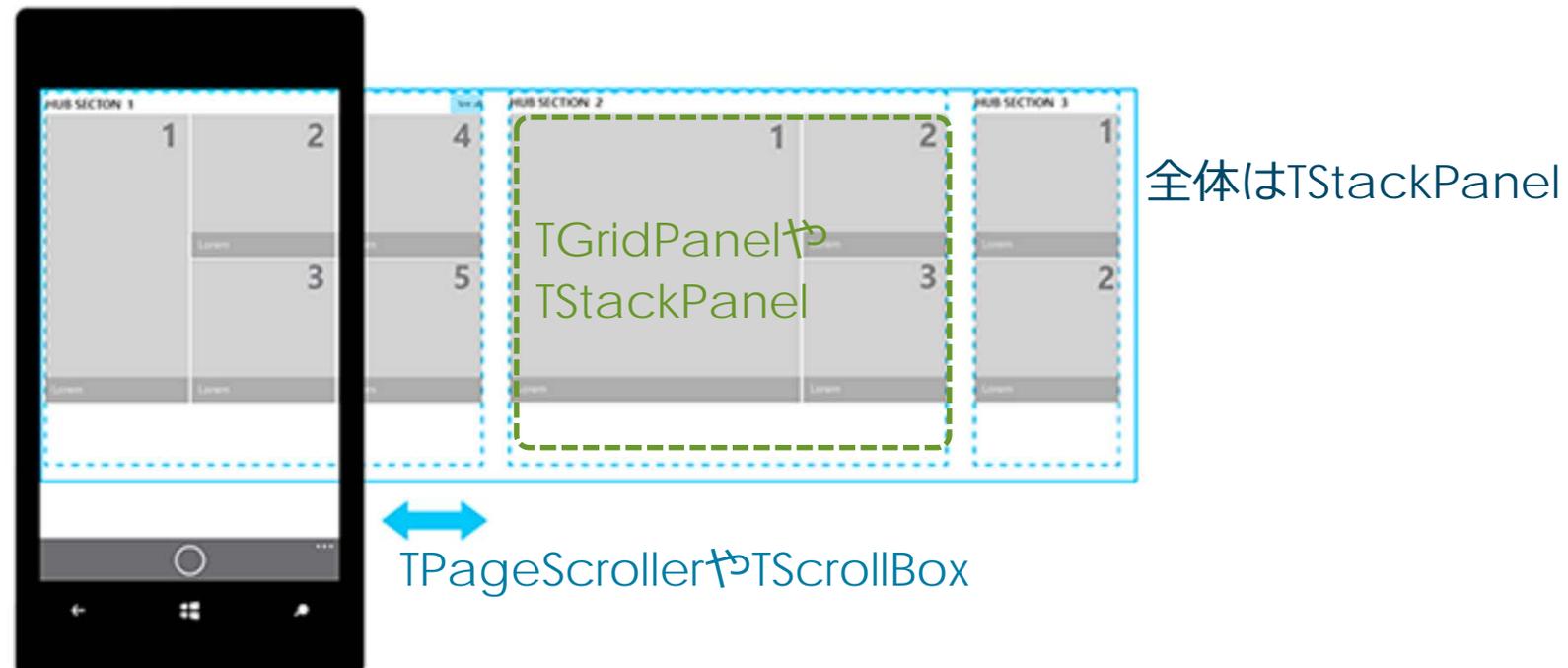


コマンドエレメントは
すべてのパンで共通にするのか？
設計時の判断が必要

Windows 10 ハブコントロール

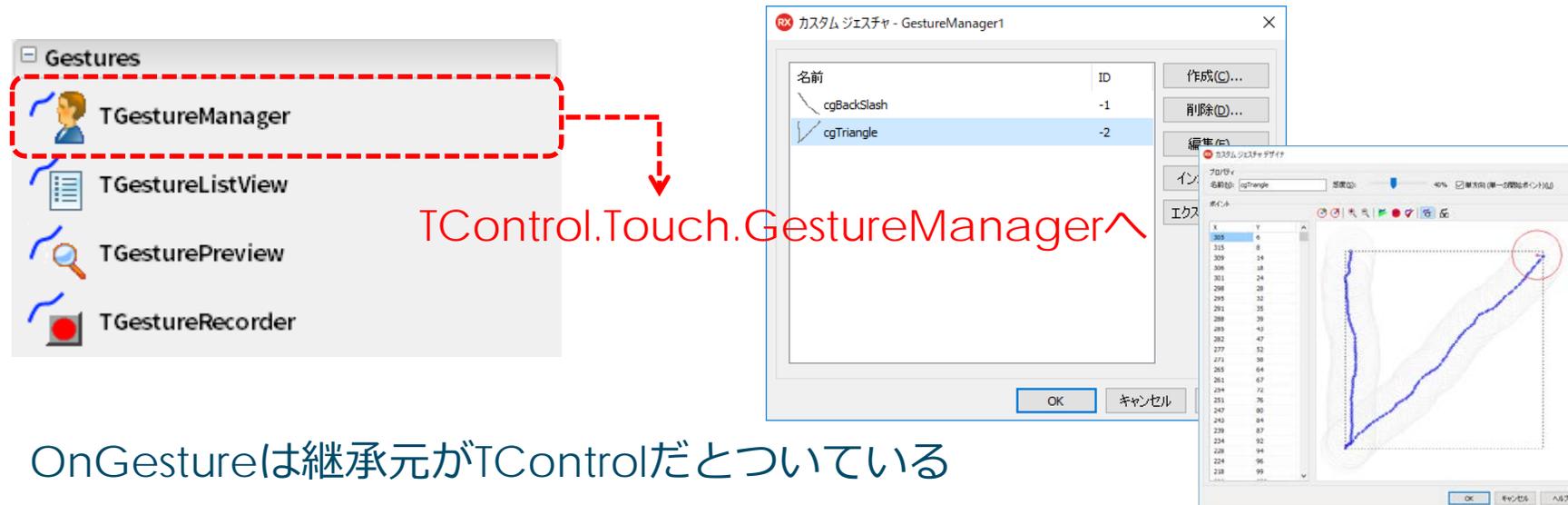
- ハブのコンテンツはパンする堅牢なビューに表示でき、新しい情報、入手可能な項目、関連する項目がひとめでわかります。

VCLでの例



Windows 10/iOS/Androidでのタッチ操作

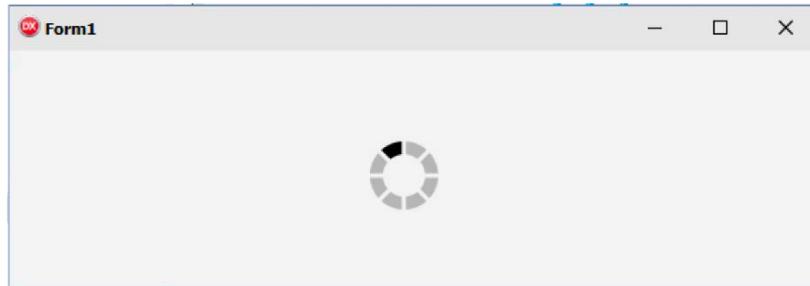
- タッチ操作時のコントロールが簡単に利用できます
- 常に指で操作されることを考慮してUI設計を行う
 - 操作部品の大きさや配置
 - (VCLでは「標準ジェスチャ(30種類以上)」と「カスタムジェスチャ」でタッチ操作を扱える)



OnGestureは継承元がTControlだとしている

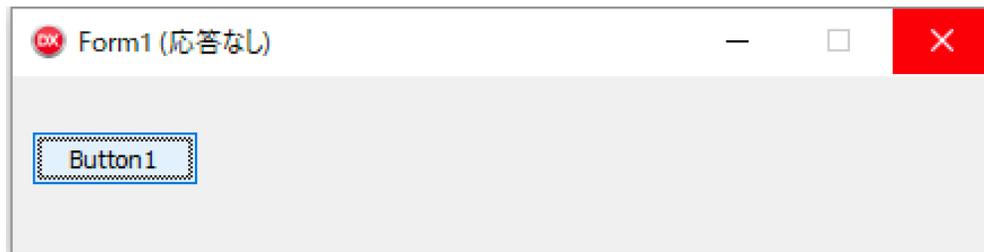
Windows 10 操作を完了までに 2秒

- 重い処理 / ダウンロード / サーバーからのレスポンス
 - ユーザーの操作をブロックするモーダルな確定的でないタスクでは、**進行状況不定バー** TActivityIndicator



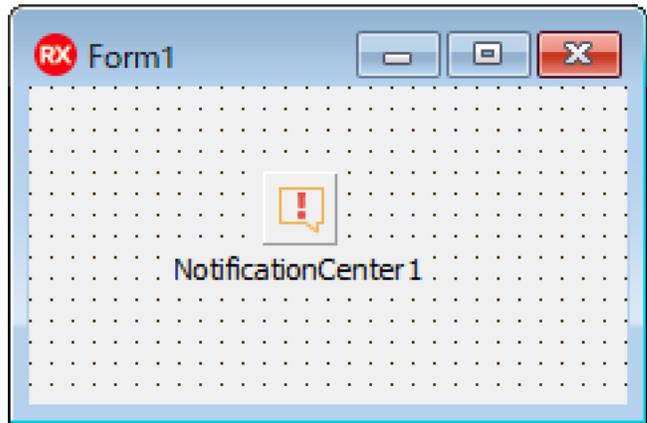
10秒より長い時間がかかる 場合は、操作を取り消す方法を用意

- 重い処理での(応答なし)もThread化



Windows 10 通知

- メッセージボックスやダイアログによる通知を `TNotificationCenter` コンポーネントを利用する



iOS/AndroidにもTNotificationCenterは付いています
スマホ操作タイミングによっては迷惑なメッセージになる

まとめ

- Windows 10/iOS/Androidそれぞれのレイアウト文化があります
- アニメーション動作には理由があります

- FireMonkey標準機能だけで実現できます

THANKS!

www.embarcadero.com/jp

第35回 エンバカデロ・デベロッパーキャンプ