

# COBOL 開発環境 利用の手引

# はしがき

本書は、COBOL 開発環境の機能と操作方法についてまとめた説明書であり、最初にお読みください。

## 対象読者

本書は、COBOL 開発環境を操作する方を対象としています。

COBOL の基本的な言語仕様と、Windows と Linux(Linux 版のみ)の操作方法について理解していることを前提としています。

## マニュアルの構成

本書は、次に示す章と付録から構成されています。

### 第 1 章 COBOL 開発環境を利用した開発

COBOL 開発環境を利用した開発について概要を説明しています。

### 第 2 章 操作環境

COBOL 開発環境を利用してプログラムを開発するときの基本的な操作について説明しています。

### 第 3 章 ソリューション

COBOL 開発環境の開発資産を管理するプロジェクトをまとめるソリューションについて説明しています。

### 第 4 章 プロジェクト

COBOL の開発資産を管理するプロジェクトについて説明しています。

### 第 5 章 ファイルの編集

COBOL 開発環境での原始プログラムを編集するエディター操作について説明しています。

### 第 6 章 アプリケーションの生成

COBOL 開発環境での COBOL アプリケーションの生成について説明しています。

### 第 7 章 ツール

COBOL 開発環境で使用可能なメニュー項目について説明しています。

### 第 8 章 付録

関連説明書

本書は次の説明書と関連があります。必要に応じてお読みください。

- COBOL プログラミングの手引
- COBOL 言語説明書
- COBOL ユーザーズガイド
- COBOL GUI ユーザーズガイド
- COBOL SQL ユーザーズガイド

キーの表記

本書でのジェネリックキーの表記とご利用のキーボードのキーの表記は異なる場合があります。キーの対応については、Windows のマニュアルを参照して確認してください。

略語一覧

本書で使用する英略語の正式名称を次に示します。

英略語	正式名称
CSV	Comma Separated Values
DLL	Dynamic Linking Library
SQL	Structured Query Language

本書で使用する記号

本書で使用する記号を次に示します。

記号	意味
[   ] キー	文字キーや F (ファンクション) キーを意味する。
[   ] + [   ] キー	+ の前のキーを押したまま、後ろのキーを押すことを意味する。
{   }	この記号で囲まれている複数の項目のうちから 1 つを選択することを意味する。項目が縦に複数行にわたって記述されている場合は、そのうちの 1 行分を選択する。
{   } +	この記号で囲まれている複数の項目のうちから 1 つ以上を選択することを意味する。同じ要素の繰り返し指定はできないが、異なる要素であれば、複数指定できる。たとえば、{ A   B   C } + と表記されている場合は、A ・ B ・ C をすべて選択することができる。項目が縦に複数行にわたって記述されている場合は、そのうちの 1 行分を選択する。
[   ]	この記号で囲まれている項目は、省略することができることを意味する。 複数の項目が縦または横に並べて記述されている場合には、すべてを省略するか、記号 {   } と同じく、どれか 1 つを選択することを意味する。
...	記述が省略されていることを意味する。 この記号の直前に示された項目を繰り返して複数指定できる。
<u>下線</u>	かっこで囲まれた複数の項目のうち 1 項目に対して使用され、かっこ内のすべてを省略したときにシステムが取る標準値を意味する。
	横に並べられた複数の項目に対して項目間の区切りを示し、「または」を意味する。
[   ]	ウィンドウのメニューバーから選択するメニュー、コマンド、またはボタンを意味する。

### 常用漢字以外の漢字の使用について

本書では、常用漢字を使用することを基本としていますが、次に示す用語については、常用漢字以外の漢字を使用しています。

罫線（けいせん） 桁（けた） 伝播（でんぱ） 貼り付け（はりつけ）

### 単位表記について

1KB（キロバイト）、1MB（メガバイト）、1GB（ギガバイト）、1TB（テラバイト）はそれぞれ1、024 バイト、1、024<sup>2</sup> バイト、1、024<sup>3</sup> バイト、1、024<sup>4</sup> バイトです。

2017 年 4 月 初 版

2018 年 10 月 第 2 版

2019 年 4 月 第 3 版

2020 年 4 月 第 4 版

2021 年 4 月 第 5 版

2022 年 4 月 第 6 版

2023 年 4 月 第 7 版

2025 年 6 月 第 8 版

## ご注意

- (1) 本書の内容の一部または全部を無断転載することは禁止されています。
- (2) 本書の内容に関しては将来予告なしに変更することがあります。
- (3) 本書は内容について万全を期して作成いたしました但、万一ご不審な点や誤り、記載もれなどお気づきのことがありましたら、ご連絡ください。
- (4) 運用した結果の影響については、(3)項にかかわらず責任を負いかねますのでご了承ください。

Microsoft、Windows、Windows Server 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。

Oracle、Pro\*COBOL は、米国 Oracle Corporation の登録商標です。

Linux は Linus Torvalds の米国およびその他の国における商標または登録商標です。

Red Hat、Red Hat Enterprise Linux は米国 Red Hat , Inc . の米国およびその他の国における商標または登録商標です。

その他、本書に記載の製品名は、各社の商標または登録商標です。

## 輸出する際の注意事項

本製品（ソフトウェア）は日本国内仕様であり、外国の規格等には準拠していません。

本製品は日本国外で使用された場合、当社は一切責任を負いかねます。また、当社は本製品に関して海外での保守サービスおよび技術サポート等は行っておりません。

## 備考

本書は以下の 64 ビット版オペレーティングシステムに対応しています。

製品名	OS
COBOL Enterprise Edition for Windows COBOL Standard Edition	Windows Server 2019
	Windows Server 2022
	Windows 10
	Windows 11
COBOL Enterprise Edition for Linux	Windows 10
	Windows 11
	Red Hat Enterprise Linux 7
	Red Hat Enterprise Linux 8

## 文字コードについて

本製品で扱う文字コードはシフト JIS のみです。

# 本書の記述について

本書では、次の記号を用いて、Enterprise Edition for Windows 版、Enterprise Edition for Linux 版、および Standard Edition 版の機能差異を説明します。

また、Enterprise Edition for Windows 版の標準動作モードおよび SE 動作モードの機能差異を説明します。

標準動作モードは、ACOS-4 COBOL85 や COBOL85 MFA からの移行に適したモードであり、Enterprise Edition for Windows 版の動作既定値です。

SE 動作モードは、A-VX COBOL85 や COBOL85 Pro からの移行に適したモードであり、Standard Edition 版と動作互換です。

【Enterprise(標準)】記号は、次の機能を表します。

Enterprise Edition for Windows 版の標準動作モードの機能

【Enterprise(SE)】記号は、次の機能を表します。

Enterprise Edition for Windows 版の SE 動作モードの機能

【Linux】記号は、次の機能を表します。

Enterprise Edition for Linux 版の機能

【Standard】記号は、次の機能を表します。

Standard Edition 版の機能

# 目次

1	COBOL 開発環境を利用した開発 .....	1
1.1	プロジェクトによる開発資産の管理 .....	2
1.2	ソリューションによるプロジェクトの管理 .....	3
1.3	開発の流れ .....	5
1.4	既存 COBOL からの移行手順の概要 .....	6
1.4.1	プロジェクトをインポートする .....	8
1.4.2	プロジェクトをまとめてインポートする .....	9
2	操作環境 .....	11
2.1	COBOL 開発環境の起動 .....	12
2.1.1	使用するファイル .....	12
2.1.2	ファイル名の規則 .....	14
2.1.3	エラー位置づけの規則 .....	15
2.1.4	起動時の状態 .....	16
2.2	COBOL 開発環境の終了 .....	17
2.3	COBOL アプリケーションの開発手順 .....	18
2.3.1	COBOL アプリケーションの開発 .....	18
2.3.2	COBOL ライブラリの開発 .....	19
2.3.3	各手順の詳細 .....	19
2.4	COBOL 開発環境の画面構成 .....	20
2.5	COBOL 開発環境のウィンドウ .....	21
2.5.1	編集ウィンドウ .....	21
2.5.2	ファイルビュー .....	21
2.5.3	ツールバー .....	22
2.5.4	ステータスバー .....	23
2.5.5	ダイアログボックス .....	23
2.5.6	出力ウィンドウ .....	23
2.6	COBOL 開発環境のメニュー .....	25

2.6.1	メニューバー .....	25
2.6.2	右クリックメニュー .....	30
3	ソリューション .....	34
3.1	ソリューションの概要 .....	35
3.2	ソリューションの新規作成 .....	36
3.3	ソリューションを開く .....	38
3.4	ソリューションを閉じる .....	40
3.5	ソリューションへの追加 .....	41
3.5.1	新しいプロジェクトの追加 .....	41
3.5.2	既存のプロジェクトの追加 .....	42
3.6	ソリューションの保存 .....	44
3.6.1	上書き保存 .....	44
3.6.2	名前を付けて保存 .....	44
3.6.3	すべてを保存 .....	46
3.7	ソリューション名の変更 .....	47
3.8	エクスプローラーでフォルダーを開く .....	48
3.9	プロジェクトの操作 .....	49
3.9.1	名前の変更 .....	49
3.9.2	削除 .....	49
3.9.3	エクスプローラーでフォルダーを開く .....	50
4	プロジェクト .....	51
4.1	プロジェクトの定義 .....	52
4.2	プロジェクトの新規作成 .....	53
4.3	プロジェクトを開く .....	54
4.4	プロジェクトへの追加 .....	58
4.4.1	新しいファイルの追加 .....	58
4.4.2	既存のファイルの追加 .....	59
4.5	プロジェクトの保存 .....	61



4.5.1	上書き保存 .....	61
4.5.2	名前を付けて保存 .....	61
4.5.3	すべてを保存 .....	62
4.6	オプション設定 .....	64
4.6.1	全般 .....	69
4.6.2	COBOL .....	72
4.6.3	リンカ .....	84
4.6.4	Pro*COBOL .....	89
4.6.5	【Standard】【Enterprise(標準)】【Enterprise(SE)】リソース .....	91
4.6.6	C/C++ .....	92
4.6.7	オプション製品 .....	93
4.6.8	【Standard】【Enterprise(標準)】【Enterprise(SE)】ディレクトリ .....	97
4.6.9	転送 .....	98
4.6.10	【Enterprise(標準)】【Linux】ACOS 互換製品 .....	103
4.6.11	コマンドライン .....	105
4.6.12	入力文字の制限一覧 .....	106
4.7	ソースファイルのオプション設定 .....	110
4.7.1	メインプログラム .....	110
4.7.2	EXTERNAL 句の参照と定義 .....	111
4.7.3	依存ファイル .....	111
4.8	ファイルの操作 .....	114
4.8.1	名前の変更 .....	114
4.8.2	削除 .....	114
4.8.3	エクスプローラーでフォルダーを開く .....	115
4.9	プロジェクトの排他制御 .....	116
5	ファイルの編集 .....	117
5.1	編集ウィンドウの概要 .....	118
5.2	ファイルの新規作成 .....	120

5.3 ファイルを開く .....	122
5.4 ファイルを保存する .....	127
5.4.1 上書き保存 .....	127
5.4.2 名前を付けて保存 .....	127
5.4.3 すべてを保存 .....	129
5.5 ファイルを閉じる .....	130
5.6 ファイルの印刷 .....	131
5.7 編集操作 .....	133
5.7.1 ウィンドウをスクロールするには .....	133
5.7.2 カーソルを移動するには .....	133
5.7.3 文字を編集するには .....	133
5.7.4 行を挿入 / 削除するには .....	134
5.7.5 行を分割 / 結合するには .....	134
5.7.6 文字列範囲を選択するには (キー操作) .....	134
5.7.7 文字列範囲を選択するには (マウス操作) .....	134
5.7.8 テキスト全体を選択するには .....	135
5.7.9 行範囲を選択するには .....	135
5.7.10 矩形範囲を選択するには .....	135
5.7.11 選択した範囲を削除するには .....	135
5.7.12 選択した範囲を切り取るには .....	136
5.7.13 選択した範囲をコピーするには .....	136
5.7.14 クリップボードの内容を貼り付けるには .....	136
5.7.15 ドラッグ&ドロップで範囲を移動するには .....	137
5.7.16 [Tab]キーでカーソルを移動するには .....	137
5.7.17 編集操作を元に戻すには .....	137
5.7.18 元に戻した編集を復元するには .....	137
5.8 文字列の検索 .....	138
5.8.1 文字列を検索するには .....	138

5.8.2 次の文字列を検索するには .....	139
5.8.3 検索する文字を貼りつけるには .....	139
5.9 文字列の置換 .....	140
5.9.1 文字列を置換するには .....	140
5.10 行番号位置づけ .....	142
5.11 部分書き出し .....	143
5.12 追加読み込み .....	144
5.13 キーワード補完 .....	145
5.14 再行番号付け .....	147
5.15 編集ウィンドウの動作変更 .....	148
5.16 文字列の一括検索 .....	152
5.16.1 文字列を一括検索するには .....	152
5.16.2 検索した文字列に位置づけるには .....	155
5.16.3 検索を中止するには .....	155
6 アプリケーションの生成 .....	156
6.1 アプリケーションを生成する .....	157
6.1.1 プロジェクト単位で生成する .....	159
6.1.2 エラーに位置づけるには .....	160
6.1.3 ソリューション全体で生成する .....	161
6.2 アプリケーションを再生成する .....	162
6.2.1 プロジェクト単位で再生成する .....	162
6.2.2 ソリューション全体で再生成する .....	163
6.3 アプリケーションを削除する .....	164
6.3.1 プロジェクト単位で削除する .....	164
6.3.2 ソリューション全体で削除する .....	164
6.4 バッチビルド .....	165
6.5 アプリケーションを実行する .....	167
6.6 【Linux】サーバー接続 .....	168

6.6.1	接続の設定 .....	168
6.6.2	ログイン .....	170
6.6.3	ログアウト .....	172
6.7	転送機能 .....	173
6.7.1	ビルド成果物のコピー .....	174
6.7.2	【Linux】ビルド成果物のクライアントへの転送 .....	174
7	ツール .....	175
7.1	既存 COBOL 資産のインポート .....	176
7.2	外部ツールの登録 .....	195
7.3	【Standard】COBOL GUI View Generator .....	197
7.4	【Standard】画面節移行ファイルの作成 .....	198
7.5	【Enterprise(標準)】【Linux】プリコンパイラのビルド設定 .....	200
7.6	データレコード情報ファイルの作成 .....	201
8	付録 .....	203
	付録 A 用語解説 .....	204
	付録 B 【Linux】ビルドサーバーの設定 .....	207

---

# 1

## COBOL 開発環境を利用した開発

この章では、COBOL 開発環境を利用した開発について説明します。

---

## 1.1 プロジェクトによる開発資産の管理

---

プログラムの開発作業を効率的に進めるために、COBOL 開発環境では、「ソリューション」や「プロジェクト」という概念で、ソースファイルなど複数の資源を管理します。ソリューションとは、複数のプロジェクトを一括して管理するための概念です。ソリューションについては、「1.2 ソリューションによるプロジェクトの管理」を参照してください。プロジェクトとは、アプリケーション開発を統合的に行うための概念です。アプリケーションの作成に必要な資源と、翻訳時オプションなどのあらゆる情報を一括して管理します。

ソリューションやプロジェクトを定義することで、複数のソースファイルのコンパイルから実行可能ファイルの作成までを一括して実行できます。また、プロジェクト内のファイル間の依存関係も管理でき、必要なファイルだけを対象とし、効率良くプログラムをコンパイルできます。

開発資産をプロジェクトとして管理するために次に示すような定義をします。

- ・ プロジェクトを構成するソースファイル
- ・ プロジェクトを格納するソリューション
- ・ コンパイル・リンク時に使用するツールのオプションや環境情報
- ・ コンパイル・リンク時に必要となるプログラム間の依存関係

これらの情報はプロジェクトごとに管理されます。

【Linux】COBOL の開発資産は、Windows クライアント上で管理します。ソリューション / プロジェクトのビルド時に Linux サーバーに開発資産を転送し、Linux サーバー上でアプリケーションを作成します。

### **注意事項**

- ・ UNC パスでソリューションファイルおよびプロジェクトファイルを開くと、アプリケーションの生成が正しく行えない場合があります。ネットワークドライブに割り当てて使用してください。

## 1.2 ソリューションによるプロジェクトの管理

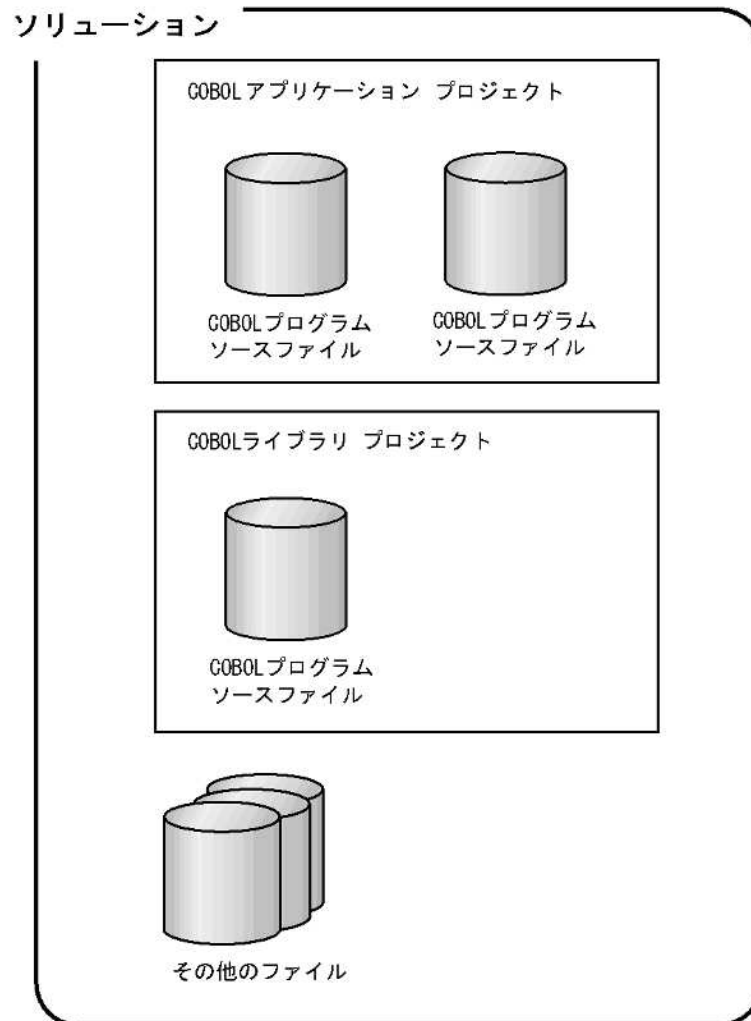
---

ソリューションとは、複数のプロジェクトを一括して管理するための概念です。COBOL開発環境を使うと、実行可能ファイルやライブラリごとに必要な複数のプロジェクトを、ソリューションで一元的に管理できます。

ソリューションの活用事例を次に示します。

- ライブラリ用プロジェクトと、そのライブラリを使用するアプリケーション用プロジェクトの一括管理

ソリューションおよびプロジェクトの概念図を次に示します。



ソリューションの定義情報、およびソリューションに含まれるプロジェクトの定義情報は、すべてソリューション定義ファイル ( .neccblsln ) に格納されます。

プロジェクトおよびソリューションは、COBOL 開発環境のファイルビューで管理します。

ファイルビューについては、「2.5.2 ファイルビュー」を参照してください。

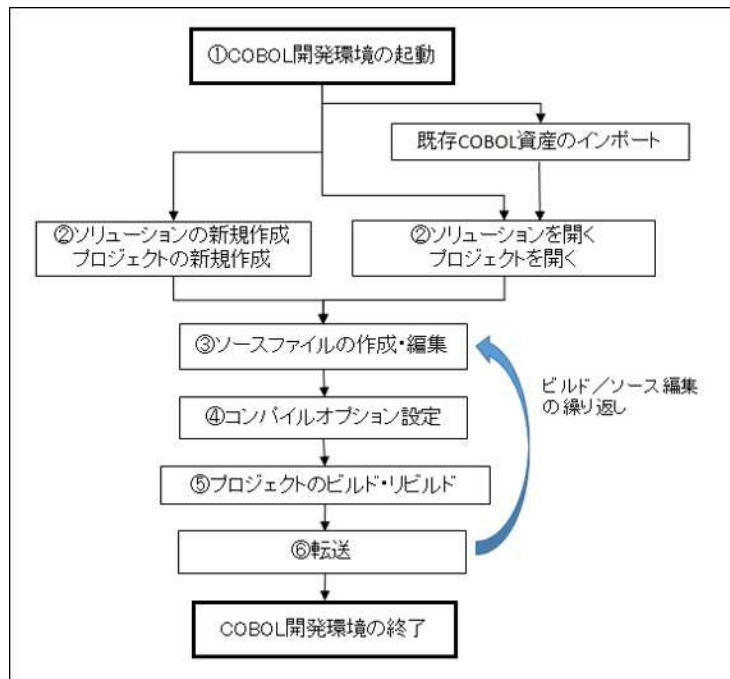
#### **注意事項**

- ・UNC パスでソリューションファイルおよびプロジェクトファイルを開くと、アプリケーションの生成が正しく行えない場合があります。ネットワークドライブに割り当てて使用してください。



## 1.3 開発の流れ

COBOL 開発環境での、COBOL アプリケーションは以下の流れで開発します。



開発フロー	マニュアル記載場所
COBOL 開発環境の起動	2.1 COBOL 開発環境の起動
プロジェクトを開く <b>新規作成</b> プロジェクトの新規作成 <b>既存のファイルを開く</b> ソリューションを開く プロジェクトを開く <b>既存 COBOL からの移行</b> 既存 COBOL 資産のインポート プロジェクトを開く	4.2 プロジェクトの新規作成 3.3 ソリューションを開く 4.3 プロジェクトを開く 7.1 既存 COBOL 資産のインポート 4.3 プロジェクトを開く
ソースファイルの作成・編集	5 ファイルの編集
コンパイルオプション設定	4.6 オプション設定
プロジェクトのビルド・リビルド	6 アプリケーションの生成
転送	6.7 転送機能

## 1.4 既存 COBOL からの移行手順の概要

---

COBOL Enterprise Edition for Windows と COBOL Standard Edition では、Open COBOL Factory 21/COBOL85 Pro (以降、Pro と表記) と Open COBOL Factory 21/COBOL85 MFA (以降、MFA と表記) の資産を移行するために、既存の開発プロジェクトを変換するインポートツールを用意しています。

既存 COBOL プロジェクトファイルから、インポートツールを使って生成した COBOL プロジェクトで、開発を行ってください。

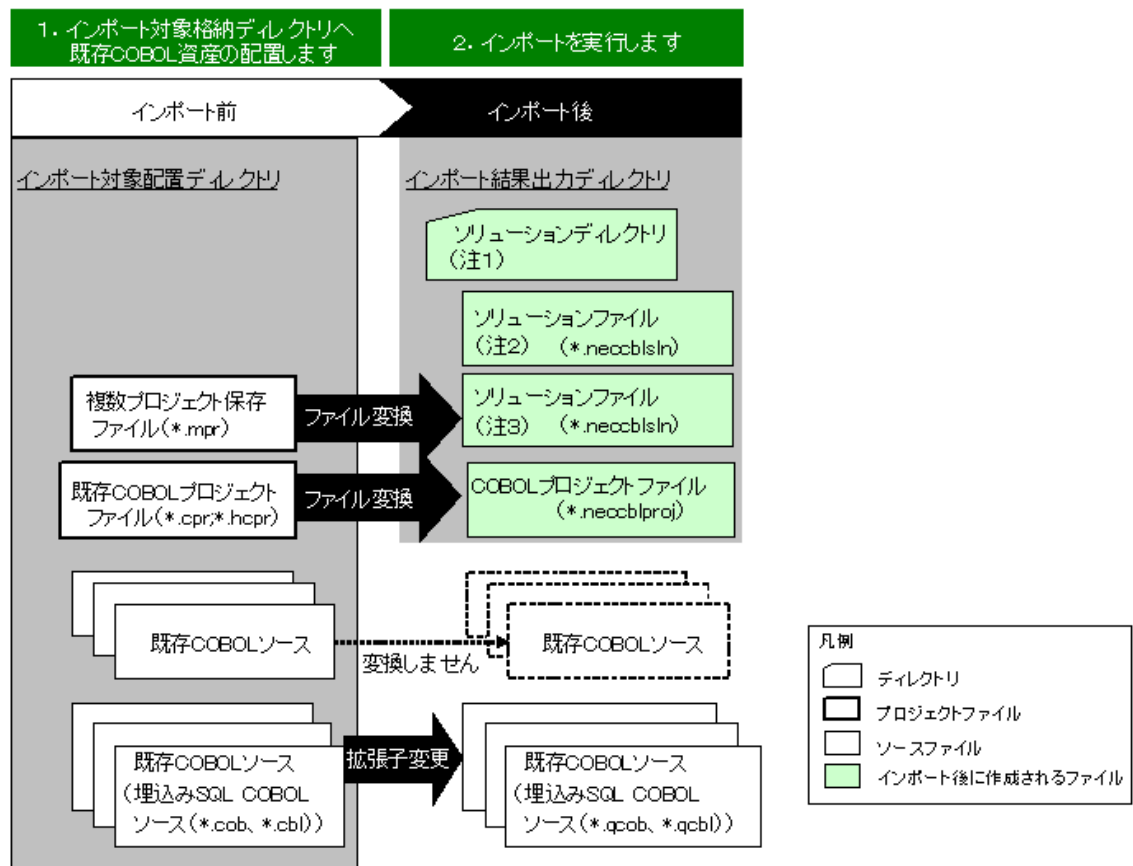
詳細は「7.1 既存 COBOL 資産のインポート」を参照してください。

### 注意事項

- ・ COBOL Enterprise Edition for Linux にインポートツールはありません。

インポートの流れは以下のようになります。

- 1 . インポート対象配置ディレクトリへ既存の COBOL 資産を配置します。
- 2 . インポートを実行します。実行すると、ソリューションディレクトリが作成されます。ソリューションディレクトリ配下には、COBOL プロジェクトファイルが作成されます。  
インポート対象配置ディレクトリに複数プロジェクト保存ファイル (\*.mpr) が存在する場合、ソリューションファイルが作成されます。  
インポート対象配置ディレクトリに埋込み SQL COBOL ソースファイルが存在する場合、拡張子変更したソースファイルが作成されます。
- 3 . COBOL プロジェクトファイルを保存します。ソリューションファイルが存在しない場合は作成され、保存されます。



(注1)

インポートのオプション「ソリューションのディレクトリを作成してインポート」が指定されている場合、インポート時に作成されます。

(注2)

インポートツールのオプション「ソリューションファイル作成」が指定されている場合、インポート時に作成されます。

オプションの詳細は「7.1 既存 COBOL 資産のインポート」を参照してください。

このファイルが存在しない場合、COBOL プロジェクトファイル保存時に作成されます。

(注3)

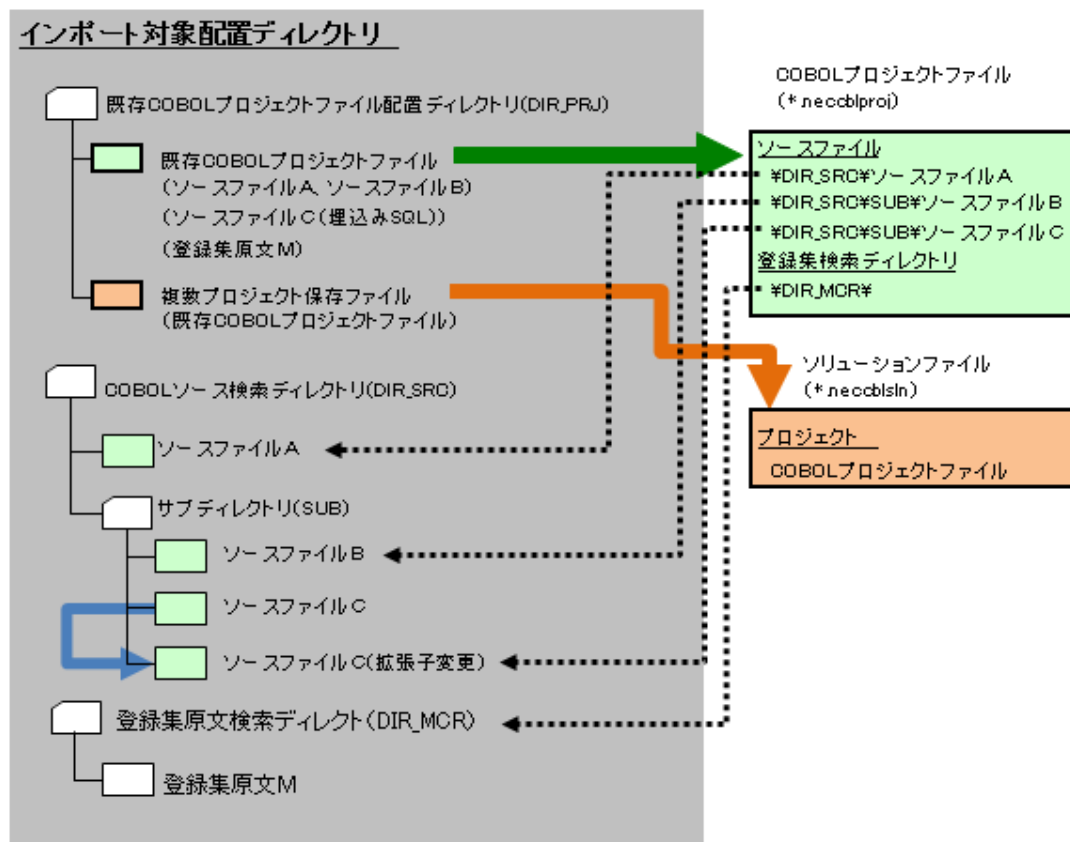
インポート時に複数プロジェクト保存ファイルから作成されます。

注2のソリューションファイルとは別のファイルになります。

### 1.4.1 プロジェクトをインポートする

インポートを行う前に、インポート対象配置ディレクトリに既存 COBOL 資産を配置しておく必要があります。ソースファイル等の配置については、インポート後も継続して参照されますので、十分に検討することが必要です。

COBOL プロジェクトファイルは以下のように変換されます。

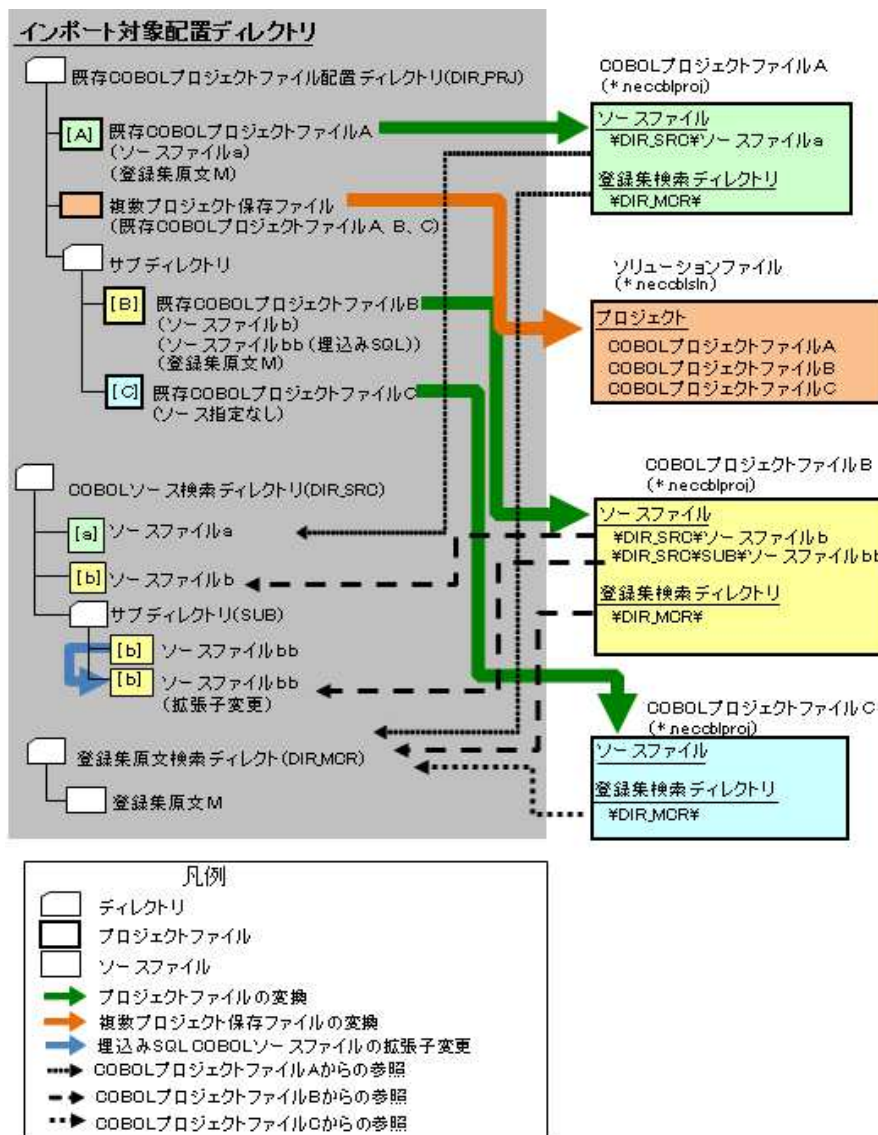


## 1.4.2 プロジェクトをまとめてインポートする

複数の既存 COBOL プロジェクトファイルは一括して変換されます。この場合、インポート対象配置ディレクトリ配下のディレクトリに分類して、配置してください。プロジェクト間でソースファイル等を混同させたくない場合には、指定ディレクトリ配下にサブディレクトリを作成して配置します。

### インポート対象配置ディレクトリ

- ・ 既存COBOLプロジェクトファイル配置ディレクトリ
- ・ COBOLソース検索ディレクトリ
- ・ 登録集原文検索ディレクトリ
- ・ 【Enterprise(標準)】COBOL/Sマクロ原文ファイル検索ディレクトリ
- ・ 【Enterprise(標準)】GMPマクロ検索ディレクトリ



---

## 2 操作環境

この章では、COBOL 開発環境を使ってプログラムを開発するときの基本的な操作について説明します。

---

## 2.1 COBOL 開発環境の起動

開発資産をプロジェクトとして管理したり、各種ツールを使って編集したりするために COBOL 開発環境を起動します。

起動方法として、以下の方法があります。

- ・ スタートメニューから「COBOL 開発環境」を選択して起動します。
- ・ ソリューションファイルをダブルクリックします。

【Enterprise(標準)】 【Enterprise(SE)】

Enterprise Edition をインストールした場合、COBOL 開発環境はインストール時に選択したモード（標準動作モードまたは SE 動作モード）で動作します。

### 2.1.1 使用するファイル

COBOL の開発環境で使用するファイルの一覧を以下に示します。

拡張子	ファイル種別	内容
.cbl 他	COBOL ソースファイル	COBOL 原始プログラムを格納するファイル。
.cbl 他	登録集原文ファイル	登録集原文を格納するファイル。
.scn	画面情報ファイル	画面定義の情報を格納するファイル。
.ssi	画面節移行ファイル	画面制御機能を View Generator で GUI に移行するための情報を格納するファイル。
.c .cpp .h	C/C++ファイル	C/C++言語で記述されたソースファイルとインクルードファイル。
.rc	リソースファイル	リソース情報を保存したファイル。
.neccb1sln	ソリューションファイル	ソリューションに関する情報を格納するファイル。（COBOL 以外の言語のプロジェクトを含むことも可能）
.neccb1proj	プロジェクトファイル	COBOL のプロジェクト構成を格納するファイル。
.neccb1suo	エディタ状態ファイル	編集ウィンドウの表示状態を格納するファイル。
.obj .o	オブジェクトファイル	オブジェクトコードを格納するファイル。
.exe	実行可能ファイル	実行可能になったプログラムを格納するファイル。
.dll	ライブラリ（DLL）ファイル	他の実行可能ファイルやライブラリから呼び出し可能な処理を格納する動的ライブラリファイル。
.lib	ライブラリ（LIB）フ	他の実行可能ファイルやライブラリから呼び



	ファイル	出し可能な処理を格納する静的ライブラリファイル。
.so	動的ライブラリ	他の実行可能ファイルやライブラリから呼び出し可能な処理を格納する動的ライブラリファイル。
.a	静的ライブラリ	他の実行可能ファイルやライブラリから呼び出し可能な処理を格納する静的ライブラリファイル。
.txt	テキストファイル	エディターで参照、更新が可能なテキストファイル。

注 ) 目的に応じて次の拡張子を使用します。

固定形式で書かれた原始プログラムをコンパイルする場合

.cob、.cbl、.pco、

.cbx、.qcob、.qcbf

【Enterprise(標準)】 【Linux】

.cblm、.cble、.cblme、.pcom、.pcoe、.pcome、.cobm、

.cblss、.cblmss、.cblsse、.pcoss、.cblmsse、.pcomss、

.pcosse、.pcomsse

自由形式で書かれた原始プログラムをコンパイルする場合

.cbf

COBOL ソースファイル、登録集原文ファイルおよびそれらの拡張子については、「COBOL プログラミングの手引」を参照してください。

### **注意事項【Linux】**

・ 拡張子が大文字の場合に、COBOL 開発環境では小文字に置き換えたファイル種別として扱われますが、アプリケーションの生成時には小文字として扱われません。アプリケーション生成時に使用するファイルの拡張子は、COBOL コンパイラまたは OS の仕様に合わせて小文字にしてください。

## 2.1.2 ファイル名の規則

COBOL 開発環境では、次の規則に従ったファイル名を指定しなければなりません。

### ファイル名の規則

- ・ファイル名には、次の文字は使用できません。

すべてのファイル : / ? : ¥ \* " < > | # \$ ^ ; ! {

プロジェクトファイル/ソリューションファイル :

/ ? : ¥ \* " < > | # \$ ^ ; % [ ,

- ・ファイル名の長さの上限は、パス名を含めて 258 桁です。
- ・Windows の予約語 ( " nul "、" aux "、" con "、" com1 "、" lpt1 " ) などは、ファイル名に使用できません。
- ・ファイル名およびディレクトリ名に、makefile で特別な意味を持つ「 % ^ & \$ 」などの文字を使用すると、コンパイルなどが正しく行えない場合があります。使用できない文字は makefile の仕様に則ります。

### 注意事項

- ・UNC パスでソリューションファイルおよびプロジェクトファイルを開くと、アプリケーションの生成が正しく行えない場合があります。ネットワークドライブに割り当てて使用してください。

### 【Linux】

- ・サーバーのファイル名およびディレクトリ名には、上記に追加して次の文字は指定できません。

すべてのファイル : % = ` ( ) & '

プロジェクトファイル/ソリューションファイル : `

ビルド成果物 : ` ( ) & ' ,

- ・サーバーのファイル名およびディレクトリ名は、大文字と小文字は区別されます。
- ・ファイル名やディレクトリ名にショートファイル名を指定できません。アプリケーションの生成が正しく行えない場合があります。
- ・システムのロケールが SJIS の環境でのみ、ファイル名とディレクトリ名に日本語を使

用できます。

- ・ファイル名およびディレクトリ名の日本語文字の中に、使用できない文字のコードを含む場合も規則に則ります。

### 2.1.3 エラー位置づけの規則

アプリケーションの生成により出力されたエラーメッセージの対象ソースファイルへの位置づけの規則を、ファイルの種類ごとに説明します。

#### (1) COBOL ファイル

編集ウィンドウにエラー対象の COBOL ファイルを開いていない場合は、コンパイラが参照した COBOL ファイルが編集ウィンドウとして開き、対象の行にカーソルが位置づきます。

登録集原文ファイルの中の記述でエラーが出力された場合は、COBOL コンパイラと同じ規則で拡張子を付加し、登録集原文ファイルを探して開きます。

#### (2) SQL 文を記述した COBOL ファイル

SQL 文が記述された COBOL ファイルの場合、SQL 文を展開するプリコンパイラが出力したエラーは位置づけの対象になります。

プリコンパイラが生成した COBOL ファイルをコンパイルして出力されたエラーは、(1)の位置づけの対象になり、編集ウィンドウに[読み取り専用]で開かれます。

エラーに対する修正は、ビルド対象のファイルに実施してください。

#### (3) 【Enterprise(標準)】【Linux】ACOS 互換製品を使用する COBOL ファイルと COBOL/S ファイル

ACOS 互換製品を使用する COBOL ファイルと COBOL/S ファイルの場合、ACOS 互換製品のプリコンパイラが出力したエラーは位置づけの対象になります。

プリコンパイラが生成した COBOL ファイルをコンパイルして出力されたエラーは、(1)の位置づけの対象になり、編集ウィンドウに[読み取り専用]で開かれます。

エラーに対する修正は、ビルド対象のファイルに実施してください。

ただし、直接入力で指定した ACOS 互換製品のオプションは、コピー原文ファイルの検索先として使用されません。

#### **(4) 外部行番号での位置づけ**

COBOL のコンパイルオプション「リスト内に印字される行番号を外部行番号で出力する [-Ce]」を指定すると、外部行番号によって位置づけます。

位置づける行は、COBOL ファイルの先頭から検索し最初に一致した外部行番号です。そのため、外部行番号が一意でない場合は、実際にエラーが出力された行に位置づかないことがあります。

自由形式の COBOL ファイルは対象外です。

画面節移行機能のコンパイルでは、外部行番号での位置づけは対象外です。

#### **注意事項**

・プリコンパイラが生成したファイルをコンパイルして出力されたエラーの位置づけでは、ビルドの対象になったファイルのみが編集ウィンドウに開かれます。ソースファイルを修正しないで再度ビルドを実行すると、プロジェクトに登録されたファイルが開かれないことがあります。ソースファイルを編集するには「5.3 ファイルを開く」の操作でファイルを開いてください。

### **2.1.4 起動時の状態**

COBOL 開発環境を起動すると、同じ利用者が前回 COBOL 開発環境を終了した時と同じ表示位置に同じ大きさで表示されます。

#### **注意事項**

・ユーティリティ起動時の既定のディレクトリは、システムのカレントディレクトリです。

## 2.2 COBOL 開発環境の終了

---

COBOL 開発環境を終了するには、[ ファイル ] - [ アプリケーションの終了 ] コマンドを選択します。

編集ウィンドウの内容を更新している場合は、保存するかを確認するメッセージが表示されます。

また、プロジェクトやソリューションの構成や設定を変更している場合にも、保存するかを確認するメッセージが表示されます。

COBOL 開発環境の終了時には、現在の COBOL 開発環境のウィンドウの表示位置と大きさ、最近開いたファイルやプロジェクトが保存され、次回の COBOL 開発環境の起動時には、それらの情報が復元されます。

### 【Linux】

COBOL 開発環境の終了時にサーバーにログイン中の場合は、ログアウト処理を実施します。

## 2.3 COBOL アプリケーションの開発手順

COBOL 開発環境では、大きく 6 つのアプリケーションが開発できます。

- ・ COBOL バッチ型アプリケーション
- ・ COBOL ライブラリ

【Standard】 【Enterprise(SE)】

- ・ COBOL 対話型アプリケーション

【Standard】

- ・ COBOL GUI アプリケーション
- ・ COBOL 画面節移行 GUI アプリケーション

これらのアプリケーション種別に対して、開発手順の概要を示します。

### 2.3.1 COBOL アプリケーションの開発

アプリケーションを新規作成する場合、下表のとおりプロジェクトの種類の選択（3.2）、プロジェクトのオプションの設定（4.6）を行います。

開発環境の動作モードによって、作成できるアプリケーションの種別は異なります。

アプリケーションの種別	プロジェクトの種類	[ オプション設定 ] 画面
COBOL バッチ型 アプリケーション	COBOL バッチ型 アプリケーション	設定の変更なし。
COBOL 対話型 アプリケーション	COBOL 対話型 アプリケーション	設定の変更なし。
COBOL GUI アプリケーション	COBOL 対話型 アプリケーション	[ オプション製品 ] タブ [ COBOL GUI アプリケーション を作成する ] をチェックする。
COBOL 画面節移行 GUI アプリケーション	COBOL 対話型 アプリケーション	[ オプション製品 ] タブ [ COBOL GUI アプリケーション を作成する ]、[ 画面節の実行 で COBOL GUI を使用する ] をチ ェックする。

アプリケーションは、以下の流れで開発します。

プロジェクトの新規作成 / プロジェクトを開く 原始プログラムの作成 / 編集 プロジェクトのオプション設定 ビルド

以後、原始プログラムの編集、ビルドを繰り返します。

## 2.3.2 COBOL ライブラリの開発

COBOL で動的ライブラリまたは静的ライブラリを作成できます。ライブラリは、以下の流れで開発します。

プロジェクトの新規作成 / プロジェクトを開く 原始プログラムの作成 / 編集 プロジェクトのオプション設定 ビルド

以後、原始プログラムの編集、ビルドを繰り返します。

【Enterprise(標準)】【Linux】

COBOL Enterprise Edition Developer/Compiler API ライセンスがインストールされている場合、「COBOL REST API」を作成できます。

開発の方法は、「COBOL ユーザーズガイド 4.5.3 COBOL REST API の生成方法」を参照してください。

## 2.3.3 各手順の詳細

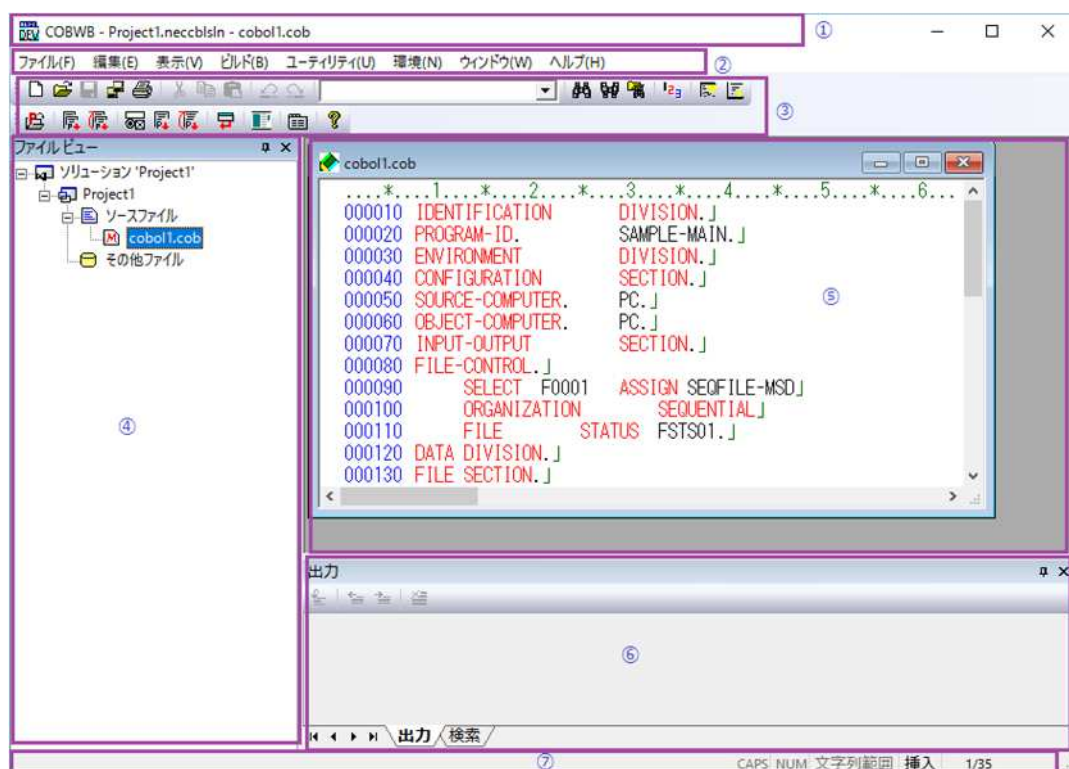
各手順の詳細は、以下を参照してください。

- ・プロジェクトの作成 / プロジェクトを開く / オプション設定・・・4 章
- ・プログラムの作成 / 編集・・・5 章
- ・ビルド・・・6 章

## 2.4 COBOL 開発環境の画面構成

COBOL 開発環境は、以下に示す画面項目から構成されています。

タイトルバー	タイトルを表示する領域
メニューバー	メニューが表示されている領域
ツールバー	COBOL 開発環境の機能へアクセスするボタン群のまとまった領域
ファイルビュー	プロジェクトを管理するためのウィンドウエリア
編集ウィンドウ	プロジェクトのファイルを編集するウィンドウ
出力ウィンドウ	ビルドの結果や一括検索の結果を表示するためのウィンドウ
ステータスバー	COBOL 開発環境の状況を示す領域
ダイアログボックス	COBOL 開発環境でメッセージを表示したり、検索を行ったりするためのウィンドウ



それぞれのウィンドウについては、次節以降にて、説明します。



## 2.5 COBOL 開発環境のウィンドウ

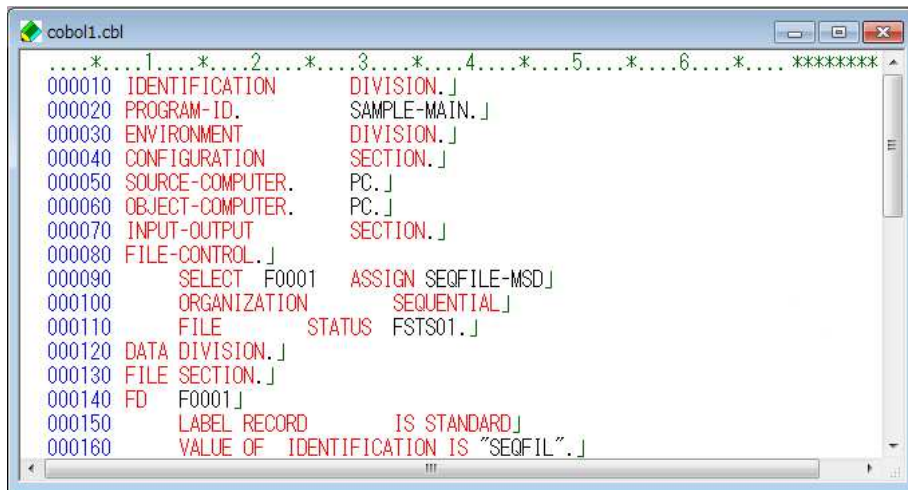
---

ここでは、COBOL 開発環境のウィンドウの種類や構成について説明します。

### 2.5.1 編集ウィンドウ

編集ウィンドウは、プログラム等の編集に用いるウィンドウです。

COBOL の原始プログラムを編集するのに適した操作が可能です。

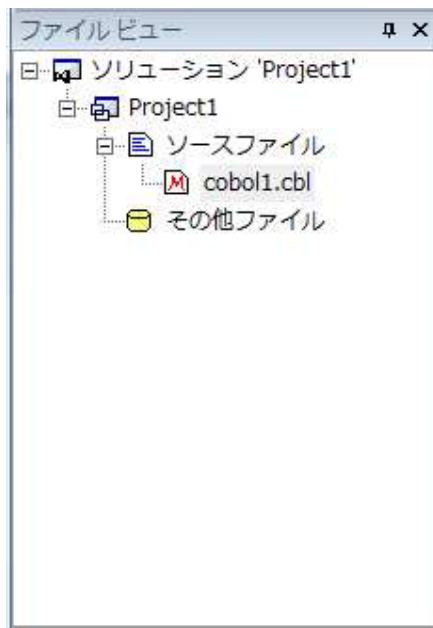
A screenshot of a COBOL editor window titled 'cobol1.cbl'. The window displays a COBOL program with line numbers 000010 through 000160. The program includes sections for IDENTIFICATION, PROGRAM-ID, ENVIRONMENT, CONFIGURATION, SOURCE-COMPUTER, OBJECT-COMPUTER, INPUT-OUTPUT, and FILE-CONTROL. It also contains a SELECT statement for F0001, an ASSIGN statement for SEQFILE-MSD, and an ORGANIZATION statement for SEQUENTIAL. The FILE section includes an FD statement for F0001, a LABEL RECORD statement, and a VALUE OF IDENTIFICATION statement. The window has a standard Windows-style title bar and a scrollbar on the right side.

```
.....*.....1.....*.....2.....*.....3.....*.....4.....*.....5.....*.....6.....*.....*****
000010 IDENTIFICATION DIVISION.
000020 PROGRAM-ID. SAMPLE-MAIN.
000030 ENVIRONMENT DIVISION.
000040 CONFIGURATION SECTION.
000050 SOURCE-COMPUTER. PC.
000060 OBJECT-COMPUTER. PC.
000070 INPUT-OUTPUT SECTION.
000080 FILE-CONTROL.
000090 SELECT F0001 ASSIGN SEQFILE-MSD
000100 ORGANIZATION SEQUENTIAL
000110 FILE STATUS FST01.
000120 DATA DIVISION.
000130 FILE SECTION.
000140 FD F0001
000150 LABEL RECORD IS STANDARD
000160 VALUE OF IDENTIFICATION IS "SEQFIL".
```

### 2.5.2 ファイルビュー

ファイルビューは、プロジェクトおよびソリューションの管理を行うエリアです。

ソリューションを構成するプロジェクトおよびファイルの関連をツリー形式で整理して表示することで、各ファイルに対して容易にアクセスすることが可能です。



ファイルビューへファイルをドラッグアンドドロップした場合には、ドロップした場所によって、以下のようになります。

- ・ プロジェクト配下：プロジェクト資産として追加されます。
- ・ [ソースファイル] 配下：ソースファイルとして追加されます。
- ・ [その他ファイル] 配下：コンパイル対象外のファイルとして追加されます。

#### **注意事項**

- ・ COBOL 開発環境を「管理者として実行」している場合は、ファイルをドラッグアンドドロップすることができません。

### **2.5.3 ツールバー**

ツールバーは、メニューバーの下に表示されます。ツールバーの表示と非表示を切り替えるには、[表示] - [ツールバー] の下の [編集ツールバー] または [プロジェクトツールバー] コマンドを選択します。

マウスポインタをボタンの上に移動させると、ツールバーボタンの用途についての短い説明をポップアップウィンドウで表示されます。

アクティブな編集ウィンドウで文字列の検索または逆検索を行うことができます。

[ 文字列の検索 ] ボタンの左側のドロップダウンリストに文字列を入力し、[ 文字列の検索 ] ボタンまたは [ 文字列の逆検索 ] ボタンをクリックします。

COBOL 開発環境の動作モードによって、ツールバーに表示されるアイコンは異なります。

## 2.5.4 ステータスバー

ステータスバーは、COBOL 開発環境の最下行に表示されます。

ステータスバーの左側の部分には、メニューコマンドを選択したときにそれぞれの簡単な説明が表示されます。また、編集ウィンドウがアクティブになっている場合には、ステータスバーの右側の部分に、挿入 / 上書きの入力状態およびカーソル位置の行、列が表示されます。

## 2.5.5 ダイアログボックス

コマンドやボタンの選択によって、ダイアログボックスが表示される場合があります。

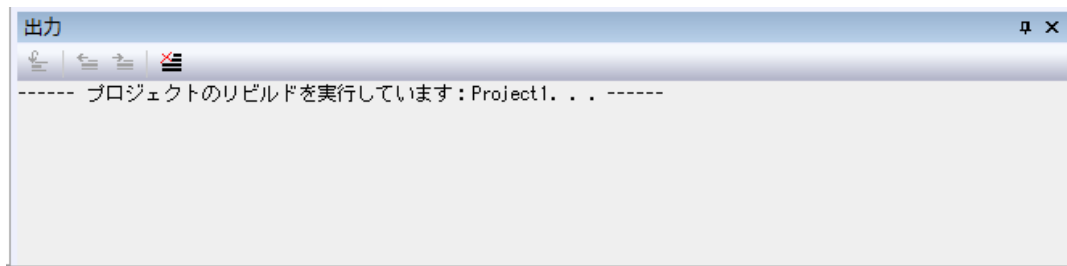
ダイアログボックスの操作方法については、OS 添付のヘルプを参照してください。

## 2.5.6 出力ウィンドウ

COBOL 開発環境での種々の情報が出力される領域です。

出力タブと検索タブで構成されます。

出力タブには、ビルド時のエラーメッセージや、アプリケーションのコンソールへの出力メッセージなどが表示されます。



検索タブには、複数ファイルの一括検索の検索結果が表示されます。



### 出力ウィンドウをクリップボードにコピーするには

- (1) 出力ウィンドウの対象のタブをアクティブにします。
- (2) コピーする部分を選択状態にします。
- (3) 出力ウィンドウ上で右クリックメニューを表示します。
- (4) [ コピー ] 項目を選択します。

### 注意事項【Linux】

・ローケルの設定が SJIS 以外の環境では、サーバー上で表示される日本語文字は出力ウィンドウに正しく表示されません。

## 2.6 COBOL 開発環境のメニュー

---

### 2.6.1 メニューバー

COBOL 開発環境での操作環境は、以下のメニューで構成されます。

- (1) ファイル
- (2) 編集
- (3) 表示
- (4) ビルド
- (5) ユーティリティ
- (6) 環境
- (7) ウィンドウ
- (8) ヘルプ

個々のメニューについて、以下に概要を記し、その詳細は併記した章節番号を参照ください。

#### [ ファイル ] メニュー

ファイルの新規作成、オープン、クローズ、保存、印刷といったファイルに関する機能が集約されたメニューです。

メニュー名	概要	章節
新規作成	プロジェクトやファイルを新規に作成する。	(3.2) (5.2)
開く	ソリューション、プロジェクトやファイルを開く。	(3.3) (4.3) (5.3)
追加	ソリューションへプロジェクトを追加する。	(3.5)
閉じる	現在アクティブなファイルを閉じる。	(5.5)
ソリューションを閉じる	現在アクティブなソリューションを閉じる	(3.4)
選択されたファイルの保存	編集中のファイルを保存する。	(3.6.1) (4.5.1) (5.4.1)
選択されたファイルに名前を付けて保存	編集中のファイルに名前をつけて保存する。	(3.6.2) (4.5.2) (5.4.2)
すべてを保存	開いているファイルをすべて保存する。	(3.6.3)

		(4.5.3) (5.4.3)
印刷	現在アクティブな編集ウィンドウを印刷する。	(5.6)
印刷プレビュー	現在アクティブな編集ウィンドウの印刷プレビュー画面を表示する。	(5.6)
印刷オプションの設定	印刷に関する各種設定を行う。	(5.6)
最近使ったファイル	最近使ったファイルの履歴を表示し、選択されたファイルを開く。	(5.3)
最近使ったプロジェクトとソリューション	最近使ったプロジェクトやソリューションの履歴を表示し、選択されたプロジェクトやソリューションを開く。	(3.3) (4.3)
アプリケーションの終了	COBOL 開発環境を終了する。	(2.2)

#### [ 新規作成 ] サブメニュー

メニュー名	概要	章節
プロジェクト	新規にプロジェクトを作成する。	(3.2)
ファイル	新規にファイルを作成する。	(5.2)

#### [ 開く ] サブメニュー

メニュー名	概要	章節
プロジェクト/ソリューション	プロジェクトまたはソリューションを開く。	(3.3) (4.3)
ファイル	ファイルを開く	(5.3)

#### [ 追加 ] サブメニュー

メニュー名	概要	章節
新しいプロジェクト	新規にプロジェクトをソリューションに追加する。	(3.5.1)
既存のプロジェクト	既存のプロジェクトをソリューションに追加する。	(3.5.2)

#### [ 編集 ] メニュー

開発環境の中でもエディターに関する機能が集約されたメニューです。

メニュー名	概要	章節
元に戻す	最後に行った編集作業を元に戻す。	(5.7.17)
やり直し	[ 元に戻す ] メニューで行った編集作業を再度実行する。	(5.7.18)
切り取り	選択されているテキストを削除し、クリップボ	(5.7.12)

	ードに保存する。	
コピー	選択されているテキストをクリップボードに保存する。	(5.7.13)
貼り付け	クリップボードに保存されているテキストをカーソル位置に貼り付ける。	(5.7.14)
削除	選択されているテキストを削除する。	(5.7.11)
すべてを選択	テキスト全体を範囲選択する。	(5.7.8)
部分書き出し	選択した行範囲をファイルに保存する。	(5.11)
追加読み込み	ファイルの内容をカーソル行の次に追加する。	(5.12)
行番号位置づけ	指定された行番号のテキストにカーソルを位置づける。	(5.10)
先頭	テキストの先頭にカーソルを移動する。	(5.7.2)
末尾	テキストの末尾( [ EOF ] の行)にカーソルを移動する。	(5.7.2)
検索	指定された文字列をアクティブな編集ウィンドウのカーソル位置から指定の向きに検索する。	(5.8)
置換	指定された置換前文字列をアクティブな編集ウィンドウのカーソル位置から指定の向きに検索し、置換後文字列に置換する。	(5.9)
キーワード補完	入力途中の文字列を COBOL キーワードに補完する。	(5.13)
文字列の一括検索	複数ファイルの文字列を検索する。	(5.16)
再行番号付け	指定された開始行番号と増分値で、アクティブな編集ウィンドウ上の COBOL 一連番号領域の行番号を振り直す。	(5.7.14)
カスタマイズ	編集ウィンドウの動作を変更する。	(5.15)

## [ 表示 ] メニュー

ツールバーや COBOL 開発環境内の各ウィンドウの表示 / 非表示を制御するメニューです。

ツールバー	各ツールバーの表示 / 非表示を切り替える。	(2.5.3)
ステータスバー	ステータスバーの表示 / 非表示を切り替える	(2.5.4)
ファイルビュー	ファイルビュー領域の表示 / 非表示を切り替える。	(2.5.2)
出力	出力ウィンドウの表示 / 非表示を切り替える。	(2.5.6)

## [ ツールバー ] サブメニュー

メニュー名	概要	章節
編集ツールバー	編集ウィンドウ用ツールバーの表示 / 非表示を切り替える。	(2.5.3)
プロジェクトツールバー	プロジェクト管理用ツールバーの表示 / 非表示を切り替える。	(2.5.3)

## [ ビルド ] メニュー

プロジェクトならびにその上位にあるソリューション単位でのビルド、リビルドを行うメニューです。

なお、このメニューはプロジェクトを開いている際に表示されます。

メニュー名	概要	章節
ソリューションのビルド	ソリューションに含まれるプロジェクトすべてをビルドする。	(6.1.3)
ソリューションのリビルド	ソリューションに含まれるプロジェクトすべてをリビルドする。	(6.2.2)
ソリューションのクリーン	ソリューションに含まれるすべてのプロジェクトの出力ファイルを削除する。	(6.3.2)
バッチビルド	ダイアログボックスにてプロジェクトやモードを選択してビルド/リビルド/クリーンを実行する。	(6.2.1)
実行	生成されたアプリケーションを実行する。 【Linux】本メニューは使用不可	(6.5)
選択されたプロジェクトのビルド	選択されたプロジェクトをビルドする。	(6.1.1)
選択されたプロジェクトのリビルド	選択されたプロジェクトをリビルドする。	(6.2.1)
選択されたプロジェクトのクリーン	選択されたプロジェクトの出力ファイルを削除する。	(6.3.1)
オプション設定	選択されたプロジェクトのコンパイルオプションを設定する。	(4.6)
キャンセル	実行中のビルド/リビルド/クリーンをキャンセルする。	(6.1.1)
ビルド成果物のコピー	ビルド成果物を指定された場所へコピーする。 【Linux】 コピー先はビルドサーバー内。	(6.7.1)
ビルド成果物のクライアントへの転送	【Linux】 ビルド成果物をクライアント内の指定された場所へ転送する。	(6.7.2)

## [ ユーティリティ ] メニュー

COBOL 開発環境の統合環境以外の別ツールの呼び出し・登録を行うメニューです。

別ツール起動時の既定のディレクトリは、システムのカレントディレクトリです。

メニュー名	概要	章節
アプリケーションカスタマイザ	【Standard】【Enterprise(SE)】 COBOL アプリケーションカスタマイザを起動する。	COBOL ユーザーズガイド
ファイルマッパー	COBOL ファイルマッパーを起動する。 【Linux】本メニューは使用不可	COBOL ユーザーズガイド



COBOL GUI View Generator	【Standard】 View Generator を起動し、プロジェクトに登録されている画面情報ファイルを開く。	(7.3)
画面節移行ファイルの作成	【Standard】 画面節移行ファイルを作成するためのサブメニュー	(7.4)
既存 COBOL 資産のインポート	既存 COBOL 資産のインポートを行う。 【Linux】本メニューは使用不可	(7.1)
データレコード情報の出力	データレコード情報ファイル作成するためのサブメニュー 本メニューは COBOL Assistant Option がインストールされている場合に表示される。 【Linux】 サーバーへのログインが必要。	(7.6)

#### [ 画面節移行ファイルの作成 ] サブメニュー

メニュー名	概要	章節
プロジェクト内のすべてのソースファイル	【Standard】 プロジェクト内のすべてのソースファイルを対象にして、画面節移行ファイルを作成する。	(7.4)
ファイルビューで選択されたソースファイル	【Standard】 ファイルビューで選択されているソースファイルを対象にして、画面節移行ファイルを作成する。	(7.4)

#### [ データレコード情報の出力 ] サブメニュー

メニュー名	概要	章節
プロジェクト内のすべてのソースファイル	プロジェクト内のすべてのソースファイルを対象にして、データレコード情報ファイルを作成する。	(7.6)
プロジェクトフォルダの COBOL ファイル	プロジェクトフォルダにあるソースファイルを対象にして、データレコード情報ファイルを作成する。	(7.6)

#### [ 環境 ] メニュー

開発環境の設定をするためのメニューです。

メニュー名	概要	章節
ユーティリティ登録	ユーティリティを登録する。	(7.2)
プリコンパイラのビルド設定	【Enterprise(標準)】【Linux】 ユーザ単位の既定値として、ACOS 互換製品に関	(7.5)

	するビルド情報を設定する。	
ログイン	【Linux】 ビルドサーバーへログインする。	(6.6.2)
ログアウト	【Linux】 ビルドサーバーからログアウトする。	(6.6.3)
接続設定	【Linux】 ビルドサーバーへの接続情報を設定する。	(6.6.1)

## [ ウィンドウ ] メニュー

COBOL 開発環境上のウィンドウの制御を行うメニューです。

メニュー名	概要	章節
新しいウィンドウを開く	現在アクティブな編集ウィンドウを複製する。	-
重ねて表示	ウィンドウを重ねて表示する。	-
上下に並べて表示	ウィンドウを上下に並べて表示する。	-
左右に並べて表示	ウィンドウを左右に並べて表示する。	-
アイコンの整列	アイコンを整列する。	-
すべて閉じる	すべてのウィンドウを閉じる。	-
ウィンドウ...	表示されているウィンドウを選択して切り替える。	-

## [ ヘルプ ] メニュー

ヘルプ情報を表示するためのメニューです。

メニュー名	概要	章節
バージョン情報	COBOL 開発環境のバージョン情報を表示する。	-

## 2.6.2 右クリックメニュー

COBOL 開発環境のウィンドウ内でマウスの右ボタンをクリックすると、マウスポインタ位置にコンテキストメニューが表示されます。

右クリックを行う場所により、初期状態で表示されるメニュー項目が変わります。

### 編集ウィンドウ内

メニュー名	概要	章節
元に戻す	現在アクティブな編集ウィンドウに対して、最後に行った作業を取り消し、元に戻す。	(5.7.17)
やり直し	直前に [ 元に戻す ] コマンドで元に戻した編集	(5.7.18)

	作業をやり直す。	
切り取り	選択されているテキストを削除し、クリップボードに保存する。	(5.7.12)
コピー	選択されているテキストをクリップボードに保存する。	(5.7.13)
貼り付け	クリップボードに保存されているテキストをカーソル位置に貼り付ける。	(5.7.14)

#### 編集ファイルなしの編集ウィンドウ

メニュー名	概要	章節
新規作成	ファイルを新規作成する。	(5.2)
開く	ファイルを開く。	(5.3)

#### ファイルビュー内

##### ソリューションを選択している場合

メニュー名	概要	章節
ソリューションのビルド	ソリューションに含まれるプロジェクトすべてをビルドする。	(6.1.3)
ソリューションのリビルド	ソリューションに含まれるプロジェクトすべてをリビルドする。	(6.2.2)
ソリューションのクリーン	ソリューションに含まれるすべてのプロジェクトの出力ファイルを削除する。	(6.3.2)
バッチビルド	ダイアログボックスにてプロジェクトやモードを選択してビルド/リビルド/クリーンを実行する。	(6.4)
追加	プロジェクトをソリューションに追加する。	(3.5)
名前の変更	ソリューションの名前を変更する。	(3.7)
エクスプローラーでフォルダーを開く	ソリューションのフォルダーをエクスプローラーで開く。	(3.8)

##### [ 追加 ] サブメニュー

メニュー名	概要	章節
新しいプロジェクト	新規にプロジェクトをソリューションに追加する。	(3.5.1)
既存のプロジェクト	既存のプロジェクトをソリューションに追加する。	(3.5.2)

### プロジェクトを選択している場合

メニュー名	概要	章節
ビルド	選択されたプロジェクトをビルドする。	(6.1.1)
リビルド	選択されたプロジェクトをリビルドする。	(6.2.1)
クリーン	選択されたプロジェクトの出力ファイルを削除する。	(6.3.1)
追加	ファイルをプロジェクトに追加する。	(4.4.1)
削除	選択されたプロジェクトをソリューションから削除する。	(3.9.2)
名前の変更	選択されたプロジェクトの名前を変更する。	(3.9.1)
プロジェクトフォルダーを開く	プロジェクトのフォルダーをエクスプローラーで開く。	(3.9.3)
オプション設定	選択されたプロジェクトのコンパイルオプションを設定する。	(4.6)

#### [ 追加 ] サブメニュー

メニュー名	概要	章節
新しいファイル	新規にファイルをプロジェクトに追加する。	(4.4.1)
既存のファイル	既存のファイルをプロジェクトに追加する。	(4.4.2)

### ソースファイルまたはその他ファイルのフォルダーを選択している場合

メニュー名	概要	章節
追加	ファイルをプロジェクトに追加する。	(4.4)

#### [ 追加 ] サブメニュー

メニュー名	概要	章節
新しいファイル	新規にファイルをプロジェクトに追加する。	(4.4.1)
既存のファイル	既存のファイルをプロジェクトに追加する。	(4.4.2)

### ソースファイルを選択している場合

メニュー名	概要	章節
開く	選択されたファイルを開く。	(5.3)
削除	選択されたファイルを削除する。	(4.8.2)
名前の変更	選択されたファイルの名前を変更する。	(4.8.1)
メインプログラムに設定	選択されたファイルをメインプログラムに設定する。	(4.7.1)
参照の設定	選択されたファイルで、EXTERNAL 句の記述されたデータを参照とする。	(4.7.2)
定義の設定	選択されたファイルで、EXTERNAL 句の記述されたデータを定義とする。	(4.7.2)
依存ファイルの追加	選択されたファイルの依存ファイルを追加する。	(4.7.3)
このアイテムのフォルダーを開く	選択されたファイルのフォルダーをエクスプローラーで開く。	(4.8.3)

### その他ファイルを選択している場合

メニュー名	概要	章節
開く	選択されたファイルを開く。	(5.3)
削除	選択されたファイルを削除する。	(4.8.2)
名前の変更	選択されたファイルの名前を変更する。	(4.8.1)
このアイテムのフォルダーを開く	選択されたファイルのフォルダーをエクスプローラーで開く。	(4.8.3)

### 依存ファイルを選択している場合

メニュー名	概要	章節
開く	選択されたファイルを開く。	(5.3)
削除	選択されたファイルを削除する。	(4.8.2)
このアイテムのフォルダーを開く	選択されたファイルのフォルダーをエクスプローラーで開く。	(4.8.3)

### 出力ウィンドウ内

メニュー名	概要	章節
コピー	出力タブ内のテキストをコピーする。	(6.1.2)
すべてクリア	出力タブ内のテキストをクリアする。	(6.1.2)
エラー位置づけ	翻訳エラーの場所へ位置づける。	(6.1.2)
次の場所へ移動	翻訳エラーの次の場所へ位置づける。	(6.1.2)
前の場所へ移動	翻訳エラーの前の場所へ位置づける。	(6.1.2)

### 検索結果ウィンドウ内

メニュー名	概要	章節
コピー	検索タブ内のテキストをコピーする。	(6.1.2)
すべてクリア	検索タブ内のテキストをクリアする。	(6.1.2)
検索結果位置づけ	検索結果の場所へ位置づける。	(6.1.2)
次の場所へ移動	検索結果の次の場所へ位置づける。	(6.1.2)
前の場所へ移動	検索結果の前の場所へ位置づける。	(6.1.2)

---

## 3 ソリューション

この章では、COBOL 開発環境の開発資産を管理するプロジェクトをまとめるソリューションについて説明します。

---

### 3.1 ソリューションの概要

---

COBOL 開発環境では、複数のプロジェクトを 1 つにまとめて管理する「ソリューション」という概念があります。このソリューションを活用することで、さまざまな開発スタイルに合わせたプロジェクトの管理を行うことができます。

#### アプリケーションの場合

アプリケーションのプロジェクトを作成し、ソリューションにて管理します。

#### ライブラリの場合

ライブラリのプロジェクトを作成し、ソリューションにて管理します。

また、そのライブラリを利用するアプリケーションのプロジェクトを同一ソリューション下に作成し、管理します。

また、既存のプロジェクトをソリューションにまとめることや、既存のソリューションへ新たにプロジェクトを追加することも可能です。なお、ソリューションへ登録可能なプロジェクトの数に制限はありません。

## 3.2 ソリューションの新規作成

新しいプロジェクトを作成すると、新しいソリューションが作成されます。

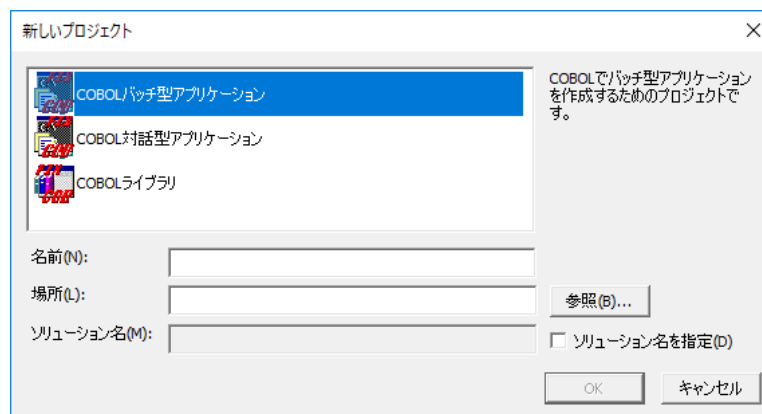
作成されたソリューションに、プロジェクトやプログラムの作成に必要なソースファイルなどを追加します。

何も含まれていない空のソリューションを作成することはできません。

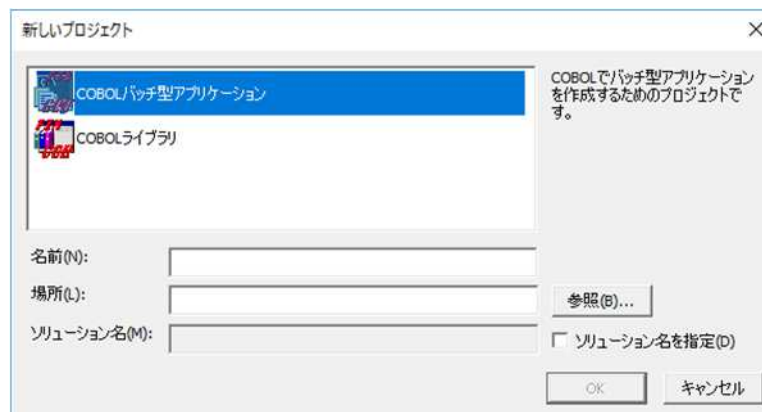
ソリューションとプロジェクトを新規に作成するには、以下の手順で行います。

1. [ファイル]メニューから[新規作成] - [プロジェクト]コマンドを選択します。
2. [新しいプロジェクト]ダイアログボックスが表示されます。
3. [新しいプロジェクト]ダイアログボックスで、テンプレートから、作成するアプリケーションまたはライブラリを選択します。

【Standard】 【Enterprise(SE)】



【Enterprise(標準)】 【Linux】



4. [名前]テキストボックスに新しいプロジェクトの名前を指定し、[場



所] テキストボックスに保存する場所を指定します。

[ 参照 ] ボタンをクリックすると、[ フォルダーの参照 ] ダイアログボックスが表示され、フォルダーを選択することができます。

5. [ ソリューション名を指定 ] チェックボックスを選択すると、[ ソリューション名 ] テキストボックスが有効になり、ソリューション名をプロジェクト名と異なる名前にすることができます。

6. [ OK ] ボタンをクリックします。

新しいソリューションとプロジェクトが作成され、ファイルビュー上に開かれます。

[ キャンセル ] ボタンをクリックすると、ソリューションとプロジェクトは作成されません。

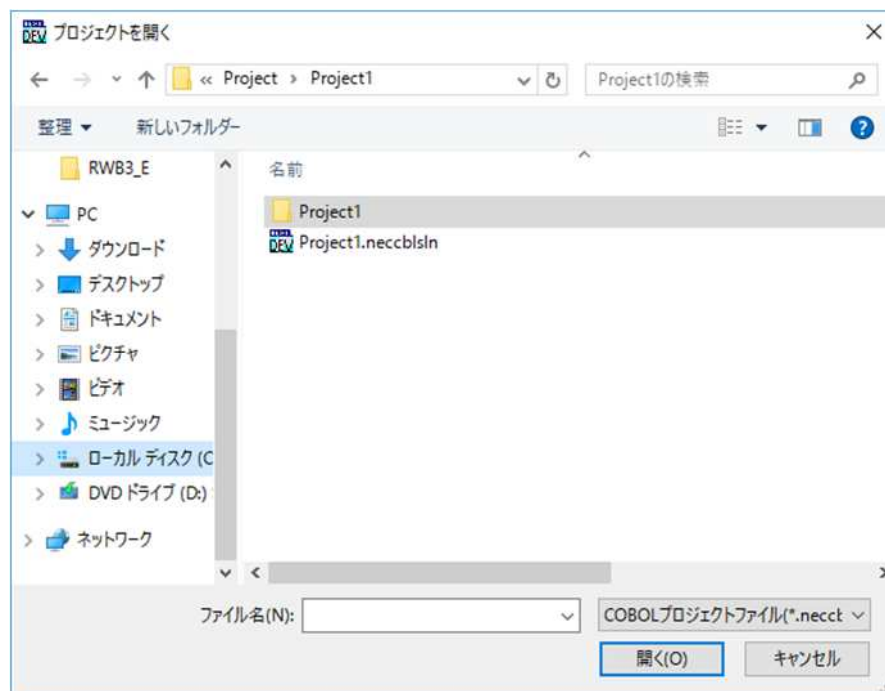
#### **注意事項**

- ・ ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。
- ・ [ 場所 ] [ 名前 ] [ ソリューション名 ] のテキストボックスに、入力なし、またはピリオドで始まる相対パスを指定にすることはできません。

### 3.3 ソリューションを開く

既存のソリューションを開く手順を次に示します。

1. [ファイル]メニューから[開く] - [プロジェクト/ソリューション] コマンドを選択します。
2. [プロジェクトを開く]ダイアログボックスが表示されます。



3. [ファイル名]ドロップダウンリストにプロジェクトファイル名またはソリューションファイル名を直接入力するか、ドロップダウンリストから選択します。

既存のファイルやフォルダーを参照するには、ナビゲーションウィンドウを操作し、フォルダーのリストから選択します。

4. [開く]ボタンをクリックします。

選択したソリューションが開いてファイルビュー上に表示されます。既にソリューションを開いている状態で別のソリューションを開いた場合、現在開いているソリューションは先に閉じられます。このとき開いているファイルに変更があり、その内容が保存されていない場合には、ファイルを保存するかどうかの確認が行われます。

[キャンセル]ボタンをクリックすると、ソリューションは開かれません。

最近開いたソリューションやプロジェクトは、[ ファイル ] - [ 最近使ったプロジェクトとソリューション ] のサブメニューに一覧で表示されます。[ ファイル ] - [ 最近使ったプロジェクトとソリューション ] から選択することで、すばやく開くことが可能です。頻繁に利用するソリューションやプロジェクトを開くときに活用します。

### **注意事項**

- ・ [ ファイル ] - [ 最近使ったプロジェクトとソリューション ] サブメニューは、利用者に登録されます。
- ・ 開発環境を閉じる時の一覧が登録され、次回に同じ利用者が開発環境を起動すると、サブメニューに同じ一覧が表示されます。

### 3.4 ソリューションを閉じる

---

ソリューションを閉じる手順を説明します。

1. [ ファイル ] メニューから [ ソリューションを閉じる ] コマンドを選択します。
2. 現在開いているソリューションが閉じられます。

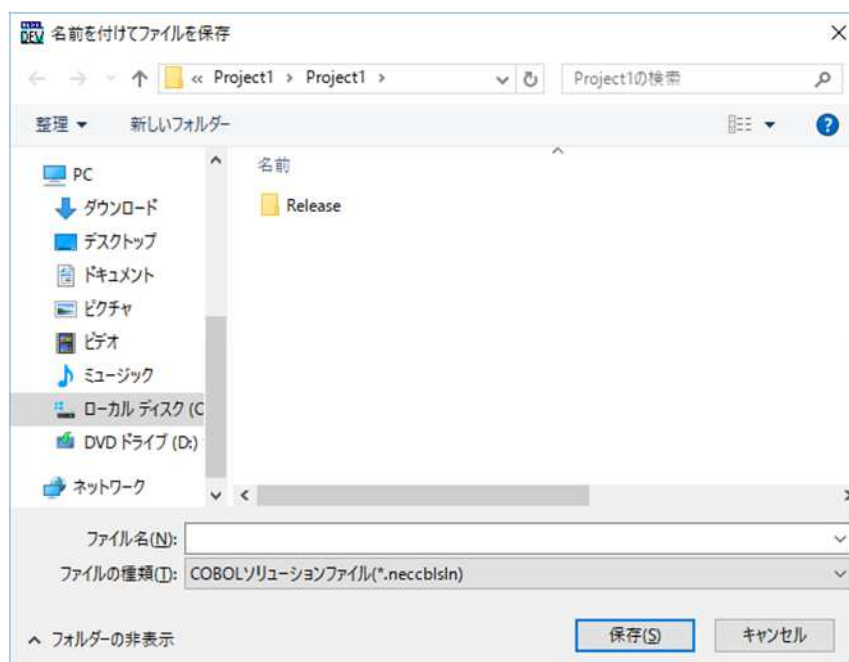
COBOL 開発環境を終了する場合にもソリューションは閉じられます。

ソリューションファイルに登録されていないプロジェクトファイルをCOBOL 開発環境で開いた場合、そのプロジェクトを含んだソリューションが一時的に作成されます。

一時的に作成されるソリューションについては、「4.3 プロジェクトを開く」を参照してください。

一時的に作成されたソリューションを閉じようとする、[ 名前を付けてファイルを保存 ] ダイアログボックスが表示されます。

ファイル名を指定して [ 保存 ] ボタンをクリックすると、ソリューションは保存されます。



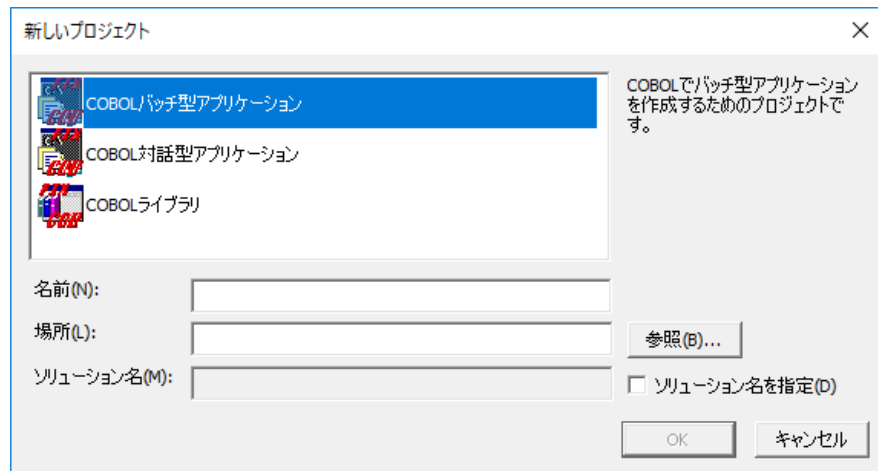
## 3.5 ソリューションへの追加

開いているソリューションに対して、プロジェクトを追加するには、2つの方法があります。

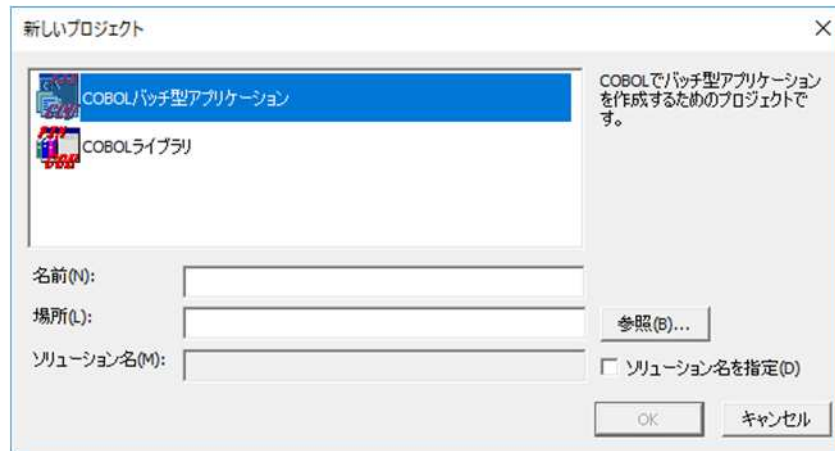
- ・ 新しいプロジェクトの追加・・・追加と同時に新たにプロジェクトを作成する場合
- ・ 既存のプロジェクトの追加・・・既に作成しているプロジェクトをソリューションで管理するために追加する場合

### 3.5.1 新しいプロジェクトの追加

1. [ファイル] - [追加] - [新しいプロジェクト] コマンドを選択します。
2. [新しいプロジェクト] ダイアログボックスが表示されます。
3. 【Standard】 【Enterprise(SE)】



## 【Enterprise(標準)】 【Linux】



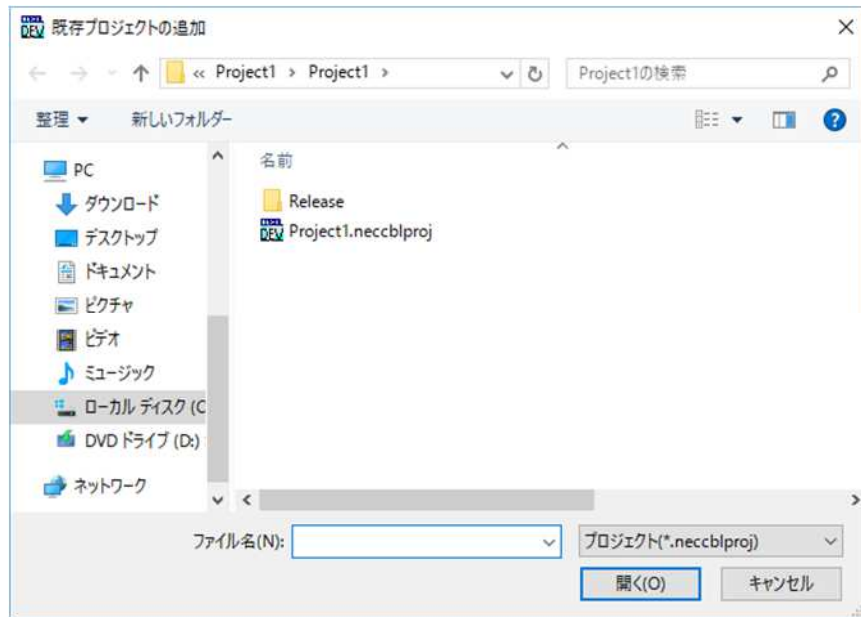
3. COBOL バッチ型アプリケーション、COBOL 対話型アプリケーションまたは COBOL ライブラリを選択します。
4. [ 名前 ] テキストボックスにプロジェクトの名前を入力します。
5. [ 場所 ] テキストボックスには、ソリューションのパスが表示されています。パスを変更する場合は、直接入力するか、[ 参照 ] ボタンをクリックして [ フォルダの参照 ] ダイアログボックスを表示し、フォルダを指定します。
6. [ OK ] ボタンをクリックすると、プロジェクトファイルが作成され、ファイルビューに追加されます。  
[ キャンセル ] ボタンをクリックすると、プロジェクトは追加されません。

### **注意事項**

- ・ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。

## 3.5.2 既存のプロジェクトの追加

1. [ ファイル ] - [ 追加 ] - [ 既存のプロジェクト ] コマンドを選択します。
2. [ 既存プロジェクトの追加 ] ダイアログボックスが表示されます。



3. ナビゲーションウィンドウでフォルダーを選択します。
4. ファイルが表示されたリストからプロジェクトファイルを選択します。  
または、[ファイル名] ドロップダウンリストに直接入力するか、ドロップダウンリストから選択します。
5. [開く] ボタンをクリックすると、プロジェクトが現在のソリューションに追加されます。  
[キャンセル] ボタンをクリックすると、既存のプロジェクトは追加されません。

### **注意事項**

- ・ソリューションに登録するプロジェクトファイル名に、使用できない文字を含むファイル名を登録すると、ソリューションの操作などが正しく行えない場合があります。
- ・ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。

## 3.6 ソリューションの保存

---

開いているソリューションを保存するには、3つの方法があります。

- ・ 上書き保存する。
- ・ 名前を付けて保存する。
- ・ すべてを保存する。

### 3.6.1 上書き保存

ソリューションを上書き保存する手順を説明します。

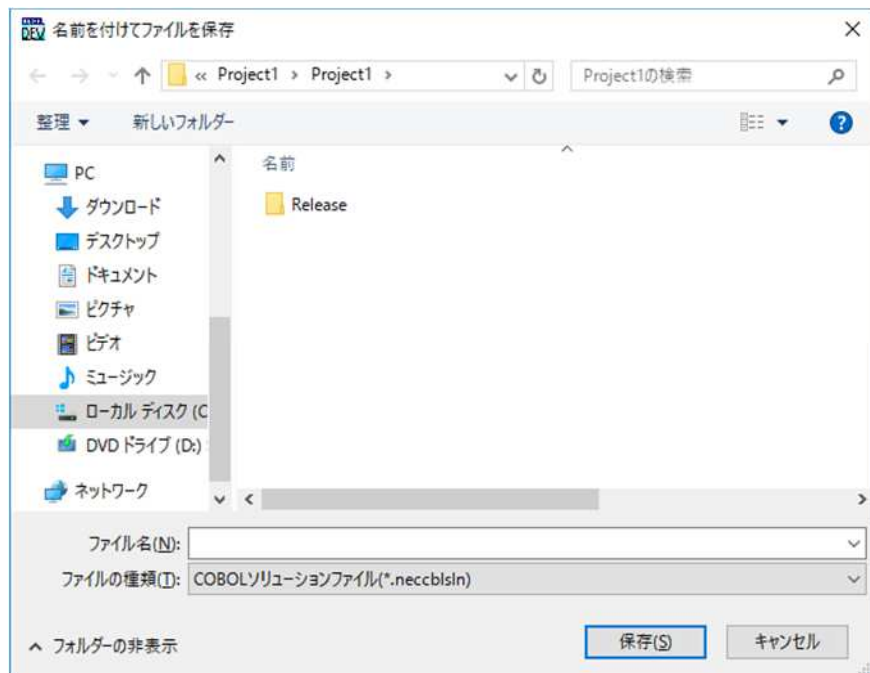
1. [ ファイルビュー ] で、ソリューションのノードを選択します。
2. [ ファイル ] - [ ソリューション名の保存 ] コマンドを選択します。  
ソリューション名には、ソリューション名が表示されます。  
または、ツールバーの [ 保存 ] ボタンをクリックします。
3. ソリューションが上書き保存されます。

### 3.6.2 名前を付けて保存

ソリューションに名前を付けて保存する手順を説明します。

1. [ ファイルビュー ] で、ソリューションのノードを選択します。
2. [ ファイル ] - [ ソリューション名に名前を付けて保存 ] コマンドを選択します。  
ソリューション名には、ソリューション名が表示されます。
3. [ 名前を付けてファイルを保存 ] ダイアログボックスが表示されます。





4. ナビゲーションウィンドウやフォルダー名のリストからフォルダーを選択します。  
保存前のソリューションと同じフォルダーに保存しない場合は、保存前のソリューションに含まれるプロジェクトファイルとプロジェクトに登録されたファイルを、保存後のソリューション配下に同じ構成で置く必要があります。
5. [ファイル名] ドロップダウンリストにファイル名を直接入力するか、ドロップダウンリストから選択します。
6. [保存] ボタンをクリックすると、指定されたファイル名でソリューションが保存されます。  
[キャンセル] ボタンをクリックすると、ソリューションは保存されません。
7. ファイルビューに、新しい名前のソリューションが開かれています。

### **注意事項**

- ・ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。

### 3.6.3 すべての保存

ソリューションをすべて上書き保存する手順を説明します。

1. [ファイル] - [すべてを保存] コマンドを選択します。  
または、ツールバーの [すべてを保存] ボタンをクリックします。
2. 開いているソリューションとすべてのプロジェクトが保存されます。

## 3.7 ソリューション名の変更

---

ソリューションの名前を変更する手順を説明します。

1. [ ファイルビュー ] で、名前を変更したいソリューションを選択します。
2. マウスを右クリックして表示されるメニューから [ 名前の変更 ] コマンドを選択します。
3. 選択したソリューションの名前の部分が編集可能になります。
4. 変更後のソリューション名を入力した後、[ Enter ] キーを押すと、ソリューション名が変更されます。

### 3.8 エクスプローラーでフォルダーを開く

---

ソリューションのフォルダーをエクスプローラーで開く手順を説明します。

1. [ ファイルビュー ] で、ソリューションを選択します。
2. マウスを右クリックして表示されるメニューから [ エクスプローラーでフォルダーを開く ] コマンドを選択します。
3. エクスプローラーが別画面に開き、ソリューションのフォルダーが表示されます。

## 3.9 プロジェクトの操作

---

ファイルビューでは、ソリューション配下のプロジェクトに対して以下の操作が可能です。

- ・ 名前の変更
- ・ 削除
- ・ エクスプローラーでフォルダーを開く

### 3.9.1 名前の変更

プロジェクトの名前を変更する手順を説明します。

1. [ ファイルビュー ] で、名前を変更したいプロジェクトを選択します。
2. マウスを右クリックして表示されるメニューから [ 名前の変更 ] コマンドを選択します。
3. 選択したプロジェクトの名前の部分が編集可能になります。
4. 変更後のプロジェクト名を入力した後、[ Enter ] キーを押すと、プロジェクト名が変更されます。

#### **注意事項**

- ・ ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。

### 3.9.2 削除

プロジェクトを削除する手順を説明します。

1. [ ファイルビュー ] で、ソリューションから削除したいプロジェクトを選択します。
2. マウスを右クリックして表示されるメニューから [ 削除 ] コマンドを選択します。

3. 選択したプロジェクトがソリューションから削除されます。

ただし、削除を実行しても、ファイル自体は物理的に削除されません。削除したプロジェクトをソリューションのツリー上へ復元するには、[ 既存のプロジェクトの追加 ] を行います。

[ 既存のプロジェクトの追加 ] については、「3.5.2 既存のプロジェクトの追加」を参照してください。

### 3.9.3 エクスプローラーでフォルダーを開く

プロジェクトのフォルダーをエクスプローラーで開く手順を説明します。

4. [ ファイルビュー ] で、プロジェクトを選択します。
5. マウスを右クリックして表示されるメニューから [ エクスプローラーでフォルダーを開く ] コマンドを選択します。
6. エクスプローラーが別画面に開き、選択したプロジェクトのフォルダーが表示されます。

---

## 4 プロジェクト

この章では、COBOL 開発環境の開発資産を管理するプロジェクトについて説明します。

---

## 4.1 プロジェクトの定義

---

プロジェクトとは、複数のソースファイルを集めて1つのアプリケーションを作成、実行するための情報（翻訳時オプションなど）をまとめることにより、アプリケーション開発を統合的に管理するための概念です。



## 4.2 プロジェクトの新規作成

---

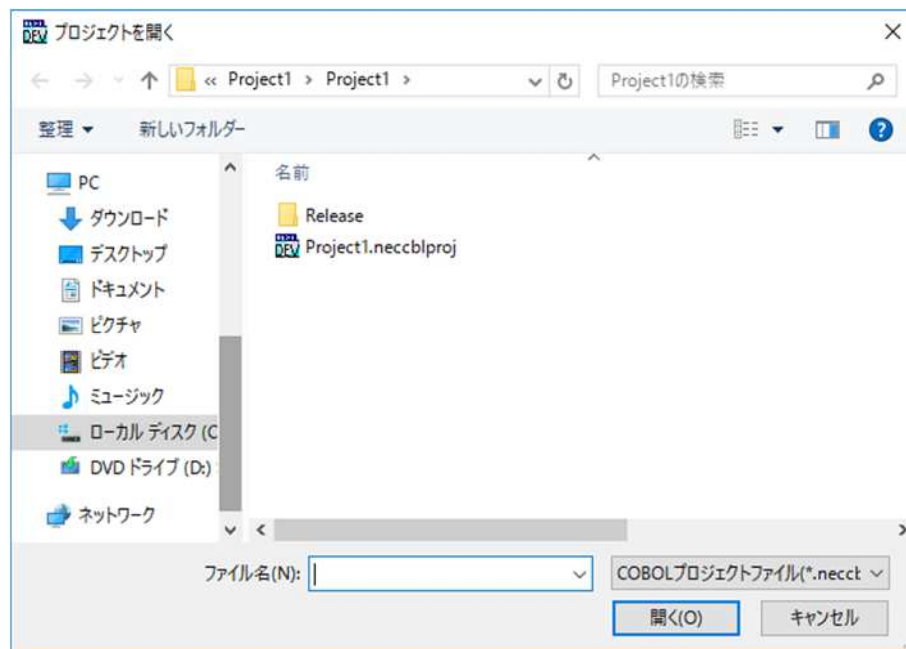
プロジェクトを新規に作成する手順については、「3.2 ソリューションの新規作成」を参照してください。

## 4.3 プロジェクトを開く

既存のプロジェクトを開く手順を説明します。

### ソリューションを開いている場合

1. [ファイル] - [開く] - [プロジェクト/ソリューション] コマンドを選択します。
2. [プロジェクトを開く] ダイアログボックスが表示されます。



3. [ファイル名] ドロップダウンリストにプロジェクトファイル名 ( \*.necbproj ) を直接入力するか、ドロップダウンリストから選択します。  
既存のファイルやフォルダーを参照するには、ナビゲーションウィンドウを操作し、フォルダーのリストから選択します。
4. [開く] ボタンをクリックすると、プロジェクトを開きます。  
[キャンセル] ボタンをクリックすると、プロジェクトは開きません。

プロジェクトを開くと、プロジェクトが登録されているソリューションも開きます。

開く対象のプロジェクトの上位のフォルダーにプロジェクトと同じ名前のソリューションファイルが存在する場合には、そのソリューションを開きます。

同じ名前のソリューションファイルが存在しない場合には、以下の両方の条件を満たすソリューションを開きます。

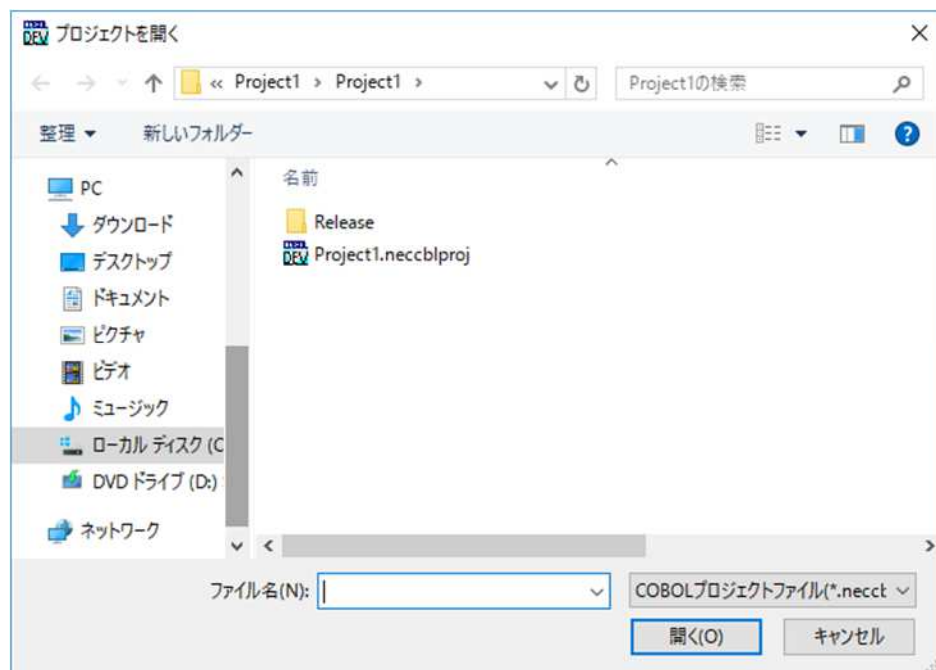
- ・別の名前のソリューションファイルが1つだけ存在する。
- ・そのソリューションに開く対象のプロジェクトが登録されている。

どちらでもない場合には、一時的なソリューションを新規に作成します。

既にソリューションを開いている場合は、そのソリューションは閉じられます。ソリューションで開いていたファイルが更新されている場合には、ファイルを保存するかを確認するダイアログボックスが表示されます。

### ソリューションを開いていない場合

1. [ファイル] - [開く] - [プロジェクト/ソリューション] コマンドを選択します。
2. [プロジェクトを開く] ダイアログボックスが表示されます。



3. [ファイル名] ドロップダウンリストにプロジェクトファイル名 ( \*.necbproj ) を直接入力するか、ドロップダウンリストから選択します。

既存のファイルやフォルダーを参照するには、ナビゲーションウィンド

ウを操作し、フォルダーのリストから選択します。

4. [開く] ボタンをクリックします。選択したプロジェクトの上位にソリューションが存在する場合は、プロジェクトとともに、そのソリューションが開きます。プロジェクトの上位にソリューションが存在しない場合は、プロジェクトを開くとともに、ソリューションが作成されます。  
[キャンセル] ボタンをクリックすると、プロジェクトは開きません。

開く対象のプロジェクトの上位のフォルダーにプロジェクトと同じ名前のソリューションファイルが存在する場合には、そのソリューションを開きます。

同じ名前のソリューションファイルが存在しない場合には、開く対象のプロジェクトが登録されている別の名前のソリューションが1つだけあれば開きます。

どちらでもない場合には、一時的なソリューションが新規に作成されます。

### **共通事項**

プロジェクトに登録されているファイルを開いた状態でプロジェクトを閉じた場合、そのプロジェクトを再度開いた時には、開いていたファイルも再度開きます。

プロジェクトに含まれているファイルが存在しない場合も、ファイルビューのプロジェクト配下にアイコンにて表示されます。

ただし、そのファイルを開こうとした場合に、エラーとなります。

エクスプローラーからファイルのアイコンを COBOL 開発環境へドラッグアンドドロップすることにより、プロジェクトファイルを開くことができます。

最近開いたソリューションやプロジェクトは、[ファイル] - [最近使ったプロジェクトとソリューション] のサブメニューに一覧で表示されます。[ファイル] - [最近使ったプロジェクトとソリューション] から選択することで、すばやく開くことが可能です。頻繁に利用するソリューションやプロジェクトを開くときに活用します。

## **注意事項**

- ・ COBOL 開発環境を「管理者として実行」している場合は、ファイルをドラッグアンドドロップすることができません。
- ・ Ver1.0 のプロジェクトを開く場合に、プロジェクトとソリューションを異なるフォルダーに格納し、ソースファイルをソリューションフォルダ配下に格納していると、そのソースファイルを開くことができません。該当のファイルをプロジェクトに再追加してください。
- ・ 【Linux】 Ver2.0 のプロジェクトを開く場合に、プロジェクトとソリューションを異なるフォルダーに格納し、ソースファイルをソリューションフォルダ配下に格納していると、そのソースファイルを開くことができません。該当のファイルをプロジェクトに再追加してください。
- ・ [ファイル] - [最近使ったプロジェクトとソリューション] サブメニューは、利用者別に登録されます。
- ・ 開発環境を閉じる時の一覧が登録され、次回に同じ利用者が開発環境を起動すると、サブメニューに同じ一覧が表示されます。
- ・ 【Linux】 拡張子「.qcb1m、.qcobm、.qcb1ss、.qcb1mss」のファイルはビルドの対象になりません。それぞれの拡張子を「.pcom、.pcom、.pcoss、.pcomss」に変更して、プロジェクトに登録し、オプション設定で SQL 文展開製品に「COBOL SQL アクセス」を指定してください。

## 4.4 プロジェクトへの追加

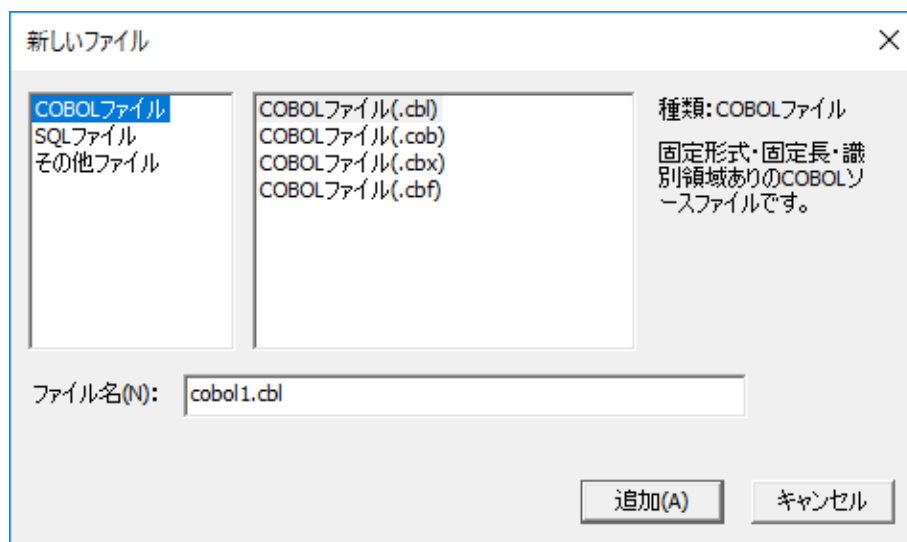
プロジェクトに対して、ファイルを追加するには、2つの方法があります。

- ・ 新しいファイルの追加・・・追加と同時に新たにファイルを作成する場合
- ・ 既存のファイルの追加・・・既に作成しているファイルをプロジェクトで管理するために追加する場合

### 4.4.1 新しいファイルの追加

現在のプロジェクトに新しいファイルを追加する手順を説明します。

1. ファイルビューで、新しいファイルを追加する場所を選択します。  
(「プロジェクト」ノードまたは「ソースファイル」ノードまたは「その他ファイル」ノードを選択します。)
2. マウスを右クリックして表示されるメニューから、[追加] - [新しいファイル] コマンドを選択します。
3. 「新しいファイル」ダイアログボックスが表示されます。  
【Standard】 【Enterprise(SE)】



【Enterprise(標準)】 【Linux】



4. ファイルの種類を選択し、ファイル名を入力します。
5. [ 追加 ] ボタンをクリックします。
6. 指定した名前でファイルが作成され、ファイルビューに追加され、編集ウィンドウが表示されます。  
[ キャンセル ] ボタンをクリックすると、新しいファイルは追加されません。

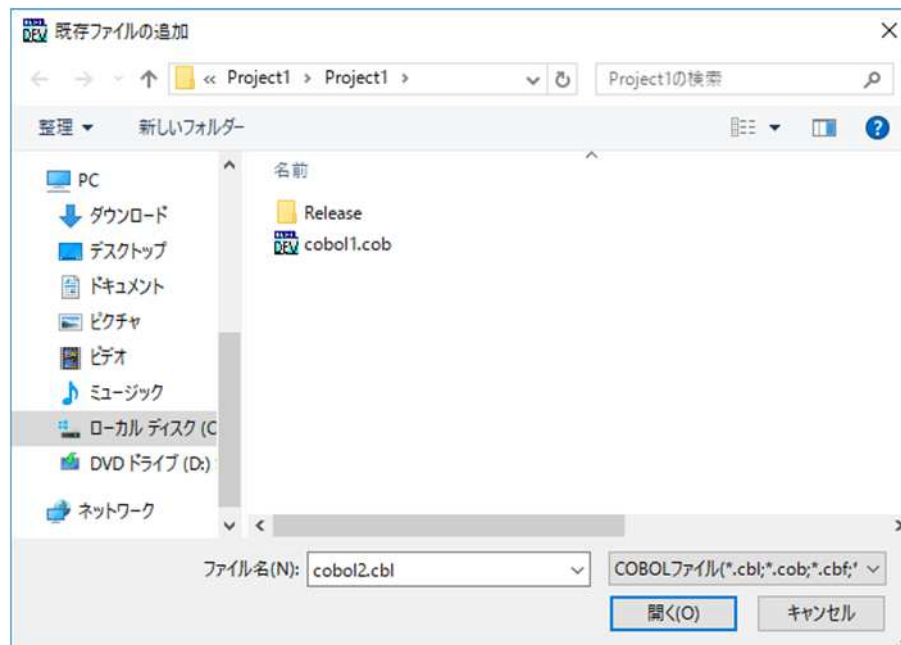
**注意事項**

- ・新規作成するファイルは、空のファイルです。
- ・ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。

## 4.4.2 既存のファイルの追加

現在のプロジェクトに既存のファイルを追加する手順を説明します。

1. ファイルビューで、既存のファイルを追加する場所を選択します。  
(「プロジェクト」ノードまたは「ソースファイル」ノードまたは「その他ファイル」ノードを選択します。)
2. マウスを右クリックして表示されるメニューから、[ 追加 ] - [ 既存のファイル ] コマンドを選択します。
3. [ 既存ファイルの追加 ] ダイアログボックスが表示されます。



4. ドロップダウンメニューで、フォルダーおよびドライブを選択します。
5. [ ファイルの種類 ] を選択して、表示するファイルを絞ることができます。
6. ファイルを選択します。[ Ctrl ] キーや [ Shift ] キーを使用しての複数ファイルの選択も可能です。  
または、[ ファイル名 ] ドロップダウンリストに直接入力するか、ドロップダウンリストから選択します。
7. [ 開く ] ボタンをクリックすると、選択したファイルがファイルビューに追加されます。  
[ キャンセル ] ボタンをクリックすると、既存ファイルは追加されません。

### **注意事項**

- ・プロジェクトに登録するファイル名に使用できない文字を含むファイル名を登録すると、プロジェクトの操作などが正しく行えない場合があります。
- ・ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。



## 4.5 プロジェクトの保存

---

開いているプロジェクトを保存するには、3つの方法があります。

- ・ 上書き保存する。
- ・ 名前を付けて保存する。
- ・ すべてを保存する。

### 4.5.1 上書き保存

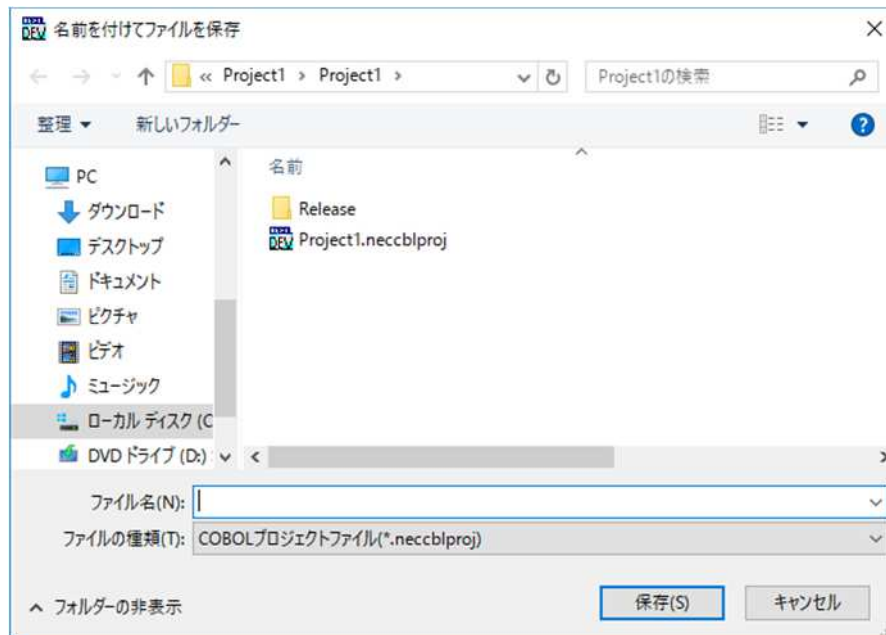
プロジェクトを上書き保存する手順を説明します。

1. [ファイルビュー]で、プロジェクトのノードを選択します。
2. [ファイル] - [プロジェクト名の保存] コマンドを選択します。  
*プロジェクト名*には、プロジェクト名が表示されます。  
または、ツールバーの[保存] ボタンをクリックします。
3. プロジェクトが上書き保存されます。

### 4.5.2 名前を付けて保存

プロジェクトに名前を付けて保存する手順を説明します。

1. [ファイルビュー]で、プロジェクトのノードを選択します。
2. [ファイル] - [プロジェクト名に名前を付けて保存] コマンドを選択します。  
*プロジェクト名*には、プロジェクト名が表示されます。
3. [名前を付けてファイルを保存] ダイアログボックスが表示されます。



4. ナビゲーションウィンドウやフォルダー名のリストからフォルダーを選択します。  
保存前のプロジェクトと同じフォルダーに保存しない場合は、保存前のプロジェクトに登録されたファイルを、保存後のソリューション配下に同じ構成で置く必要があります。
5. [ファイル名] ドロップダウンリストにファイル名を直接入力するか、ドロップダウンリストから選択します。
6. [保存] ボタンをクリックすると、指定されたファイル名でプロジェクトが保存されます。  
[キャンセル] ボタンをクリックすると、プロジェクトは保存されません。
7. ファイルビューに、新しい名前のプロジェクトが開かれています。

### **注意事項**

- ・ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。

## 4.5.3 すべてを保存

プロジェクトをすべて上書き保存する手順を説明します。

1. [ファイル] - [すべてを保存] コマンドを選択します。

または、ツールバーの [すべてを保存] ボタンをクリックします。

2. 開いているソリューションとすべてのプロジェクトが保存されます。

## 4.6 オプション設定

COBOL コンパイラ、C/C++コンパイラ、リソースコンパイラ、リンカを起動する際のオプションは、[ビルド] - [オプション設定] コマンドを選択すると表示される [オプション設定] ダイアログボックスで設定します。

現在のプロジェクトの「オプション設定」ダイアログボックスを表示する手順を説明します。

1. ファイルビューで、「プロジェクト」ノードまたはプロジェクト配下のノードを選択します。
2. メインメニューで [ビルド] - [オプション設定] コマンドを選択します。または、マウスを右クリックして表示されるメニューから [オプション設定] コマンドを選択するか、ツールバーの [オプション設定] ボタンをクリックします。
3. 「オプション設定」ダイアログボックスが表示されます。

【Standard】 【Enterprise(SE)】



## 【Enterprise(標準)】

オプション設定

ディレクトリ		ACOS互換製品		コマンドライン	
全般	COBOL	リンカ	Pro*COBOL	リソース	C/C++
				オプション製品	

初期化

成果物形式

アプリケーション

☒ バッチ型

ライブラリ

☐ 動的

☐ 静的

☐ オブジェクトファイル

成果物のファイル名 [-o]

プリコンパイラのビルド設定

全初期化 OK キャンセル

## 【Linux】

オプション設定

転送		ACOS互換製品		コマンドライン	
全般	COBOL	リンカ	Pro*COBOL	C/C++	オプション製品

初期化

成果物形式

アプリケーション

☒ バッチ型

ライブラリ

☐ 動的

☐ 静的

☐ オブジェクトファイル

成果物のファイル名 [-o]

プリコンパイラのビルド設定

全初期化 OK キャンセル

「オプション設定」ダイアログボックスは、次のカテゴリに大別され、タブを選択することで画面を切り替えます。

カテゴリ	設定できる内容
全般	アプリケーションに関する設定を行います。
COBOL	コンパイラの翻訳時動作に関する設定を行います。 コンパイルモード、実行モード、実行モード、リスト出力、ディレクトリ、従来製品互換の6画面で構成されます。 【Enterprise(標準)】【Linux】ACOS-4 互換の画面を含みます。 【Linux】用語機能の画面を含みます。
リンカ	リンカに関する設定を行います。
Pro*COBOL	Pro*COBOL に関する設定を行います。
リソース	【Standard】【Enterprise(SE)】【Enterprise(標準)】 リソースコンパイラに関する設定を行います。
C/C++	C/C++コンパイラに関する設定を行います。
オプション製品	オプション製品に関する設定を行います。
ディレクトリ	【Standard】【Enterprise(SE)】【Enterprise(標準)】 ディレクトリに関する設定を行います。
転送	成果物の転送に関する設定を行います。
ACOS 互換製品	【Enterprise(標準)】【Linux】 ACOS 互換製品に関する設定を行います。
コマンドライン	設定されたオプションでのコマンドライン文字列を表示します。

[ OK ] ボタンをクリックすると、設定内容が有効となり、ダイアログボックスが閉じられます。

一方、[ キャンセル ] ボタンをクリックすると、すべての変更を取り消して、ダイアログボックスが閉じられます。

[ 初期化 ] ボタンをクリックすると、タブの中の設定内容が初期値に設定されます。

[ 全初期化 ] ボタンをクリックすると、すべてのタブの設定内容が初期値に設定されます。

### ディレクトリの選択方法

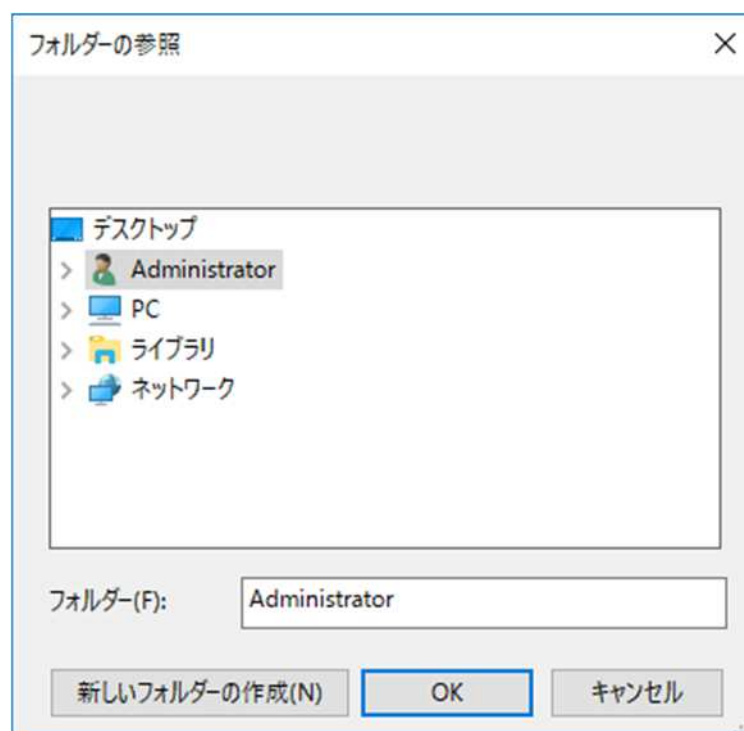
ディレクトリを入力する各テキストボックスにおいて、フォルダー選択用の [ 一覧 ] ボタンが右端に表示されます。

[ 一覧 ] ボタンをクリックすると、以下のような [ フォルダー一覧 ] ダイアログボックスが表示されます。

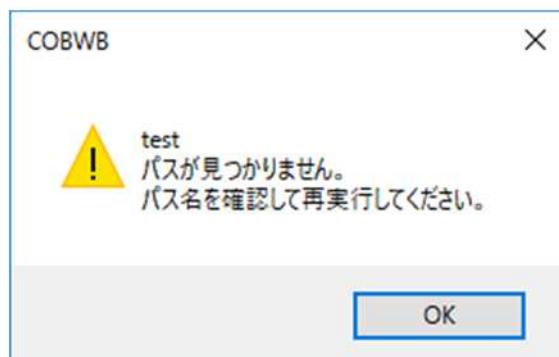


フォルダーを追加する場合には、[ 追加 ] ボタンをクリックし、表示される [ フォルダーの参照 ] ダイアログボックスでフォルダーを選択し、[ OK ] ボタンをクリックします。

[ キャンセル ] ボタンをクリックすると、フォルダーの追加がキャンセルされます。



ただし、[ フォルダー名 ] ドロップダウンリストに存在しないフォルダーなどアクセスできないフォルダーを指定すると、以下のようなエラーメッセージが表示されます。



選択したフォルダーをリストから削除する場合には、削除するフォルダー名を選択し、[ 削除 ] ボタンをクリックします。

選択したフォルダーの順序を変更する場合には、変更するフォルダー名を選択し、[ ] [ ] ボタンをクリックします。

[ OK ] ボタンをクリックすると、その内容がオプション設定画面に反映されます。

[ キャンセル ] ボタンをクリックすると、行なった変更が取り消されます。

### **注意事項**

- ・セミコロンを含むディレクトリ名を指定すると、コンパイルなどが正しく行えない場合があります。
- ・オプション設定のファイル名に、改行コードや使用できない文字を含むファイル名を登録すると、プロジェクトの操作などが正しく行えない場合があります。
- ・ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。
- ・オプション設定時に、makefile で特別な意味を持つ「 % ^ & \$ 」などの文字を使用すると、コンパイルなどが正しく行えない場合があります。使用できない文字は makefile の仕様に則ります。

次節以降で、各ページにおいて、設定できる内容を示し、対応するコンパイルオプションを併記します。



## 4.6.1 全般

アプリケーションに関する設定ページです。

【Standard】【Enterprise(SE)】

The screenshot shows a Windows-style dialog box titled 'オプション設定' (Option Settings) with a close button (X) in the top right corner. The dialog has a tabbed interface with three tabs: 'ディレクトリ' (Directory), '転送' (Transfer), and 'コマンドライン' (Command Line). The '全般' (General) tab is selected. Below the tabs is a '初期化' (Initialize) button. The main area is divided into two columns. The left column is titled '成果物形式' (Output Format) and contains a group box 'アプリケーション' (Application) with three radio buttons: 'パッチ型' (Patch type) which is selected, '対話型' (Interactive), and 'オブジェクトファイル' (Object file). The right column is titled 'ライブラリ' (Library) and contains two radio buttons: '動的' (Dynamic) and '静的' (Static). Below these columns are two text input fields: '成果物のファイル名 [-o]' (Output file name [-o]) and 'レジストリキー名 [-Yr]' (Registry key name [-Yr]). At the bottom of the dialog are three buttons: '全初期化' (Initialize all), 'OK', and 'キャンセル' (Cancel).

ディレクトリ	転送	コマンドライン				
全般	COBOL	リンク	Pro*COBOL	リソース	C/C++	オプション製品

初期化

成果物形式

アプリケーション

- ☒ パッチ型
- ☐ 対話型
- ☐ オブジェクトファイル

ライブラリ

- ☐ 動的
- ☐ 静的

成果物のファイル名 [-o]

レジストリキー名 [-Yr]

全初期化 OK キャンセル

【Enterprise(標準)】 【Linux】

オプション設定

転送		ACOS互換製品		コマンドライン	
全般	COBOL	リンカ	Pro*COBOL	C/C++	オプション製品

初期化

成果物形式

アプリケーション	ライブラリ
<input checked="" type="radio"/> バッチ型	<input type="radio"/> 動的 <input type="radio"/> 静的
<input type="radio"/> オブジェクトファイル	

成果物のファイル名 [-o]

プリコンパイラのビルド設定

全初期化 OK キャンセル

項目名	内容
成果物形式 【Enterprise(標準)】 【Linux】	次のいずれかを選択します。 「COBOL バッチ型アプリケーション」 「COBOL 動的ライブラリ」 「COBOL 静的ライブラリ」 「COBOL REST API」 「オブジェクトファイル」
成果物形式 【Standard】 【Enterprise(SE)】	次のいずれかを選択します。 「COBOL バッチ型アプリケーション」 「COBOL 対話型アプリケーション」 「COBOL 動的ライブラリ」 「COBOL 静的ライブラリ」 「オブジェクトファイル」
成果物のファイル名 [-o]	成果物のファイル名 成果物形式がアプリケーションまたはライブ

	ラリの場合に指定が有効。
レジストリキー名 [-Yr] 【Standard】 【Enterprise(SE)】	最大 128 バイトの任意の文字列
プリコンパイラのビルド設定 【Enterprise(標準)】 【Linux】	「プリコンパイラのビルド設定」ダイアログボックスで、プロジェクトごとにビルド順と SQL 文展開製品を指定します。 ユーザ単位の既定値は、[ 環境 ] メニューの [ プリコンパイラのビルド設定 ] コマンドで設定します。

### 注意事項

・成果物のファイル名にセミコロンを含むと、特定の条件下においてビルド時にエラーが発生するため、セミコロンを使用しないでください。

・直接入力した設定値が、オプションのチェックボックスを外すことでグレー表示されることがありますが、オプション設定画面を閉じるとグレー表示の設定値は保存されません。

#### 【Enterprise(標準)】

・COBOL REST API は、COBOL Enterprise Edition Developer/Compiler API ライセンスがインストールされている場合に作成できます。

#### 【Linux】

・COBOL REST API は、COBOL Enterprise Edition Developer/Compiler API ライセンスがサーバーにインストールされている場合に作成できます。サーバーにログインすると、ライブラリグループに「REST API」が表示され選択が可能になります。

詳細については、「COBOL ユーザーズガイド 4.5.3 COBOL REST API の生成方法」を参照してください。

## 4.6.2 COBOL

コンパイラの翻訳時動作に関する設定を行うページです。

分類のドロップダウンリストから次のいずれかを選択します。

- ・「コンパイルモード」
- ・「実行モード」
- ・「実行モード」
- ・「リスト出力」
- ・「ディレクトリ」
- ・【Linux】「用語機能」
- ・「従来製品互換」
- ・【Enterprise(標準)】【Linux】「ACOS-4 互換」

【Standard】 【Enterprise(SE)】

オプション設定

ディレクトリ | 転送 | コマンドライン

全般 | COBOL | リンカ | Pro\*COBOL | リソース | C/C++ | オプション製品

分類 **コンパイルモード** 初期化

プログラムの固有文字集合

☒ SJIS ☐ JIPS(J) [-Cj]

☐ JIS2004 [-CF]

選択行指定

☐ すべての選択文字を有効にする [-Na]

☐ 指定した選択文字を有効にする [-N[0-9]]

デバッグ行

☐ 有効にする [-Dy]

☐ 無効にする [-Dn]

☐ 外部リポトリを更新する [-CG]

☐ 外部リポトリとの差分チェックを行う [-CH]

☐ 外部名の英小文字を英大文字と等価としない [-Cs]

☐ 外部データ名中のハイフンを下線と等価とする [-Cu]

☐ INITIALIZEの展開順序を85規格仕様とする [-Xi]

☐ 文字列を予約語としない [-U]

予約語表

☒ COBOL既定 [-w0]

☐ COBOL85 MFA [-w1]

☐ COBOL85 Pro [-w2]

暗黙の初期値

☐ すべてゼロで初期化 [-vi]

☐ SPACEで初期化 [-vs]

☐ ZEROで初期化 [-vz]

☐ 16進値で初期化 [-vnn]

16進値

マニフェストファイル

☐ 分離する [-q]

☐ 分離しない [-p]

☐ 警告エラーを抑制する [-Hw]

全初期化 OK キャンセル

【Enterprise(標準)】

オプション設定

ディレクトリ | 転送 | ACOS互換製品 | コマンドライン  
 全般 | COBOL | リンカ | Pro\*COBOL | リソース | C/C++ | オプション製品

分類 **コンパイルモード** 初期化

プログラムの固有文字集合

☒ SJIS ☐ JIPS(J) [-Cj]  
☐ JIS2004 [-CF] ☐ UNICODE [-CU]

選択行指定

☐ すべての選択文字を有効にする [-Na]  
☐ 指定した選択文字を有効にする [-N[0-9]]

デバッグ行

☐ 有効にする [-Dy]  
☐ 無効にする [-Dn]

☐ 外部リポトリを更新する [-CG]  
☐ 外部リポトリとの差分チェックを行う [-CH]  
☐ 外部名の英小文字を英大文字と等価としない [-Cs]  
☐ 外部データ名中のハイフンを下線と等価とする [-Cu]  
☐ INITIALIZEの展開順序を85規格仕様とする [-Xi]  
☐ 文字列を予約語としない [-U]

予約語表

☒ COBOL既定 [-w0]  
☐ COBOL85 MFA [-w1]  
☐ COBOL85 Pro [-w2]

暗黙の初期値

☐ すべてゼロで初期化 [-vi]  
☐ SPACEで初期化 [-vs]  
☐ ZEROで初期化 [-vz]  
☐ 16進値で初期化 [-vnn]  
 16進値

マニフェストファイル

☐ 分離する [-q]  
☐ 分離しない [-p]

CODE-SET句指定

☒ UTF-8 [-CC0]  
☐ UCS-2 [-CC1]  
☐ STANDARD-2 [-CC2]

☐ 警告エラーを抑制する [-Hw]

全初期化 OK キャンセル

【Linux】

オプション設定

転送	ACOS互換製品	コマンドライン
全般	COBOL	リンカ
	Pro*COBOL	C/C++
		オプション製品

分類 **コンパイルモード** 初期化

プログラムの固有文字集合

☒ SJIS
 ☐ JIPS(J) [-Cj]
 ☐ UNICODE [-CU]

選択行指定

☐ すべての選択文字を有効にする [-Na]
 ☐ 指定した選択文字を有効にする [-N[0-9]]

デバッグ行

☐ 有効にする [-Dy]
 ☐ 無効にする [-Dn]

☐ 外部リポトリを更新する [-CG]
 ☐ 外部リポトリとの差分チェックを行う [-CH]
 ☐ 警告エラーを抑制する [-Hw]

☐ 外部名の英小文字を英大文字と等価としない [-Cs]
 ☐ 外部データ名中のハイフンを下線と等価とする [-Cu]

☐ INITIALIZEの展開順序を85規格仕様とする [-Xi]
 ☐ 文字列を予約語としない [-U]

予約語表

☒ COBOL既定 [-w0]
 ☐ COBOL85 MFA [-w1]
 ☐ COBOL85 Pro [-w2]

暗黙の初期値

☐ すべてゼロで初期化 [-vi]
 ☐ SPACEで初期化 [-vs]
 ☐ ZEROで初期化 [-vz]
 ☐ 16進値で初期化 [-vnn]

16進値

CODE-SET句指定

☒ UTF-8 [-CC0]
 ☐ UCS-2 [-CC1]
 ☐ STANDARD-2 [-CC2]

全初期化 OK キャンセル

(1) コンパイルモード

項目名	内容
プログラムの固有文字集合	プログラムの固有文字集合
	・SJIS (既定値)
[-Cj]	・JISP(J)
【Standard】	
【Enterprise(SE)】	
【Enterprise(標準)】	
[-CF]	・JIS2004
【Enterprise(標準)】	
【Linux】	
[-CU]	・UNICODE

使用する予約語表 [ -w0] [ -w1] [ -w2]	使用する予約語表 ・ 本 COBOL 既定の予約語表を使用する（既定値） ・ COBOL85 MFA の予約語表を使用する ・ COBOL85 Pro の予約語表を使用する
選択行指定 [ -Na] [ -N[0-9]]	原始プログラム中の選択行を翻訳行とするか、注記行とするか。 ・ すべての選択文字を有効にする ・ 指定した選択文字を有効にする 「指定した選択文字を有効にする」を指定する場合には、テキストボックスに数字を直接入力して指定する。なお、複数の数字を指定する場合は、まとめて指定する。
暗黙の初期値 [ -vi] [ -vs] [ -vz] [ -vnn]	暗黙の初期値 ・ すべて ZERO で初期化 ・ SAPCE で初期化 ・ ZERO で初期化 ・ 16 進値で初期化 「16 進値で初期化」を指定する場合には、テキストボックスに 2 バイトの数字を入力できる。
デバッグ行 [ -Dy] [ -Dn]	デバッグ行 ・ デバッグ行を有効にする ・ デバッグ行を無効にする
【Standard】 【Enterprise(SE)】 【Enterprise(標準)】 マニフェストファイル [ -q] [ -p]	マニフェストファイル ・ マニフェストファイルを分離する ・ マニフェストファイルを分離しない ・ チェックしない（既定値）
【Enterprise(標準)】 【Linux】 CODE-SET 句指定 [ -CC0] [ -CC1] [ -CC2]	CODE-SET 句指定 ・ UTF-8 ・ UCS-2 ・ STANDARD-2 ・ チェックしない（既定値）
外部リポジトリを更新する [ -CG]	外部リポジトリを更新する ・ チェックする ・ チェックしない（既定値）
外部リポジトリとの差分チェックを行う [ -CH]	カレントフォルダーに存在する外部リポジトリとコンパイル時に生成したリポジトリ情報とに差分があるかをチェックし、差分があれば、警告エラーとする。 外部リポジトリを更新しない場合に有効。 ・ チェックする ・ チェックしない（既定値）
警告エラーを抑制する	警告(WARNING)および注意(OBSERVE)レベルの



[-Hw]	エラー出力を抑制する。 ・チェックする ・チェックしない (既定値)
外部名の英小文字を英大文字と等価としない [-Cs]	外部公開される利用者定義語のうち AS 指定のないものに対して、ソースプログラム中に書かれたとおりにオブジェクトファイルに出力する。 ・チェックする ・チェックしない (既定値)
外部データ名中のハイフンを下線と等価とする [-Cu]	外部公開名に含まれる語中のハイフンを下線と等価とみなす。 ・チェックする ・チェックしない (既定値)
INITIALIZE の展開順序を 85 規格仕様とする [-Xi]	INITIALIZE の展開順序を 85 規格仕様とする。 ・チェックする ・チェックしない (既定値)
文字列を予約語としない [-Ustr]	文字列 str を予約語とせずに翻訳する。 ・チェックする ・チェックしない (既定値)

## (2) 実行モード

項目名	内容
USAGE COMP を USAGE COMP-5 とする [-C5]	COMP データの内部表現形式を COMP-5 互換とする。 ・チェックする ・チェックしない (既定値)
不正 10 進データチェックを行わない [-Cd]	不正 10 進データの実行時チェックを行わない。 ・チェックする ・チェックしない (既定値)
【Standard】 順ファイル使用時に COBOL 独自ファイルを使用する [-Cf]	順ファイル使用時に COBOL 独自ファイルを使用する。 ・チェックする ・チェックしない (既定値)
【Standard】 相対ファイル使用時に COBOL 独自ファイルを使用する [-Cg]	相対ファイル使用時に COBOL 独自ファイルを使用する。 ・チェックする ・チェックしない (既定値)
【Standard】 索引ファイル使用時に ISAM を使用する [-Ch]	索引ファイル使用時に ISAM を使用する。 ・チェックする ・チェックしない (既定値)
添字、可変長集団項目および部分参照の範囲外参照のチェックを行わない	実行時に添字、可変長集団項目、および部分参照の範囲外参照チェックを行わない。

[-Co]	<ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
同桁同符号の外部 10 進同士の転記を英数字転記で行う [-Cr]	同桁同符号の外部 10 進数同士の転記を英数字転記で行う。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
BINARY 項目の 10 進数桁落しをしない [-Ct]	BINARY 項目へのデータ格納時、データが割り付けられた記憶領域に対応した値まで格納し、PICTURE 句桁数での桁落としを行わない。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
重ね符号桁の空白文字を不正 10 進エラーとしない [-CB]	実行時に重ね符号桁の空白文字を不正 10 進エラーとしない。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
システム日付を取得する [-CD]	システム日付を取得する。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
重ね符号桁の表現を Pro*COBOL 互換とする [-CS]	重ね符号桁の表現を Pro*COBOL 互換とする。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
実行可能プログラムに静的リンクした外部プログラムを一意名呼び出しできるようにする [-CY]	実行可能プログラムに静的リンクした外部プログラムを一意名呼び出しできるようにする。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
【Enterprise(標準)】 【Linux】 複数スレッドにて動作できるスレッドセーフなオブジェクトを生成する [-MT]	複数スレッドにて動作できるスレッドセーフなオブジェクトを生成する。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
【Linux】 位置独立コード(Position Independent Code)を作成する [-Cz]	位置独立コード(Position Independent Code)を作成する。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>

### (3) 実行モード

項目名	内容
データ記述項に SYNCHRONIZED 句が指定されたものとみなす [-Ca]	実行効率が高くなるようにデータを割り付ける。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
INITIAL 句と CANCEL 文は未使用のため、生成オブジェクトを効率化する	INITIAL 句および CANCEL 文は未使用のため、生成オブジェクトを効率化する。

[-Ci]	<ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない(既定値)</li> </ul>
実行効率を低下させる記述を指摘する [-Cm]	<p>実行効率を低下させる場所に対してメッセージを出力する。</p> <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない(既定値)</li> </ul>
CALL 定数における特定の引数を BYREFERENCE 指定とみなす [-Cv]	<p>CALL 定数における引数について、BY VALUE 指定のデータ項目が下記である場合、BY REFERENCE 指定とみなす。</p> <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない(既定値)</li> </ul> <ul style="list-style-type: none"> <li>・2文字以上の英字項目または英数字項目</li> <li>・2桁以上の外部10進数項目 (独立符号の場合は1桁以上)</li> <li>・2桁以上の内部10進数項目 (符号領域なしの場合は3桁以上)</li> <li>・日本語項目</li> <li>・日本語編集項目</li> <li>・数字編集項目</li> <li>・英数字編集項目</li> <li>・指標データ項目</li> </ul>
プリンタファイルの APPLY SHIFT-CODE 句をコメントとする [-Cy]	<p>プリンタファイルの APPLY SHIFT-CODE 句をコメントとする。</p> <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない(既定値)</li> </ul>
SYSOUT ファイルの印刷データに、プリンタ制御コードを埋め込む [-CA]	<p>SYSOUT ファイルの印刷データにプリンタ制御コードを埋め込む機能を使用する。</p> <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない(既定値)</li> </ul>
トレース情報にカバレッジ情報を出力する [-CI]	<p>トレース情報にカバレッジ情報を出力する。COBOL Assistant Option インストール時のみ有効。</p> <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない(既定値)</li> </ul>
偶数桁で符号 S がない COMP-3 項目を符号領域なしとして動作させる [-CN]	<p>項目を符号領域なし COMP-3 項目として動作させる。</p> <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない(既定値)</li> </ul>
【Enterprise(標準)】 【Linux】 SYSOUT ファイルの印刷データに、KLP 用のプリンタ制御コードを埋め込む [-CP]	<p>PPR-CONTROL-3 指定の SYSOUT ファイルの印刷データに、KLP 用のプリンタ制御コードを埋め込む</p> <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない(既定値)</li> </ul>
ON SIZE ERROR 句なしのゼロ除算結果を 0 とする [-CX]	<p>ON SIZE ERROR 句なしのゼロ除算結果を 0 とする。</p> <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない(既定値)</li> </ul>

英数字転記の作用対象が重なっている場合の動作を変更する [-Cw]	英数字転記の作用対象が重なっている場合の動作を変更する。 ・処理速度を優先した転記（既定値） ・1バイト単位で転記
最適化 [-00] [-01] [-02]	最適化。 ・最適化を行わない ・局所的に最適化(不要文の削除など)を行う（既定値） ・広域的な最適化(中間結果の有効利用など)を行う

#### (4) リスト出力

項目名	内容
リスト内に印字される行番号を外部行番号で出力する [-Ce]	診断メッセージリスト、データマップリスト、クロスリファレンスリストの中に印字される行番号を外部行番号で出力する。 ・チェックする ・チェックしない（既定値）
登録集原文展開リスト [-Hc]	登録集原文展開リストを出力する。 ・チェックする ・チェックしない（既定値）
データマップリスト [-Hd]	データマップリストを出力する。 ・チェックする ・チェックしない（既定値）
【Linux】 用語値名参照リスト [-He]	用語値名参照リストを出力する。 ・チェックする ・チェックしない（既定値）
【Linux】 用語の展開リスト [-Hf]	用語の展開リストを出力する。 ・チェックする ・チェックしない（既定値）
【Linux】 用語名リスト [-Hm]	用語名リストを出力する。 ・チェックする ・チェックしない（既定値）
パラメータリスト [-Hp]	パラメータリストを出力する。 ・チェックする ・チェックしない（既定値）
原始プログラムリスト [-Hs]	原始プログラムリストを出力する。 ・チェックする ・チェックしない（既定値）
クロスリファレンスリスト [-Hx]	クロスリファレンスリストを出力する。 ・チェックする ・チェックしない（既定値）
【Standard】 【Enterprise(SE)】 翻訳リストファイル	翻訳リストファイルまたは翻訳リストファイルの格納場所を指定する。

[-Yf]	<ul style="list-style-type: none"> <li>・パスありファイル名</li> <li>・パスなしファイル名</li> <li>・パス名のみ</li> </ul>
-------	--

#### (5) ディレクトリ

項目名	内容
登録集原文ディレクトリ [-I]	登録集原文ディレクトリ 直接入力可能。 相対パスと絶対パス両方で指定が可能。
外部リポジトリ検索ディレクトリ [-R]	リポジトリ段落で参照する外部リポジトリファイルの検索フォルダー 直接入力可能。 相対パスと絶対パス両方で指定が可能。

#### (6) 用語機能

項目名	内容
【Linux】 用語辞書データベース名 [-T]	用語辞書データベース名を最大 256 バイトで指定する。 直接入力可能。

#### (7) 従来製品互換

項目名	内容
連絡節に GLOBAL 句の記述を可能とする [-Qa]	連絡節に GLOBAL 句の記述を可能とする。 ・チェックする ・チェックしない (既定値)
ベースデータ節に GLOBAL 句の記述を可能とする [-Qb]	ベースデータ節に GLOBAL 句の記述を可能とする。 ・チェックする ・チェックしない (既定値)
指標名や指標データ項目を構成する出現番号と指標値を COMP-2 とする [-Qc]	指標名や指標データ項目を構成する出現番号と指標値を COMP-2 とする。 ・チェックする ・チェックしない (既定値)
【Standard】 【Enterprise(SE)】 可変繰り返し回数が最大として COMBINE 文を実行する [-Qd]	可変繰り返し回数が最大として COMBINE 文を実行する。 ・チェックする ・チェックしない (既定値)
行順ファイルのレコード右端の空白を削除して書き出す [-Qe]	行順ファイルのレコード右端の空白を削除して書き出す。

	<ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
<b>【Standard】</b> <b>【Enterprise(SE)】</b> LINAGE 句の論理ページサイズの評価タイミングを変更する [-Qg]	LINAGE 句の論理ページサイズの評価タイミングを変更する。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
<b>【Standard】</b> <b>【Enterprise(SE)】</b> NUMERIC 検査の結果を変更する [-Qh]	NUMERIC 検査の結果を変更する。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
<b>【Standard】</b> <b>【Enterprise(SE)】</b> SYSIN ファイルの入力終了条件を指定する [-Qi]	SYSIN ファイルの入力終了条件を指定する。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
<b>【Standard】</b> <b>【Enterprise(SE)】</b> UNSTRING 文の P 付き整数項目の計数結果を変更する [-Qj]	UNSTRING 文の P 付き整数項目の計数結果を変更する。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
<b>【Standard】</b> <b>【Enterprise(SE)】</b> CLOSE 文の WITH LOCK 句をコメントとする [-Ql]	CLOSE 文の WITH LOCK 句をコメントとする。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
プログラム名と利用者定義語の同名定義を可能とする。 [-Qm]	プログラム名と利用者定義語の同名定義を可能とする。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
OPEN 文の入出力状態および永続誤り発生後の動作を変更する [-Qo]	OPEN 文の入出力状態および永続誤り発生後の動作を変更する。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
行順ファイル(CSV なし)に対するファイル共用機能を使用可能とする [-Qp]	行順ファイル(CSV なし)に対するファイル共用機能を使用可能とする。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>
<b>【Linux】</b> 行順ファイル(CSV あり)に対するファイル共用機能を使用可能とする [-Qq]	行順ファイル(CSV あり)に対するファイル共用機能を使用可能とする。 <ul style="list-style-type: none"> <li>・チェックする</li> <li>・チェックしない (既定値)</li> </ul>

(8) ACOS-4 互換

項目名	内容
【Enterprise(標準)】 【Linux】 データ項目の大きさとアラインを ACOS-4 互換とする [-Fa]	データ項目の大きさとアラインを ACOS4 互換とする。 ・チェックする ・チェックしない (既定値)
【Enterprise(標準)】 【Linux】 データポインタデータ項目の割り付け境界を ACOS-4 互換とする [-Fb]	データポインタデータ項目の割り付け境界を ACOS-4 互換とする。 ・チェックする ・チェックしない (既定値)
【Enterprise(標準)】 【Linux】 COMP 項目の内部表現形式を ACOS-4 互換の外部 10 進数とする [-Fc]	COMP 項目の内部表現形式を ACOS-4 互換の外部 10 進数とする。 ・チェックする ・チェックしない (既定値)
【Enterprise(標準)】 【Linux】 比較規則を ACOS-4 互換とする [-Fd]	比較規則を ACOS-4 互換とする。 ・チェックする ・チェックしない (既定値)
【Enterprise(標準)】 【Linux】 NUMERIC 検査を ACOS-4 互換とする [-Fe]	NUMERIC 検査を ACOS-4 互換とする。 ・チェックする ・チェックしない (既定値)
【Enterprise(標準)】 【Linux】 小入出力先の既定値を ACOS-4 互換とする [-Fj]	小入出力先の既定値を ACOS-4 互換とする。 ・チェックする ・チェックしない (既定値)
【Enterprise(標準)】 【Linux】 STOP 文の ABORT 指定を可能とする [-FI]	STOP 文の ABORT 指定を可能とする。 ・チェックする ・チェックしない (既定値)
【Enterprise(標準)】 【Linux】 中間結果格納領域を小数部桁数に基づく固定値とする [-Fr]	中間結果格納領域を小数部桁数に基づく固定値とする。 ・チェックする ・チェックしない (既定値) 「小数部桁数に基づく固定値」を指定する場合には、テキストボックスに 1 から 30 の数字を入力できる。

<b>【Enterprise(標準)】</b> <b>【Linux】</b> 英数字から外部 10 進数への 数字転記においてゾーン部の 補正を行わない [-Fs]	英数字から外部 10 進数への数字転記におい てゾーン部の補正を行わない。 ・チェックする ・チェックしない(既定値)
<b>【Enterprise(標準)】</b> <b>【Linux】</b> 符号なし外部 10 進数を符号 付きとして参照する [-Ft]  [-Fx]	符号なし外部 10 進数を符号付きとして参照 する。 ・英数字項目または日本語項目への転記を除 かない ・英数字項目または日本語項目への転記を除 く ・チェックしない(既定値)
<b>【Enterprise(標準)】</b> SYSOUT ファイルに改行位置の 絶対行指定を可能とする [-Fv]	SYSOUT ファイルに改行位置の絶対行指定を 可能とする。 ・チェックする ・チェックしない(既定値)
<b>【Enterprise(標準)】</b> PPR-CONTROL-1 指定の SYSOUT ファイルの印刷データにシフ トコードおよびプリンタ制御 コードを埋め込む [-Fy]	PPR-CONTROL-1 指定の SYSOUT ファイルの印 刷データにシフトコードおよびプリンタ制御 コードを埋め込む。 ・チェックする ・チェックしない(既定値)
<b>【Enterprise(標準)】</b> ROUNDED 句指定のある COMPUTE 文において中間結果 格納領域の算出方法を ACOS-4 互換とする [-Fw]	ROUNDED 句指定のある COMPUTE 文において中 間結果格納領域の算出方法を ACOS-4 互換と する。 ・チェックする ・チェックしない(既定値)
<b>【Enterprise(標準)】</b> データ例外が発生する手続き 文を ACOS-4 互換とする [-Fz]	データ例外が発生する手続き文を ACOS-4 互 換とする。 ・チェックする ・チェックしない(既定値)

### 4.6.3 リンカ

リンクに関する設定をするページです。

**【Standard】【Enterprise(SE)】【Enterprise(標準)】**

このページで指定するファイル名については、オブジェクトファイルの拡張子は.obj、  
ライブラリファイルの拡張子は.lib です。





#### 【Linux】

このページで指定するファイル名については、オブジェクトファイルの拡張子は.o、静的ライブラリファイルの拡張子は.a、動的ライブラリの拡張子は.so です。

オプション設定

転送		ACOS互換製品		コマンドライン	
全般	COBOL	リンカ	Pro*COBOL	C/C++	オプション製品
					初期化
リンクライブラリ検索ディレクトリ					
<input type="text"/>					一覧...
リンクライブラリ					
<input type="text"/>					一覧...
リンカオプション					
<input type="text"/>					
全初期化					OK    キャンセル

## 全般

項目名	内容
リンクライブラリ検索ディレクトリ	リンクライブラリ検索ディレクトリ。
リンクライブラリ	リンクライブラリ。 最大 4096 バイトの任意の内容の文字列で指定する。
リンカオプション	リンカのオプションを指定する。 最大 255 バイトの任意の文字列で指定する。 リンカオプションに入力した内容に対するチェックは行いません。リンカで使用可能なオプションを確認の上、入力してください。
<b>【Standard】</b> <b>【Enterprise(SE)】</b> <b>【Enterprise(標準)】</b> 使用する製品名	関連製品がインストールされている場合に製品名が表示される。 ・チェックする ・チェックしない(既定値)

	：チェックした場合、リンカオプションに関連製品(例：COBOL 拡張システムサブルーチン)のライブラリを追加する。
--	---

#### <ディレクトリの選択>

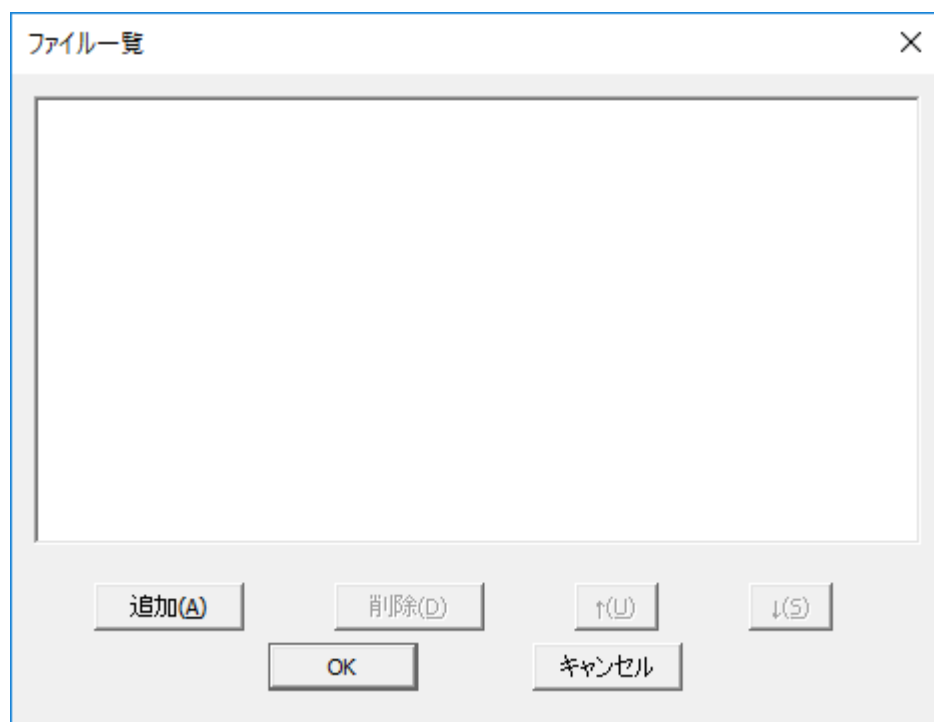
ディレクトリを入力するテキストボックスにおいて、フォルダー選択用の〔一覧〕ボタンが右端に表示されます。

〔一覧〕ボタンをクリックした場合の操作については、「**ディレクトリの選択方法**」の説明を参照してください。

#### <リンクファイルの選択>

リンクライブラリファイルを入力するテキストボックスにおいて、ファイル選択用の〔一覧〕ボタンが右端に表示されます。

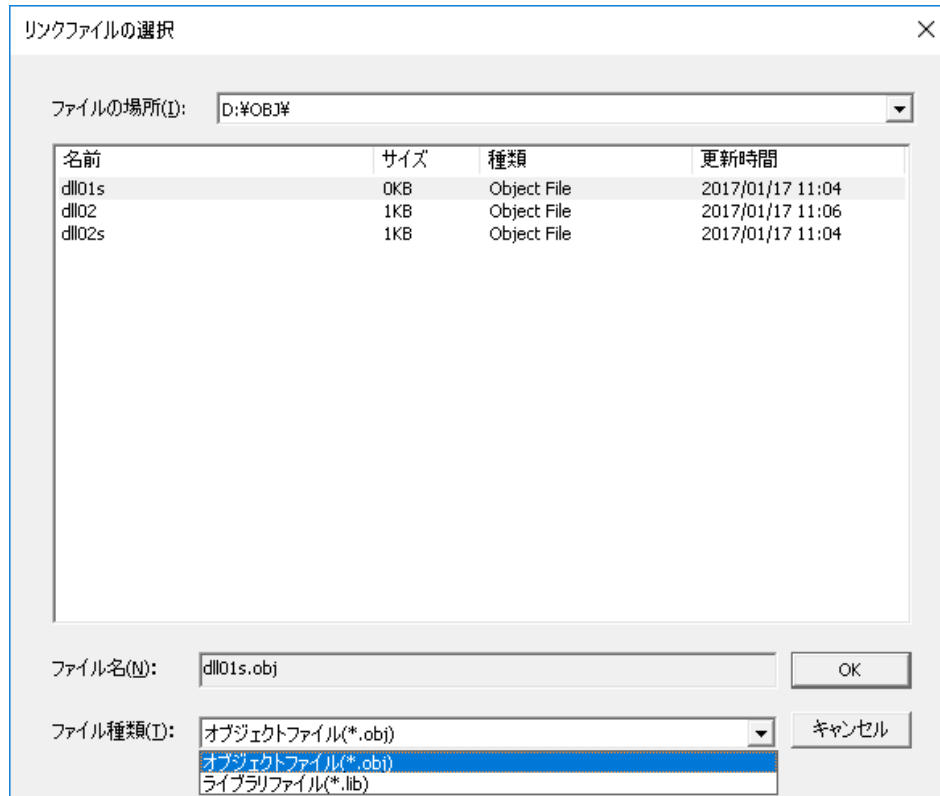
〔一覧〕ボタンをクリックすると、以下のような〔ファイル一覧〕ダイアログボックスが表示されます。



ファイルを追加する場合には、〔追加〕ボタンをクリックすると〔リンクファイルの選択〕ダイアログボックスが表示されます。〔ファイルの場所〕のリストからリンクライブラリ検索ディレクトリに指定したフォルダーを選択し、〔ファイル種類〕を選択すると、対象のファイル名が一覧に表示されます。リンクするファイルを選択し、〔OK〕ボタン

をクリックします。

[ キャンセル ] ボタンをクリックすると、ファイルの追加がキャンセルされます。



選択したファイルをリストから削除する場合には、削除するファイル名を選択し、[ 削除 ] ボタンをクリックします。

選択したファイルの順序を変更する場合には、変更するファイル名を選択し、[ ] [ ] ボタンをクリックします。

[ OK ] ボタンをクリックすると、その内容がオプション設定画面に反映されます。

[ キャンセル ] ボタンをクリックすると、行なった変更が取り消されます。

### 注意事項

・「;」を含むディレクトリおよびリンクライブラリは指定しないでください。指定された場合、アプリケーションの生成が失敗する場合があります。

### 【Linux】注意事項

・ビルド時に存在しないファイルはリンクの対象になりません。

#### 4.6.4 Pro\*COBOL

Pro\*COBOL に関する設定をするページです。

【Standard】【Enterprise(SE)】【Enterprise(標準)】

オプション設定

ディレクトリ | 転送 | ACOS互換製品 | コマンドライン

全般 | COBOL | リンカ | Pro\*COBOL | リソース | C/C++ | オプション製品

初期化

Pro\*COBOL オプション

COBOLファイル拡張子選択 .cob

全初期化 OK キャンセル

【Linux】

オプション設定

転送 | ACOS互換製品 | コマンドライン

全般 | COBOL | リンカ | Pro\*COBOL | C/C++ | オプション製品

初期化

Pro\*COBOL オプション

COBOLファイル拡張子選択: .cob

ライブラリ選択: build

全初期化 OK キャンセル

全般

項目名	内容
Pro*COBOL オプション	Pro*COBOL オプション。 最大 255 バイトの任意の内容の文字列で指定する。 Pro*COBOL オプションに入力した内容に対するチェックは行わない。Pro*COBOL で使用可能なオプションを確認の上、入力する。
COBOL ファイル拡張子選択	COBOL 原始ファイル拡張子選択 ・ .cob (既定値) ・ .cbl
【Linux】 ライブラリ選択	ライブラリ選択 ・ build (既定値) ・ build_static

4.6.5 【Standard】【Enterprise(標準)】【Enterprise(SE)】リソース

リソースに関する設定をするページです。

オプション設定

ディレクトリ | 転送 | ACOS互換製品 | コマンドライン  
全般 | COBOL | リンカ | Pro\*COBOL | リソース | C/C++ | オプション製品

初期化

リソースコンパイラオプション

全初期化 OK キャンセル

**全般**

項目名	内容
リソースコンパイラオプション	リソースコンパイラオプション。 最大 255 バイトの任意の内容の文字列で指定する。 リソースコンパイラオプションに入力した内容に対するチェックは行いません。リソースコンパイラで使用可能なオプションを確認の上、入力する。

### 4.6.6 C/C++

C/C++に関する設定をするページです。

オプション設定

ディレクトリ | 転送 | ACOS互換製品 | コマンドライン  
全般 | COBOL | リンカ | Pro\*COBOL | リソース | C/C++ | オプション製品

初期化

C/C++オプション

インクルードディレクトリ

一覧...

全初期化 OK キャンセル

**全般**

項目名	内容
C/C++オプション	C/C++オプション。 最大 255 バイトの任意の内容の文字列で指定する。 C/C++オプションに入力した内容に対するチェックは行わない。C/C++で使用可能なオプションを確認の上、入力する。
インクルードディレクトリ	インクルードディレクトリ



#### <ディレクトリの選択>

ディレクトリを入力するテキストボックスにおいて、フォルダー選択用の[一覧]ボタンが右端に表示されます。

[一覧]ボタンをクリックした場合の操作については、  
「ディレクトリの選択方法」の説明を参照してください。

### 4.6.7 オプション製品

オプション製品に関する設定をするページです。

【Standard】

【Enterprise(標準)】【Linux】

オプション設定

ディレクトリ | ACOS互換製品 | コマンドライン

全般 | COBOL | リンカ | Pro\*COBOL | リソース | C/C++ | オプション製品

初期化

COBOL SQLアクセス  
SQLオプション

GMPとの連携時の拡張子選択 .qcob

全初期化 OK キャンセル

## 【Enterprise(SE)】

オプション設定

ディレクトリ			転送		コマンドライン	
全般	COBOL	リンカ	Pro*COBOL	リソース	C/C++	オプション製品

初期化

COBOL SQLアクセス  
SQLオプション

全初期化      OK      キャンセル

## COBOL GUI

項目名	内容
<b>【Standard】</b> COBOL GUI アプリケーションを作成する [-Pg]	GUI画面機能を利用したCOBOL GUIアプリケーションを作成する。 ・チェックする ・チェックしない（既定値）
<b>【Standard】</b> 画面節の実行で COBOL GUIを使用する [-P2]	画面節の実行でCOBOL GUIを使用する。 ・チェックする ・チェックしない（既定値）
<b>【Standard】</b> 画面情報ファイル名	COBOL GUIアプリケーションを作成するときに

	<p>必要な画面情報ファイル名。          拡張子を省略した場合、自動で .scn が補完される。          最大 255 バイトの任意の内容の文字列で指定する。</p>
--	---

#### COBOL SQL

項目名	内容
SQL オプション	<p>COBOL SQL オプション。          最大 255 バイトの任意の内容の文字列で指定する。          SQL オプションに入力した内容に対するチェックは行わない。COBOL SQL で使用可能なオプションを確認の上、入力する。</p>
<b>【Enterprise(標準)】</b> <b>【Linux】</b> GMP との連携時の拡張子選択	<p>GMP との連携時の拡張子選択。</p> <ul style="list-style-type: none"> <li>・ qcob (既定値)</li> <li>・ qcb1</li> </ul>

# 4.6.8 【Standard】【Enterprise(標準)】【Enterprise(SE)】ディレクトリ

中間成果物と成果物の格納場所に関する設定をするページです。

オプション設定

全般

COBOL

リンカ

Pro\*COBOL

リソース

C/C++

オプション製品

ディレクトリ

転送

コマンドライン

初期化

オブジェクトファイル

参照...

実行可能ファイル

参照...

全初期化

OK

キャンセル

## 全般

項目名	内容
オブジェクトファイル	オブジェクトファイルの格納場所 最大 255 バイトの任意の内容の文字列で指定する。
実行可能ファイル	実行可能ファイルの格納場所 最大 255 バイトの任意の内容の文字列で指定する。

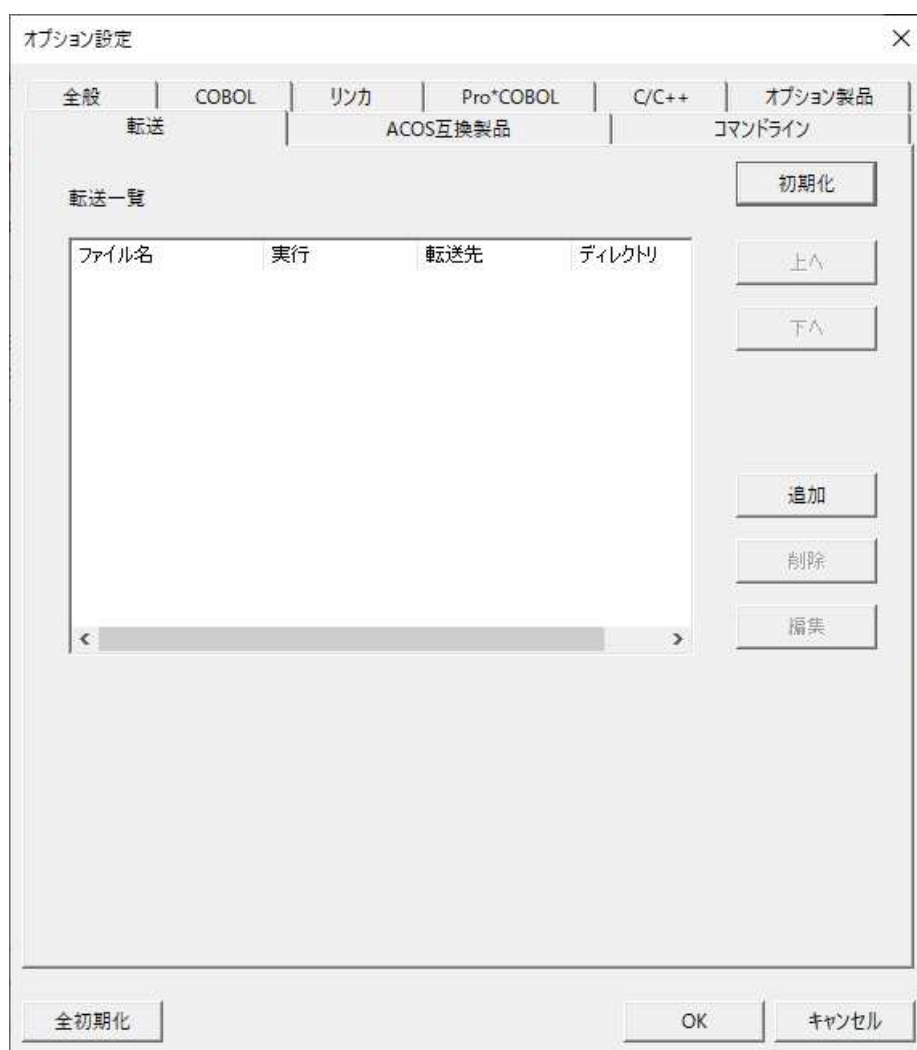
## 注意事項

- ・ 既定の生成ディレクトリ「Release」と同じディレクトリを指定すると、成果物を正しく生成できないため、別の名前を指定してください。

## 4.6.9 転送

COBOL 開発環境のリモートビルドによって作成されたファイルを、所定の場所に転送する機能について、設定するページです。自動転送を設定すると、ビルド後に転送が行われます。

【Enterprise(標準)】 【Linux】



【Enterprise(SE)】 【Standard】

オプション設定

全般 | COBOL | リンカ | Pro\*COBOL | リソース | C/C++ | オプション製品

ディレクトリ | 転送 | コマンドライン

転送一覧

ファイル名	実行	転送先	ディレクトリ
-------	----	-----	--------

初期化

上へ

下へ

追加

削除

編集

全初期化

OK

キャンセル

全般

項目名	内容
ファイル名	<p>ファイル名を直接入力する、 ファイル名をワイルドカードで直接入力、または次のいずれかをドロップダウンリストから選択する。</p> <p>【Linux】</p> <p>「実行可能形式ファイル」</p> <p>「オブジェクトファイル(*.o)」</p> <p>「静的ライブラリ(*.a)」</p> <p>「動的ライブラリ(*.so)」</p> <p>「外部リポジトリ(*.rp?)」</p> <p>「コンパイルリスト(*.list)」</p> <p>「COBOL/S リスト(*.lst)」</p> <p>「Pro*COBOL リスト(*.lis)」</p>

	<p>「拡張部品リスト(*.rst)」</p> <p>「拡張部品エラーファイル(*.experr)」</p> <p>「&lt;参照&gt;」</p> <p>【Enterprise(標準)】</p> <p>「実行可能形式ファイル」</p> <p>「オブジェクトファイル(*.obj)」</p> <p>「静的ライブラリ(*.lib)」</p> <p>「動的ライブラリ(*.dll)」</p> <p>「外部リポジトリ(*.rp?)」</p> <p>「コンパイルリスト(*.list)」</p> <p>「COBOL/S リスト(*.lst)」</p> <p>「Pro*COBOL リスト(*.lis)」</p> <p>「拡張部品リスト(*.rst)」</p> <p>「拡張部品エラーファイル(*.experr)」</p> <p>「&lt;参照&gt;」</p> <p>【Enterprise(SE)】</p> <p>【Standard】</p> <p>「実行可能形式ファイル」</p> <p>「オブジェクトファイル(*.obj)」</p> <p>「静的ライブラリ(*.lib)」</p> <p>「動的ライブラリ(*.dll)」</p> <p>「外部リポジトリ(*.rp?)」</p> <p>「コンパイルリスト(*.list)」</p> <p>「&lt;参照&gt;」</p> <p>「実行可能ファイル」を選択すると、設定時のファイル名を登録する。設定後にプロジェクト名や成果物の名前を変更しても、連動して転送対象のファイル名は変更されない。</p> <p>「&lt;参照&gt;」を選択して指定するには、ビルドサーバーにログインしておく必要がある。</p>
自動転送する	「自動転送する」選択した場合、ビルド終了時に自動的に転送される。
転送先	<p>次のいずれかをドロップダウンリストから選択する。</p> <p>「コピー」</p> <p>【Linux】</p> <p>「クライアントへ転送」</p>
ディレクトリ	<p>【Enterprise(標準)】</p> <p>【Enterprise(SE)】</p> <p>【Standard】</p> <p>転送先を「コピー」とした場合 任意のディレクトリを絶対パスで指定する。</p> <p>【Linux】</p> <p>転送先を「コピー」とした場合 ビルドサーバー内の任意のディレクトリを絶対パスで指定する。UNC パス指定には対応しない。</p>



	<p>ビルドサーバー内の任意のディレクトリを絶対パスを指定する。  [ 参照 ] ボタンを選択して指定するには、ビルドサーバーにログインしておく必要がある。</p> <p>転送先を「クライアントへ転送」とした場合クライアント内の任意のディレクトリを指定する。  プロジェクトファイルがあるフォルダーからの相対パスを記述することができる。</p>
--	--

### < 転送一覧 >

転送の設定を追加する場合には、[ 追加 ] ボタンをクリックします。

選択した設定をリストから削除する場合には、削除する設定の行を選択し、[ 削除 ] ボタンをクリックします。

選択した設定の内容を編集する場合には、編集する設定の行を選択し、[ 編集 ] ボタンをクリックします。

選択した設定の順序を変更する場合には、変更する設定の行を選択し、[ 上へ ] [ 下へ ] ボタンをクリックします。

### < 転送設定の追加 / 編集 >

転送画面で [ 追加 ] ボタンをクリックすると、以下のような「転送の追加」ダイアログボックスが表示されます。[ 編集 ] ボタンの場合は、「転送の編集」ダイアログボックスが表示されます。

The image shows a dialog box titled "転送の追加" (Add Transfer). It has a close button (X) in the top right corner. Inside the dialog, there are four main input areas: "ファイル名" (File Name) with a dropdown arrow, a checkbox labeled "自動転送する" (Automatic Transfer), "転送先" (Transfer Destination) with a dropdown arrow, and "ディレクトリ" (Directory) with a text input field and a "参照" (Reference) button to its right. At the bottom of the dialog, there are two buttons: "OK" and "キャンセル" (Cancel).

転送の編集の場合は編集対象の設定内容が項目に表示されます。

【Linux】

サーバーのファイルやディレクトリを選択する場合は、転送設定の前にサーバーへログインしてください。

[ファイル名] リストから「<参照>」を選択すると、サーバーの「ファイルの選択」ダイアログボックスが表示されます。

[参照]ボタンをクリックすると、[転送先]が「コピー」の場合はサーバーの「ディレクトリの選択」ダイアログボックスが表示され、「クライアントへ転送」の場合はクライアントの「フォルダーの参照」ダイアログボックスが表示されます。

【Enterprise(標準)】【Enterprise(SE)】【Standard】

[ファイル名] リストから「<参照>」を選択すると、「ファイルの選択」ダイアログボックスが表示されます。

[参照]ボタンをクリックすると、「フォルダーの参照」ダイアログボックスが表示されます。

すべての項目に入力して [OK] ボタンをクリックします。

[キャンセル] ボタンをクリックすると、入力内容がキャンセルされます。

#### 4.6.10 【Enterprise(標準)】【Linux】ACOS 互換製品

GMP、COBOL/S、COBOL 拡張部品に関する設定をするページです。

分類のドロップダウンリストから次のいずれかを選択します。

- ・「GMP」
- ・「COBOL/S」
- ・「COBOL 拡張部品」

オプション設定

全般 | COBOL | リンカ | Pro\*COBOL | リソース | C/C++ | オプション製品

ディレクトリ | ACOS互換製品 | コマンドライン

分類: GMP

初期化

GMPオプション

マクロオブジェクト登録ディレクトリ

一覧...

全初期化 OK キャンセル

**(1) GMP**

項目名	内容
GMP オプション	GMP オプション。 最大 255 バイトの任意の内容の文字列で指定する。 GMP オプションに入力した内容に対するチェックは行わない。GMP で使用可能なオプションを確認の上、入力する。
マクロオブジェクト登録ディレクトリ	マクロオブジェクト登録ディレクトリ。 ; で区切り最大 4 つまで指定する。

**(2) COBOL/S**

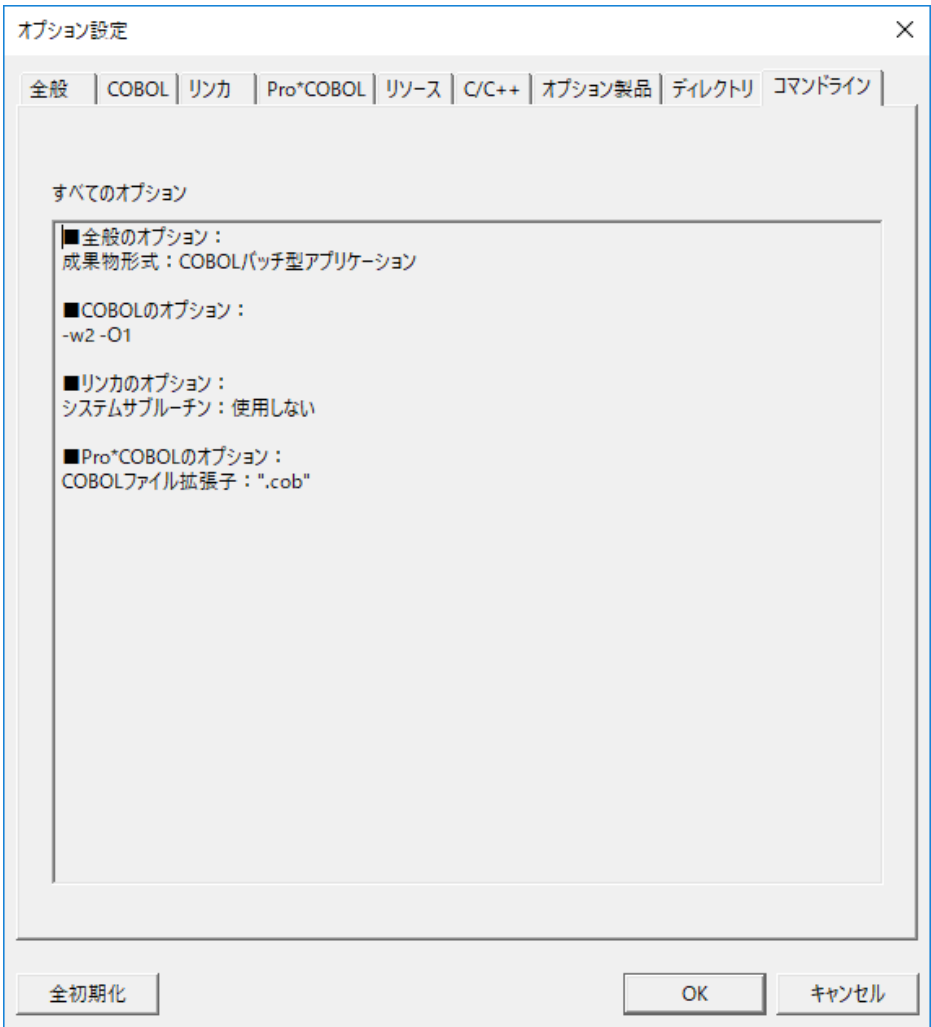
項目名	内容
COBOL/S オプション	COBOL/S オプション。 最大 255 バイトの任意の内容の文字列で指定する。 COBOL/S オプションに入力した内容に対するチェックは行わない。COBOL/S で使用可能なオプションを確認の上、入力する。
マクロ原文ファイル検索ディレクトリ	マクロ原文ファイル検索ディレクトリ。 ; で区切り最大 4 つまで指定する。

**(3) COBOL 拡張部品**

項目名	内容
【Enterprise(標準)】 エラーファイル格納場所指定	エラーファイル格納場所指定。 最大 255 バイトのパス名を指定する。
【Enterprise(標準)】 作業ディレクトリ	作業ディレクトリ。 最大 255 バイトのパス名を指定する。
部品仕様格納場所指定	部品仕様格納場所指定。 最大 780 バイトのパス名を指定する。 ; で区切り最大 3 つまで指定する。
データ仕様格納場所指定	データ仕様格納場所指定。 最大 780 バイトのパス名を指定する。 ; で区切り最大 3 つまで指定する。
リスト情報指定	リスト情報指定。 ・チェックする ・チェックしない(既定値)

### 4.6.11 コマンドライン

設定されたすべてのオプションを表示するページです。



#### 全般

項目名	内容
すべてのオプション	設定されたすべてのオプションを表示する。 同一タブにあるオプションをスペースで区別して、表示する。

## 4.6.12 入力文字の制限一覧

オプションのテキストボックス項目について、入力文字数および、入力可能文字の制限を示します。

### 〔全般〕

項目名	入力文字数の制限	入力可能文字の制限
成果物のファイル名	255 バイト	ファイル名には 「 / ? : , ¥ ¥ * ¥ " < >   # % ; \$ ^ [ 」が入力不可  【Linux】「 ( ) & ` ' 」も入力不可
レジストリキー名	128 バイト	

### 〔COBOL〕 - 〔コンパイルモード〕

項目名	入力文字数の制限	入力可能文字の制限
選択文字	10 バイト	0 から 9 までの文字列
16 進定数	2 バイト	
文字列を予約語としない	256 バイト	

### 〔COBOL〕 - 〔リスト出力〕

項目名	入力文字数の制限	入力可能文字の制限
翻訳リストファイル	258 バイト	ファイル名には「 ¥ / : * ? " < >   , ; # % 」が入力不可 パス名には「 / : * ? " < >   , ; # % 」が入力不可

### 〔COBOL〕 - 〔ACOS-4 互換〕

項目名	入力文字数の制限	入力可能文字の制限
中間結果格納領域を小数部桁数に基づく固定値とする	2 バイト	1 ~ 30 までの文字列

### 〔リンカ〕

項目名	入力文字数の制限	入力可能文字の制限
リンクライブラリ検索ディレクトリ		「 / ? : , * " < >   # % 」が入力不可
リンクライブラリ	4096 バイト	「 / ? : , ¥ * " < >   # % ; \$ ^ [ 」が入力不可
リンクオプション	255 バイト	

#### 【 Pro\*COBOL 】

項目名	入力文字数の制限	入力可能文字の制限
Pro*COBOL オプション	255 バイト	

#### 【 リソース 】

項目名	入力文字数の制限	入力可能文字の制限
リソースコンパイラオプション	255 バイト	

#### 【 C/C++ 】

項目名	入力文字数の制限	入力可能文字の制限
C/C++オプション	255 バイト	

#### 【 転送 】

項目名	入力文字数の制限	入力可能文字の制限
ファイル名	255 バイト	<b>【Standard】</b> <b>【Enterprise(SE)】</b> <b>【Enterprise(標準)】</b> 「 ¥ / : " < >   」が入力不可  <b>【Linux】</b> クライアントへ転送には 「 ¥ / : " < >   」 コピーには 「 /   」が入力不可
ディレクトリ	247 バイト	<b>【Standard】</b> <b>【Enterprise(SE)】</b> <b>【Enterprise(標準)】</b> 「 / : * ? " < >   」が入力不可  <b>【Linux】</b> クライアントへ転送には

		「 / : * ? " < >   」 コピーには 「   」が入力不可
--	--	--

**〔オプション製品〕 - 〔COBOL GUI〕**

項目名	入力文字数の制限	入力可能文字の制限
画面情報ファイル名	258 バイト	ファイル名には「 / : * ? " < >   , ; # % 」が入力不可 パス名には「 / : * ? " < >   , ; # % 」が入力不可
SQL オプション	255 バイト	

**〔ACOS 互換製品〕 - 〔GMP〕**

項目名	入力文字数の制限	入力可能文字の制限
GMP オプション	255 バイト	
マクロオブジェクト登録 ディレクトリ	1041 バイト	「 / ? : , * " < >   # % 」 が入力不可

**〔ACOS 互換製品〕 - 〔COBOL/S〕**

項目名	入力文字数の制限	入力可能文字の制限
COBOL/S オプション	255 バイト	
マクロ原文ファイル検索 ディレクトリ	1041 バイト	「 / ? : , * " < >   # % 」 が入力不可

**〔ACOS 互換製品〕 - 〔COBOL 拡張部品〕**

項目名	入力文字数の制限	入力可能文字の制限
エラーファイル格納場所 指定	255 バイト	「 / ? : , * " < >   # % ; 」が入力不可
作業ディレクトリ	255 バイト	「 / ? : , * " < >   # % ; 」が入力不可
部品仕様格納場所指定	780 バイト	「 / ? : , * " < >   # % 」 が入力不可
データ仕様格納場所指定	780 バイト	「 / ? : , * " < >   # % 」 が入力不可

制限のあるテキストボックス項目に対するチェックは、各テキストボックスがアクティブでなくなった場合、または [ OK ] ボタンで確定した場合に行なわれます。

チェックによって、エラーと判定された場合には、エラーダイアログボックスが表示さ



れます。

#### **注意事項**

- ・ COBOL 開発環境の動作モードによって、ファイル名に使用できる文字が異なります。  
ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。

## 4.7 ソースファイルのオプション設定

---

プロジェクトのオプション設定では、登録されたすべてのソースファイルに対して同じオプションが指定されますが、ソースファイル固有で設定できるオプションがあります。

成果物の形式がアプリケーションの場合に、1つのCOBOLソースファイルをメインプログラムであることを指定するオプションと、EXTERNAL 句の記述されたデータを参照とするか定義とするかを指定するオプションです。

また、ソースファイルをビルド対象にするかの判断のための依存ファイルを指定します。

以下に各指定について説明します。

### 4.7.1 メインプログラム

成果物がアプリケーションの場合に、プロジェクトに登録されたCOBOLソースファイルの中の1つをメインプログラムに指定できます。

COBOLソースファイルをメインプログラムに指定する手順を説明します。

1. ファイルビューで、メインプログラムにするソースファイルを選択します。  
(「ソースファイル」ノードの下にCOBOLソースファイルが対象です)
2. マウスを右クリックして表示されるメニューから、[メインプログラムに設定] コマンドを選択します。
3. ファイルビューに表示されるソースファイルのアイコンが変更されます。  
別のソースファイルがメインプログラムに指定されていた場合は、そのファイルのメインプログラムの指定が取り消され、アイコンが変更されます。
4. メインプログラムの指定を取り消すには、メインプログラムの指定と同じ操作を行います。

## 4.7.2 EXTERNAL 句の参照と定義

プロジェクトに登録された COBOL ソースファイルに対して、EXTERNAL 句の記述されたデータを実行ファイル間で共有するために、データ定義を参照とするか定義とするかを指定できます。

COBOL ソースファイルのデータ定義を指定する手順を説明します。

1. ファイルビューで、参照または定義を指定するソースファイルを選択します。  
(「ソースファイル」ノードの下 COBOL ソースファイルが対象です)
2. マウスを右クリックして表示されるメニューから、[ 参照の設定 ] コマンドまたは [ 定義の設定 ] コマンドを選択します。
3. ファイルビューに表示されるソースファイル名の右側に、参照の設定の場合は[-Si]、定義の設定の場合は[-Sx]の文字が追加されます。  
1つのソースファイルに対してはどちらか1つの指定しかできないため、一方が設定されると、もう一方の指定は取り消されます。
4. 指定を取り消すには、指定と同じ操作を行います。

## 4.7.3 依存ファイル

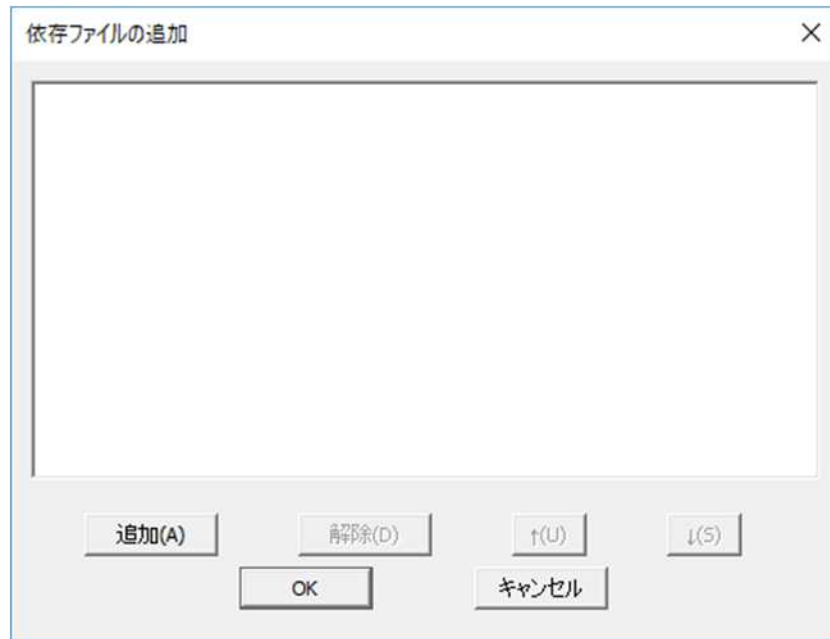
依存ファイルとは、直接コンパイルの対象ではないものの、ビルド時に必要となるファイルです。

依存ファイルを指定する目的は、マクロや登録集原文ファイルが変更された場合、それを参照しているソースファイルもコンパイルする必要があるためです。

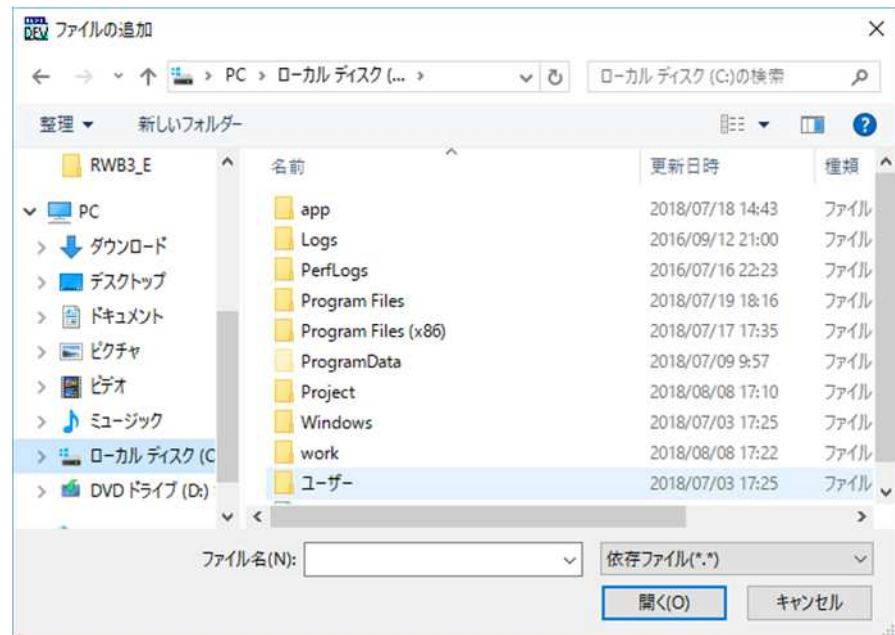
たとえば、マクロや登録集原文ファイルを依存ファイルとして指定することにより、依存関係によりソースファイルのコンパイルが行われるようになります。依存ファイルの日付がソースファイルの日付より新しい場合のみコンパイルが行われます。

依存ファイルを指定する手順を説明します。

1. ファイルビューで、依存ファイルの追加を行いたいファイルを選択します。  
(「ソースファイル」ノードの下のソースファイルが対象です)
2. マウスを右クリックして表示されるメニューから、[ 依存ファイルの追加 ] コマンドを選択します。
3. 「依存ファイルの追加」ダイアログボックスが表示されます。



4. 依存ファイルを選択します。依存ファイルは複数選択することができます。
- ・ ファイルを追加する場合には、[ 追加 ] ボタンをクリックし、そこで表示される以下のような [ ファイルの追加 ] ダイアログボックスでファイルを選択し、[ 開く ] ボタンをクリックします。  
[ キャンセル ] ボタンをクリックすると、ファイルの追加がキャンセルされます。



- ・ 選択したファイルを依存ファイルから解除する場合には、解除するファイル名を選択し、[ 解除 ] ボタンをクリックします。
- ・ 選択したファイルの順序を変更する場合には、変更するファイル名を選択し、[ ] [ ] ボタンをクリックします。
- ・ [ OK ] ボタンをクリックすると、その内容がファイルビューに反映されます。
- ・ [ キャンセル ] ボタンをクリックすると、行なった変更が取り消されます。

### **注意事項**

- ・ プロジェクトに登録するファイル名に使用できない文字を含むファイル名を登録すると、プロジェクトの操作などが正しく行えない場合があります。
- ・ ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。

## 4.8 ファイルの操作

---

ファイルビューでは、プロジェクト配下のファイルに対して以下の操作が可能です。

- ・ 名前の変更
- ・ 削除
- ・ エクスプローラーでフォルダーを開く

### 4.8.1 名前の変更

ファイルの名前の変更は、COBOL 開発環境 の [ ファイルビュー ] 上で実行できます。

ソースファイルなどの名前を変更する手順を説明します。

1. [ ファイルビュー ] で、名前を変更したいファイルを選択します。
2. マウスを右クリックして表示されるメニューから [ 名前の変更 ] コマンドを選択します。
3. 選択したファイルの名前の部分が編集可能になります。
4. 変更後のファイル名を入力した後、[ Enter ] キーを押すと、ファイル名が変更されます。

#### **注意事項**

- ・ ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。

### 4.8.2 削除

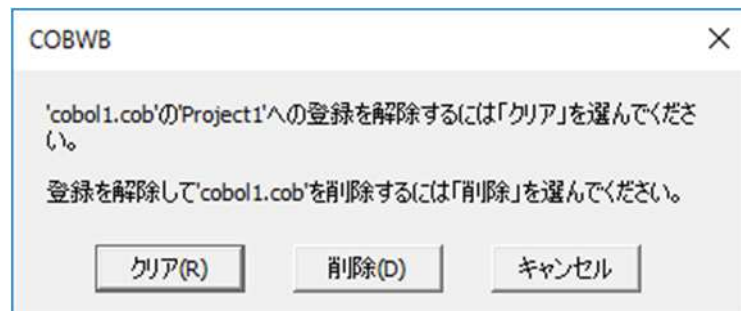
ファイルの削除は、COBOL 開発環境 の [ ファイルビュー ] 上で実行できます。

プロジェクトからソースファイルなどを削除する場合には、プロジェクトから除外して物理ファイルは削除しない方法と、物理ファイルを削除する方法の2種類があります。

どちらの削除にするかは、コマンド選択後に表示されるダイアログボックスで選択します。

ファイルを削除する手順を説明します。

1. [ ファイルビュー ] で、プロジェクトから削除したいファイルを選択します。
2. マウスを右クリックして表示されるメニューから [ 削除 ] コマンドを選択します。
3. 以下の確認ダイアログボックスが表示されます。



4. [ クリア ] ボタンをクリックすると、プロジェクトからファイルを除外します。物理ファイルは削除しません。
5. 「削除」ボタンをクリックすると、プロジェクトからファイルを除外し、物理ファイルを削除します。
6. 「キャンセル」ボタンをクリックすると、削除を中止します。

### 4.8.3 エクスプローラーでフォルダーを開く

[ ファイルビュー ] 上から、ファイルフォルダーをエクスプローラーで開くことができます。

ファイルフォルダーをエクスプローラーで開く手順を説明します。

1. [ ファイルビュー ] で、ファイルを選択します。
2. マウスを右クリックして表示されるメニューから [ エクスプローラーでフォルダーを開く ] コマンドを選択します。
3. エクスプローラーが別画面に開き、選択したファイルのフォルダーが表示されます。

## 4.9 プロジェクトの排他制御

---

プロジェクトが格納されているディレクトリが複数の環境から参照できる場合に、複数の COBOL 開発環境から同時に同じプロジェクトファイルを開いてしまうことがあります。

このような状況を抑止するために、COBOL 開発環境ではプロジェクトの排他制御を行っています。

プロジェクトの排他は以下のように行われます。

- ・ 開発環境で既に関われているプロジェクトを、他の開発環境で開こうとすると、プロジェクトは「読み取り専用」で開かれます。  
この状態ではプロジェクトの情報を参照することはできますが、編集することはできません。
- ・ 開発環境で既に関われている COBOL ソースファイルを、他の開発環境で開こうとすると、「読み取り専用」で開かれます。  
この状態では COBOL ソースファイルの参照はできますが、編集することはできません。



---

## 5 ファイルの編集

この章では、COBOL 開発環境で COBOL ソースファイルを編集する操作について説明します。

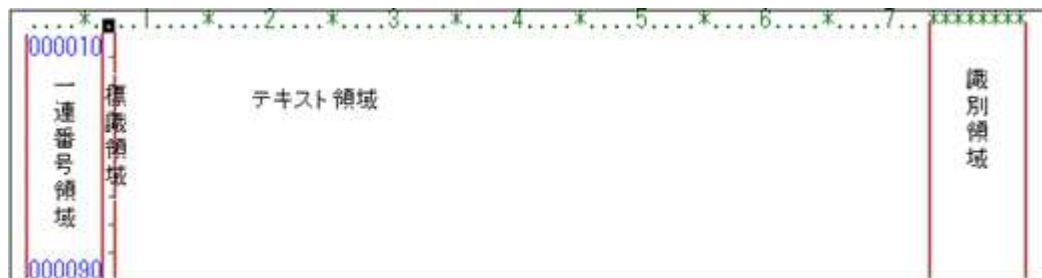
---

## 5.1 編集ウィンドウの概要

COBOL 開発環境にて、COBOL の原始プログラムを編集するのに適したエディターを提供します。

なお、編集ウィンドウは、固定形式および自由形式によって、異なります。

固定形式の場合は、ウィンドウ内の罫線にて、一連番号領域、標識領域、テキスト領域、識別領域に区分けされます。



固定形式エディタ



自由形式エディタ

固定形式、自由形式については、「COBOL プログラミングの手引」を参照してください。

固定形式の場合は、テキスト領域の編集を行うことで、一連番号領域と識別領域は自動的に追加されます。

自由形式の場合は、一連番号領域や識別領域を自動的に付加することはありません。

また、COBOL 言語の予約語は色分け表示することができます。

予約語については、「COBOL 言語説明書」を参照してください。

なお、予約語は、テキスト領域にて色分け表示されます。

キーワード補完についても、テキスト領域で有効となります。キーワード補完については「5.13 キーワード補完」を参照してください。

## 5.2 ファイルの新規作成

COBOL プログラムの開発に使用するファイルを、COBOL 開発環境 から新規に作成できます。

新規にファイルを作成する手順を説明します。

1. [ファイル] - [新規作成] - [ファイル] コマンドを選択します。
2. 「新しいファイル」ダイアログボックスが表示されます。  
【Standard】 【Enterprise(SE)】



【Enterprise(標準)】 【Linux】



3. COBOL ソースを選択し、[ 開く ] ボタンをクリックすると、指定したファイル名のファイルが作成され、編集ウィンドウが表示されます。  
[ キャンセル ] ボタンをクリックすると、ファイルは作成されません。

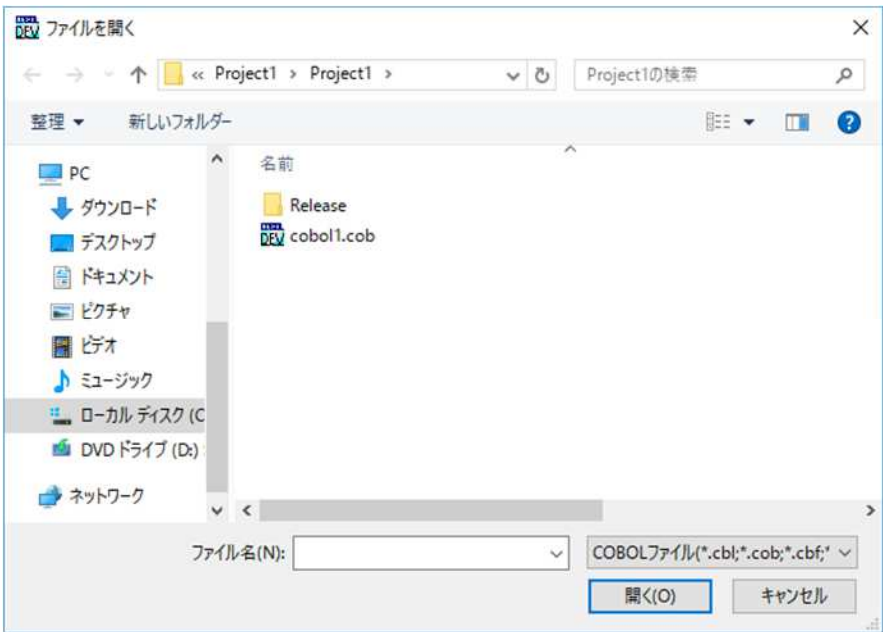
#### **注意事項**

- ・ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。

## 5.3 ファイルを開く

COBOL 開発環境 からファイルを開く手順を説明します。

1. [ファイル] - [開く] - [ファイル] コマンドを選択します。
2. [ファイルを開く] ダイアログボックスが表示されます。



3. ドロップダウンリストで、表示するファイルの種類を指定します。ドロップダウンリストには「すべてのファイル」を含む COBOL ソースファイルの組み合わせが登録されています。COBOL ソースファイルの組み合わせは以下のとおりです。

分類	拡張子
COBOL ファイル	*.cob
	*.cbl
	*.cbf
	*.cbx
	*.cb1m 【Enterprise(標準)】 【Linux】
	*.cobm 【Enterprise(標準)】 【Linux】
	*.cble 【Enterprise(標準)】 【Linux】

	*.cblme 【Enterprise(標準)】 【Linux】
SQL ファイル	*.pco *.qcbI *.qcob *.pcom 【Enterprise(標準)】 【Linux】 *.pcoe 【Enterprise(標準)】 【Linux】 *.pcome 【Enterprise(標準)】 【Linux】
COBOL/S ファイル	*.cblss 【Enterprise(標準)】 【Linux】 *.cblmss 【Enterprise(標準)】 【Linux】 *.cblsse 【Enterprise(標準)】 【Linux】 *.cblmsse 【Enterprise(標準)】 【Linux】 *.pcoss 【Enterprise(標準)】 【Linux】 *.pcomss 【Enterprise(標準)】 【Linux】 *.pcosse 【Enterprise(標準)】 【Linux】 *.pcomsse 【Enterprise(標準)】 【Linux】
すべてのファイル	*.*

なお、任意のファイルの種類を指定したい場合は、[ ファイル名 ] ドロップダウンリストに直接入力してください。（例：“\*.cob”）複数の種類を指定したい場合は、セミコロン(;)で区切って指定してください。（例：“\*.cob;\*.cbl”）

4. ドロップダウンメニューで、フォルダーおよびドライブを選択します。
5. 開くファイルを選択します。[ Ctrl ] キーや [ Shift ] キーを使用しての複数ファイルの選択も可能です。また、[ ファイル名 ] ドロップダウンリストに直接入力することもできます。
6. [ 開く ] ボタンをクリックすると、選択したファイルが開きます。  
[ キャンセル ] ボタンをクリックすると、ファイルは開きません。

### ファイルビューから開く

ファイルビュー上でファイルをダブルクリックすることで、そのファイルの編集ウィンドウを開くことができます。

また、以下の手順でも編集ウィンドウを開くことができます。

1. [ ファイルビュー ] でファイルを選択する。
2. マウスの右ボタンをクリックし表示されるメニューから [ 開く ] コマンドを選択します。
3. ファイルの編集ウィンドウが表示されます。

### エディタが開く際の拡張子の形式

COBOL ソースファイルおよび登録集原文ファイルの場合、編集ウィンドウが開く際の形式は COBOL ソースファイルの拡張子で決定します。拡張子を元に固定形式と自由形式に分類します。

形式	エディター カラム	識別 領域	拡張子	種類 ( COBOL )
定形式	80		.cbl	COBOL
			.cblm	COBOL 【Enterprise(標準)】 【Linux】
			.cble	
			.cblme	
			.pco	Pro*COBOL
			.pcom	埋込み SQL COBOL 【Enterprise(標準)】 【Linux】
			.pcoe	
			.pcome	
			.qcbi	埋込み SQL COBOL
			.cblss	COBOL/S 【Enterprise(標準)】 【Linux】
			.cblmss	
			.cblsse	
			.cblmsse	
			.pcoss	
			.pcomss	
			.pcosse	
			.pcomsse	
固定形式	MAX	-	.cob	COBOL



	255		.cobm	COBOL【Enterprise(標準)】【Linux】
			.qcob	埋込み SQL COBOL
固定形式	MAX 255		.cbx	COBOL
自由形式	MAX 255	-	.cbf	COBOL

登録集原文ファイルは COBOL ソースファイル形式にあわせ、以下のいずれかを利用します。

cbl、cob、cbf、cbx

### エクスプローラーから開く

エクスプローラーからファイルをダブルクリックすることで、COBOL 開発環境を起動し、次の状態に移行します。

- ・ COBOL ソースファイルおよび登録集原文ファイルの場合、COBOL プログラム用の編集ウィンドウが開きます。
- ・ COBOL プロジェクトファイルの場合、そのプロジェクトを開きます。
- ・ COBOL ソリューションファイルの場合、そのソリューションを開きます。

エクスプローラーからファイルのアイコンを COBOL 開発環境へドラッグアンドドロップすることにより、ファイルを開くことができます。

- ・ COBOL ソースファイルおよび登録集原文ファイルの場合、COBOL プログラム用の編集ウィンドウが開きます。
- ・ COBOL プロジェクトファイルの場合、そのプロジェクトを開きます。
- ・ COBOL ソリューションファイルの場合、そのソリューションを開きます。

また、最近使ったファイルは、[ ファイル ] - [ 最近使ったファイル ] 配下に登録されていますので、その配下を選択することで開くことができます。

### **注意事項**

- ・ COBOL 開発環境を「管理者として実行」している場合は、ファイルをドラッグアンドドロップすることができません。

### **制限事項**

- ・ 1 行の文字数が制限を超える COBOL ソースファイルまたは登録集原文ファイルを開くと、COBOL 形式とみなされずに一般のファイルとして編集ウィンドウに表示されます。その状態でファイルを保存すると、行の先頭に空白文字列を追加してファイルに保存されます。
- ・ COBOL ソースファイルを編集して保存する場合は、1 行の文字数を制限内になるように修正してから、ファイルを開いてください。

## 5.4 ファイルを保存する

---

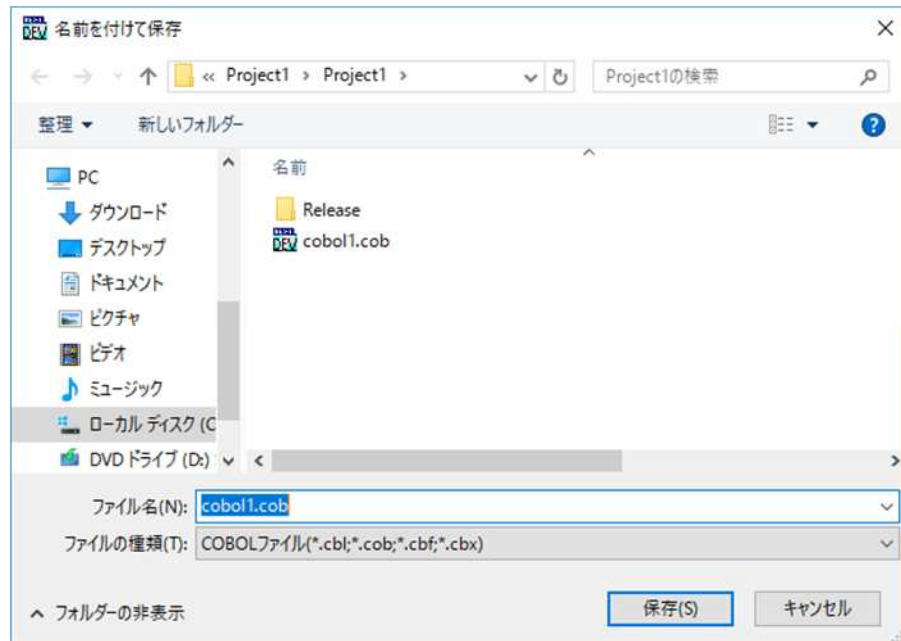
編集ウィンドウの内容をファイルに保存する手順を説明します。

### 5.4.1 上書き保存

1. 保存する編集ウィンドウをアクティブにします。
2. [ファイル]メニューから、アクティブな編集ウィンドウのファイル名が書かれた[ ~の保存 ]コマンドを選択します。
3. 現在アクティブな編集ウィンドウがファイルに保存されます。

### 5.4.2 名前を付けて保存

1. 保存する編集ウィンドウをアクティブにします。
2. [ファイル]メニューから、アクティブな編集ウィンドウのファイル名を含む[ ~に名前を付けて保存 ]コマンドを選択します。
3. [名前を付けて保存]ダイアログボックスが表示されます。[ファイル名]ドロップダウンリストにはアクティブな編集ウィンドウのファイル名が表示されています。



4. [ ファイルの種類 ] ドロップダウンリストで、表示するファイルの種類を指定します。

ドロップダウンリストにはあらかじめ COBOL で使用する拡張子が登録されています。

任意のファイルの種類を指定したい場合は、[ ファイル名 ] ドロップダウンリストに直接入力してください。(例: “ \*.cob ”) 複数の種類を指定したい場合は、セミコロン(;)で区切って指定してください。(例: “ \*.cob;\*.cbt ”)

エクスプローラーのフォルダーオプションにて、[ 登録されている拡張子は表示しない ] を選択している場合、[ ファイルの種類 ] ドロップダウンリストに拡張子が表示されません。

5. [ 保存先 ] ドロップダウンメニューで、フォルダー、ドライブを選択します。
6. 既存のファイル名にするには、そのファイル名を選択します。また、[ ファイル名 ] ドロップダウンリストに直接入力するか、ドロップダウンリストから選択することもできます。
7. [ 保存 ] ボタンをクリックすると、入力したファイル名で保存されます。  
[ キャンセル ] ボタンをクリックすると、別の名前をつけて保存しません。

### **注意事項**

- ・ファイル名の規則については、「2.1.2 ファイル名の規則」を参照してください。
- ・COBOL ファイルを開いて、COBOL ファイル以外の拡張子の名前を付けて保存する場合、画面表示上はカスタマイズの「キーワード強調表示」の設定が引き続き有効になります。
- ・ファイルの拡張子を変更した場合、拡張子に合わせた表示には変更されません。

### **5.4.3 すべてを保存**

1. [ファイル] - [すべてを保存] コマンドを選択する。
2. 変更のあったファイルがすべて保存されます。

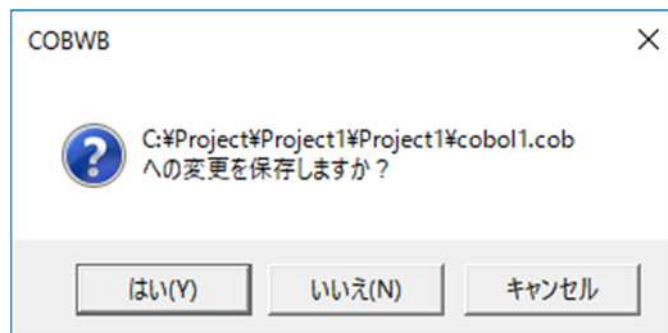
## 5.5 ファイルを閉じる

---

ファイルを閉じる手順を説明します。

1. [ファイル] - [閉じる] コマンドを選択します。
2. 現在アクティブな編集ウィンドウのファイルを閉じます。

閉じるウィンドウが編集ウィンドウで、その内容が変更されている場合は、確認メッセージが表示されます。



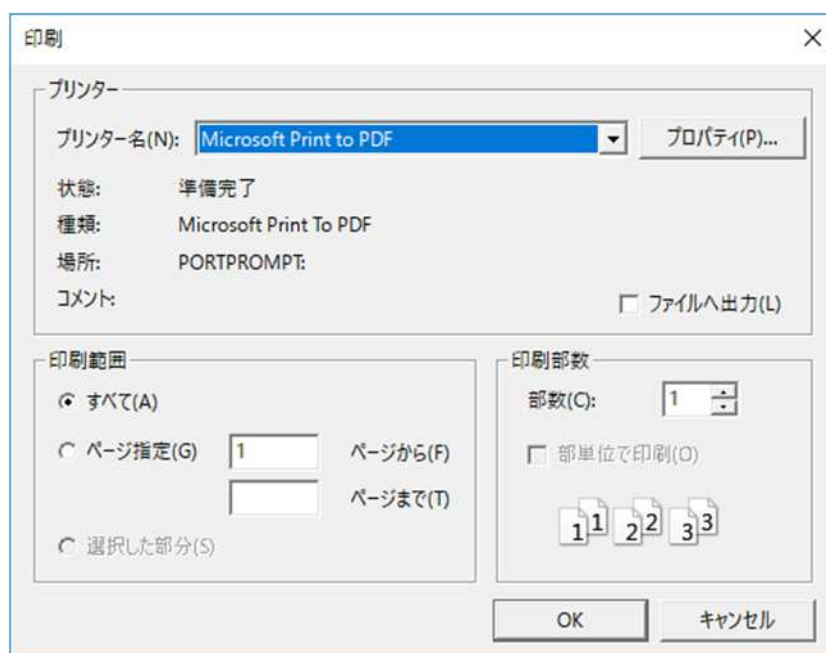
[ はい ] ボタンをクリックするとファイルに上書き保存し、[ いいえ ] ボタンをクリックすると保存せずにウィンドウを閉じます。[ キャンセル ] ボタンをクリックすると、ファイルを閉じません。

## 5.6 ファイルの印刷

開いているファイルを COBOL 開発環境から印刷することができます。

印刷する手順を説明します。

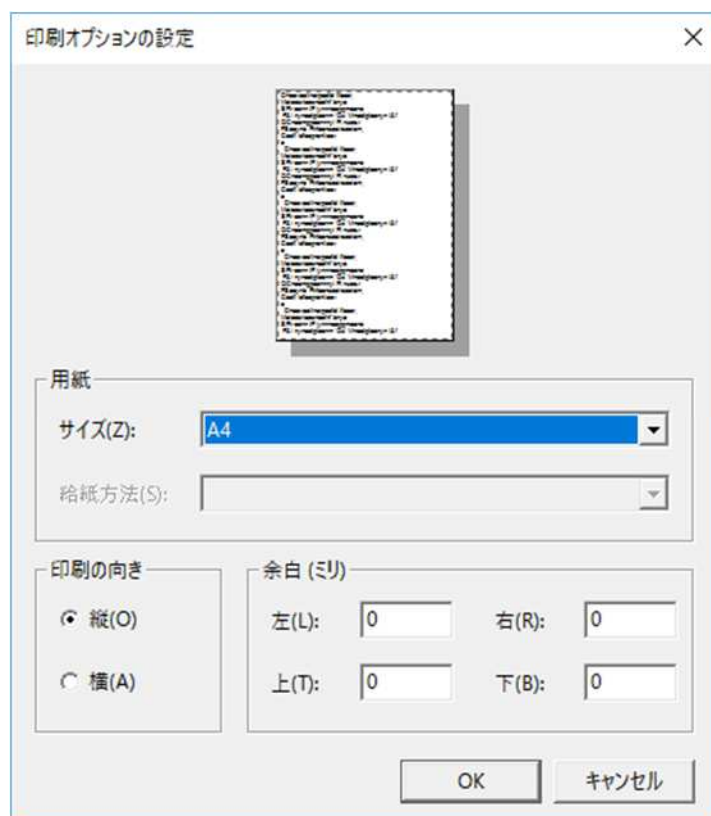
1. [ファイル] - [印刷] コマンドを選択すると、以下のような印刷ダイアログボックスが表示されます。



2. 画面の内容にしたがって、条件を設定します。  
[OK] ボタンをクリックすると、対象のファイルが印刷されます。罫線  
は印刷されません。  
[キャンセル] ボタンをクリックすると、ファイルは印刷されません。

印刷プレビュー画面の表示を行う場合は、[ファイル] - [印刷プレビュー] コマンド  
を選択します。

印刷にあたって、オプションの設定を行う場合には、[ファイル] - [印刷オプション  
設定] コマンドを選択します。選択すると、以下のようなダイアログボックスが表示さ  
れます。



必要に応じて、設定を変更してください。

[ OK ] ボタンをクリックすると、設定が反映されます。

[ キャンセル ] ボタンをクリックすると、オプション設定は取り消されます。



## 5.7 編集操作

---

### 5.7.1 ウィンドウをスクロールするには

1. [ Page Up ] キーまたは [ Page Down ] キーで上下に 1 画面ずつ移動します。
2. カーソル位置をウィンドウの表示外の部分に移動させると、その方向にスクロールします。
3. マウスでスクロールバーを移動します。

### 5.7.2 カーソルを移動するには

1. 矢印キーを押します。[ Ctrl ] + [ 左右矢印 ] キーで単語単位に移動します。
2. カーソルを位置づけるポイントでマウスの左ボタンをクリックします。
3. 左端または右端に位置づける場合は、[ Home ] キーまたは [ End ] キーを押します。
4. テキストの先頭または末尾( [ EOF ] の行)に位置づける場合は、[ 編集 ] - [ 先頭 ] コマンドまたは [ 編集 ] - [ 末尾 ] コマンドを選択します。または [ Ctrl ] + [ Home ] キー、[ Ctrl ] + [ End ] キーを押します。

### 5.7.3 文字を編集するには

1. 文字キーは、挿入モードではカーソル位置に文字を挿入し、上書きモードではカーソル位置の文字を置換します。挿入モードと上書きモードを切り換えるには [ Insert ] キーを押します。
2. [ Backspace ] キーでカーソルの左の文字を削除します。
3. [ Delete ] キーでカーソル位置の文字を削除します。

#### 5.7.4 行を挿入 / 削除するには

1. [ Shift ] + [ Enter ] キーでカーソルのある行の次の行に新たな行を挿入します。
2. 終了行( [ EOF ] を表示)にカーソルを位置づけて文字キーを入力すると、テキストの末尾に新たな行を追加します。
3. [ Shift ] + [ Delete ] キーでカーソルのある行を削除します。

#### 5.7.5 行を分割 / 結合するには

1. 行を分割するには、分割したい桁位置にカーソルを位置づけて挿入モードで [ Enter ] キーを押します。上書きモード時の [ Enter ] キーは、カーソルを次行の先頭に移動する機能で行の分割は行いません。
2. 行を結合するには、テキスト右端(復改コードの位置)で [ Delete ] キーまたはその次の行の先頭 (フリーカーソルモードの場合は復改の次の桁)で [ Backspace ] キーを押します。

#### 5.7.6 文字列範囲を選択するには ( キー操作 )

1. 文字列の開始位置から終了位置まで [ Shift ] + [ 矢印 ] キーでカーソル位置を移動します。
2. カーソル位置から行の左端または右端まで選択するには、[ Shift ] + [ Home ] キーまたは [ Shift ] + [ End ] キーを押します。
3. カーソル位置からテキストの先頭または末尾まで選択するには、[ Shift ] + [ Ctrl ] + [ Home ] キーまたは [ Shift ] + [ Ctrl ] + [ End ] キーを押します。

#### 5.7.7 文字列範囲を選択するには ( マウス操作 )

1. 文字列の開始位置でマウスの左ボタンを押して、終了位置までドラッグします。

2. 単語を選択するには、選択したい単語の位置でマウスの左ボタンをダブルクリックします。
3. 現在のカーソル位置からマウスポインタの位置まで選択するには、  
[ Shift ] + マウスの左ボタンをクリックします。

#### 5.7.8 テキスト全体を選択するには

1. [ 編集 ] - [ すべて選択 ] コマンドを選択します。  
または、[ Ctrl ] + [ A ] キーを押します。

#### 5.7.9 行範囲を選択するには

1. ウィンドウ左端の余白でマウスの左ボタンを押して、終了行までドラッグします。

#### 5.7.10 矩形範囲を選択するには

1. 矩形の開始位置で [ Ctrl ] + マウスの左ボタンを押して、終了位置(矩形の対角の位置)までドラッグします。

#### 5.7.11 選択した範囲を削除するには

1. 範囲を選択します。
2. [ 編集 ] - [ 削除 ] コマンドを選択します。  
または [ Backspace ] キー、[ Delete ] キーを押します。

文字列範囲、行範囲選択中に文字キーを押した場合は、選択範囲を入力文字で置換します。

なお、削除した文字列はクリップボードには保存されません。

### 5.7.12 選択した範囲を切り取るには

1. 範囲を選択します。
2. [ 編集 ] - [ 切り取り ] コマンドを選択します。  
または [ Ctrl ] + [ X ] キーを押します。  
  
切り取った文字列は、選択した形式でクリップボードに保存されます。

### 5.7.13 選択した範囲をコピーするには

1. 範囲を選択します。
2. [ 編集 ] - [ コピー ] コマンドを選択します。  
または [ Ctrl ] + [ C ] キーを押します。  
  
コピーした文字列は選択した形式でクリップボードに保存されます。

### 5.7.14 クリップボードの内容を貼り付けるには

1. クリップボードに文字列がない場合は、切り取るかコピーし文字列を保存します。
2. 貼り付ける位置にカーソルを移動します。
3. [ 編集 ] - [ 貼り付け ] コマンドを選択します。  
または [ Ctrl ] + [ V ] キーを押します。

クリップボードの内容が文字列形式と矩形の場合はカーソル位置に、行形式の場合はカーソルのある行の次に、保存した時の形式で貼り付けます。他のアプリケーションで保存または参照するクリップボードの内容は文字列形式になります。

文字列範囲、行範囲選択中に貼り付けを実行すると、選択範囲をクリップボードの文字列で置換します。

### 5.7.15 ドラッグ&ドロップで範囲を移動するには

1. 範囲を選択します。
2. 範囲選択した部分でマウスの左ボタンを押してドラッグします。
3. 移動先の位置でマウスの左ボタンを離します。

離すときに [Ctrl] キーを押していると、移動ではなくコピーになります。

### 5.7.16 [Tab]キーでカーソルを移動するには

1. マウスポインタをスケール行に移動します。
2. タブ位置を設定したい桁位置でマウスの左ボタンをダブルクリックします。既に設定している桁位置の場合は、タブ位置の解除になります。
3. COBOL フォーマットの場合に [Tab] キーを押すと、カーソルはA領域、B領域および識別領域の先頭と設定したタブ位置を順次移動します。COBOL フォーマット以外の場合は、行の先頭と設定したタブ位置を順次移動します。  
[Shift] + [Tab] キーで逆方向に移動します。

### 5.7.17 編集操作を元に戻すには

1. [編集] - [元に戻す] コマンドを選択します。  
または [Ctrl] + [Z] キーを押します。

### 5.7.18 元に戻した編集を復元するには

1. [編集] - [やり直し] コマンドを選択します。

## 5.8 文字列の検索

---

### 5.8.1 文字列を検索するには

1. 検索開始位置にカーソルを位置づけます。  
一連番号領域、テキスト領域、識別領域を区別して検索します。したがって、テキスト領域にカーソルを位置づけた場合は一連番号領域と識別領域は検索しません。
2. [編集] - [検索] コマンドを選択するか、[Ctrl] + [F] キーを押すと、「文字列の検索」ダイアログボックスが表示されます。



3. [検索する文字列] テキストボックスに文字列を入力します。  
入力すると [次を検索] ボタンが有効となります。
4. 検索する方向を指定します。
5. 大文字と小文字を区別するかを指定します。
6. [単語単位で検索] をチェックすると単語のみを検索します。
7. [次を検索] ボタンをクリックします。
8. 現在のカーソル位置から指定された方向に向かって検索を開始します。見つかると文字列を選択状態にします。
9. ダイアログボックスを閉じるには [キャンセル] ボタンをクリックします。

なお、ダイアログボックスを閉じても検索文字列は保存され、次の検索の際の検索文字列として、また文字列の置換の際の置換前文字列の初期値として使用されます。

### 5.8.2 次の文字列を検索するには

1. 検索開始位置にカーソルを位置づけます。
2. [ 文字列の検索 ] ダイアログボックスの [ 次を検索 ] ボタンをクリックします。

### 5.8.3 検索する文字を貼りつけるには

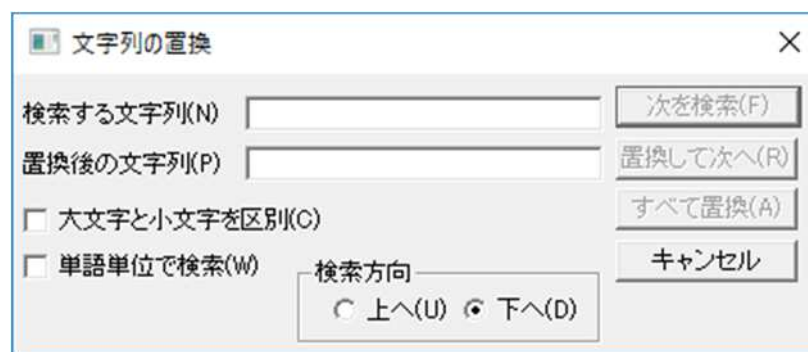
1. 検索する文字列を [ Ctrl ] + [ C ] キーでコピーします。
2. [ 検索する文字列 ] テキストボックスにカーソルを位置づけます。
3. [ Ctrl ] + [ V ] キーを押します。

## 5.9 文字列の置換

---

### 5.9.1 文字列を置換するには

1. 置換開始位置にカーソルを位置づけます。
2. [編集] - [置換] コマンドを選択します。  
または [Ctrl] + [R] キーを押します。
3. 「文字列の置換」ダイアログボックスが表示されます。



[ 検索する文字列 ] テキストボックスに文字列を入力します。  
入力するとすべてのボタンが有効となります。

4. [ 置換後の文字列 ] テキストボックスに文字列を入力します。  
入力しないと長さ 0 の文字列となります。
5. 検索する方向を指定します。
6. 大文字と小文字を区別するかを指定します。
7. [ 単語単位で検索 ] をチェックすると単語のみを検索します。
8. 置換の方法を以下に示します。
  - ・ 検索のみで置換しない場合は、[ 次を検索 ] ボタンをクリックします。
  - ・ 最初に見つかった文字列を 1 つだけ置換する場合には、[ 置換して次に ] ボタンをクリックします。
  - ・ カーソル位置から指定された検索方向に向かって文字列をすべて置換する場合には、[ すべて置換 ] ボタンをクリックします。



- ・ 行範囲指定されている場合に、[ すべて置換 ] ボタンをクリックすると、その範囲内だけが置換の対象となります。

9. ダイアログボックスを閉じるには [ キャンセル ] ボタンをクリックします。

なお、ダイアログボックスを閉じても、検索する文字列と置換後の文字列は保存され、次回の置換の際の初期値として、また次回の検索の際の検索文字列として使用されます。

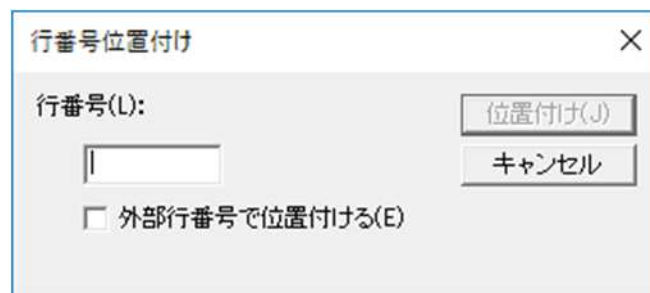
### **注意事項**

- ・ 置換対象はテキスト領域のみで、一連番号領域と識別領域は置換しません。

## 5.10 行番号位置づけ

---

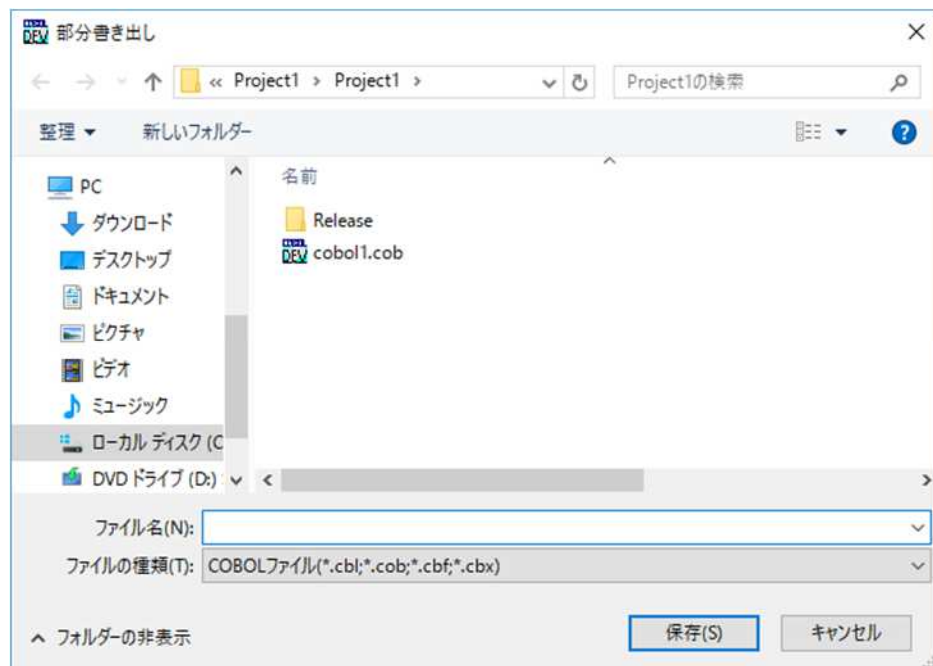
1. 位置づけ対象となる編集ウィンドウをアクティブにします。
2. [ 編集 ] - [ 行番号位置づけ ] コマンドを選択します。
3. [ 行番号位置づけ ] ダイアログボックスが表示されます。



4. [ 行番号 ] テキストボックスに行番号を入力します。  
符号付きの行番号を入力することにより、カーソル位置からの相対行数で位置づけることができます。
5. COBOL 一連行番号領域の行番号で位置づけるには、[ 外部行番号で位置づける ] を指定します。  
ファイル先頭からの相対行番号に位置づけるには、[ 外部行番号で位置づける ] を指定しません。
6. [ 位置づけ ] ボタンをクリックします。
7. 指定した行番号を持つ行を編集ウィンドウに表示して、カーソルを位置づけます。なお、ダイアログボックスは閉じられます。

## 5.11 部分書き出し

1. 対象となる編集ウィンドウをアクティブにし、ファイルに保存する範囲を行範囲選択します。
2. [編集] - [部分書き出し] コマンドを選択すると、[部分書き出し] ダイアログボックスが表示されます。

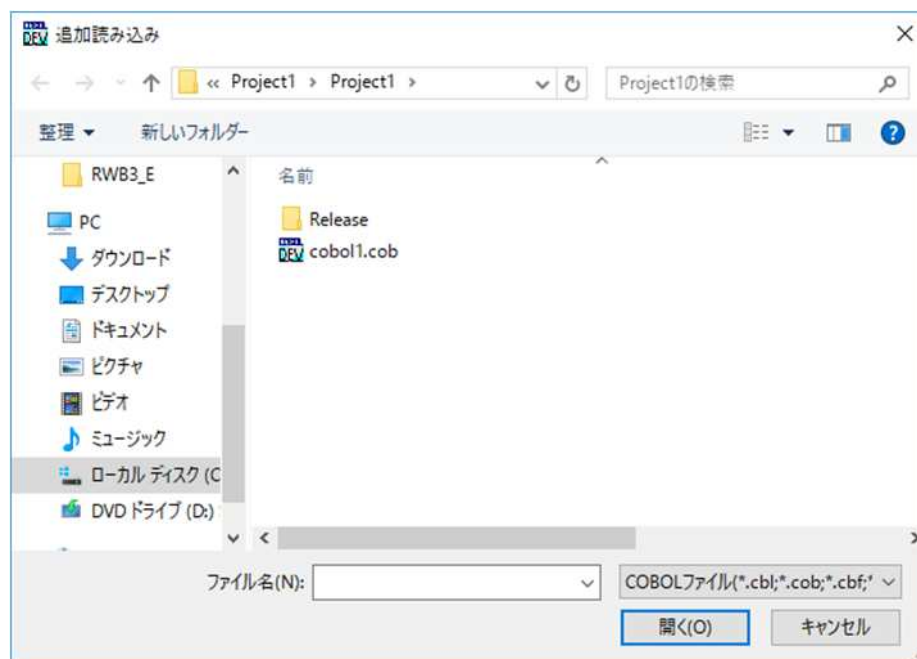


3. [ファイルの種類] リストボックスで、表示するファイルの種類を指定します。
  4. [保存する場所] リストボックスで、フォルダー、ドライブを選択します。
  5. [保存] ボタンをクリックすると、選択した範囲がファイルに保存されます。
- [キャンセル] ボタンをクリックすると、ファイルに保存されません。

## 5.12 追加読み込み

---

1. 挿入先のファイルをアクティブに、挿入する場所へカーソルを移動します。
2. [編集] - [追加読み込み] コマンドを選択すると、[追加読み込み] ダイアログボックスが表示されます。



3. 挿入するファイルを選択し、[開く] ボタンをクリックすると、カーソル位置に選択したファイルが挿入されます。  
[キャンセル] ボタンをクリックすると、ファイルは挿入されません。

## 5.13 キーワード補完

---

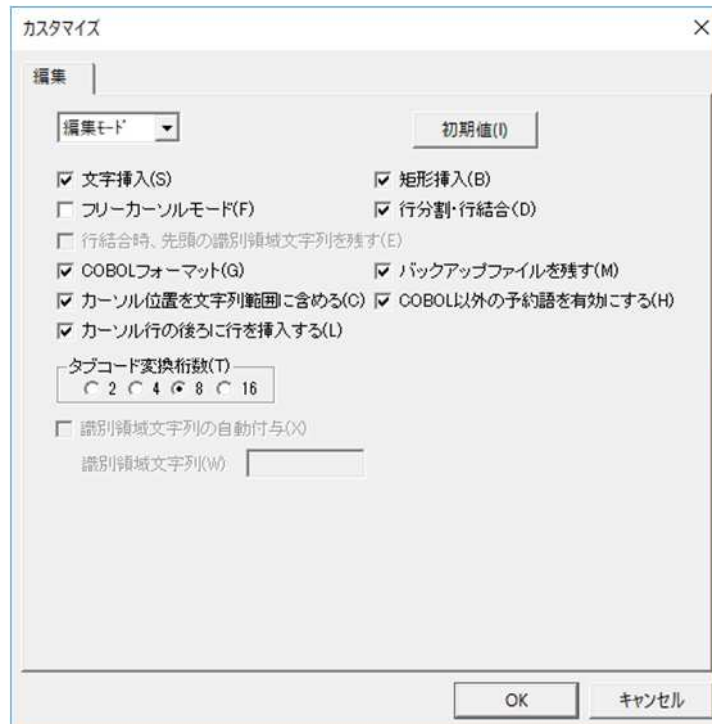
入力途中の文字列を COBOL のキーワードに補完することができます。

1. 編集ウィンドウに COBOL 予約語の途中までキー入力します。
2. [ 編集 ] - [ キーワードの補完 ] コマンドを選択します。  
または、[ CTRL ] + [ T ] を押します。
3. 候補が 1 件の場合は、カーソル位置直前の文字列を候補文字列に置換します。

候補が複数ある場合は、一覧をリスト形式で表示します。目的のキーワードを選択して [ Enter ] キーまたはマウスの左ボタンをダブルクリックすることにより選択文字列が決定され、カーソル位置直前の文字列と置換されます。一覧をキャンセルするには [ Esc ] キーを押します。

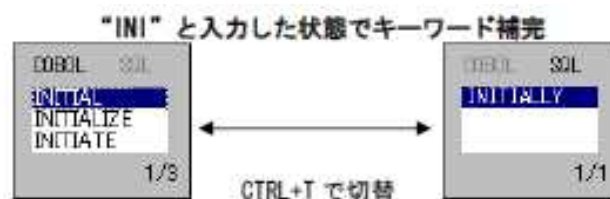
入力途中の文字列を SQL のキーワードに補完することができます。

1. [ 編集 ] - [ カスタマイズ ] コマンドを選択します。
2. 「カスタマイズ」ダイアログボックスの [ 編集 ] タブ画面から [ 編集モード ] を選択します。
3. [ COBOL 以外の予約語を有効にする ] チェックボックスをチェックします。



4. 編集ウィンドウに SQL 文予約語の途中までキー入力します。
5. [編集] - [キーワードの補完] コマンドを選択します。
6. 候補が 1 件の場合は、カーソル位置直前の文字列を候補文字列に置換します。候補が複数ある場合は、一覧をリスト形式で表示します。目的のキーワードを選択して [Enter] キーまたはマウスの左ボタンをダブルクリックすることにより、選択文字列が決定され、カーソル位置直前の文字列と置換されます。一覧をキャンセルするには [Esc] キーを押します。候補が COBOL 予約語にもある場合、一覧には COBOL 予約語が優先的に表示されます。

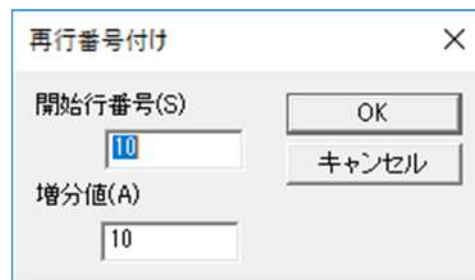
ここで、[Ctrl]+[T] キーを押すことで SQL 文予約語の一覧に切り替えることができます。[Ctrl]+[T] キーを押すたびに、COBOL 予約語と SQL 文予約語の一覧を交互に切り替えることができます。



## 5.14 再行番号付け

---

1. 再行番号付けする範囲を行範囲選択します。  
範囲選択をしなければファイル全体の再行番号付けをします。
2. [ 編集 ] - [ 再行番号付け ] コマンドを選択します。
3. [ 再行番号付け ] ダイアログボックスが表示されます。

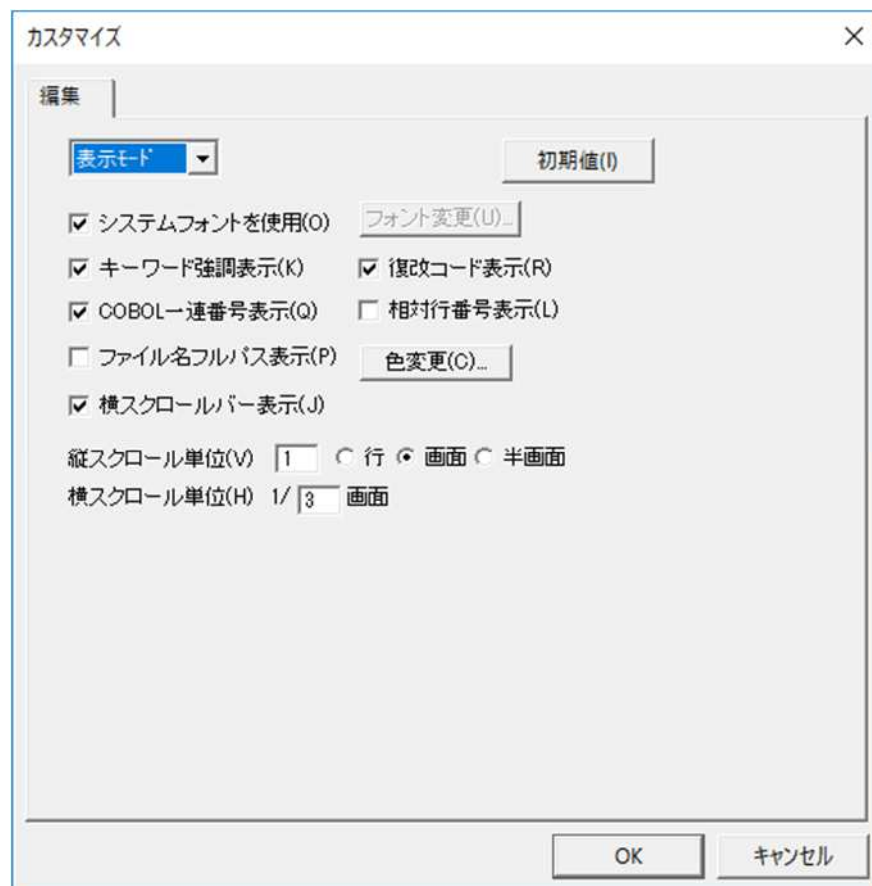


4. [ 開始行番号 ] テキストボックスに先頭行に付与する行番号を入力します。
5. [ 増分値 ] テキストボックスに 2 行目以降に付与する行番号の増分値を指定します。  
  
[ OK ] ボタンをクリックします。  
[ キャンセル ] ボタンをクリックすると、一連番号は変更されません。
6. 再番号付けの結果が範囲の前後の行番号に対して昇順でなくなる場合は、実行するか中止するか問い合わせます。

## 5.15 編集ウィンドウの動作変更

1. [編集] - [カスタマイズ] コマンドを選択します。
2. 「カスタマイズ」ダイアログボックスの編集タブ画面から [編集モード] を選択します。

先頭のリストボックスで [編集モード] [表示モード] を選択することにより、それぞれに分類されている機能項目を表示します。



3. [初期値] ボタンは、すべての設定を初期値に戻します。
4. [文字挿入] は、編集ウィンドウを新規に開いた場合の文字入力モードを指定します。チェックすると挿入モード、非チェックにすると上書きモードになります。
5. [矩形挿入] をチェックすると、矩形を切り取った位置に右側のテキスト



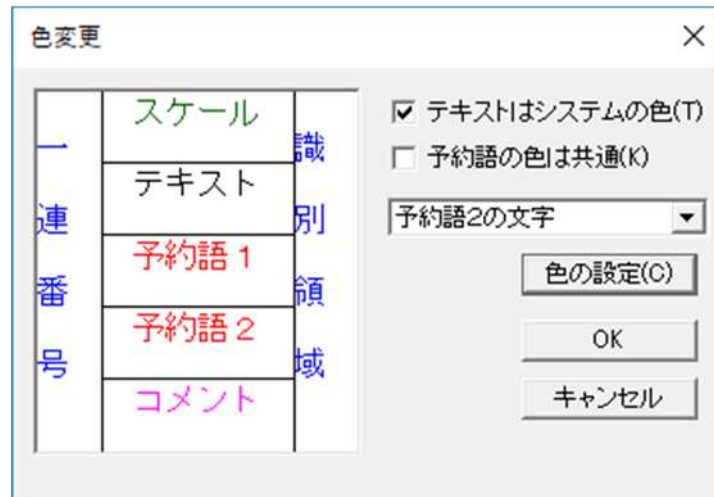
がシフトされ、貼り付けた位置にあったテキストは右側にシフトされます。非チェックの場合は、矩形を切り取った位置は空白になり、貼り付けた位置にあったテキストは上書きされます。

6. [フリーカーソルモード] をチェックすると、復改コード以降の桁にカーソルを移動することができます。非チェックの場合は、復改コードの位置から次の行のテキストの先頭桁に移動します。
7. [行分割・行結合] をチェックすると、[Enter] キー(挿入モード)によりカーソル位置で行を分割、[Delete] または [Backspace] キーによる復改コードの削除で次行との行結合を行います。非チェックにすると、これらのキーによる行分割、行結合は行いません。
8. [行結合時、先頭の識別文字を残す] をチェックすると、行結合したテキストの識別領域に前側の行の識別文字列を引き継ぎます。オフにすると、後ろ側の行の識別文字列を引き継ぎます。
9. [COBOL フォーマット] をチェックすると、COBOL 原始プログラムファイルを開く場合に、COBOL 正書法に従った編集ウィンドウを表示させる指定です。COBOL 原始プログラムファイルの一連番号領域、テキスト領域、識別領域を区分けしないで編集したい場合は非チェックにします。
10. [バックアップファイルを残す] をチェックすると、編集ウィンドウの保存時に作成されるバックアップファイル(拡張子が “.bak” )を残します。非チェックの場合は、保存処理が正常に終了するとバックアップファイルを削除します。  
いずれの場合も拡張子が “.bak” のファイルは、作業用に上書きまたは削除される可能性があるので、明示的に作成しないでください。
11. [カーソル位置を文字列範囲に含める] をチェックすると、文字列範囲指定時にカーソル位置までが選択範囲になります。非チェックにすると、カーソル位置の直前までが文字列範囲になります。行範囲、矩形範囲、および識別領域での文字列範囲選択では常にカーソル位置までが範囲です。
12. [カーソル行の後ろに行を挿入する] をチェックすると、カーソルのある行の次に行を挿入します。非チェックにすると、カーソルのある行の前に行を挿入します。
13. [タブコード変換桁数] ラジオボタンをクリックすると、入力ファイルにタブコードが含まれていた場合に、何桁の空白に拡張するかを変更できます。拡張された空白はファイルに保存するときも空白のままになります。

14. [ 識別領域文字列の自動付与 ] をチェックすると、 COBOL フォーマットのときにテキスト領域を編集した行の識別領域に自動的に文字列 ( [ 識別領域文字列 ] で指定 ) を設定します。非チェックにすると、自動設定は行いません。
15. [ 識別領域文字列 ] テキストボックスには、識別領域に自動設定する文字列を指定します。
16. [ COBOL 以外の予約語を有効にする ] をチェックすると、入力途中の文字列を SQL 文の予約語に補完することができます。補完方法については、「5.13 キーワード補完」を参照してください。  
【Enterprise(標準)】【Linux】COBOL 以外の予約語には COBOL/S の予約語が含まれます。
17. [ システムフォントを使用 ] をチェックすると、編集ウィンドウに表示するフォントとしてシステムフォントを使用することを指定します。非チェックにすると、[ フォント変更 ] ボタンが選択可能なり、フォントを変更することができます。
18. [ フォント変更 ] ボタンをクリックすることにより、[ フォント ] ダイアログボックスを表示して編集ウィンドウのフォントを変更することができます。
19. [ キーワード強調表示 ] をチェックすると、COBOL の予約語および COBOL 以外の予約語をテキストと異なる色で表示します。非チェックの場合は、テキストと同じ色で表示します。
20. [ 復改コード表示 ] をチェックすると、復改コードの位置を画面に記号 (「」) で表示します。非チェックの場合は、空白と同じ表示になります。
21. [ COBOL 一連番号表示 ] をチェックすると、 COBOL フォーマットの一連番号領域を表示します。非チェックにすると、テキスト領域(7 桁目)から表示します。
22. [ 相対番号表示 ] をチェックすると、先頭からの相対行番号を表示します。非チェックにすると、表示しません。
23. [ ファイル名フルパス表示 ] をチェックすると、タイトルバーに編集集中のファイル名をフルパスで表示します。非チェックにすると、ファイル名のみ表示します。
24. [ 横スクロールバー表示 ] をチェックすると、表示テキスト種別の最大長

が編集ウィンドウの幅を超えた場合に自動的に横スクロールバーを表示します。非チェックにすると、横スクロールバーは表示されません。

25. [色変更] ボタンをクリックすることにより、[色変更] ダイアログボックスが表示され、編集ウィンドウの文字色および背景色を変更できます。



文字色あるいは背景色を変更する領域をセレクションボックスで選択した後、[色の設定] ボタンをクリックします。[予約語 1] は COBOL の予約語を、[予約語 2] は COBOL 以外の予約語を示します。[色の設定] ダイアログボックスで設定する色をクリックして [OK] ボタンをクリックします。テキストの文字色は [テキストはシステムの色] をチェックすることにより、システム色に連動します。また、テキストの背景色はシステム色から変更することはできません。

26. [縦スクロール単位] は、縦スクロールの単位を変更します。
27. [横スクロール単位] は、横スクロールの単位を変更します。
28. 最後に [OK] ボタンをクリックすることで編集ウィンドウの環境を変更します。

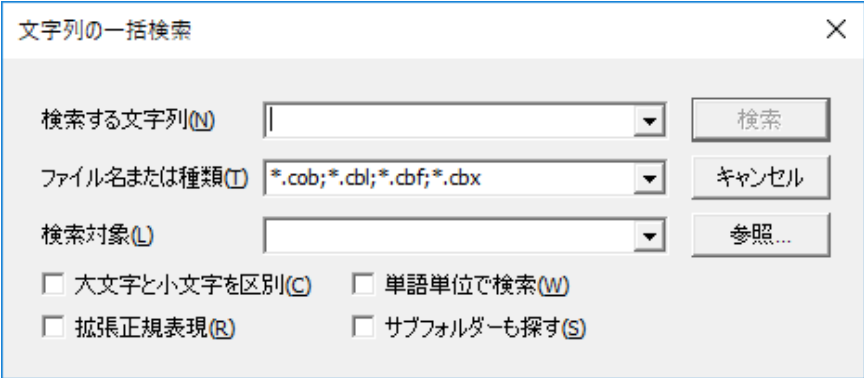
### **注意事項**

- ・ 編集ウィンドウの動作変更は、利用者別に上書き登録されます。

## 5.16 文字列の一括検索

### 5.16.1 文字列を一括検索するには

1. [編集] - [一括検索] コマンドを選択するか、[Ctrl] + [Shift] + [F] キーを押すと、「文字列の一括検索」ダイアログボックスが表示されます。



文字列の一括検索

検索する文字列(N)  検索

ファイル名または種類(T) \*.cob;\*.cbl;\*.cbf;\*.cbx キャンセル

検索対象(L)  参照...

☐ 大文字と小文字を区別(C) ☐ 単語単位で検索(W)

☐ 拡張正規表現(R) ☐ サブフォルダーも探す(S)

[ 検索する文字列 ] テキストボックスに文字列を入力します。

入力すると [ 検索 ] ボタンが有効となります。

検索する文字列は 1 つのみ入力が可能です。

2. [ ファイル名または種類 ] リストボックスから種類を選択するか、直接入力でファイル名を指定します。  
ファイルの種類を指定する場合は、「\*.拡張子」の形式のみが有効です。
3. [ 検索対象 ] リストボックスから検索対象を選択するか、直接入力で 1 つのフォルダー名を指定します。  
リストボックスには、開発環境の状態に応じて選択が可能な項目が表示されます。

リストの選択肢	開発環境の状態
ソリューション	ソリューションを開いている
現在のプロジェクト	ソリューションを開き、[ ファイルビュー ] のプロジェクトノード配下にフォーカスがある
開いている全ファイル	編集ウィンドウが 1 つ以上開いている
アクティブな編集ウィンドウ	編集ウィンドウが 1 つ以上開いて

	いる
--	----

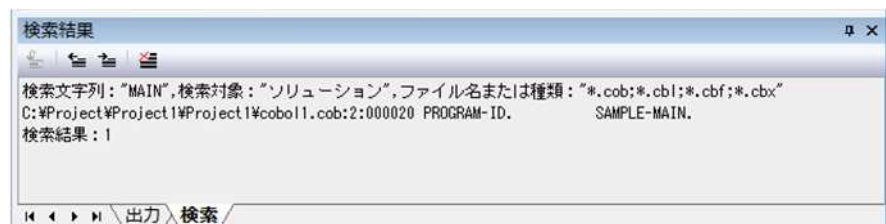
[ 検索する文字列 ]、[ ファイル名または種類 ]、[ 検索対象 ] を入力すると、[ 検索 ] ボタンが有効となります。

4. [ 大文字と小文字を区別 ] をチェックすると、大文字と小文字を区別して検索します。
5. [ 単語単位で検索 ] をチェックすると、単語のみを検索します。
6. [ 拡張正規表現 ] をチェックすると、検索する文字列に正規表現を記述できます。

正規表現で利用できるメタ文字は次のとおりです。

メタ文字	値
.	ワイルドカード - 任意の文字
*	繰り返し - 前の文字またはクラスが 0 個以上出現します
^	開始行の位置 - 行の先頭
\$	行末の位置 - 行末
[class]	文字クラス - セット内の任意の 1 文字
[^class]	逆クラス - セット内に含めない任意の 1 文字
[x-y]	範囲 - 指定した範囲内の任意の 1 文字
\x	エスケープ - メタ文字のリテラル使用
\<string	先頭の単語の位置 - 単語の先頭
string\>	末尾の語の位置 - 単語の末尾

7. [ サブディレクトリも探す ] をチェックすると、[ 検索対象 ] で指定したフォルダー配下の階層も検索対象とします。
8. [ 検索 ] ボタンをクリックします。
9. 検索を開始します。検索タブを選択すると開く [ 検索結果 ] ウィンドウに、検索結果が表示されます。



10. ダイアログボックスを閉じるには [ キャンセル ] ボタンをクリックします。

なお、ダイアログボックスを閉じても検索文字列は保存され、次の検索の際の検索文字列として使用されます。

#### **注意事項**

- ・ 検索結果の件数が多い場合に、初回の位置づけ操作が可能になるまでに時間がかかり、画面がグレー表示になることがあります。

### 5.16.2 検索した文字列に位置づけるには

1. [ 検索結果 ] ウィンドウをアクティブにします。
2. 位置づけたい検索結果の行にカーソルを移動します。
3. 以下の3つの方法のいずれかを実行します。
  - ・マウスの左ボタンをダブルクリックします。
  - ・右クリックメニューから [ 検索結果位置づけ ] コマンドを選択します。
  - ・ツールバーの [ 検索結果位置づけ ] のボタンをクリックします。
4. 原始プログラムファイルを開いていない場合は、コンパイラが参照した原始プログラムファイルが編集ウィンドウとして開かれます。
5. 前の検索結果に位置づけるには、ツールバーの [ 前の場所へ移動 ] のボタンをクリックします。  
または、右クリックメニューから [ 前の場所へ移動 ] コマンドを選択します。
6. 次の検索結果に位置づけるには、ツールバーの [ 次の場所へ移動 ] のボタンをクリックします。  
または、右クリックメニューから [ 次の場所へ移動 ] コマンドを選択します。
7. 表示されている内容を消去するには、ツールバーの [ すべてクリア ] のボタンをクリックします。  
または、右クリックメニューから [ すべてクリア ] コマンドを選択します。

### 5.16.3 検索を中止するには

1. 検索中に [ 一括検索処理中 ] ダイアログが表示されます。
2. 「中止」ボタンをクリックします。
3. 検索結果の表示中に「一括検索結果表示中」ダイアログが表示されます。
4. 「中止」ボタンをクリックします。

検索の時間が短い場合は、ダイアログがすぐに閉じられることがあります。

---

## 6 アプリケーションの生成

この章では、COBOL 開発環境での COBOL アプリケーションの生成について説明します。

---

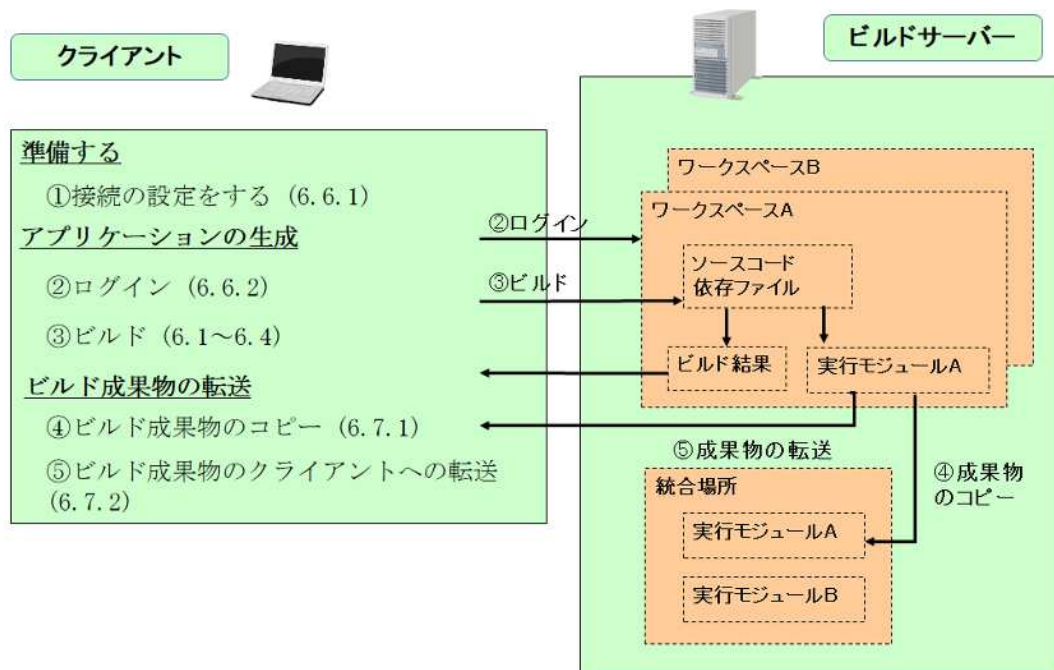


## 6.1 アプリケーションを生成する

### 【Linux】

COBOL 開発環境では、ビルド機能はリモートビルドのみを提供しています。

アプリケーションを生成する際のクライアントの操作の流れと、対応するビルドサーバー側の構成イメージを下記に示します。



ワークスペースとは、ビルドの対象のソースファイル、依存ファイルなど、また、ビルド後の成果物やビルド結果を格納する作業領域です。

### 注意事項

- ・ビルドサーバーはクライアントを識別するために、ワークスペースを使用することによって、異なるユーザーで同じワークスペースを使用することは可能です。ただし、同一ワークスペースへの複数ユーザーの同時ログインできません。

- ・ワークスペース配下には COBOL 開発環境が使用する制御情報が含まれています。ログイン中にこれらの情報を変更または削除した場合には、通信エラーが発生するなど意図しない動作となる可能性があります。

- ・プロジェクトに登録したソースファイルが存在するフォルダーにあるすべてのファイルが転送対象になります。ビルドに必要なないファイルを置く場合は、ファイルサイズ

に注意してください。

- ・プロジェクトフォルダーに存在するすべてのオブジェクトファイルがリンク対象になります。リンク対象外のオブジェクトファイルを置かないでください。

- ・UNC パスでソリューションファイルおよびプロジェクトファイルを開くと、アプリケーションの生成が正しく行えない場合があります。ネットワークドライブに割り当てて使用してください。

#### 【Enterprise(標準)】【Enterprise(SE)】【Standard】

COBOL 開発環境では、ビルド機能はローカル環境でのビルドのみを提供しています。

#### 【Enterprise(標準)】

##### **注意事項**

- ・プロジェクトフォルダーに存在するすべてのオブジェクトファイルがリンク対象になります。リンク対象外のオブジェクトファイルを置かないでください。

- ・プロジェクトフォルダーにあるオブジェクトファイルのうちオブジェクトファイル名 \*.obj のファイルは、アプリケーションの生成時に成果物の生成フォルダーに移動されリンク対象となり、プロジェクトフォルダーから削除されます。リンク対象にしたい固有のオブジェクトファイルのうちオブジェクトファイル名 \*.obj のファイルは、削除の対象にならないように別のフォルダーに置いてプロジェクトに登録してください。

### 6.1.1 プロジェクト単位で生成する

1. [ファイルビュー]で、アプリケーションを生成するプロジェクトを選択します。
2. 次の3つの方法のいずれかを実行します。
  - ・ [ビルド] - [プロジェクト名のビルド] コマンドを選択します。
  - ・ 右クリックメニューで [ビルド] を選択します。
  - ・ ツールバーから [ビルド] のボタンをクリックします。
3. プロジェクトのビルドが始まります。

ビルドをキャンセルする場合には、[ビルド] - [キャンセル] コマンドを選択します。

ビルド情報は出力タブに表示されます。

#### ビルド対象について

ビルド対象は、ソースファイル配下に登録されているソースファイルのみです。

#### 依存ファイルについて

- ・ ソースファイルのプロパティの依存ファイルに指定されているファイルが依存ファイルになります。依存ファイルの指定については、「4.7.3 依存ファイル」を参照してください。
4. ビルドが終了すると、出力タブに翻訳メッセージが表示されます。

## 6.1.2 エラーに位置づけるには

1. 出力タブをアクティブにします。
2. 位置づけたい翻訳メッセージ本文の行にカーソルを移動します。
3. 以下の3つの方法のいずれかを実行します。
  - ・マウスの左ボタンをダブルクリックします。
  - ・右クリックメニューから [ エラー位置づけ ] コマンドを選択します。
  - ・ツールバーの [ エラー位置づけ ] のボタンをクリックします。
4. 原始プログラムファイルを開いていない場合は、コンパイラが参照した原始プログラムファイルが編集ウィンドウとして開かれます。  
プリコンパイラによる中間ファイルにエラーが存在する場合、中間ファイルが開かれます。  
エラー位置づけの規則は、「2.1.3 エラー位置づけの規則」を参照してください。。
5. 前のエラーに位置づけるには、ツールバーの [ 前の場所へ移動 ] のボタンをクリックします。  
または、右クリックメニューから [ 前の場所へ移動 ] コマンドを選択します。
6. 次のエラーに位置づけるには、ツールバーの [ 次の場所へ移動 ] のボタンをクリックします。  
または、右クリックメニューから [ 次の場所へ移動 ] コマンドを選択します。
7. 表示されている内容を消去するには、ツールバーの [ すべてクリア ] のボタンをクリックします。  
または、右クリックメニューから [ すべてクリア ] コマンドを選択します。

### 6.1.3 ソリューション全体で生成する

ソリューション全体を対象にしてアプリケーションを生成する場合も、プロジェクト単位と手順は同じです。

1. 次の3つの方法のいずれかを実行します。
  - ・ [ビルド] - [ソリューションのビルド] コマンドを選択します。
  - ・ 右クリックメニューで [ソリューションのビルド] を選択します。
  - ・ ツールバーから [ソリューションをビルド] のボタンをクリックします。
2. ソリューションのビルドが始まり、含まれるプロジェクトが順番にビルドされ、出力タブにビルド情報が表示されます。

ビルドが終わった後にエラー行に位置づける方法は、「6.1.2 エラーに位置づけるには」を参照してください。

## 6.2 アプリケーションを再生成する

---

ビルドを行うと、前回のビルドからソースファイルや登録集原文ファイルが変更されている場合にコンパイルされます。

無条件にアプリケーションを生成しなおす場合に、再生成（リビルド）を行います。

アプリケーションの再生成は、生成時と同様に、プロジェクト単位およびソリューション全体で行うことが可能です。

### 6.2.1 プロジェクト単位で再生成する

1. [ファイルビュー] で、アプリケーションを生成するプロジェクトを選択します。
2. 次の3つの方法のいずれかを実行します。
  - ・ [ビルド] - [プロジェクト名のリビルド] コマンドを選択します。
  - ・ 右クリックメニューで [リビルド] を選択します。
  - ・ ツールバーから [リビルド] のボタンをクリックします。
3. プロジェクトのリビルドが始まります。

ビルドをキャンセルする場合には、[ビルド] - [キャンセル] コマンドを選択します。

ビルド情報は出力タブに表示されます。

リビルドが終わった後にエラー行に位置づける方法は、「6.1.2 エラーに位置づけるには」を参照してください。

## 6.2.2 ソリューション全体で再生成する

ソリューション全体を対象にしてアプリケーションを生成する場合も、プロジェクト単位と手順は同じです。

1. 次の3つの方法のいずれかを実行します。
  - ・ [ビルド] - [ソリューションのリビルド] コマンドを選択します。
  - ・ 右クリックメニューで [ソリューションのリビルド] を選択します。
  - ・ ツールバーから [ソリューションをリビルド] のボタンをクリックします。
2. ソリューションのリビルドが始まり、含まれるプロジェクトが順番にリビルドされ、出力タブにビルド情報が表示されます。

リビルドが終わった後にエラー行に位置づける方法は、「6.1.2 エラーに位置づけるには」を参照してください。

## 6.3 アプリケーションを削除する

---

ビルドしたアプリケーション（出力ファイル）を削除することが可能です。

### 6.3.1 プロジェクト単位で削除する

1. 次の2つの方法のいずれかを実行します。
  - ・ [ビルド] - [プロジェクト名のクリーン] コマンドを選択します。
  - ・ 右クリックメニューで [クリーン] を選択します。
2. クリーンが終了すると、プロジェクトのアプリケーション（出力ファイル）が削除されます。

### 6.3.2 ソリューション全体で削除する

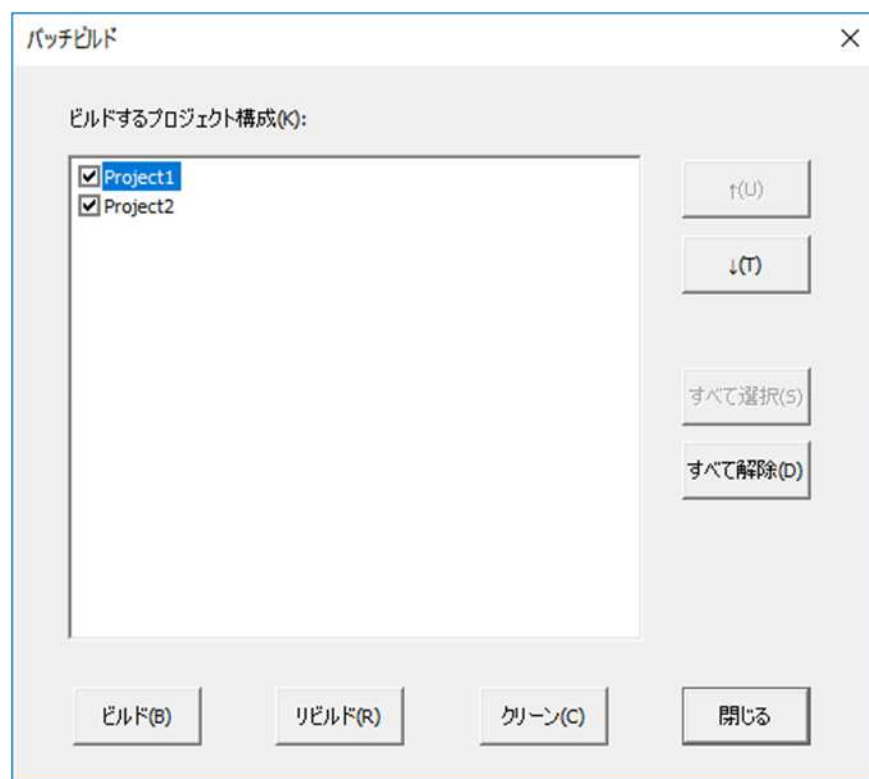
1. 次の2つの方法のいずれかを実行します。
  - ・ [ビルド] - [ソリューションのクリーン] コマンドを選択します。
  - ・ 右クリックメニューで [ソリューションのクリーン] を選択します。
2. ソリューションに含まれているプロジェクトのアプリケーションの削除が始まります。
3. クリーンが終了すると、ソリューションに含まれているプロジェクトのアプリケーション（出力ファイル）が削除されます。



## 6.4 バッチビルド

ソリューション内の複数のプロジェクトや構成を組み合わせ、ビルドやリビルドやクリーンを実行することができます。

1. [ファイルビュー]で、アプリケーションを生成したいプロジェクトが登録されているソリューションを選択します。
2. [ビルド] - [バッチビルド]コマンドを選択すると、[バッチビルド]ダイアログボックスが表示されます。



3. リストに表示されているプロジェクトの中から、ビルドやリビルドやクリーンをしたいプロジェクトのチェックボックスをチェックします。
4. リストの上から順にビルドが実行されますので、順番を変更する場合はリストの中のプロジェクト名をクリックして選択した状態で [ ] [ ] ボタンをクリックします。
5. チェックボックスをすべてチェックするには、[すべて選択] ボタンをクリックします。

6. チェックボックスをすべて解除するには、[すべて解除] ボタンをクリックします。
7. [ビルド] ボタンをクリックすると、対象のプロジェクトのビルドが始まり、出力タブにビルド情報が表示されます。
8. [リビルド] ボタンをクリックすると、対象のプロジェクトのリビルドが始まり、出力タブにビルド情報が表示されます。
9. [クリーン] ボタンをクリックすると、対象のプロジェクトのアプリケーションの削除が始まり、出力タブに削除情報が表示されます。
10. [閉じる] ボタンをクリックすると、何も行わずにダイアログボックスを閉じます。

## 6.5 アプリケーションを実行する

---

### 【Enterprise(標準)】【Enterprise(SE)】【Standard】

プロジェクトから生成されたアプリケーションを、メニューから実行することができます。

アプリケーション実行時の既定のディレクトリは、実行するアプリケーションが存在するディレクトリです。

プロジェクトのビルド/リビルドが成功し、アプリケーションが生成されると、アクティブなプロジェクトごとにアプリケーションファイル名が[ビルド]メニューに表示されます。

アプリケーションを実行する手順を説明します。

1. [ファイルビュー]で、プロジェクトを選択します。
2. [ビルド] - [実行]コマンドを選択すると、アプリケーションが実行されます。

### 【Linux】

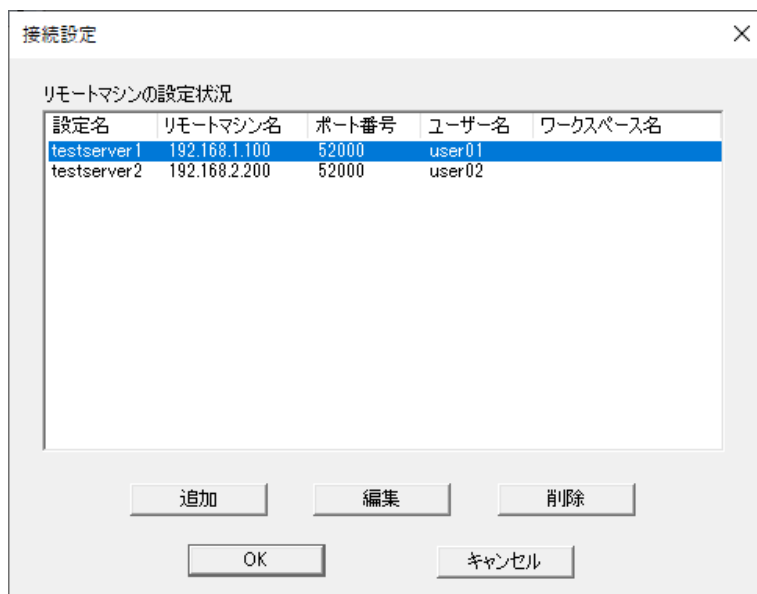
プロジェクトから生成されたアプリケーションを、メニューから実行することはできません。

## 6.6 【Linux】サーバー接続

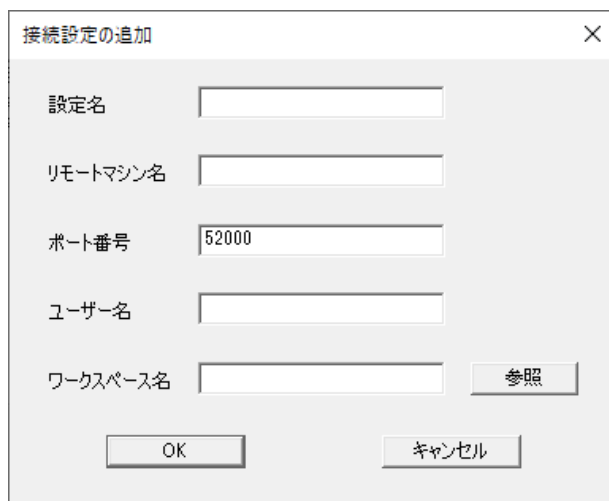
---

### 6.6.1 接続の設定

1. [ 環境 ] - [ 接続設定 ] コマンドを選択します。



2. [ 追加 ] ボタンを選択すると、[ 接続設定の追加 ] ウィンドウが表示されます。
3. 設定を入力します。



設定名	リモートマシン、ポート番号、ユーザー名、ワークセットのセットに対して指定した名前。設定名はログイン画面で接続先の一覧として表示され、利用者は設定名を選択してログイン先を決定する。 省略不可
リモートマシン名	ログインするビルドサーバーのホスト名またはIPアドレスを入力する。IPアドレスはIPv4のみが指定できる。 省略不可。
ポート番号	ビルドサーバーとの間で使用するポート番号を指定する。既定値は52000。 変更する場合はビルドサーバーのポート番号と同じ値を設定する。 省略不可。
ユーザー名	ビルドサーバーにログインするユーザー名を指定する。ユーザー名はビルドサーバーに登録されている必要がある。 省略不可。
ワークスペース名	ワークスペースとはビルドサーバー上の作業領域であり、ビルドサーバー上のディレクトリの絶対パスで指定します。ワークスペース名はクライアントの識別にも使用する。 省略した場合、ログインユーザーの「ホームディレクトリ/COBOL/WORKSPACE」がワークスペースになる。 入力制限の文字に加えて、「\」、「 」の文字を使用できない。 ワークスペース名は、ビルドサーバー上でのフルパスが108バイトまでの長さである必要がある。

4. [編集] ボタンを選択すると、[接続設定の編集] ウィンドウが表示されます。操作方法は[接続設定の追加]と同様です。
5. リストの中の行を選択した状態で[削除] ボタンを選択すると、接続情報が削除されます。
6. [OK] ボタンを選択します。

#### 【入力文字の制限】

項目名	入力文字数の制限	入力可能文字の制限
設定名	32 バイト	「タブコード、;」が入力不可
リモートマシン名	128 バイト	「タブコード、;」が入力不可
ポート番号	6 バイト	数字
ユーザー名	256 バイト	「タブコード、;」が入力不可
ワークスペース名	93 バイト	「タブコード、;」が入力不可

### 6.6.2 ログイン

利用者はリモートビルドを行う前に、ログインを行ってビルドサーバーとの接続を確立する必要があります。

ログインは同時に 1 つのサーバーに対してのみ行うことが可能です。

ログイン

設定名: testserver1

リモートマシン名: 192.168.1.100

ポート番号: 52000

ユーザー名: user01

パスワード:

ワークスペース名:

OK キャンセル

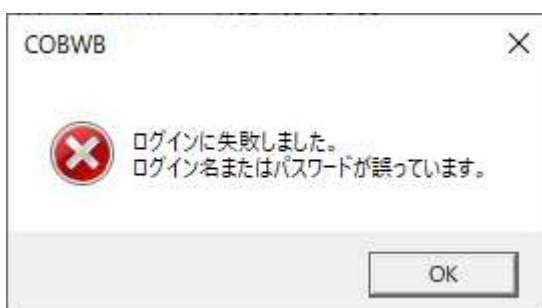
設定名	「接続設定」の画面で設定した設定名がドロップダウンリストで表示される。
リモートマシン名	設定名に対応したリモートマシン名が表示される。この画面では変更できない。 ドロップダウンリストの設定名の選択を変更すると、連動して表示される。
ポート番号	設定名に対応したポート番が表示される。この画面では変更できない。 ドロップダウンリストの設定名の選択を変更すると、連動して表示される。
ユーザー名	設定名に対応したユーザー名が表示される。 変更可能。 ドロップダウンリストの設定名の選択を変更すると、連動して表示される。
パスワード	ユーザー名に対応したパスワードを入力する。パ

	スワードは記憶されないため、ログインのたびに毎回入力する必要がある。パスワードは暗号化されてから送信される。
ワークスペース	設定名に対応したワークスペースが表示される。変更可能。 ドロップダウンリストの設定名の選択を変更すると、連動して表示される。 入力制限の文字に加えて、「\、"、 」の文字を使用できない。 ワークスペース名は、ビルドサーバー上でのフルパスが 108 バイトまでの長さである必要がある。

4. [OK] ボタンを選択します。ログインに成功したかどうかをダイアログボックスで表示します。



エラーがあった場合は、エラーの原因をあらわすメッセージが表示されます。



#### 【入力文字の制限】

項目名	入力文字数の制限	入力可能文字の制限
ユーザー名	256 バイト	「タブコード、;」が入力不可
パスワード	256 バイト	
ワークスペース名	93 バイト	「タブコード、;」が入力不可

### **注意事項**

- ビルドサーバーにログインしている状態で COBOL 開発環境を終了させ、次回起動後にログインせずに「ビルド」や「クライアントへ転送」などの機能を実行した場合、通信エラーが発生します。ログイン後に作業を継続してください。
- ワークスペース名は、ビルドサーバー上でのフルパスで 108 バイト以下の長さになる必要があります。画面での設定では 93 バイトまでを上限としていますが、相対パスで指定した場合は、サーバー上でのフルパスのファイル名が 108 バイトより大きいとサーバーへのログインはできません。

## 6.6.3 ログアウト

1. [ COBOL ] - [ ログアウト ] コマンドを選択します。

または、COBOL 開発環境を終了することでも自動的にログアウトします。



## 6.7 転送機能

---

【Standard】 【Enterprise(SE)】 【Enterprise(標準)】

COBOL 開発環境のビルドによって作成されたファイルを所定の場所にコピーする機能を提供します。

【Linux】

COBOL 開発環境のリモートビルドによって作成されたファイルは、ビルドサーバー上のワークスペース配下に格納されます。ビルドで作成されたファイルをビルドサーバー上の所定の場所にコピーする機能と、クライアント上の所定の場所に転送する機能を提供します。

転送機能を利用するには、あらかじめ転送内容を設定しておく必要があります。転送の設定については、「4.6.9 転送」を参照してください。

### **注意事項**

- ・ビルド中に「ビルド成果物のコピー」や「ビルド成果物のクライアントへの転送」を実行した場合、成果物が正しくコピーまたは転送されない可能性があります。ビルドが正常終了してから「ビルド成果物のコピー」や「ビルド成果物のクライアントへの転送」を実行してください。

### **注意事項【Linux】**

- ・コピーまたは転送の対象は、ビルドによる成果物の格納ディレクトリ(ワークスペースの BUILD 配下のプロジェクト/PROJECT/SRC)にあるファイルのみです。

【ワークスペース/BUILD】

プロジェクト名

PROJECT

SRC

### 6.7.1 ビルド成果物のコピー

【Standard】【Enterprise(SE)】【Enterprise(標準)】

ビルドによって作成されたファイルを所定の場所にコピーすることができます。

【Linux】

ワークスペース配下にあるビルド成果物を、ビルドサーバー内の任意のディレクトリにコピーすることができます。

あらかじめ、ログインしておく必要があります。

1. ファイルビューでコピーの対象のプロジェクトをクリックして選択状態にします。
2. [ビルド] - [ビルド成果物のコピー] コマンドを選択します。
3. コピーが終了すると、出力ウィンドウにログを出力します。

### 6.7.2 【Linux】ビルド成果物のクライアントへの転送

ワークスペースの下にあるビルド成果物を、クライアントに転送することができます。

あらかじめ、ログインしておく必要があります。

1. ファイルビューで転送の対象のプロジェクトをクリックして選択状態にします。
2. [ビルド] - [ビルド成果物のクライアントへの転送] コマンドを選択します。
3. 転送が終了すると、出力ウィンドウにログを出力します。

.

---

# 7 ツール

この章では、COBOL 開発環境で使用可能なツールの実行について説明します。

---

## 7.1 既存 COBOL 資産のインポート

---

### 【Linux】

既存 COBOL 資産をインポートするためのインポートツールを提供しません。

### 【Enterprise(標準)】【Enterprise(SE)】【Standard】

既存COBOL資産のインポートは、既存のプロジェクトファイル（\*.cpr、\*.hcpr）や複数プロジェクト保存ファイル（\*.mpr）から、COBOL 開発環境のプロジェクトファイル（\*.neccbproj）やソリューションファイル（\*.neccbisln）への変換を行います。

COBOL85 Proのプロジェクトファイル(\*.cpr)は、Standard Editionのプロジェクトファイルに変換します。

COBOL85 MFAのプロジェクトファイル(\*.hcpr)は、Enterprise Editionの標準動作モードのプロジェクトファイルに変換します。

インポートするには、あらかじめ、インポート対象配置ディレクトリへ、既存COBOL資産を配置しておく必要があります。

概要については、「1.4 既存COBOLからの移行手順の概要」を参照してください。

既存COBOL資産のインポートは、以下の手順で行います。

1. [ユーティリティ] - [既存 COBOL 資産のインポート] を選択します。

**既存COBOL資産のインポート**

**インポート対象ディレクトリ**  
 インポートするプロジェクトが置かれているディレクトリを指定してください。  
 ※指定ディレクトリ配下に存在するファイルは、インポートツールにより「インポート結果出力ディレクトリ」に自動的にコピーされます。  
 既存COBOLプロジェクトファイル配置ディレクトリ名 (必須)

**インポート結果出力ディレクトリ**  
 プロジェクトのインポート先ディレクトリを指定してください。  
 COBOLプロジェクトファイル出力ディレクトリ名 (必須)

**追加の検索ディレクトリ**  
 追加の検索ディレクトリがある場合に指定してください。  
 インポートツールは、指定ディレクトリからプロジェクト内のファイルを検索します。  
 ※検索ディレクトリが複数指定された場合は、最初にファイルが存在したディレクトリを有効とします。  
 COBOLソース検索ディレクトリ名 (セミコロン「;」で複数指定可)

登録集原文検索ディレクトリ名 (セミコロン「;」で複数指定可)

リンクライブラリ検索ディレクトリ名 (セミコロン「;」で複数指定可)

COBOL/Sマクロ原文ファイル検索ディレクトリ (セミコロン「;」で複数指定可)

GMPマクロオブジェクト登録ディレクトリ (セミコロン「;」で複数指定可)

インポート処理のオプション

実行 キャンセル

2. インポート対象配置ディレクトリを指定します。以下のディレクトリを指定してください。

- ・ 既存 COBOL プロジェクトファイル配置ディレクトリ名

パスを直接入力するか、参照ボタンを押下して、ディレクトリの参照ダイアログボックスでパスを指定します。指定したディレクトリ配下のプロジェクトファイルは階層を問わずすべて取得します。ディレクトリ指定は絶対パスで指定します。相対パスを設定した場合の動作は不定です。

プロジェクトファイルと同じディレクトリ以外に参照しているファイルがある場合、ファイル種別に応じて以下の入力項目に外部ディレクトリを指定する必要があります。

3. インポート結果出力ディレクトリを指定します。以下のディレクトリを指定してください。

- ・ COBOL プロジェクトファイル出力ディレクトリ名

パスを直接入力するか、参照ボタンを押下して、ディレクトリの参照ダイアログボックスでパスを指定します。ディレクトリ指定は絶対パスで指定します。

4. 追加の検索ディレクトリを指定します。以下のディレクトリを指定してください。

- COBOL ソース検索ディレクトリ名

プロジェクトファイルと異なるディレクトリにソースファイルが配置されている場合、配置ディレクトリを指定してください。

インポートツールは、プロジェクトの「ソースファイル」を検索し、ファイルパスを指定ディレクトリに更新します。

- 登録集原文検索ディレクトリ名

登録集原文が配置されているディレクトリを指定してください。

インポートツールは、出力プロジェクトファイルの「登録集原文ディレクトリ」に、ここで指定されたディレクトリを追加します。

またインポートツールは、プロジェクトの「依存ファイル」を検索し、ファイルパスを指定ディレクトリに更新します。

- リンクライブラリ検索ディレクトリ名

リンクライブラリが配置されているディレクトリを指定してください。

インポートツールは、出力プロジェクトの「リンクライブラリ検索ディレクトリ」に、ここで指定されたディレクトリを追加します。

またインポートツールは、プロジェクトの「その他ファイル」にあるリンクライブラリを検索し、ファイルパスを指定ディレクトリに更新します。

- COBOL/S マクロ原文ファイル検索ディレクトリ名

COBOL/S マクロ原文ファイルが配置されているディレクトリを指定してください。

インポートツールは、出力プロジェクトの「マクロ原文ファイル検索ディレクトリ」に、ここで指定されたディレクトリと、続けて既存プロジェクトファイル内の「COBOL/S マクロ原文ファイル検索ディレクトリ」に指定されたディレクトリを追加します。

- GMP マクロオブジェクト登録ディレクトリ名

GMP マクロオブジェクトが配置されているディレクトリを指定してください。

インポートツールは、出力プロジェクトの「マクロオブジェクト登録ディレクトリ」に、ここで指定されたディレクトリと、続けて既存プロジェクトファイル内の「GMP マクロオブジェクト登録ディレクトリ」に指定されたディレクトリを追加します。

これらのディレクトリは COBOL 開発環境のプロジェクトファイル中に、それぞれの参照先ディレクトリとして記録されます。

パスを直接入力するか、参照ボタンを押下して、ディレクトリの参照ダイアログボックスでパスを指定します。

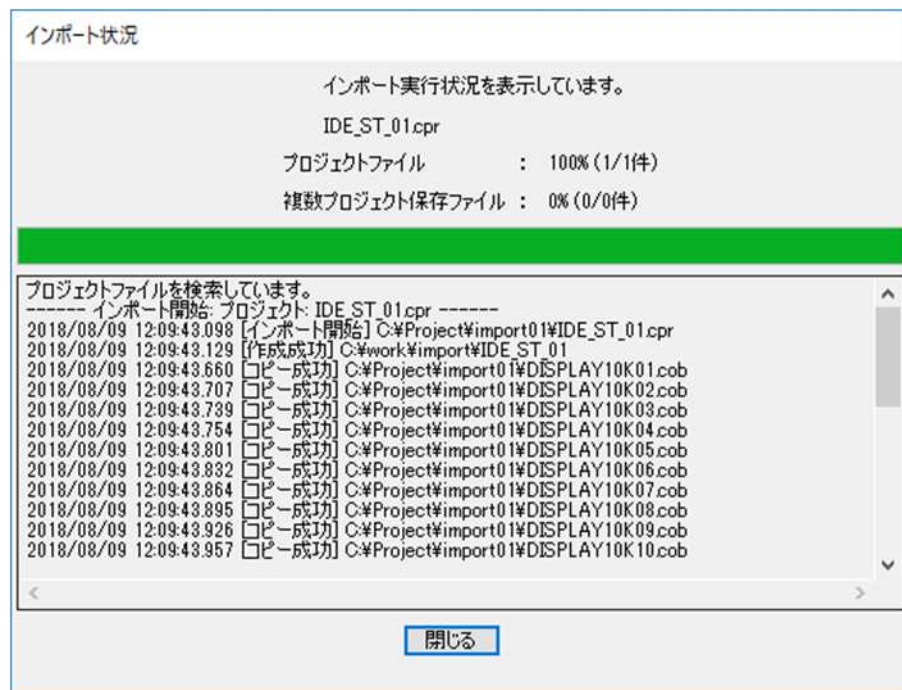
セミコロンによる区切りにより手入力の複数指定可とします。既に指定されて

いる場合に[参照]ダイアログボックスでディレクトリを指定した場合は、入力欄は上書きされます。

5. インポートを実行します。

[ 実行 ] ボタンをクリックすると、簡易チェックが行われ、確認ダイアログボックスが表示されますので[OK]を選択します。

[ キャンセル ] ボタンをクリックすると、インポートは行われません。インポート実行中およびインポート処理結果は「インポート状況」ダイアログ画面に表示されます。確認後、[閉じる]ボタンをクリックし、画面を閉じます。



2 にて指定した既存 COBOL プロジェクトファイル配置ディレクトリ配下のファイルは、インポート時に COBOL 開発環境のプロジェクトファイルの出力先にコピーされます。

ただし、以下の拡張子を持つファイルはコピーされません。

.exe、.obj、.dll、.lib、拡張子なし

埋込み SQL COBOL ソースファイルの場合は拡張子変更したファイルを出力先に作成するため、以下の拡張子の元ファイルはコピーされません。

.cbl、cob

同時に実行ログとして新プロジェクトディレクトリ配下にインポート時のログファイルが出力されます。ログファイルの自動削除などは行われませんので、利用者にて整理を行ってください。ログファイル名の形式は以下のようになっています。

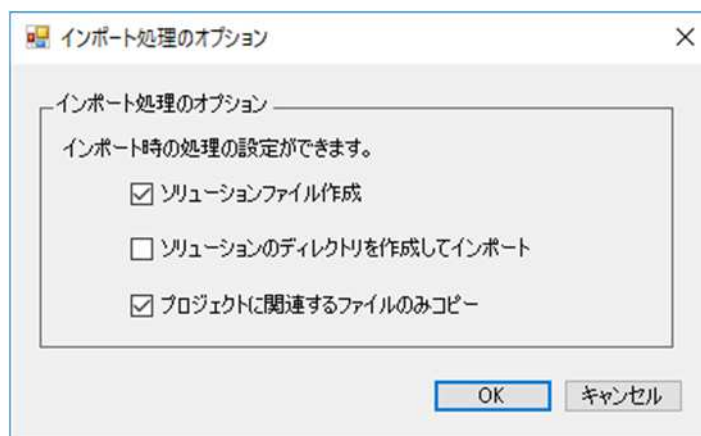
インポート時のログファイル名	説明
COBOL_Import_(yyyymmdd_hhmmss).log	yyyymmdd_hhmmss はログ取得終了時刻を表します。

## インポート処理のオプション



インポート処理の出力内容に対するオプションを以下の手順で指定できます。

1. [インポート処理のオプション]ボタンをクリックすると以下の画面が表示されます。



2. 各項目のチェックボックスを切り替え、[OK]ボタンをクリックします。  
[キャンセル]ボタンをクリックすると、変更内容は破棄されます。  
各項目の詳細は、画面上の項目名にマウスオーバーすることでポップアップ表示されます。

項目名	機能	デフォルト
ソリューションファイル作成	ソリューションファイル(.neccb1s1n)をインポート時に作成するかを選択します。 チェックあり ：プロジェクトごとにソリューションファイルを作成する チェックなし ：プロジェクトごとにソリューションファイルを作成しない	チェックあり
ソリューションのディレクトリを作成してインポート	インポート時にソリューションディレクトリを作成するかどうかを指定します。 チェックあり ：作成する （ソリューションディレクトリ配下に COBOL プロジェクトが出力される） チェックなし ：作成しない （出力先ディレクトリ配下に COBOL プロジェクトが出力される）	チェックなし
プロジェクトに関連するファイルのみコピー	インポート時にプロジェクトファイルに関連するファイルのみコピーするかどうかを指定します。	チェックあり

	チェックあり : プロジェクトに登録されたファイルのみコピー チェックなし : プロジェクトに登録されたファイル以外もコピー	
--	---	--

## ファイル分類

既存 COBOL 資産のインポートでは、拡張子によってファイルを「ソースファイル」と「その他ファイル」に分類して管理します。

COBOL85 Pro のプロジェクトファイルのインポートでは、ソースファイルの拡張子は、.cbl、.cob、.cbf、.cbx、.pco、.c、.cpp、.rc で、ビルド対象として認識されるファイルです。

COBOL85 MFA のプロジェクトファイルのインポートでは、ソースファイルの拡張子は、.cbl、.cob、.cbf、.cbx、.cblm、.cobm、.pco、.pcom、.cblss、.cblmss、.pcoss、.pcomoss、.c、.cpp、.rc で、ビルド対象として認識されるファイルです。



その他ファイルの拡張子は、上記以外で、ビルド対象外として認識されるファイルです。

また、埋込み SQL COBOL ソースファイルについては、以下のように拡張子を変更したファイルを「ソースファイル」に分類して管理します。

拡張子 (変更前)	拡張子 (変更後)
.cbl	.qcbI
.cob	.qcob
.cbx	変換しません (非対応)










## エラーおよび警告の対処





メッセージダイアログ、インポート状況ダイアログボックスをよび、実行ログにエラーまたは警告が表示された場合以下の対処を行ってください。














種別 メッセージボックス: 、インポート状況および実行ログ: 











種別	メッセージ内容	対処
----	---------	----

	既存 COBOL プロジェクトファイル配置ディレクトリを指定してください。	インポート対象プロジェクトファイルの配置ディレクトリを、既存 COBOL プロジェクトファイル配置ディレクトリ名コンボボックスに指定し、再度インポートを実行してください。
	既存 COBOL プロジェクトファイル配置ディレクトリの複数指定はできません。	インポート対象プロジェクトファイルの配置ディレクトリを、1つだけ既存 COBOL プロジェクトファイル配置ディレクトリ名コンボボックスに指定し、再度インポートを実行してください。
	既存 COBOL プロジェクトファイル配置ディレクトリが存在しません。	インポート対象プロジェクトファイルの配置ディレクトリが正しいか確認し、再度インポートを実行してください。
	COBOL ソース検索ディレクトリが存在しません。	インポート対象 COBOL ソースの検索ディレクトリが正しいか確認し、再度インポートを実行してください。
	登録集原文検索ディレクトリが存在しません。	インポート対象登録集原文検索ディレクトリが正しいか確認し、再度インポートを実行してください。
	リンクライブラリ検索ディレクトリが存在しません。	インポート対象リンクライブラリ検索ディレクトリが正しいか確認し、再度インポートを実行してください。
	COBOL プロジェクトファイル出力ディレクトリを指定してください。	インポート結果を出力するディレクトリを COBOL プロジェクトファイル出力ディレクトリ名コンボボックスに指定し、再度インポートを実行してください。
	COBOL プロジェクトファイル出力ディレクトリの複数指定はできません。	インポート結果を出力するディレクトリを、1つだけ COBOL プロジェクトファイル出力ディレクトリ名コンボボックスに指定し、再度インポートを実行してください。
	[作成失敗]<COBOL プロジェクトファイル出力ディレクトリ名>	COBOL プロジェクトファイル出力ディレクトリ名コンボボックスに指定したディレクトリにアクセスできることを確認し、再度インポートを実行してください。
	[読込失敗]<既存 COBOL プロジェクトファイル配置ディレクトリ名>	既存 COBOL プロジェクトファイル配置ディレクトリ名コンボボックスに指定したディレクトリにアクセスできることを確認し、再度インポートを実行してください。
	[変換失敗]<.neccbproj ファイル>	インポート対象プロジェクトファイル (*.cpr) が正常でない可能性があります。インポート対象プロジェクトファイルを確認し、再度インポートを実行してください。
	[更新失敗]<.neccbproj ファイル>	インポート対象プロジェクトファイル (*.cpr) が正常でない可能性があります。インポート対象プロジェクトファイルを確認し、再度インポートを実行してください。
	[コピー失敗]<ファイル名>	インポート対象ファイルのコピー処理で失敗。インポート対象配置ディレクトリおよび、インポート結果出力ディレクトリにアクセスできることを確認し、再度インポートを実行してください。

	[プロジェクト情報取得失敗] <プロジェクトファイルパス>	オプション「プロジェクトに関連するファイルのみコピー」チェック時のインポート対象ファイルのコピー処理で失敗。 インポート対象プロジェクトファイル（*.cpr）が正常でない可能性があります。インポート対象プロジェクトファイルを確認し、再度インポートを実行してください。
	プロジェクトに関連するファイル情報の取得に失敗したため、すべてのコピー対象ファイルをコピーします。	オプション「プロジェクトに関連するファイルのみコピー」チェック時のインポート対象ファイルのコピー処理で失敗。 インポート対象プロジェクトファイル（*.cpr）が正常でない可能性があります。インポート対象プロジェクトファイルを確認し、再度インポートを実行してください。
	[コピー失敗]<ファイルパス> コピー対象ファイルが存在しません。	オプション「プロジェクトに関連するファイルのみコピー」チェック時のインポート対象ファイルのコピー処理で失敗。 インポート対象配置ディレクトリおよび、インポート結果出力ディレクトリにアクセスできることを確認し、再度インポートを実行してください。 各検索ディレクトリが正しいことを確認し、再度インポートを実行してください。
	[参照失敗]ファイル名 指定ディレクトリ内にファイルは存在しませんでした。	プロジェクトファイルで指定したディレクトリにファイルが存在しているか、インポート対象プロジェクトファイルおよびディレクトリ構成を確認し、再度インポートを実行してください。
	同名のプロジェクトファイル名が存在するため、インポートをキャンセルします。	既存 COBOL プロジェクトファイル配置ディレクトリ名コンボボックスに指定したディレクトリ配下に、同名のプロジェクトファイルが存在すると、インポートがキャンセルされます。インポート時に後からインポートする内容で上書きしてしまうためです。同名のプロジェクトファイルをリネームするなどして、再度インポートを実行してください。
	プロジェクトファイルは見つかりませんでした。	インポート対象プロジェクトファイルの配置ディレクトリを確認して、再度インポートを実行してください。
	警告：<製品名>は非対応です。	COBOL 拡張システムサブルーチン以外の製品が、使用する製品として選択されていますが、非対応なのでインポート後のプログラムを確認してください。
	警告：<拡張子>ファイルは非対応です。	インポートファイルにビルド対象外のファイルが含まれていた場合、「その他ファイル」としてインポートされます。インポート後のプロジェクトを確認してください。ビルド対象のファイル拡張子は、.cbl、.cob、.cbf、.cbx、.c、.cpp、.rc です。
	警告：アプリケーション種別が「標準」以外のプロジェクトは非対応です。	アプリケーション種別が、コンポーネント（クライアント）、または、コンポーネント（サーバ）は非対応なので、コンポーネントではないとみなしてインポートしています。

		インポート後のプログラムを確認してください。
	警告：<ファイル名>リンクライブラリ検索ディレクトリの優先順を確認ください。"	<p>インポート元.cpr ファイルのリンカオプションに、パスを含むファイル名の指定があった場合、この警告が出力されます。</p> <p>警告メッセージの&lt;ファイル名&gt;には、パスを含むファイル名のファイル名部分が出力されます。</p> <p>インポート元.cpr ファイルのリンカオプションに指定した、パスを含むファイル名は、パス部分がリンクライブラリ検索ディレクトリ、ファイル名がリンクライブラリ、に分割されインポートされます。また、インポート元.cpr ファイルのリンカオプションに /LIBPATH:&lt;パス&gt;で指定したパスは、リンクライブラリ検索ディレクトリにインポートされます。ファイル名のみ指定したファイル名は、リンクライブラリにインポートされます。</p> <p>インポート元.cpr ファイルのリンカオプションに以下のように指定された場合、インポート後対処が必要となる可能性があります。</p> <p>例) /libpath:"C:¥linklib1" "C:¥linklib2¥Object2.obj" Object1.obj</p> <p>この場合、インポートされたプロジェクトでは、リンクライブラリ検索ディレクトリは、C:¥linklib1;C:¥linklib2 と登録され、リンクライブラリは、Object2.obj;Object1.obj; と登録されます。</p> <p>ビルド時、リンクライブラリの Object2.obj ファイルは、C:¥linklib1 を初めに検索します。そのため、同名の別ファイル Object2.obj が存在した場合、本来期待する "C:¥linklib2¥Object2.obj" と異なるファイルが選択されビルドエラーとなります。</p> <p>この場合、[プロジェクト]-[プロパティ]でプロパティページを開き、[リンカ]ノードのリンクライブラリ検索ディレクトリの[...]ボタンを押下し、[フォルダ選択]画面で優先順を変更し対処します。</p>
	警告：このソースファイルは SQLEXTENSION の指定に従い拡張子変換が必要です。<ファイル名>	インポート対象の埋込み SQL COBOL ソースファイルの検索ディレクトリを確認して、再度インポートを実行してください。
	警告：拡張子が cbx の COBOL ソースに対する「SQL 記述ソース」の指定は非対応です。<ファイル名>	拡張子が cbx の COBOL ソースは埋込み SQL COBOL ソースファイルとして指定できません。インポート後のプログラムを確認してください。
	[変換失敗]<mpr ファイル名>同名の複数プロジェクト保存ファイルが存在するため、変換対象外です。	既存 COBOL プロジェクトファイル配置ディレクトリ名コンボボックスに指定したディレクトリ配下に、同名のファイルが存在すると、複数プロジェクト保存ファイル (*.mpr) がインポートできません。インポート時に後から

		<p>インポートする内容で上書きしてしまうためです。</p> <p>同名のファイルをリネームするなどして、再度インポートを実行してください。</p>
	<p>[変換失敗] : &lt;mpr ファイル名&gt;ファイルに登録されたプロジェクトが存在しません。&lt;プロジェクトファイル名&gt;</p>	<p>インポート対象の複数プロジェクト保存ファイル (*.mpr) に記載されているプロジェクトファイルが確認できません。</p> <p>プロジェクトファイルで指定したディレクトリにファイルが存在しているか、インポート対象プロジェクトファイルおよびディレクトリ構成を確認し、再度インポートを実行してください。</p>
	<p>[変換失敗]&lt;mpr ファイル名&gt;複数プロジェクト保存ファイルの変換に失敗しました。</p>	<p>インポート対象の複数プロジェクト保存ファイル (*.mpr) が正常でない可能性があります。インポート対象の複数プロジェクト保存ファイルを確認し、再度インポートを実行してください。</p>
	<p>警告：画面情報ファイル &lt;画面情報ファイル名&gt; が見つかりません</p>	<p>インポート対象プロジェクトファイル (*.cpr) に設定されている画面情報ファイル (*.scn) が確認できません。</p> <p>プロジェクトファイルで指定したディレクトリにファイルが存在しているか、インポート対象プロジェクトファイルおよびディレクトリ構成を確認し、再度インポートを実行してください。</p>
	<p>警告：「呼び出し規約対応リスト [/Hb]」は移行できません。</p>	<p>指定されたオプションはインポートできません。インポート後のプログラムを確認してください。</p>
	<p>警告：「翻訳規格 74(PC-COBOL) [/C74]」は移行できません。</p>	<p>指定されたオプションはインポートできません。インポート後のプログラムを確認してください。</p>
	<p>警告：「デバッグ情報を出力する [/g]」は移行できません。</p>	<p>指定されたオプションはインポートできません。インポート後のプログラムを確認してください。</p>
	<p>警告：「データ属性に合わせた小入力を行う [/Ca]」は移行できません。</p>	<p>指定されたオプションはインポートできません。インポート後のプログラムを確認してください。</p>
	<p>警告：「等価規則適用 [/Ce]」は移行できません。</p>	<p>指定されたオプションはインポートできません。インポート後のプログラムを確認してください。</p>
	<p>警告：「ハードタブ間隔の変更 [/T]」は移行できません。</p>	<p>指定されたオプションはインポートできません。インポート後のプログラムを確認してください。</p>
	<p>警告：「ヘッダーと改ページの抑制 [/Pn]」は移行できません。</p>	<p>指定されたオプションはインポートできません。インポート後のプログラムを確認してください。</p>
	<p>警告：「SORTKIT を使用しない [/Ckn]」は移行できません。</p>	<p>指定されたオプションはインポートできません。インポート後のプログラムを確認してください。</p>
	<p>警告：「IFASPRO 一般索引ファイルを使用する [/Cfii]」は移行できません。</p>	<p>指定されたオプションはインポートできません。インポート後のプログラムを確認してください。</p>
	<p>警告：端末内部コード [/Cn] が使用できません。</p>	<p>指定されたオプションは、[プログラムの固有文字集合 (内部コード) を指定する]=SJIS と</p>

		してインポートされました。インポート後のプログラムを確認してください。
	警告：「翻訳規格 74(PTOS,A-VX5)[/C74P]」は移行できません。	指定されたオプションはインポートできません。インポート後のプログラムを確認してください。
	警告：「COBOL データアクセスオプションをパースする」は移行できません。	指定されたオプションはインポートできません。インポート後のプログラムを確認してください。
	警告：「環境変数 PATH」は移行できません。	指定されたオプションはインポートできません。インポート後のプログラムを確認してください。
	警告：「環境変数 TMP」は移行できません。	指定されたオプションはインポートできません。インポート後のプログラムを確認してください。
	警告：「オブジェクトファイル[/Fo]」は移行できません。	指定されたオプションはインポートできません。インポート後のプログラムを確認してください。
	警告：「アイコンファイル[/Fi]」は移行できません。	指定されたオプションはインポートできません。インポート後のプログラムを確認してください。
	警告：「COBOL ビルド指示ファイル[/Fb]」は移行できません。	指定されたオプションはインポートできません。インポート後のプログラムを確認してください。
	警告：「作業用ディレクトリ[/t]」は移行できません。	指定されたオプションはインポートできません。インポート後のプログラムを確認してください。
	警告：「メイク時に帳簿 Ex を起動」は移行できません。	指定されたオプションはインポートできません。インポート後のプログラムを確認してください。
	警告：「フォーム定義ファイル一覧」は移行できません。	指定されたオプションはインポートできません。インポート後のプログラムを確認してください。

## 注意事項

・以下に COBOL85 Pro のプロジェクトファイルのインポートで移行されるオプションと移行されないオプションの一覧を示します。（○：移行する。×：移行しない。）

画面項目名[翻訳時オプション]	移行
パラメータリスト[/Hp]	
原始プログラムリスト[/Hs]	
登録集原文展開リスト[/Hsc]	
データマップリスト[/Hd]	
クロスリファレンスリスト[/Hx]	

呼び出し規約対応リスト[/Hb]	×
デバッグ行	
メインプログラム[/M]	
最適化[/00][/01][/02]	
選択行指定[/N]	
選択行指定[/Na]	
翻訳規格 74(PC-COBOL) [/C74]	×
不正 10 進データチェックを行わない[/Cd]	
添字の範囲外参照チェックを行わない[/Co]	
デバッグ情報を出力する[/g]	×
BINARY 項目の 10 進数桁落しをしない[/Ct]	
データ属性に合わせた小入力を行う[/Ca]	×
等価規則適用[/Ce]	×
ハードタブ間隔の変更[/T]	×
ヘッダーと改ページの抑制[/Pn]	×
SORTKIT を使用しない[/Ckn]	×
IFASPRO 一般索引ファイルを使用する[/Cfii]	×
翻訳規格 74(PTOS、A-VX5) [/C74P]	×
作成ファイル種別	
メインプログラム	
画面節の実行で View Generator/WEBCOBOL を使用する [/Cg]	
コンソールアプリケーションを作成する[/Ccon]	
警告エラーを抑制する[/NOWARNING]	
COBOL WEB アプリケーションを作成する[/Cweb]	×



COMP 項目の用途変更[/C5]	
アプリケーション種別	×
COBOL データアクセスオプションを指定する	×
数字と符号の重ね合わせ形式を変更する[/CS]	
重ね符号桁の空白文字を不正 10 進エラーとしない[/CB]	
すべて ZERO で初期化[/vi]	
SPACE で初期化[/vs]	
ZERO で初期化[/vz]	
16 進値で初期化[/vnn]	
APPLY SHIFT-CODE 句をコメントとする[/Cy]	
マニフェストファイルを分離する[/q]	
環境変数 PATH	×
環境変数 CBLCPY	
環境変数 LIB	
環境変数 TMP	×
リスト出力ディレクトリの変更[/lf]	
オブジェクトファイル[/Fo]	×
実行可能ファイル[/Fe]	
アイコンファイル[/Fi]	×
レジストリキー名[/Fr]	
COBOL ビルド指示ファイル[/Fb]	×
登録集原文ディレクトリ[/l]	
作業用ディレクトリ[/t]	×
C コンパイルオプション	
リソースコンパイルオプション	

リンカオプション	
オブジェクトファイル出力先ディレクトリ	
実行可能ファイル出力先ディレクトリ	
依存ファイル名	
0-9 のうち複数 [/N[0-9]] 選択行指定の文字列	
16 進値で初期化 [/vnn] 16 進値	
登録ファイル名一覧（プロジェクトの編集画面）	
使用する製品名	
Pro*COBOL オプション（オプション設定画面）	
メイク時に帳楽 Ex を起動	×

・以下に COBOL85 MFA のプロジェクトファイルのインポートで移行されるオプションと移行されないオプションの一覧を示します。（ ：移行する。×：移行しない。）

画面項目名[翻訳時オプション]	移行
メインプログラム[-M]	
パラメータリスト[-Hp]	
原始プログラムリスト[-Hs]	
登録集原文展開リスト[-Hc]	
データマップリスト[-Hd]	
クロスリファレンスリスト[-Hx]	
用語値名参照リスト[-He]	×
用語の展開リスト[-Hf]	×
用語名リスト[-Hm]	×
デバッグモード：	

有効[-Dy] 無効[-Dn]	
最適化[-O[012]]	
選択行指定[-N[0-9]]	
選択行指定[-Na]	
不正 10 進データチェックを行わない[-Cd]	
添字の範囲外参照チェックを行わない[-Co]	
BINARY 項目の 10 進数桁落としをしない[-Ct]	
外部名の英小文字を英大文字と等価としない[-Cs]	
警告エラーを抑制する[-Hw]	
プログラム固有文字集合を Unicode とする[CU]	
CODE-SET 句が指定されたものとして扱う[-CC]	
CODE-SET 句が指定されたものとして扱う[-CC]	
文字列を予約語としない[-U]	
16 進値で初期化する場合の 16 進値[-v"nn"]	
同桁同符号の外部 10 進数同士の転記を英数字転記で行う [-Cr]	
符号 S がない COMP-3 項目を符号領域なしとして動作させる [-CN]	
USAGE COMP を USAGE COMP-5 とする[-C5]	
INITIAL 句と CANCEL 命令は未使用のため、生成オブジェクトを効率化する[-Ci]	
重ね符号桁の表現を Pro*COBOL 互換とする[-CS]	
システム日付を取得する[-CD]	
重ね符号桁の空白文字を不正 10 進数エラーとしない[-CB]	
外部データ名中のハイフンを下線と等価とする[-Cu]	

実行効率を低下させる記述を指摘する [-Cm]	
オブジェクトコードを PIC 形式で出力する [-Cz]	
データ記述項に SYNCHRONIZED 句が指定されたものとみなす [-Ca]	
登録集原文ディレクトリパス	
データ項目のサイズとアラインを ACOS-4 互換とする [-Fa]	
ポインタデータ項目の割り付け境界を ACOS-4 互換とする [-Fb]	
COMP データの内部表現形式を ACOS-4 互換とする [-Fc]	
比較規則を ACOS-4 互換とする [-Fd]	
NUMERIC および不正 10 進データの検査を ACOS-4 互換とする [-Fe]	
小入出力先の既定値を ACOS-4 互換とする [-Fj]	
STOP 命令の ABORT 指定を可能とする [-Fl]	
中間結果領域の小数部桁数を指定する [-Fr]	
英数字から外部 10 進数への転記時にゾーン部の補正を行わない [-Fs]	
符号なし外部 10 進数の最右端バイトの符号部(S)のチェックを行う [-Ft]	
リンカオプション	
実行可能ファイル名	
Pro*COBOL オプション	
COBOL 原始ファイル拡張子選択	
GMP オプション	
マクロオブジェクト登録ディレクトリ名	
COBOL/S マクロオプション (COBOL/S マクロ展開ツール)	

COBOL/S オプション	
マクロオブジェクト登録ディレクトリ名	

・インポートを行うと既存 COBOL プロジェクトのディレクトリ配下のファイルを COBOL プロジェクトのディレクトリ配下にコピーします。登録集原文ファイルなどが既存 COBOL プロジェクトのディレクトリ配下に存在する場合は、インポート後に適切に配置してください。

・「;」を含むディレクトリは指定しないでください。指定された場合、インポートが失敗する場合があります。

・インポート対象プロジェクトファイル(cpr)の配下に存在するディレクトリを COBOL ソース検索ディレクトリに指定することはできません。指定されると正常にインポートされない場合があります。

例として、以下のようなディレクトリ指定はできません。

既存COBOL資産のインポート

インポート対象ディレクトリ  
 インポートするプロジェクトが置かれているディレクトリを指定してください。  
 ※指定ディレクトリ配下に存在するファイルは、インポートツールにより「インポート結果出力ディレクトリ」に自動的にコピーされます。  
 既存COBOLプロジェクトファイル配置ディレクトリ名 (必須)  
 C:\Project\Import01

インポート結果出力ディレクトリ  
 プロジェクトのインポート先ディレクトリを指定してください。  
 COBOLプロジェクトファイル出力ディレクトリ名 (必須)  
 C:\work\Import

追加の検索ディレクトリ  
 追加の検索ディレクトリがある場合に指定してください。  
 インポートツールは、指定ディレクトリからプロジェクト内のファイルを検索します。  
 ※検索ディレクトリが複数指定された場合は、最初にファイルが存在したディレクトリを有効とします。  
 COBOLソース検索ディレクトリ名 (セミコロンで複数指定可)  
 C:\Project\Import01\src

登録集原文検索ディレクトリ名 (セミコロンで複数指定可)

リンクライブラリ検索ディレクトリ名 (セミコロンで複数指定可)

COBOL/Sマクロ原文ファイル検索ディレクトリ (セミコロンで複数指定可)

GMPマクロオブジェクト登録ディレクトリ (セミコロンで複数指定可)

インポート処理のオプション

実行 キャンセル

この場合、COBOL ソース検索ディレクトリに指定したディレクトリ (またはディレクト

リ内のファイル)を別の場所に配置し、配置し直したディレクトリを COBOL ソース検索ディレクトリに指定してください。

- ・複数プロジェクトのインポート時、インポート対象のあるプロジェクトに登録されたファイルが、同時にインポートする他プロジェクトの配下に存在する場合、正常にインポートされない場合があります。

この場合、他プロジェクト配下に配置されたファイルについて、以下のいずれかの対処を実施した後再度インポートを実施してください。

- ・ファイルが登録されたプロジェクト配下にファイルを配置し、プロジェクトのファイル参照先を変更する。
- ・既存 COBOL プロジェクトファイル配置ディレクトリ以外のフォルダーにファイルを配置し、そのフォルダーを検索ディレクトリに指定する。

## 7.2 外部ツールの登録

COBOL 開発環境では、ユーティリティ（外部ツール）を簡単に呼び出せるように登録しておくことが可能です。

1. [環境] - [ユーティリティ登録] コマンドを選択すると、以下のよう  
なダイアログボックスが表示されます。

ユーティリティ登録

コマンドライン

メニュー項目名

☐ 起動時にパラメータを入力する

参照...  
追加

登録メニュー情報

変更  
削除

OK  
キャンセル

2. 実行するコマンドを [コマンドライン] テキストボックスに入力しま  
す。[参照] ボタンで実行可能ファイルを選択することも可能です。
3. メニューに表示される文字列を [メニュー項目名] テキストボックスに  
入力します。
4. 起動時に引数を入力する場合には、[起動時にパラメータを入力する]  
をチェックします。
5. [ユーティリティ] メニューから呼び出せるようにコマンドを登録する  
場合には、[追加] ボタンをクリックします。
6. [ユーティリティ] メニューからコマンドを削除する場合には、[登録  
メニュー情報] リストボックスのメニュー項目名を選択して [削除] ボ  
タンをクリックします。
7. 登録したユーティリティ情報を変更する場合には、[登録メニュー情  
報] リストボックスのメニュー項目名を選択して [変更] ボタンをクリ  
ックします。
8. [OK] ボタンをクリックすると、[登録メニュー情報] リストボックス  
に表示された内容が登録されます。

[ キャンセル ] ボタンをクリックすると、設定内容は登録されません。

登録しているユーティリティの項目名は、[ ユーティリティ ] メニューを選択すると項目名として表示されます。この項目名を選択することで、ユーティリティを起動することができます。起動時に引き数を指定するように設定していれば、[ パラメータ指定 ] ダイアログボックスが表示されます。



#### **注意事項**

- ・ユーティリティ情報は利用者別に書き登録されます。



### 7.3 【Standard】 COBOL GUI View Generator

---

COBOL GUI View Generator を使用してプロジェクトのファイルの作成、編集を行います。

[ユーティリティ] - [COBOL GUI View Generator] を選択すると、COBOL GUI View Generator が起動します。

プロジェクトのプロパティのオプション製品「画面情報ファイル名」にファイル名が指定されている場合、指定の画面情報ファイルを開いた状態で COBOL GUI View Generator が起動します。

指定がない場合、COBOL GUI View Generator のみが起動します。

COBOL GUI View Generator の詳細は「COBOL GUI ユーザーズガイド」を参照してください。

## 7.4 【Standard】画面節移行ファイルの作成

---

[ユーティリティ] - [画面節移行ファイルの生成] メニューによる、画面節移行ファイル生成の手順を説明します。

画面節移行ファイルの詳細は「COBOL GUI ユーザーズガイド」を参照してください。

1. [ユーティリティ] - [画面節移行ファイルの生成] を選択すると、以下のサブメニューが表示されます。
  - ・プロジェクト内のすべてのソースファイル
  - ・ファイルビューで選択されたソースファイル
2. [プロジェクト内のすべてのソースファイル] を選択すると、プロジェクトに属するソースファイルすべてを移行対象として、画面節移行ファイルが生成されます。

[ファイルビューで選択されたソースファイル] を選択すると、ファイルビューの[ソースファイル]で選択したファイルを移行対象として、画面節移行ファイルが生成されます。

以下のように、移行対象とするソースファイルをファイルビュー上で選択してからメニューを実行してください。



### 注意事項

- ・ソリューションに複数のプロジェクトが存在する場合は、1つのプロジェクト内のソースファイルを選択してください。
- ・複数のプロジェクトのファイルを選択すると、画面節移行ファイルは作成されません。

3. [出力]ウィンドウの[出力]タブに処理結果が表示されます。  
エラーが発生した場合は、「COBOL プログラミングの手引」にしたがって  
対処を行ってください。

また、メニュー実行時、COBOL プロジェクトディレクトリ配下に以下の構成で作業用ファイル、ディレクトリが作成されます。

**【プロジェクトディレクトリ】**

generatessx.bat

ssxwork

移行対象となるソースファイルのコピー

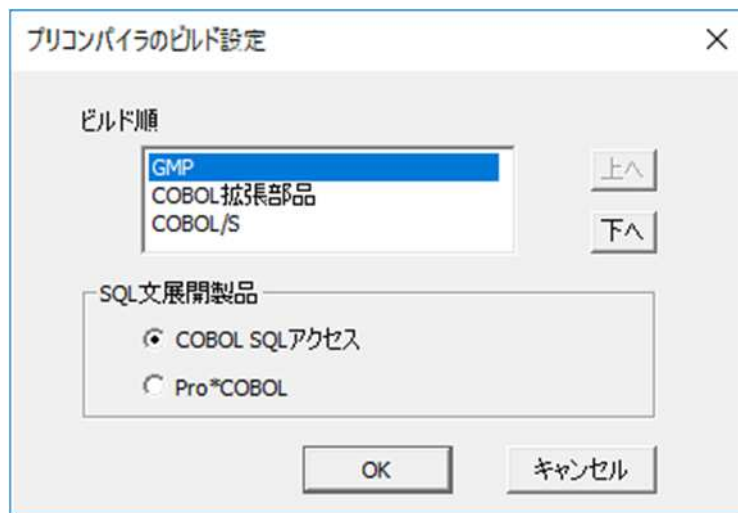
4. 処理が正常終了すると、画面節移行ファイルは COBOL プロジェクトファイルと同じディレクトリに作成されます。  
ファイル名は、プロジェクトのプロパティのオプション製品「画面情報  
ファイル名」に設定されたファイル名から、拡張子を ".SSI" に変更した  
ものになります。

## 7.5 【Enterprise(標準)】【Linux】プリコンパイラのビルド設定

[ユーティリティ] - [プリコンパイラのビルド設定]メニューによる、ユーザ単位にプリコンパイラ起動順序と SQL 文展開製品の既定値を登録する手順を説明します。

ビルド順指定対象のプリコンパイラは、「GMP」「COBOL/S」「COBOL 拡張部品」です。

1. [ユーティリティ] - [プリコンパイラのビルド設定]を選択すると、「プリコンパイラのビルド設定」ダイアログボックスが表示されます。



2. ビルド順を、「上へ」「下へ」ボタンで変更します。
3. ソースプログラム内に SQL 文が記述されている拡張子「.pcom」、  
「.pcoss」、「.pcoe」、「.pcomss」、「.pcome」、「.pcosse」、  
「.pcomsse」のファイルについて、ビルド時に COBOL SQL アクセスか  
Pro\*COBOL のどちらのプリコンパイラを起動するかを設定します。
4. 「OK」ボタンをクリックすると、ユーザ単位のレジストリ情報に登録されます。
5. ダイアログボックスを閉じるには [キャンセル] ボタンをクリックします。

設定した既定値は、プロジェクトを新規作成する場合や、プリコンパイラのビルド設定がされていないプロジェクトを開く場合に使用されます。

## 7.6 データレコード情報ファイルの作成

---

本機能の使用には、COBOL Assistant Option のインストールが必要です。

[ユーティリティ] - [データレコード情報の出力] メニューによる、データレコード情報ファイル生成の手順を説明します。

データレコード情報ファイルの詳細は「COBOL ユーザーズガイド」を参照してください。

### 【Linux】

本機能の使用の前に、COBOL Assistant Option がインストールされているサーバーにログインしてください。

1. ファイルビューで対象のプロジェクトをクリックし選択状態にします。  
プロジェクトが選択対象でない場合は、メニューを選択してもデータレコード情報ファイルは作成されません。
2. [ユーティリティ] - [データレコード情報ファイルの生成] を選択すると、以下のサブメニューが表示されます。
  - ・プロジェクト内のすべてのソースファイル
  - ・プロジェクトフォルダの COBOL ファイル
3. [プロジェクト内のすべてのソースファイル] を選択すると、プロジェクトに属するソースファイルすべてを移行対象として、データレコード情報ファイルが生成されます。
4. [プロジェクトフォルダの COBOL ファイル] を選択すると、プロジェクトフォルダにある COBOL ファイル(拡張子 .cbl、.cob、.cbx、.cbf)を対象として、データレコード情報ファイルが生成されます。
5. [出力] ウィンドウの [出力] タブに処理結果が表示されます。  
エラーが発生した場合は、「COBOL プログラミングの手引」にしたがって対処を行ってください。

6. 処理が正常終了すると、データレコード情報ファイルはプロジェクトの Release フォルダーに作成されます。  
ファイル名は、COBOL ファイル名の拡張子を ".csv" に変更したのになります。

【Linux】

サーバー上では、COBOL プロジェクト配下の datarecwork ディレクトリに作成され、クライアントのプロジェクトの Release フォルダーに転送されます。

---

## 8 付録

付録 A 用語解説

付録 B ビルドサーバーの設定

---

## 付録A 用語解説

(英字)

COBOL GUI View Generator

COBOL 言語による GUI アプリケーションの開発ツール。

CSV ファイル

CSV (Comma Separated Values) 形式でデータが記述されたファイルのこと。CSV 形式とは、表計算プログラムやリレーショナルデータベースでデータを扱うテキストデータの形式をいう。データの区切りをコンマ(、)、レコードの区切りを改行で表す。レコードは可変長形式になる。コンマで区切られた個々のデータは、セルと呼ばれる。

Visual Studio

Microsoft 社の製品で、原始プログラムの編集に適したテキストエディターやデバッグ機能、その他を備えた、アプリケーションを開発するための統合開発環境。プログラミング言語に依存しないでアプリケーションを作成できる。

(ア行)

オブジェクト

データとそのデータを操作するメソッド(手続き)を一体化させたモジュール。オブジェクトは、ある特定の仕事を受け持ち、所定のメッセージに対してメソッドを動作させることで問題を処理する。オブジェクト指向のプログラムでは、オブジェクト同士がメッセージをやり取りすることによって、問題が処理される。

(カ行)

画面節移行ファイル

COBOL 言語の画面制御機能を、COBOL GUI View Generator を使用して GUI に移行するための情報ファイル。

環境変数

実行可能ファイルおよびその実行環境へのオプションを変数として設定しておくもの。



## コンパイル

原始プログラムを翻訳すること。

COBOL 開発環境のコンパイルには、次の 2 つの方法がある。

ビルド

リビルド

(サ行)

## ソリューション

複数のプロジェクトを管理するための概念。ソリューションにはプロジェクト間の親子関係や、すべてのプロジェクトに共通する設定を定義する。

## ソリューションファイル

ソリューションの情報を保管するファイル。

(タ行)

## 登録集原文

コンパイルするプログラムとは別のファイルに登録した部分的なプログラム、または 1 つのプログラム。

COBOL プログラム中でよく利用される標準化した手続き、ファイル記述、レコード記述を別のファイルに登録する。

(ハ行)

## ビルド

プロジェクトに登録されたソースファイルや定義ファイルから最終生成ファイル (実行可能ファイル、ライブラリなど) を生成する方法。ビルドの対象となるファイルは、変更のあったファイルだけである。

## ファイルビュー

開発環境のウィンドウの 1 つ。ファイルビューには、ソリューションに登録されているプロジェクト、ファイルの情報がツリー形式で表示される。

## プロジェクト

実行可能ファイルやライブラリを作成するための資源を一括して管理するための概念。1 つのプロジェクトは、1 つの実行可能ファイル、またはライブラリに対応している。

## プロジェクトファイル

ソリューションに含まれるプロジェクトの情報を出力したファイル。

(ラ行)

## リビルド

プロジェクトに登録されたソースファイルや定義ファイルから最終生成ファイル(実行可能ファイル、ライブラリなど)を生成する方法。リビルドの対象となるファイルは、プロジェクトに登録されたすべてのファイルである。

---

## 付録B 【Linux】ビルドサーバーの設定

### (1) 通信設定

root でデーモン起動スクリプトを編集します。

デーモン起動スクリプトのファイル名および格納場所は、以下のとおりです。

ファイル名	neccobold
格納場所	/etc/init.d

neccobold に定義している環境変数によって設定を行います。

```
export NECCOBOL_MAXCONNECTNUM=nn
export NECCOBOL_PORT=nnnnnn
export NECCOBOL_TIMEOUT=nn
```

nn は数値を表します。

各設定項目を以下に説明します。

#### (a) NECCOBOL\_MAXCONNECTNUM

クライアントから同時にログインできる接続数の上限を指定します。

1 から 32 まで値を指定可能です。既定の値は 10 です。

0 以下の値を指定すると、既定の値である 10 が指定されたと見なされます。

32 より大きい値を指定すると、最大値の 32 が指定されたと見なされます。

注意：ログインした状態のまま、クライアントが終了した場合でも、接続中とみなされるため、見た目の接続数と実際の接続数が異なる場合があります。

このような場合の接続は、終了したクライアントが再接続を行ってログアウトを実行するか、タイムアウトが発生するかで、切断されます。

#### (b) NECCOBOL\_PORT

クライアント開発環境と接続するときのポート番号を指定します。

一般的には 49152 から 65535 までの値を指定可能です。既定の値は 52000 です。

負の値を指定すると、既定の値である 52000 が指定されたと見なされます。

#### (c) NECCOBOL\_TIMEOUT

クライアントとの接続のタイムアウト時間を秒単位で指定します。

0 から  $2^{31}-1$  までの値を指定可能です。既定の値は 0 です。0 はタイムアウトが発生しないことを表します。

負の値を指定すると、既定の値である 0 が指定されたと見なされます。

## (2) Pro\*COBOL を使用するときの設定

Pro\*COBOL のソースファイルをコンパイルするには、デーモンスクリプトの編集が必要です。デーモンスクリプトファイル名および格納場所は、以下のとおりです。

デーモンスクリプトファイル名	neccobold
格納場所	/etc/init.d

以下の環境変数を設定します。

環境変数名	説 明
ORACLE_HOME	ORACLE をインストールしたディレクトリ
LD_LIBRARY_PATH	ORACLE のライブラリのディレクトリ
PROMAKEFILE	Pro*COBOL のソースファイル用の makefile へのパス
PROCOB	Pro*COBOL プリプロセッサへのパス

環境変数の設定の記述例を以下に示します。

```
export ORACLE_HOME=/app/oracle/product/11.2.0/client_1
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
export PROMAKEFILE=$ORACLE_HOME/precomp/demo/procob2/demo_procob.mk
export PROCOB=$ORACLE_HOME/bin/procob
```

また、環境変数 PROMAKEFILE に設定する Pro\*COBOL のソースファイル用の makefile に、相対パスや COBOL コンパイラでサポートしないコンパイルオプションが指定されている場合は、絶対パスへの変更やサポート外のオプション文字列の削除が必要です。

## 注意事項

・ビルドサーバーの設定を変更した場合は、設定ファイルの再読み込みとデーモンの再起動が必要です。

COBOL 開発環境

利用の手引

2017 年 4 月 初 版

2025 年 6 月 第 8 版

日本電気株式会社

東京都港区芝五丁目 7 番 1 号

TEL (03) 3454-1111 (大代表)

(C)NEC Corporation 2017-2025

日本電気株式会社の許可なく複製・改変などを行うことはできません。

本書の内容に関しては将来予告なしに変更することがあります。