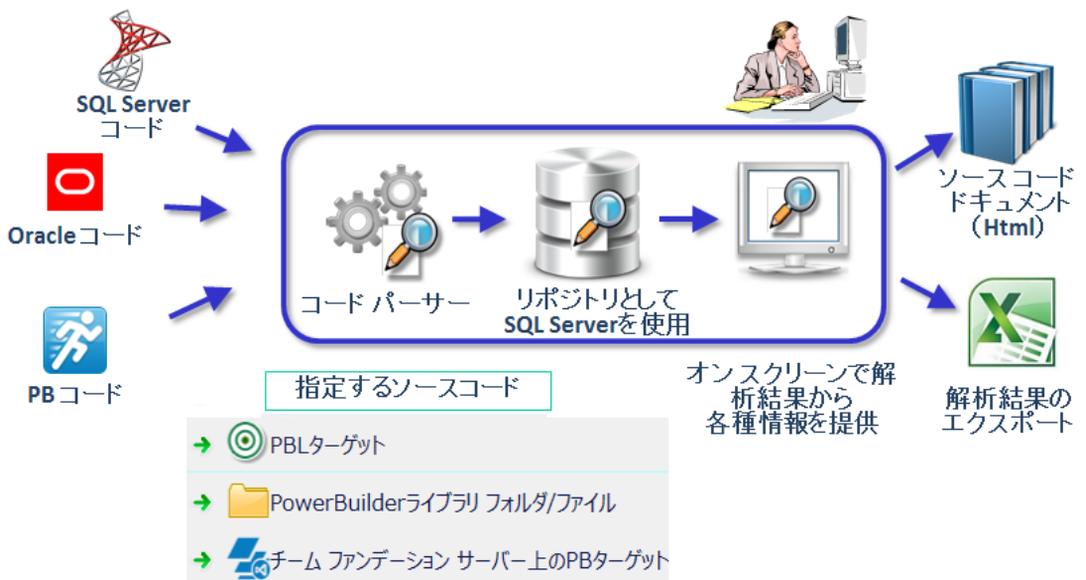




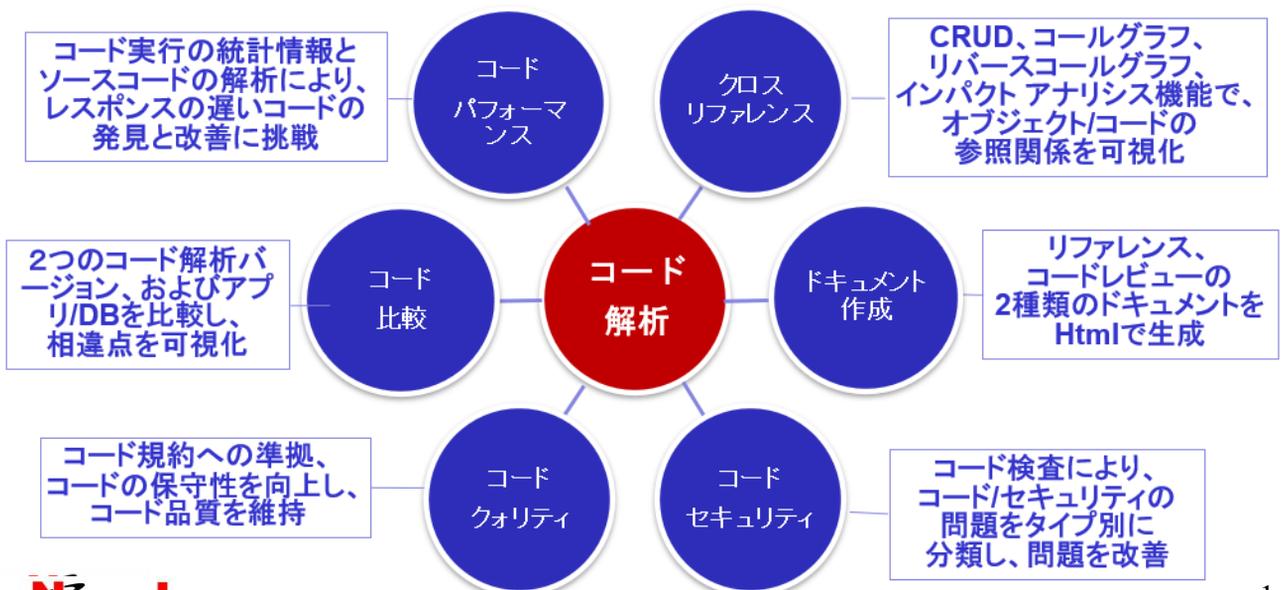
Visual Expert はコード解析結果をあらゆる視点から可視化！
C/S構成ではWebブラウザからもアクセス可能！
2025年はAI活用プラットフォームの実現へ！

■ **PowerBuilder, ORACLE, SQLServerのコード解析パーサーを搭載**

- ソースコードの解析結果を、スクリーン上であらゆる角度から調査
- ドキュメント作成および各種コード解析情報をExcel/HTMLでエクスポート可能



■ **Visual Expertの機能概要； +200以上の機能を搭載**



■2025年はAIを活用した機能を提供予定

-  **ビジネス説明**; 選択したコードのビジネス指向の説明を生成
-  **コードロジック説明**; コード構造を分析して、ロジックに関する技術的な説明を提供
-  **コードコメント追加**; コード内にコメントを挿入して、ロジックと実装を文書化
-  **問題修復**; コード欠陥やセキュリティの脆弱性を特定し、修正されたコードバージョンを生成
-  **SQLとコードの最適化**; 非効率的なコードを検出し、最適化されたコード代替を生成

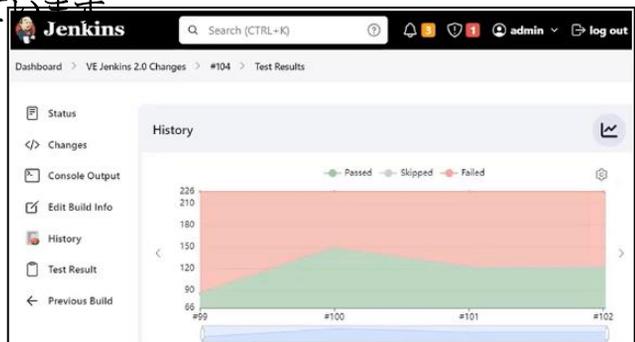
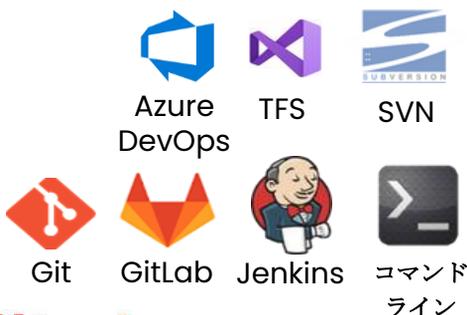


■コード検査

PowerBuilderでは79の定義済みルールにより、コード解析時にセキュリティの脆弱性、バグ、保守性の問題を特定し、深刻度とタイプ別に分類します。

■継続的インテグレーション; DevOpsのサポート

Visual Expert は、最も人気のあるソース管理および自動化サーバーとシームレスに統合できます。新バージョンでは、**PowerBuilder 2022**以降へのシームレスな移行、**DevOps**統合のための**JSON**および**JUnit**形式の包括的なコード検査レポート、そして大幅なパフォーマンス向上もサポートしています。



■ CRUDマトリクスによるテーブル/カラム操作の可視化;

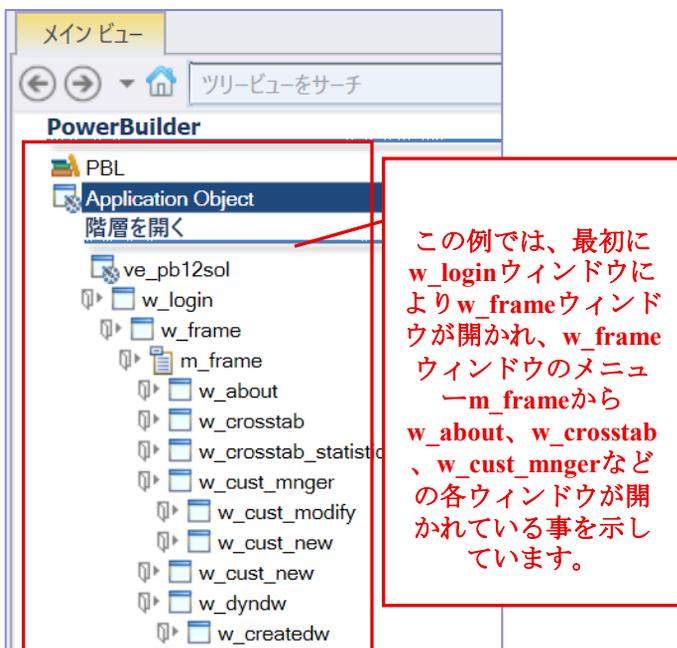
Visual Expertは、複数Table/オブジェクトを選択してCRUDマトリクスを自動生成できます。

CRUD Matrix		TABLES & VIEWS				
Calling Item	Item Type	customer	product	sales_order_items	sales_order	states
soladvanced.pbl.d_dddw_states	Datawindow Object					RU
soladvanced.pbl.d_order_form_select	Datawindow Object				RU	
comadvanced.pbl.w_product_report.open	Event		R			
comadvanced.pbl.w_sales_report.ue_retrieve	Event		R	R	R	
soladvanced.pbl.w_cust_mnger.cb_delete.clicked	Event	UD				
soladvanced.pbl.w_cust_mnger.open	Event	R				
soladvanced.pbl.w_cust_new.open	Event	C				
soladvanced.pbl.w_order_mnger.cb_add_item.clicked	Event		C	C		
soladvanced.pbl.w_order_mnger.cb_add.clicked	Event				C	
soladvanced.pbl.w_order_mnger.cb_save.clicked	Event			U	U	
soladvanced.pbl.w_order_mnger.cb_save.clicked	Event		C	C		
soladvanced.pbl.w_order_mnger.cb_save.clicked	Event			U	U	
soladvanced.pbl.w_order_mnger.cb_save.clicked	Event		C	C		
soladvanced.pbl.w_order_mnger.cb_save.clicked	Event				C	
soladvanced.pbl.w_select_customer.open	Event	R				
soladvanced.pbl.u_dw.uf_get_custid	Function	R				
soladvanced.pbl.u_dw.uf_get_orderid	Function				R	

DWオブジェクト、イベント/関数スクリプトによるDBテーブル/カラムの操作がマトリクスで表示されます。

■ PBアプリケーション ウィンドウのオープン階層

- アプリケーション ウィンドウのオープン階層で、アプリケーション全体の処理概要を把握できます。
- 各グローバル変数を使用しているすべてのオブジェクト、スクリプトがリストされ、目的のスクリプトを簡単に特定できます。



■インパクト アナリシス

Visual Expertは、オブジェクト/テーブル/カラムのコード変更によってどの要素が影響を受けるかを可視化できます。

インパクトアナリシスは、ツリービュー形式、およびダイアグラムで表現できます。

w_cust_newウィンドウを変更する場合の影響範囲を調べます。

このウィンドウは、**m_frame**メニュー、**w_cust_mgr**ウィンドウから表示されます。これらのオブジェクトが影響範囲になります。

■ オブジェクトの依存性をあらゆる視点から可視化；

- コールグラフにより、コール チェーンを可視化
- リバース コールグラフにより、呼び出し元を可視化

コールグラフ、リバースコールグラフで、呼出し階層または呼出し元を簡単に探る事ができます。

すべてのユーザオブジェクト

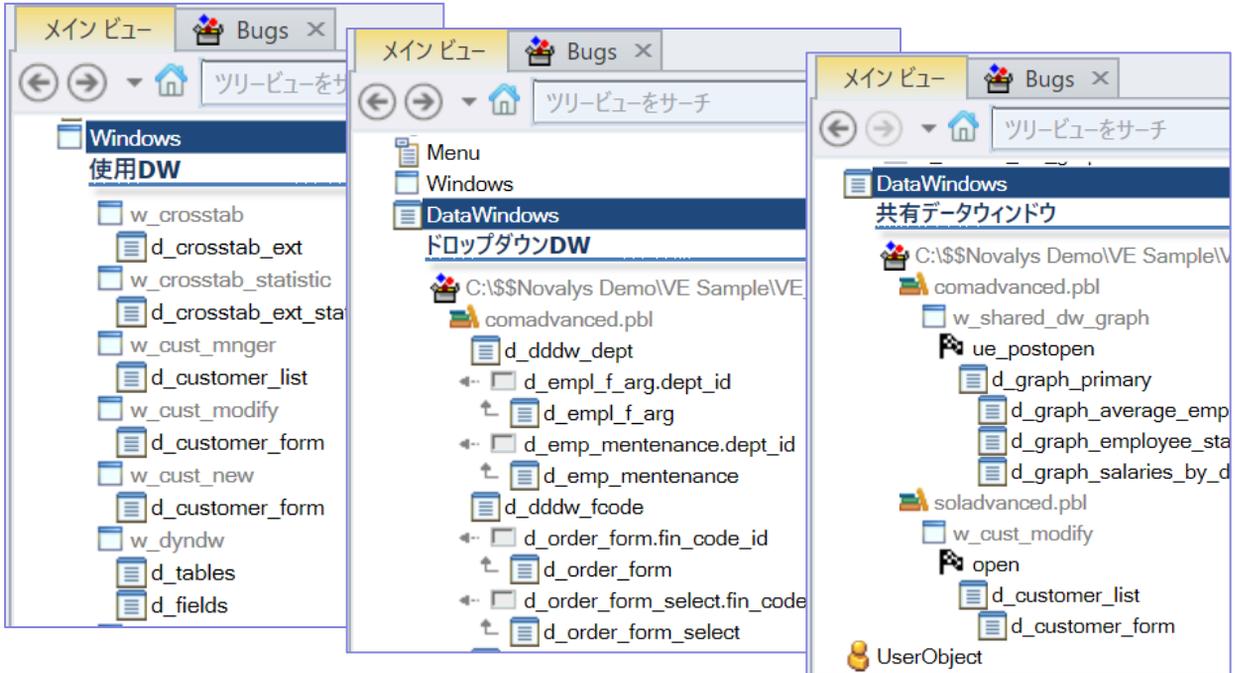
f(*) of_setsecurity
リバース コールグラフ

- f(*) n_cst_appmanager.of_setsecurity
- w_logon_demo.cbx_activatesecurity.clicked
- n_cst_demo.pfc.constructor
- w_logon_demo.cbx_activatesecurity.clicked
- pfc_logon
- n_cst_appreference

■ ウィンドウ、データウィンドウ、DDDWの使用状況を可視化

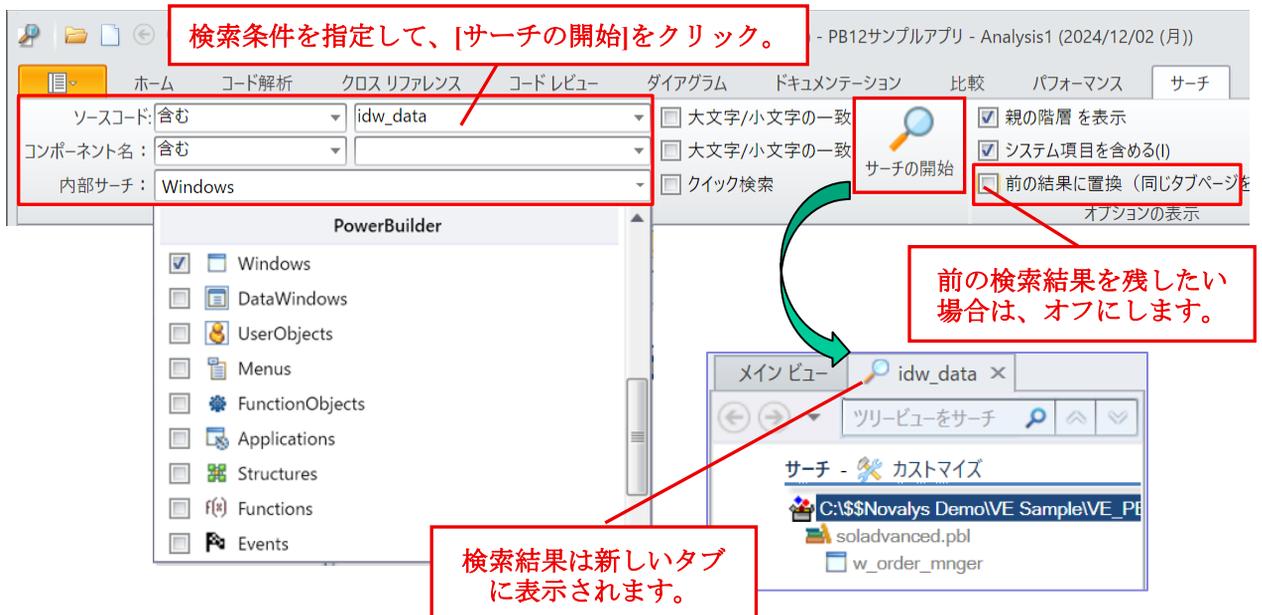
- 各ウィンドウが使用しているDWは?、データウィンドウに設定されたDDDWは?、共有データウィンドウが使用されているウィンドウは?

--- Visual Expertは様々な質問に答えます ---



■ グローバル サーチ機能を搭載!

- グローバル サーチは様々な検索が可能!
検索結果は新しいタブに表示されます。



■DataWindow構造の把握

- 個別DataWindows を選択して、各項目をクリックして設定内容を確認できます。

The image shows a screenshot of the DataWindow Designer interface with several panels and callouts explaining the structure of the `d_customer_form` DataWindow.

Left Panel (Properties): A list of properties for the DataWindow, with red boxes highlighting `データソース`, `検索引数`, `更新可能カラム`, `コンテナ階層`, and `配置`.

Right Panel (Structure): A tree view showing the internal structure of `d_customer_form`, with callouts explaining each level:

- データソース (Data Source):** Shows `Retrieve SQL` and `customer`. Callout: "このDWのSQLとアクセスしているテーブルが判ります。" (You can see the SQL and the table being accessed by this DW.)
- 検索引数 (Search Arguments):** Shows `custid`. Callout: "このDWは、検索引数として custid を使用しています。" (This DW uses custid as a search argument.)
- 更新可能カラム (Updatable Columns):** Shows `id`, `lname`, and `fname`. Callout: "このDWは、更新可能カラムとして、これらのカラムが定義されています。" (This DW defines these columns as updatable columns.)
- コンテナ階層 (Container Hierarchy):** Shows a hierarchy including `w_cust_new.dw_1`, `w_cust_new`, `soladvanced.pbl`, `w_cust_modify.dw_1`, and `w_cust_modify`. Callout: "このDWのコンテナ階層が解ります。w_cust_newのdw1、w_cust_modifyのdw1で使用されています。" (You can see the container hierarchy of this DW. It is used in dw1 of w_cust_new and dw1 of w_cust_modify.)

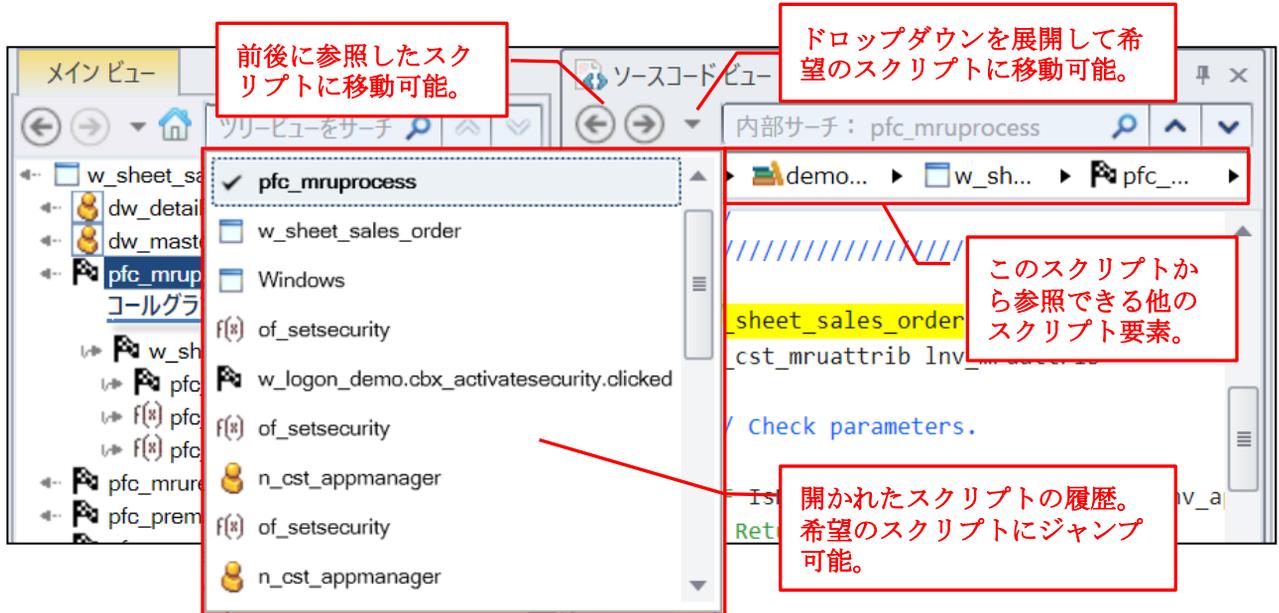
Bottom Left Panel (Cross Reference): A list of cross-reference items, with red boxes highlighting `インパクトアナリシス (ツリー)`, `使用されているコントロール`, and `呼出されたDDDWW`.

Bottom Right Panels (Cross Reference Details): Three panels showing the details of the selected cross-reference items:

- インパクトアナリシス (ツリー):** Shows a tree view of dependencies. Callout: "d_customer_formのインパクトアナリシスです。変更する場合は、w_cust_modifyとw_cust_newへの影響を考慮します。" (This is the impact analysis of d_customer_form. When changing, consider the impact on w_cust_modify and w_cust_new.)
- 使用されているコントロール:** Shows the controls used by the DataWindow. Callout: "このDWは、w_cust_modifyとw_cust_newで使用されています。" (This DW is used in w_cust_modify and w_cust_new.)
- 呼出されたDDDWW:** Shows the DataWindows called by this one. Callout: "d_customer_formはstateカラムでd_dddw_statesを使用しています。" (d_customer_form uses the state column of d_dddw_states.)

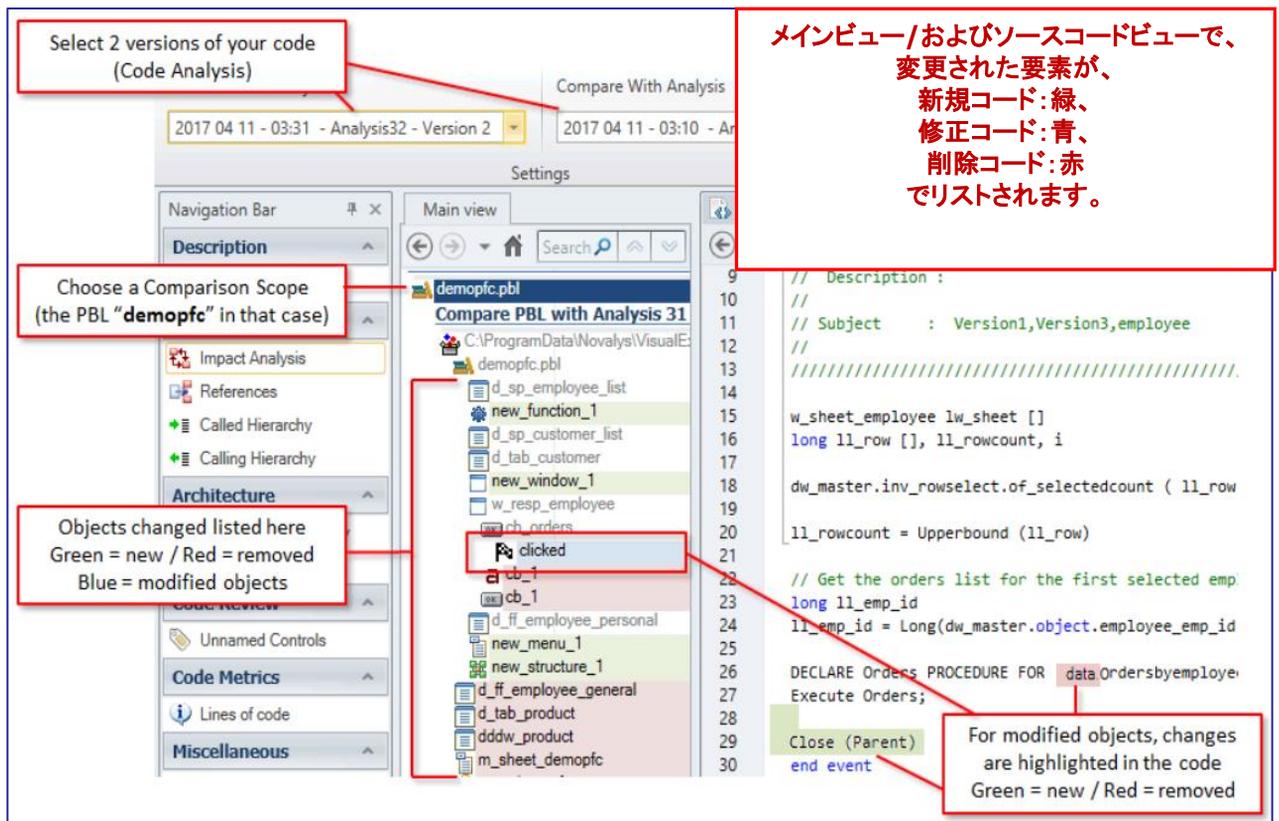
■ ソースコードをドリルダウンして自由に移動可能

- ソースコードビューはハイパーリンクが貼られ、次々とドリルダウンして自由に移動できます。



■ アプリケーション バージョン間のソースコード比較;

- アプリケーション バージョン間、プロジェクトの解析結果間のコードを比較できます。



■ アプリケーションドキュメントの生成;

- リファレンスドキュメント/コードレビューの2種類のドキュメントをhtmlで作成できます。ドキュメント内はハイパーリンクが張り巡らされ、ドリルダウンで各種情報を表示します。



プロジェクトのドキュメント

PB12サンプルアプリ

VE_PB12sol

PBL

ウィンドウ

ユーザオブジェクト

データウィンドウ

メニュー

構造体

アプリケーション オブジェクト

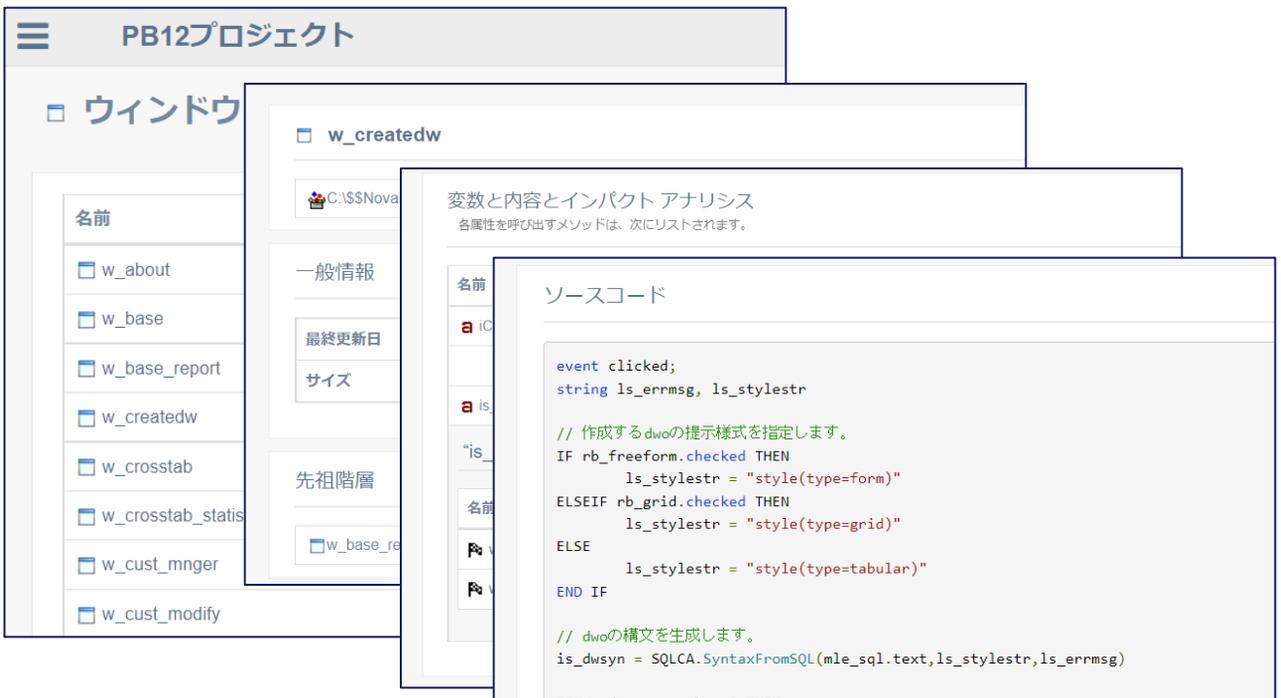
PB12サンプルアプリ

ドキュメント作成日: 日曜日 27 7月 2025 09:54

このドキュメント内のコンポーネントの詳細を表示するには、左フレーム内のコンポーネントをクリックしてください。

ダッシュボード

Application VE_PB12sol		
PBL	3 components	
ウィンドウ	23 components	
ユーザオブジェクト	14 components	
データウィンドウ	29 components	
メニュー	3 components	
構造体	1 components	
アプリケーション オブジェクト	1 components	
グローバル変数	6 components	



PB12プロジェクト

ウィンドウ

名前

- w_about
- w_base
- w_base_report
- w_createdw
- w_crosstab
- w_crosstab_statis
- w_cust_mgrer
- w_cust_modify

w_createdw

C:\\$Novas

変数と内容とインパクト アナリシス

各属性を呼び出すメソッドは、次にリストされます。

名前

一般情報

最終更新日

サイズ

先祖階層

名前

ソースコード

```

event clicked;
string ls_errmsg, ls_stylestr

// 作成するdwoの提示様式を指定します。
IF rb_freeform.checked THEN
    ls_stylestr = "style(type=form)"
ELSEIF rb_grid.checked THEN
    ls_stylestr = "style(type=grid)"
ELSE
    ls_stylestr = "style(type=tabular)"
END IF

// dwoの構文を生成します。
is_dwsyn = SQLCA.SyntaxFromSQL(mle_sql.text,ls_stylestr,ls_errmsg)
    
```