

---

# Windows COBOL2002 04-10 新機能のご紹介

2019年6月

株式会社 日立製作所

サービスプラットフォーム事業本部 IoT・クラウドサービス事業部  
フローマネジメント本部 基盤ソフトウェア部

## 1. 資産分析への活用

- 1 – 1 COBOLソース解析とは
- 1 – 2 COBOLソース解析の活用例

## 2. 影響調査への活用

- 2 – 1 データ影響波及分析とは
- 2 – 2 データ影響波及分析の活用例

## 3. Windows COBOL2002 04-10 新機能のご紹介

- 3 – 1 COBOL2002 04-10 新機能一覧
- 3 – 2 COBOL2002 Developer Professional 04-10 新機能
- 3 – 3 COBOL2002 Net Developer 04-10 新機能
- 3 – 4 カバレッジの新機能を活用したサービスのご紹介

---

# 1. 資産分析への活用

COBOL資産全体の構造を把握できます。  
プログラムの呼出し関係や未使用プログラムも把握できます。

- プログラム構造全体が見える化

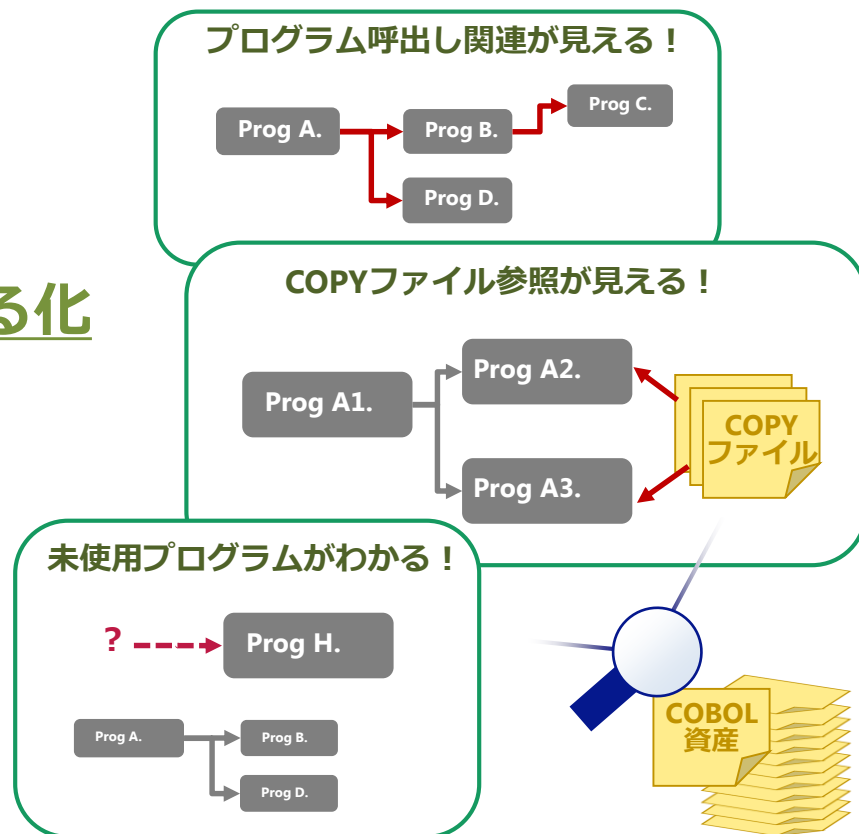
アプリ構造理解や影響調査を効率化

- 未使用プログラムや処理が見える化

資産棚卸を効率化

- HTML形式で見える化

使い慣れたブラウザで参照できて  
導入コスト小



## 1 COBOL資産の棚卸

長年改修を続けてきたシステムはブラックボックス化してしまい、使われていない資産も含まれていることがあります。本機能では、使われていない登録集原文や、デッドコードなどが表示できるため、COBOL資産の棚卸に活用できます。

## 2 COBOLアプリケーションの構造理解

システム改修や保守の新しい担当者はアプリケーションの構造理解が必要になります。本機能では、COBOLアプリケーション内のプログラムの一覧や、プログラム内で使われているファイル/データベース（表）/データ項目などの一覧が表示できるため、COBOLアプリケーションの構造理解に活用できます。

## 3 COBOLアプリケーションの関連要素の調査

システム改修や保守業務では、アプリの関連要素の調査が必要になります。本機能では、プログラムとプログラムの関連や、プログラムとデータベース（表）の関連などが表示できるため、関連要素の調査に活用できます。

## 4 ドキュメントのメンテナンス

度重なるプログラム改修においては、ドキュメントの修正が漏れることがあります。本機能では、COBOLソースからプログラム構造に関するドキュメントを生成できるため、ドキュメントのメンテナンスに活用できます。

---

## 2. 影響調査への活用

COBOLアプリ全体へのデータ項目の影響範囲を調査できます。  
従来の手作業による調査に比べ、漏れを防止できます。

### ● コンパイラの解析情報を基に影響個所を追跡\*

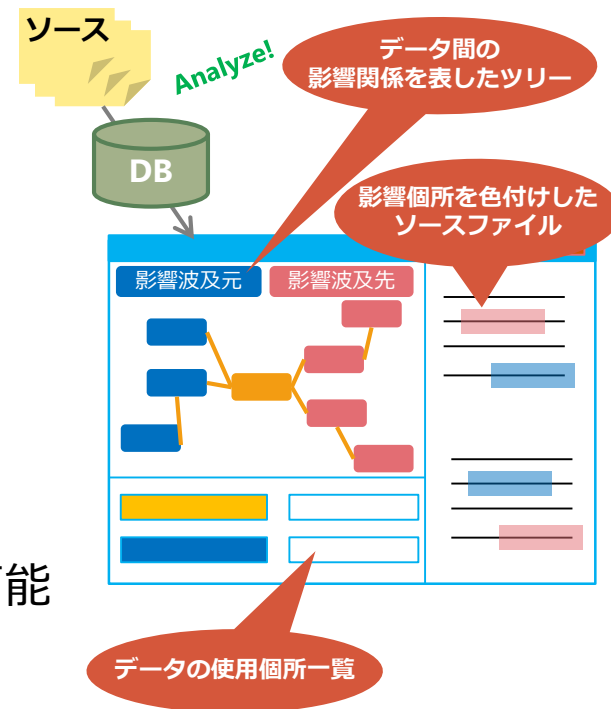
プログラム間の呼び出しで渡されるデータや別名で使用されるデータも追跡ができて影響個所洗い出しの漏れ防止

### ● 影響個所を見やすく表示

ソースコード上の影響個所が直ぐにわかる

### ● 影響個所はCSV形式でバッチ出力可能

調査結果を書き込んで、エビデンスとしても活用可能



\* : プログラム内で使用している項目の別名の数が多い、影響範囲が広すぎる、などの理由で解析できない場合があります。

### 1 影響調査

従来手法のgrepコマンドによる文字列検索を駆使した影響調査では調査範囲が漏れることがあります。本機能では、コンパイラによる解析情報を基に影響範囲を自動的に抽出できるため、従来手法に比べて漏れを防止できます。

### 2 障害発生時の調査

データ不正による障害が発生することがあります。本機能では、データ項目の転記元を追跡することができるため、障害発生時の調査の効率化に役立ちます。

### 3 プログラムテストの範囲絞り込み

システム改修におけるテストでは多くの工数が必要になる場合があります。本機能では、プログラムをまたいだデータ項目の影響範囲を抽出できるため、テスト範囲の絞り込みに役立ちます。



---

## 3. Windows COBOL2002 04-10 新機能のご紹介

## COBOLソース解析

SQLプリプロセス前COBOLソース対応

解析結果の可読性向上／SQLプリプロセス実行不要

未使用資産情報の拡充

棚卸作業の効率化（未使用のファイルや制御が渡らない節などの情報表示）

性能改善※1

プログラム解析処理からドキュメント生成までの時間を短縮※2

※1：COPY文解析、デッドコード解析処理の性能改善を実施

※2：約700万ステップ(COPY文展開後約900万ステップ)の当社性能測定用プログラムを用いた前バージョン04-00との比較

## データ影響波及分析

SQLプリプロセス前COBOLソース対応

解析結果の可読性向上／SQLプリプロセス実行不要

実行単位の範囲に絞り込んだ解析

解析精度の向上（解析範囲の絞り込み）

出力ファイル選択オプション

使い易さの向上（必要なファイルだけ出力）

## カバレッジ

カバレッジ情報リストのCSV形式出力コマンド

外部ツールによる  
カバレッジ情報の利活用が容易

## XML連携機能

XMLドキュメントのWindows-31Jエンコーディング対応

Windows-31Jエンコーディングの  
XMLドキュメントの扱いが可能

## コンパイラ

通信記述項の初期値をメインフレームに合わせるオプション

メインフレームからの移行工数を削減

## COBOLソース解析・データ影響波及分析の新機能

## SQLプリプロセッサ前COBOLソース対応

これまでは・・・

SQLプリプロセッサが変換したCOBOLソースに対する解析結果のため  
可読性が低かった。

プリプロセッサ

プリプロセッサ変換  
COBOLソース

04-00以前  
COBOLソース解析  
データ影響波及分析

解析結果

ユーザ作成のCOBOLソース

EXEC SQL

...

END-EXEC

04-10製品  
COBOLソース解析\*  
データ影響波及分析\*

解析結果

これからは！

ユーザ作成のCOBOLソースをそのまま解析！解析結果の可読性が向上！  
SQLプリプロセッサによる変換の手間も不要！

\*：動的SQLやSQL手続きの構文を使用している、DBMS固有の特殊構文を使用している、SQLの予約語を識別子や利用者定義語に使用している、などの理由で解析できない場合があります。

## カバレッジの新機能

# カバレッジ情報リストのCSV形式出力コマンド

これまででは・・・

読みやすさ重視のテキスト形式は外部ツール\*では利活用しにくかった。

カバレッジ情報表示  
コマンド  
(テキスト形式)

カバレッジ情報リスト テキスト形式

```

COBOL2002 (d) VV-RR *****
                * カバレッジ情報 *                2003-01-01 09:30:00
-----
クラス名      MAKELINE
コンパイル日時 2003-01-01 09:00:00      変更回数    :   1
テスト日時    2003-01-01 09:15:00      テスト回数    :   2
-----
<全ソース表示>
メソッド名    INIT_MAKELINE.F
変更<CO><C1>
0000021 PROCEDURE DIVISION
*   0000022 INVOKE SUPER  INIT-COLORS-F
+   0000023 CALL  'OBLEXEC' USING EXEC-NAME-LEN EXEC-NAME EXEC-PARM
*   0000024 EXIT  METHOD
    
```



外部ツール\*

ツール用に  
データ整形

カバレッジ情報の  
利活用

プログラム情報  
ファイル

カバレッジ情報を格納するファイル

カバレッジ情報表示  
コマンド  
(CSV形式)

カバレッジ情報リスト CSV形式

```

バージョン,"出力日時","コンパイル日時","プログラム情報ファイル格納先","プログラム情報
"COBOL2002 (X) 04-10","2019-04-01 11:00:00","2019-04-01 09:00:00","C:\COBOL\COBOLPV",a
"顧客単位名","ソース要素種別","ソース要素名","未実行ソース","差分ソース","差分未実行ソ
"MAKELINE","METHOD","INIT_MAKELINE.F",0000021,"PROCEDURE DIVISION,"
"MAKELINE","METHOD","INIT_MAKELINE.F",ACT,0000022,"INVOKE SUPER ",
"MAKELINE","METHOD","INIT_MAKELINE.F",ACT,0000023,"CALL  'OBLEXEC' U
"MAKELINE","METHOD","INIT_MAKELINE.F",0000024,"EXIT  METHOD,"
"MAKELINE","METHOD","INIT_MAKELINE.F",0000025,"END  METHOD  INIT_MAKELINE.F
"MAKELINE","METHOD","INIT_MAKELINE.F",0000026,"
"MAKELINE","METHOD","INIT_MAKELINE.F",0000027,"END  FACTORY,"
"MAKELINE","METHOD","INIT_MAKELINE.F",ACT,0000043,"PROCEDURE DIVISION,"
"MAKELINE","METHOD","INIT_MAKELINE.F",ACT,0000044,"COMPUTE  MISOOUNT =
"MAKELINE","METHOD","INIT_MAKELINE.F",ACT,0000045,"EXIT  METHOD,"
"MAKELINE","METHOD","INIT_MAKELINE.F",0000046,"END  METHOD  INIT_MAKELINE.F
"MAKELINE","METHOD","INIT_MAKELINE.F",0000047,"PROCEDURE DIVISION USING 1-COLOR
"MAKELINE","METHOD","INIT_MAKELINE.F",ACT,0000048,"LAUNCH  SELF  'COBOL2002
    
```

これからは！

カバレッジ情報の利活用が簡単！

データを  
そのまま活用！

外部ツール\*

カバレッジ率の推移を表示



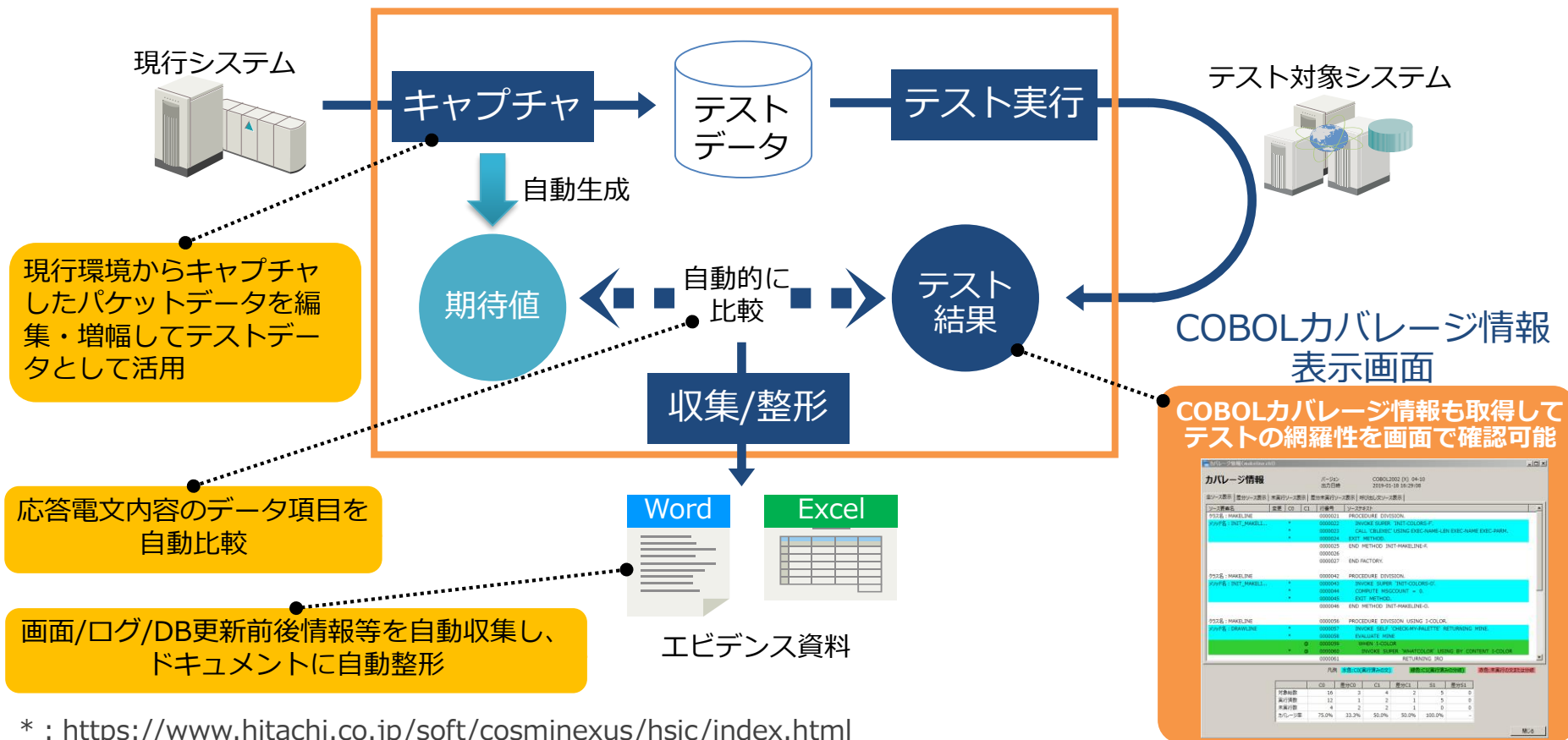
データ可視化ツールなどで表示できます。

\* : データ可視化ツールなど、ユーザ側でご用意いただく必要があります。

## システム検証支援サービス\*

システム検証支援サービスは、システムの品質確保にかかる工数を削減することで、システム構築・維持管理の効率化を支援するサービスです。

システム検証支援サービス



\* : <https://www.hitachi.co.jp/soft/cosminexus/hsic/index.html>

- ・HITACHI、Cosminexusは、株式会社 日立製作所の商標または登録商標です。
- ・Excel, およびWindowsは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
- ・製品仕様は、改良のため変更することがあります。

**HITACHI**  
Inspire the Next

# END

---

## Windows COBOL2002 04-10 新機能のご紹介

2019年6月

株式会社 日立製作所

サービスプラットフォーム事業本部 IoT・クラウドサービス事業部

フローマネジメント本部 基盤ソフトウェア部

**HITACHI**  
Inspire the Next 