

FICO™ Xpress Optimization Suite

# 評価者のためのガイド

最終更新 2009年5月26日

# FICO<sup>TM</sup> Xpress Optimization Suite

## 評価者のためのガイド

2009年5月26日

### はじめに

FICO<sup>TM</sup> Xpress へようこそ!!このガイドでは Xpress 評価の枠組みをご案内します。  
このガイドを使って、次のことが可能となります。

1. Xpress のインストールが正常であることの確認
2. 評価する Xpress 製品の決定
3. Xpress の評価 .

### 1 Xpress のインストールが正常であることの確認

Xpress-IVE の起動 (Windows 版のみ) またはコンソールコマンドの実行 (全 OS) により、Xpress のインストールが正常であることをテストできます。

1. Xpress-IVE の起動 : Xpress-IVE は、Windows 用の統合ビジュアル開発環境です。  
Xpress-IVE を実行するには、デスクトップ上のアイコンをクリックするか、[スタート]メニュー → [プログラム] → [Xpress-IVE] を選択します。また DOS ウィンドウで「ive」と入力するか、Xpress-Mosel モデルファイル (拡張子が .mos のファイル) をダブルクリックすることでも Xpress-IVE を起動させることができます。

2. コンソールコマンドの実行 : コマンドプロンプト 1 にて、以下の一連のコマンドを入力します。

```
optimizer  
(Enter)  
quit
```

次のような出力が表示されます。

```
C:\> optimizer  
Xpress-Optimizer v19.00.0
```

Hyper capacity, MIP (20 threads), Barrier (20 threads), Network, QP/MIQP  
 (c) Copyright Fair Isaac Corporation 2008

```
Using Xpress-Optimizer [C:\XpressMPnbinxprs.dll]
Enter problem name >
[xpress C:\] quit
```

1 Windows でコマンドプロンプトを表示させるには、[スタート]メニュー→[プログラム]→[アクセサリ]→[コマンド プロンプト] を選択します。

## 2 評価する Xpress 製品の決定

この判断をするためには、解決したい問題のタイプ、モデル生成に使用したいツール、使用予定のプラットフォームを考慮する必要があります。

### 2.1 問題のタイプ

Xpress は様々なタイプの問題のためのソルバーエンジンを備えています。

問題のタイプ	ソルバーエンジン
線形計画 (LP)	Xpress-Optimizer
混合整数計画 (MIP)	
二次計画 (QP)	
混合整数二次計画 (MIQP)	
二次制約・二次計画 (QCQP)	
線形制約凸最適化 (LCCO)	
非線形計画 (NLP)	Xpress-SLP
混合整数非線形計画 (MINLP)	
確率計画 (SP)	Xpress-SP
制約計画 (CP)	Xpress-Kalis

### 補足情報

次の Xpress のドキュメンテーション (Xpress インストール先の docs サブディレクトリ、または <http://research-rules.fico.com/>) を参照。

- ・数理計画問題の分類については『Xpress 入門 (Getting Started with Xpress)』1.1「数理計画」
- ・Mosel によるモデリング、特に LP/MIP については、『Xpress-Mosel ユーザーガイド』
- ・数理計画問題の定式化については、書籍『Applications of Optimization with Xpress-MP (Xpress-MP による最適化の適用)』Part I「Developing Linear and Integer Programming models (線形および整数計画モデルの作成)」を参照
- ・確率計画については、『確率計画の利用分野に関する概要 (Overview of Stochastic Programming Applications)』ホワイトペーパー
- ・制約計画については、『Xpress-Kalis ユーザーガイド』

## 2.2 モデル開発ツールおよび仕様

Xpress はモデル開発仕様として下記のツールを提供しています。ではモデル開発ツール。

- ・Xpress-Mosel/Xpress-IVE：Mosel は、先進的なモデリング・プログラミング言語。Xpress-IVE は、Mosel モデル開発用の統合ビジュアル環境
- ・Xpress-BCL：サポート対象言語からの呼び出しが可能なオブジェクト指向ライブラリ
- ・Xpress-Optimizer ライブラリ：Optimizer サポート対象のプログラミング言語で開発されたアプリケーションとの低レベルでの統合を行うために関数およびプロシージャを提供しています。

また、モデルは MPS または LP 形式の行列で記述できます。

### 補足情報

『Xpress によるモデリング』ホワイトペーパーを参照 (<http://research-rules.fico.com/> にて入手可能)。

## 2.3 モデル展開ツール

モデル展開は、モデルの詳述に使用するツールに大きく左右されます。モデル展開は、Xpress ライブラリの使用により行われます。

モデルの記述に使用	展開に使用
Mosel	Mosel ライブラリ (C++/C、Java、VB、.NET)
BCL	BCL ライブラリ (C、C++、Java、VB、.NET)
Optimizer ライブラリ	Optimizer ライブラリ (C++/C、Java、VB、.NET)

確率計画 (Xpress-SP) および制約計画 (Xpress-Kalis) の場合、モデルは Xpress-Mosel を使用して記述しなければならないので注意してください。また、Java、.NET で展開した Xpress-SLP の場合は、Xpress サポート部門にお問い合わせください。.NET で展開した Xpress-BCL についても、Xpress サポート部門までお問い合わせください。

## 補足情報

次の Xpress のドキュメンテーション (Xpress インストール先の docs サブディレクトリ、または <http://research-rules.fico.com/>) を参照。

- ・導入の例 (Mosel/BCL/Optimizer) については、『Xpress 入門 (Getting Started with Xpress)』
- ・Mosel ライブラリの例については、『Xpress-Mosel ユーザーガイド』 Part III 「Mosel ライブラリの利用」
- ・アプリケーション一式が揃った事例 (Mosel) については、『最適化アルゴリズムの組み込み (Embedding optimization algorithms)』 ホワイトペーパー (<http://research-rules.fico.com/> にて入手可能)
- ・Mosel ライブラリのドキュメンテーションについては、『Mosel ライブラリ・リファレンスマニュアル』
- ・BCL については、『BCL リファレンス・マニュアル』
- ・Optimizer については、『Optimizer リファレンスマニュアル』

## 2.4 プラットフォーム

Xpress-IVE は、マイクロソフト Windows プラットフォームでのみ利用可能です。Xpress-Kalis は、Windows、Linux、Solaris の 32-bit 版でのみ利用可能です。その他の製品については、いずれもサポート対象のプラットフォームすべてでご利用いただけます。

## 3 Xpress の評価

以下の 4 つの代表的な評価シナリオは、評価する製品の選択ごとに定められています (質問 2

を参照)。

- **シナリオ 1**：任意のタイプの問題を Mosel でモデル化し、任意のプログラミング言語で展開

モデル開発ツール： Mosel  
 問題のタイプ： 任意  
 モデル展開ツール： Mosel ライブラリ (C++/C、Java、VB、.NET)

- **シナリオ 2**：Xpress-Optimizer を必要とする問題を BCL でモデル化し、任意のプログラミング言語で展開

モデル開発ツール： BCL  
 問題のタイプ： LP、MIP、QP、MIQP  
 モデル展開ツール： BCL ライブラリ (C、C++、Java、VB、.NET)

- **シナリオ 3**：カスタムアプリケーションを用いてモデル作成し、Xpress-Optimizer ライブラリを呼び出し。Xpress-Optimizer で求解可能な問題に適用可能

モデル開発ツール： Optimizer ライブラリを呼び出すカスタムアプリケーション  
 問題のタイプ： LP、MIP、QP、MIQP、NLP、MNLP  
 モデル展開ツール： Optimizer ライブラリ (C++/C、Java、VB、.NET)

- **シナリオ 4**：LP または MPS 形式で既に利用できる行列を実行。このようなモデルは、Xpress-IVE、コンソールコマンド、または Xpress-Optimizer ライブラリ関数を呼び出すアプリケーションにより実行可能である

シナリオごとに、それぞれ一連の評価ステップが定められています。

### 3.1 シナリオ 1 の評価ステップ

モデル開発ツール： Mosel  
 問題のタイプ： 任意  
 モデル展開ツール： Mosel ライブラリ (C++/C、Java、VB、.NET)

このシナリオでは、\XpressMP\examples\getting\_started\Mosel ディレクトリにある『Xpress 入門 (Getting Started with Xpress)』ドキュメントに記載されている例を使用します。

### 3.1.1 Xpress-IVE の起動

Xpress-IVE は、Windows 用の統合ビジュアル開発環境です。Xpress-IVE を実行するには、デスクトップ上の Xpress-IVE アイコンをダブルクリックするか、[スタート]メニュー→[プログラム]→[Xpress]→[Xpress\_IVE] と選択します。そのほか、IVE は DOS ウィンドウで `ive` と入力するか、モデルファイル（拡張子が `.mos` のファイル）をダブルクリックすることでも起動できます。

### 3.1.2 Xpress-IVE で Mosel モデルを開く

Mosel モデルファイルを開くには、[ファイル]→[開く]を選択します。評価用のサンプルを含むディレクトリ (Examples\Getting Started\Mosel) に進み、`foliolp.mos` ファイルを開きます。「LP によるポートフォリオ最適化」モデルが、中央のペイン (IVE エディタ) に開きます。このモデルでは、10 種類の証券 (株式) を使い、いくつかのリスク面および地理的な制約を条件に入れて最適な投資ポートフォリオを求めます。コード内のコメント (感嘆符 (!) から後ろのテキスト) で、記述の意味を説明しています。RET は、期待される株式のリターンを表す実数値の配列ですので注意してください。モデル内の決定変数は、`mpvar` タイプの配列 `frac` で与えられます。maximize プロシージャは、Xpress-Optimizer を呼び、目的関数を最大化します。このコードには、最適解および解の値をプリントするための記述も入っています。

### 補足情報

- この例でのモデルの定式化に関する詳細については、『Xpress 入門 (Getting Started with Xpress)』Chapter 2 「モデル構築 (Building models)」を参照
- この例でのモデルの Mosel 上での表現に関する詳細は、『Xpress 入門 (Getting Started with Xpress)』3.2 「LP モデル」を参照
- Xpress-IVE エディタに関する詳細は、Xpress-IVE のヘルプを参照 ([ヘルプ]→[Xpress-IVE ヘルプ]→[エディタ]を選択)
- Mosel および Mosel の事例に関する詳細は、『Xpress-Mosel ユーザーガイド』Chapter 1 「Mosel 入門 (Getting started with Mosel)」1.1 「モデルの入力 (Entering a model)」を参照

### 3.1.3 Mosel モデルのコンパイルと実行

Mosel モデルをコンパイルするには、[ビルド]→[コンパイル]を選択します。コンパイルの状態は、Xpress-IVE の下部 (情報バーのビルド画面) に表示されます。「`foliolp.mos` は正常にコンパイル

ルされました」と表示されるはずですが。コンパイルが正常終了したら、左側のプロジェクトバーにエンティティツリーが表示されます。ここには Mosel モデル内のすべての識別子が含まれます。コンパイルが正常終了すると、Mosel のコンパイル済みファイルである foliolp.bim も作成されます。

### 補足情報

- ・コンパイルおよびコンパイル時のエラーに関する詳細については、『Xpress 入門 (Getting Started with Xpress)』3.3 「エラーの修正とモデルのデバッグ (Correcting errors and debugging a model)」を参照
- ・Xpress-IVE の詳細については、Xpress-IVE ヘルプ (プロジェクトバー、情報バー) を参照。

Mosel モデルを実行するには、[ビルド]→[実行]を選択します。モデルが事前にコンパイルされていない場合は、モデルの実行によりコンパイルも行われます。実行が正常に終了すると、右側のウィンドウ (実行バー) にて、Mosel コードおよび Xpress-Optimizer からの出力結果が各タブ上に表示されます。具体的に述べると、[出力/入力]タブでは、Mosel モデルで解の印刷用の記述により印字された出力を表示し、[統計および行列]タブでは Xpress-Optimizer からの出力結果を表示します。また、エンティティツリーの識別子の値は、問題の求解後に、[ヒント]または[ダイアログの表示] (識別子のクリック時) 上で表示されます。

文字列インデックス：モデルデータおよび解は、文字列インデックスを使用するとより簡単に理解できます。モデルファイルの foliolps.mos を開いて実行します。[出力]タブおよびエンティティツリーを照会してください。

### 補足情報

- ・このサンプルでの Mosel モデルの実行に関する詳細については、『Xpress 入門 (Getting Started with Xpress)』3.4 「求解、最適化表示ならびに解の表示 (Solving, optimization displays and viewing the solution)」を参照
- ・Xpress-IVE に関する詳細は、Xpress-IVE の [ヘルプ]→[実行バーのタブ/ペイン] を参照
- ・この事例での文字列インデックスの詳細については、『Xpress 入門 (Getting Started with Xpress)』3.4.1 「文字列インデックス」および『Xpress-Mosel ユーザーガイド』2.1.3 「burglar 問題の再考」を参照

#### 3.1.4 Mosel 内のデータの扱い

Mosel モデルでは、シンプルなテキストファイルや、データベースなどのファイル形式から、



メモリ内で Mosel モデルとホストアプリケーション間、または同時にいくつかの Mosel モデル同士で交換されたデータに至るまで、多種多様なデータソースを扱えます。ここでは、よくあるケース、すなわちテキストファイルと ODBC によるデータベースアクセスについて記述します。

## テキストファイル

モデル foliodata.mos を開いて実行します。このモデルには parameters ブロックと、initializations によるブロックがあることに注意してください。parameters にはデータ入力ファイル、出力ファイル、その他のモデルの定数値が含まれています。parameters ブロックは実行時にリセットできるので、モデルがビジネスアプリケーション内で展開される場合に、特に重要なものとなります。initializations によるブロックは、folio.dat ファイルからデータを読み込みます。このファイルは、Xpress-IVE で開く ([ファイル]→[開く]→[ファイルの種類: データファイル (\*.dat)]) ことで参照可能な、Mosel 特有の形式です。モデル内のインデックスセットとデータ配列は、入力ファイル内のデータに基づいて動的に生成されます。最後にこのモデルは、fopen および fclose プロシージャを呼ぶことにより、解の出力を外部のフリーフォーマットのファイル result.dat へ行うように指示します。

## 補足情報

データの扱いおよび Mosel のパラメータの使用に関する詳細については、『Xpress 入門 (Getting Started with Xpress)』Chapter 4 「データの扱い」を参照。パラメータ、データ入力、テキスト出力については、Xpress-IVE ウィザードも参照。

## スプレッドシートおよびデータベース

モデル folioexcel.mos を開いて実行します。このプログラムの initializations f によるブロックは、MS Excel スプレッドシートからデータを読み込みます。モデルには folio.xls のデータが付け加えられます。注：この例ではファイルからデータを受け取るために、Boolean 配列内の集合 RISK および NA を変換しています。initializations to ブロックは、問題の結果をスプレッドシートへ出力します。書き込み時に該当の Excel ファイルが開いていると、出力データはセーブされず、結果を保存するかどうかを選択します。モデルを繰り返し実行すると、対象範囲の以前の出力結果は上書きされます。

2 つめのよく似たモデルである folioodbc.mos は、ODBC データソース（たとえば、MS Access データベース folio.mdb）からデータを読み込み、問題の結果をデータベースに出力します。現在、initializations ブロック内のファイル名のプレフィクスは、取り扱っているデータの種類に

対応し、mmodbc.odbc です。そのほかはすべて Excel モデルと同様です。ODBC データベースアクセス機能は、MS Excel スプレッドシートで展開することもできます。しかし、いくつかの制約事項があるため、folioexcel.mos モデルで示したように、Excel 専用のデータアクセスを使用することを推奨しています。

警告：この例題を実行するためには、対応するデータソース用の ODBC ドライバが必要です。また Windows においては、[ODBC データ ソース アドミニストレータ] (Windows 2000 または XP：[スタート]メニュー→[設定]→[コントロールパネル]→[管理ツール]→[データソース(ODBC)]) の中に「MS Access データベース」という名前でユーザーデータソースが設定されていることを必ずご確認ください。

### 補足情報

- ODBC モジュールの使用については、『Xpress-Mosel ユーザーガイド』Chapter 2「具体的事例」、および『Mosel での ODBC およびその他データベースインタフェースの利用 (Using ODBC and other database interfaces with Mosel)』ホワイトペーパーを参照
- ODBC モジュールのドキュメンテーションについては、『Mosel 言語リファレンスマニュアル』Chapter 8「mmodbc」
- その他外部ソースとのデータ交換の概略については、『Xpress-Mosel ユーザーガイド』16.1「一般的なファイル操作」
- 外部データソースとの高度な通信方式の例については、『Mosel での一般的なファイル操作 (Generalized file handling in Mosel)』ホワイトペーパーを参照

### 3.1.5 Mosel MIP と二次モデル

以下の Mosel モデルを開き、実行します。

foliomip1.mos：このモデルはバイナリ変数の配列 buy を使い、ポートフォリオに組み入れる株式の数を限定する制約を課します。

foliomip2.mos：このモデルは、配列変数 frac を半連続に再定義します。したがって、少なくとも最小限の額の予算が各株式の購入に投資されます。。

folioqp.mos：このモデルでは二次定式化を使用し、目標とする期待リターンの実現を制約としてポートフォリオの分散を最小化します。Mosel プログラムは問題の求解を 2 度行い、2 回目

の実行時にポートフォリオに組み入れる株式数の上限を課します。

各モデルについては、実行バーのタブに表示される解および情報を参照してください。

## 補足情報

- ・『Xpress 入門 (Getting Started with Xpress)』 Chapter 6 「混合整数計画」 および Chapter 7 「二次計画」
- ・MIP 変数の全種類のリストについては、『Xpress-Mosel ユーザーガイド』 Chapter 4 「整数計画」

### 3.1.6 その他の問題：制約計画、非線形および確率モデル

これまで見てきたモデルはすべて、問題の求解に Xpress-Optimizer を使用します (モデルの先頭の `uses "mmxprs"` ステートメントで選択)。別のソルバーを使用したい場合は、対応するソルバーのモジュール名を指定する必要があります。

`assign.mos` モデルを開く：このモデルは、Xpress-Kalis で、制約計画 (CP) の手法を用いて割当問題を実装し、求解を行います。問題は、労働者および機械の配置対して、1 人の労働者をすべての機械に正確に割り当て、全体の生産性を最大化させることです。労働者 1 人あたりの生産性は、割り当てられた機械に依存します。

これまで見てきた問題と、いくつかの違いが見られます。

- ・ここではソルバー選択のステートメントは、`uses "kalis"` となっています。
- ・CP 決定変数は、`cpvar` タイプであり、ドメイン (=許容値) は `setdomain` プロシージャで設定できます。
- ・CP モデルは (「Total productivity (全体の生産性)」の制約として) 線形制約を有することがあります。しかしこのモデルでは、他のタイプの制約関係、いわゆる「グローバル制約」も使用します。制約条件 `element` は、1 つの変数で離散関数を定式化し、`all_different` リレーションで制約条件内のすべての変数が異なる値をとらなければならないと規定します。
- ・CP 問題は、木探索法により求解されます。デフォルトの探索戦略は使用せずに、通常はより問題に特化した戦略を選ぶ (`cp_set_branching` プロシージャを使用する) 方が好ましいです。
- ・`cp_maximize` 関数は、最適化を行うために使用されています。

その他 (一般構造、宣言、データへのアクセス、出力の印刷) は、すべてこれまで見てきたものと同様です。

IVE エンティティツリー表示でこのモデルを実行すると、追加された、モデルの出力結果がワークスペースの右側にある [出力/入力] ペインに表示されます。モデル実行についての詳細情報は、[CP 統計] (問題の統計情報の概要) および [CP 探索] (CP 探索木のグラフィカル表示) タブの選択により見ることができます。

Mosel で使用可能なその他のソルバータイプには、非線形計画問題を解決する Xpress-SLP、確率最適化問題向けの Xpress-SP があります。これらのソルバーモジュールは、それぞれ問題のタイプ専用の機能 (変数や制約のタイプなど) を備えています。対応のマニュアルを参照してください。また、一部の IVE 表示は、Mosel モデルで使用するソルバーに適応しています。

### 補足情報

- Xpress-Kalis は、Mosel 環境から制約計画ソルバー Kalis の機能へのアクセスを提供します。Xpress-Kalis に関する詳細は、『Xpress-Kalis ユーザーガイド』『Xpress-Kalis リファレンスマニュアル』ドキュメントを参照してください。
- Xpress-SP は、確率計画問題を定式化するための Mosel 言語の拡張版です。Xpress-SP に関する詳細は、『Xpress 確率計画ガイド』ドキュメントを参照してください。
- Xpress-SLP は、逐次線形近似の技法を基盤とした、非線形および混合整数非線形計画向けのソルバーです。Xpress-SLP に関する詳細は、『Xpress-Mosel SLP リファレンスマニュアル』ドキュメントを参照してください。

### 3.1.7 Mosel モデルの展開

Mosel モデルを開き、[Deploy]→[Deploy] を選択するか、[Deploy] ボタンをクリックします。この操作により、[Deploy] ダイアログボックスが開きます。Mosel モデルを実行させたいプログラミング言語を選択し、[次へ] ボタンをクリックします。この操作により、展開用のコードを含んだソースコードのダイアログが開きます。

### 補足情報

- Xpress-IVE のヘルプ ([ヘルプ]→[Xpress-IVE ヘルプ]→[ダイアログ]→[Deploy ダイアログ]) を選択してください。
- 入門的な例題 (VB) については、『Xpress 入門 (Getting Started with Xpress)』Chapter 10 「Mosel モデルのアプリケーションへの組み込み」
- その他の例 (C/Java/VB) については、『Xpress-Mosel ユーザーガイド』Part III 「Mosel ライブラリの利用」

- Mosel C ライブラリのドキュメンテーションは、『Xpress-Mosel ライブラリリファレンスマニュアル』
- Mosel Java ライブラリのドキュメンテーションは、『Xpress-Mosel ライブラリリファレンスマニュアル JavaDoc』
- Mosel .NET インタフェースのドキュメンテーションは、『Xpress-Mosel .NET インタフェース』

### 3.1.8 行列ファイルのエクスポート

Mosel モデルの実行後、LP または MPS によりモデルを表現したファイルを生成するため [ビルド] → [行列ファイルのエクスポート] を選択します。

#### 補足情報

- Xpress-IVE ヘルプ ([ヘルプ] → [Xpress-IVE ヘルプ] → [ダイアログ] → [行列ダイアログへエクスポート]) を選択)
- 『Xpress 入門 (Getting Started with Xpress)』 9.4 「行列ファイル」
- Xpress-IVE ヘルプ ([ヘルプ] → [Xpress-IVE ヘルプ] → [ダイアログ] → [Optimizer ダイアログ]) を選択)

### 3.1.9 Mosel コンソールコマンド

Xpress-IVE 内部で Mosel モデルを実行させる別の方法として、モデルを Mosel スタンド・アロンバージョンで実行させることができます。例えば、テストや実験目的の場合、バッチファイルから一連のモデルを実行させたい場合は、このモードの方が望ましいことが多いです。

コマンドプロンプトで、次の一連のコマンドを入力します。

```
mosel
compile foliolp
load foliolp
run
quit
```

以下のような出力結果が得られます。

```
mosel
** Xpress-Mosel **
```

(c) Copyright Fair Isaac Corporation 2008

```
>compile foliopl
```

```
Compiling 'foliopl' ...
```

```
>load foliopl
```

```
>run
```

```
Total return: 14.0667
```

```
treasury: 30%
```

```
hardware: 0%
```

```
theater: 20%
```

```
telecom: 0%
```

```
brewery: 6.66667%
```

```
highways: 30%
```

```
cars: 0%
```

```
bank: 0%
```

```
software: 13.3333%
```

```
electronics: 0%
```

```
Returned value: 0
```

```
>quit
```

```
Exiting.
```

上記の一連のコマンドは、次のように 1 行に集約させることができます。

```
mosel -c "exec foliopl"
```

## 補足情報

『Xpress-Mosel ユーザーガイド』 1.1 「モデルの入力」を参照。あわせて、『Xpress-Mosel リファレンスマニュアル』 1.1 「Mosel とは」 — 「Mosel の実行」も参照。

## 評価ガイド

### 3.1.10 その他 Mosel に関して

このガイドで述べてきた内容のほか、『Xpress 入門 (Getting Started with Xpress)』ドキュメントでは Xpress-IVE でのユーザーグラフの描画方法 (Chapter 5 「ユーザーグラフの描画」) および Mosel によるヒューリスティクスのプログラム方法 (Chapter 8 「ヒューリスティクス」) に



ついて記述しています。これらのトピックの詳細については、『Xpress-Mosel ユーザーガイド』 Part II 「高度な言語機能」にあります。そこでは Mosel のプログラミング機能について記載されているほか、他のヒューリスティクスの例題があります。また、Part IV 「拡張機能とツール」 — 16.3 「mmive および mmxad によるグラフィクス」にも詳細があります。

Xpress Application Developer (XAD) は、標準的な GUI 生成用の関数およびプロシージャにより、Mosel 機能を拡張します。XAD に関する詳細を知るには、『Xpress Application Development リファレンスマニュアル』ドキュメントを参照してください。

Mosel によるモデリングおよびプログラミングのその他の事例は、\XpressMP\examples\ ディレクトリ内にあります。また、書籍『Applications of Optimization with Xpress-MP (Xpress-MP による最適化の適用)』(Dash Optimization, 2002) では、Xpress による多数の応用問題の定式化と求解の方法について記載しています。http://research-rules.fico.com/ を参照ください。

### 3.2 シナリオ 2 の評価ステップ

Xpress-Optimizer を必要とする問題を BCL でモデル化し、任意のプログラミング言語で展開します。

モデル作成ツール： BCL  
問題のタイプ： LP、MIP、QP、MIQP  
モデル展開ツール： BCL ライブラリ (C、C++、Java、VB、.NET)

『Xpress 入門 (Getting Started with Xpress)』 Part II 「BCL 入門」を参照ください。C++ での例を詳細な説明付きで記載しています。これらの事例は、C、Java、VB でも実装できます。対応する以下のディレクトリにもあります。

\XpressMP\examples\bcl\\*\UGExpl

#### 補足情報

その他の例や BCL のドキュメンテーションすべてについては、『Xpress-BCL リファレンスマニュアル』および『Xpress-BCL Javadoc』にあります。

### 3.3 シナリオ 3 の評価ステップ

カスタムアプリケーションを用いてモデル生成し、Xpress-Optimizer ライブラリを呼び出します。

Xpress-Optimizer で求解可能な問題に適用できます。

モデル作成ツール：	Optimizer ライブラリを呼び出すカスタムアプリケーション
問題のタイプ：	LP、MIP、QP、MIQP、NLP、MNLP
モデル展開ツール：	Optimizer ライブラリ (C++/C、Java、VB、.NET)

『Xpress 入門 (Getting Started with Xpress)』 Part III 「Optimizer 入門」を参照。C 言語での例が詳細な説明付きで記載されています。

### 補足情報

- ・『Xpress-Optimizer リファレンスマニュアル』
- ・非線形計画 (NLP、MNLP) については、『Xpress-SLP プログラムリファレンスマニュアル』を参照ください。

## 3.4 シナリオ 4 の評価ステップ

LP または MPS 形式で迅速に利用できる行列を実行します。こうしたモデルは Xpress-IVE、コンソールコマンド、または Xpress-Optimizer ライブラリ関数を呼び出すアプリケーションにより実行できます。

### 3.4.1 Xpress-IVE

[ファイル]→[開く]を選択し、行列ファイル (通常は拡張子が .mat、.mps、.lp のファイル) を開きます。[ビルド]→[行列ファイルの最適化]を選択し、問題を実行します。

### 補足情報

- ・Xpress-IVE ヘルプ ([ヘルプ]→[Xpress-IVE ヘルプ]→[ダイアログ]→[Optimizer ダイアログ])
- ・『Xpress 入門 (Getting Started with Xpress)』 9.4 「行列ファイル」

### 3.4.2 コンソールコマンド

コマンドプロンプトで、以下の一連のコマンドを入力し、foliolp.mps ファイル内の MPS 行列を実行させます。

```
optimizer  
foliolp
```



```
readprob
maxim
printsol
quit
```

次のような出力結果が得られます。

```
>optimizer
Xpress-Optimizer v19.00.00
Hyper capacity, MIP (20 threads), Barrier (20 threads), Network, QP/MIQP
(c) Copyright Fair Isaac Corporation 2008
Using Xpress-Optimizer [C:\XpressMPnbinxprs.dll]
Enter problem name > foliolp
[xpress C:\] readprob

Reading Problem moselP

Problem Statistics
4 ( 0 spare) rows
10 ( 0 spare) structural columns
29 ( 0 spare) non-zero elements
Global Statistics
0 entities    0 sets  0 set members
[xpress C:\] maxim
Presolved problem has:   3 rows      10 cols      19 non-zeros

Its   Obj Value   S   Ninf  Nneg  Sum Inf   Time
0    42.600000   D    2    0    3.166667   0
5    14.066659   D    0    0    .000000   0
Uncrunching matrix
5    14.066659   D    0    0    .000000   0
Optimal solution found
[xpress C:\] printsol
```

Problem Statistics

Matrix moselP

Objective \*OBJ\*

RHS \*RHS\*

Problem has 4 rows and 10 structural columns

Solution Statistics

Maximization performed

Optimal solution found after 5 iterations

Objective function value is 14.066659

type c/r to continue, anything else to finish >

Rows Section

Number	Row	At	Value	Slack Value	Dual Value	RHS
N 1	*OBJ*	BS	14.066659	-14.066659	.000000	.000000
E 2	_R1	EQ	1.000000	.000000	8.000000	1.000000
G 3	_R2	LL	.500000	.000000	-5.000000	.500000
L 4	_R3	UL	.333333	.000000	23.000000	.333333

ype c/r to continue, anything else to finish >

Columns Section

Number	Column	At	Value	Input Cost	Reduced Cost
C 5	frac(1)	UL	.300000	5.000000	2.000000
C 6	frac(2)	LL	.000000	17.000000	-9.000000
C 7	frac(3)	BS	.200000	26.000000	.000000
C 8	frac(4)	LL	.000000	12.000000	-14.000000
C 9	frac(5)	BS	.066667	8.000000	.000000
C 10	frac(6)	UL	.300000	9.000000	1.000000
C 11	frac(7)	LL	.000000	7.000000	-1.000000
C 12	frac(8)	LL	.000000	6.000000	-2.000000
C 13	frac(9)	BS	.133333	31.000000	.000000

```
C    14    frac(10)    LL    .000000    21.000000    -10.000000
[xpress C:\] quit
```

補足情報

『Xpress-Optimizer リファレンスマニュアル』 Chapter 6 「コンソール・ライブラリ関数」

### 3.4.3 Xpress-Optimizer ライブラリ関数

『Xpress 入門 (Getting Started with Xpress)』 Chapter 14 「行列の入力」を参照。

C 言語での例が詳細な説明付きで記載されています。