



# IBM ILOG Views V5.3

## 製品の概要

2009年6月

© Copyright International Business Machines Corporation 1987, 2009.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.



## 著作権の告知

©Copyright International Business Machines Corporation 1987, 2009.

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

### 商標

IBM、IBM ロゴ、ibm.com、Websphere、ILOG、ILOG のデザイン、および CPLEX は、世界中の多くの国の管轄権で登録されている International Business Machines Corp. の商標または登録商標です。その他の製品およびサービス名は、IBM またはその他の企業の商標です。IBM 社の現在の商標一覧は、<http://www.ibm.com/legal/copytrade.shtml> にある Copyright and trademark information (著作権と商標についての情報) にあります。

Adobe、Adobe のロゴ、PostScript、および PostScript のロゴは、米国およびその他の国における Adobe Systems Incorporated の商標または登録商標です。

Linux は、米国およびその他の国における Linus Torvalds の登録商標です。

Microsoft、Windows、Windows NT、および Windows のロゴは、米国およびその他の国における Microsoft Corporation の商標です。

Java およびすべての Java に基づいた商標とロゴは、米国およびその他の国の Sun Microsystems, Inc. の商標です。

その他の企業、製品およびサービス名は、その他の企業の商標またはサービス商標です。

### 告知

詳細は、インストールした製品の <installdir>/license/notices.txt を参照してください。

## 目次

前書き	本書について .....	8
	前提事項.....	8
	表記法 .....	8
<b>IBM ソフトウェア・サポート・ハンドブック</b> .....		<b>10</b>
	ソフトウェア・サポートへのアクセス .....	10
	<b>Web</b> でのお問い合わせ .....	11
	電話でのお問い合わせ.....	12
<b>IBM ILOG Views コンポーネント・スイートへようこそ</b> .....		<b>14</b>
	<b>IBM ILOG Views</b> コンポーネント・スイート .....	15
	内容 .....	16
	IBM ILOG Views Studio .....	16
	ディレクトリ .....	16
	IBM ILOG Views Data Access: バージョン 5.2 との既知の非互換性.....	17
	<b>基本製品</b> .....	<b>17</b>
	IBM ILOG Views Controls .....	17
	IBM ILOG Views 2D Graphics Standard .....	18
	IBM ILOG Views 2D Graphics Professional.....	18
	IBM ILOG Views.....	19
	<b>オプション製品</b> .....	<b>20</b>

<b>IBM ILOG Views</b> コンポーネント・スイートを選択する理由 .....	<b>21</b>
<b>IBM ILOG Views</b> コンポーネント・スイートからメリットを受けるアプリケーション .....	<b>22</b>
地図アプリケーション .....	23
専用エディタ .....	23
制御・処理アプリケーション .....	24
ネットワーク監視アプリケーション .....	25
金融アプリケーション .....	27
交通・運輸アプリケーション .....	28
<b>IBM ILOG Views</b> コンポーネント・スイート・パッケージ .....	<b>30</b>
IBM ILOG Views Foundation .....	30
IBM ILOG Views Studio .....	32
IBM ILOG Views Gadgets .....	35
IBM ILOG Views Application Framework .....	36
IBM ILOG Views Manager .....	39
IBM ILOG Views Grapher .....	41
IBM ILOG Views Prototypes .....	42
IBM ILOG Views Gantt .....	43
IBM ILOG Views Charts .....	45
IBM ILOG Views Maps .....	47
IBM ILOG Views Data Access .....	49
IBM ILOG Views Graph Layout .....	50
<b>IBM ILOG Views</b> のセットアップ .....	<b>51</b>
ディスク・スペースおよびメモリ .....	52
ディレクトリ構造 .....	53
UNIX 上でのセットアップ .....	53
Windows 上でのセットアップ .....	54
<b>実行要件</b> .....	<b>56</b>
<b>コンパイル</b> .....	<b>57</b>
Windows でのコンパイルおよびリンクのオプション .....	58
ライブラリの従属性 .....	60
Motif および共有ライブラリ .....	66

ライブラリ構築情報 (Unix プラットフォームのみ) .....	66
ライセンス .....	67
<b>Unix ユーザの方へ</b> .....	67
<b>Windows Server 2003/XP/Vista ユーザの方へ</b> .....	67
<b>Microsoft Visual C++ ユーザの方へ</b> .....	68
異なるバージョン .....	69
索引 .....	74



## 本書について

本書では、IBM® ILOG® Views Component Suite の概要を説明します。

---

### 前提事項

本書では、特定のウィンドウシステムを含め、ユーザが IBM ILOG Views を使用する Microsoft® Windows® や UNIX® 環境について精通していることが前提となっています。IBM ILOG Views は C++ 開発者用に作成されているため、このマニュアルでは、ユーザが C++ のコードを作成できること、および C++ の開発環境について精通しており、ファイルやディレクトリの操作、テキスト・エディタの使用、C++ プログラムのコンパイルおよび実行ができることも前提となっています。

---

### 表記法

---

#### 書体の規則

以下の書体に関する規則は、このマニュアル全体に適用されます。

- ◆ コードの引用やファイル名は `courier` 書体で記載されます。
- ◆ ユーザが入力する項目は、`courier` 書体で記載されます。



- ◆ 初出の斜体の用語には、ユーザ・マニュアルの用語集で解説されているものがあります。

---

## 命名規則

以下の命名規則は、マニュアル全体を通して API に適用されます。

- ◆ IBM ILOG Views Foundation ライブラリで定義されている型、クラス、関数、マクロの名前は **Ilv** で始まります。
- ◆ クラス名、およびグローバル関数は、最初の文字が大文字で表された連結語として記載されます。

```
class IlvDrawingView;
```

- ◆ 仮想および通常メソッドの名前は小文字で始まります。スタティック・メソッドの名前は大文字で始まります。例：

```
virtual IlvClassInfo* getClassInfo() const;  
static IlvClassInfo* ClassInfo() const;
```

# IBM ソフトウェア・サポート・ハンドブック

このガイドには、IBM 製品に関するサービスおよびサポートを受けるための手順と方法についての重要な情報が含まれています。これは、特定の IBM 製品またはサービス購入時の契約内容や条件に取って代わるものではありません。内容を熟読してください。このサイトにブックマークを付けて、必要に応じて最新情報を参照することができます。IBM のサポートの質を継続的に向上させるため、各ページのナビゲーション・バーの左にあるフィードバック・リンクをクリックして皆様からのフィードバックを気軽にお寄せください。IBM ソフトウェア・サポート・ハンドブックへは、次の URL からアクセスできます。

<http://www14.software.ibm.com/webapp/set2/sas/f/handbook/home.html>

---

## ソフトウェア・サポートへのアクセス

特定のサービス依頼について IBM ソフトウェア・サポートに連絡したり問題を送付する場合は、次の情報が必要となりますので、あらかじめ手元にご用意ください。

IBM お客様番号

マシンのタイプ、モデル、シリアル番号 (サブスクリプションおよびサポートに関して連絡する場合)

御社名

担当者名

希望する連絡方法 ( 電話または電子メール )

電話番号 ( 電話連絡を希望する場合 )

関連製品とバージョン情報

関連 OS およびデータベース情報

問題の詳細説明

問題がお客様のビジネス・ニーズに与える影響の重大度

---

## Web でのお問い合わせ

**製品のサポートサービス依頼**は、IBM 社のあるどの国からでもハードウェアまたはソフトウェアに関する任意の問題のサポートを依頼する適切な場所を検索するためのツールです。これは、サービス依頼をどこに連絡するかが明確でないときの出発点となります。

サービス・リクエスト (SR) ツールは、パスポート・アドバンテージ契約のお客様に、お客様番号を使って PMR (問題報告書) をオープン、編集、および現在オープンまたはクローズされた PMR を追跡できる配布プラットフォームのオンライン問題管理システムを提供します。時間を節約するオプション: あらかじめ入力済みの人口統計フィールドを使用して新規 PMR を作成する、問題を説明し、その重大度を選択する、適切なサポート・キューに PMR を直接提出する、トラブルシューティング・ファイルを直接 PMR に添付する、IBM が PMR を更新するときにアラートを受け取る、オープンおよびクローズされた PMR に関するレポートを表示する。

SR に関するアシスタンス情報は、<http://www.ibm.com/software/support/help-contactus.html> を参照してください。

**システム・サービス・リクエスト (SSR)** ツールは、エレクトロニック・サービス・リクエストと類似しています。System i、System p、System z、TotalStorage 製品、Linux、Windows、Dynix/PTX、Retail、OS/2、Isogon、OS/390 の Candle、および Consul z/OS レガシー製品に対するサポート契約を結んでいるお客様にオンラインでの問題管理システムを提供します。

**IBMLink - SoftwareXcel** サポート契約は、System z プラットフォームをご使用のお客様に System z ソフトウェア製品に関する問題記録をオープンして、使用方法についての質問を行う IBMLink オンライン問題管理ツールを提供します。欠陥または問題記録のオープン、追跡、更新およびクローズ、訂正 / 防止 / 許容保守の注文、既知の問題やテクニカル・サポート情報の検索、該当する問題レポートの追跡、重大度の高い問題およびエラーのフィックスに対するアラートの受信、新規リリースおよび防止保守についての計画情報の表示を行うことができます。

## 電話でのお問い合わせ

現在有効なサービス保守契約をお持ちの場合、またはプログラム・サービスで補償されている場合、カスタマー・サポート・チームに電話で問い合わせることができます。各国の連絡先情報の詳細は、[IBM Directory of worldwide contacts](#) (各国の連絡先一覧) のテクニカル・サポート・セクションを参照してください。



# IBM ILOG Views コンポーネント・スイートへ ようこそ

IBM® ILOG® Views コンポーネント・スイート へようこそ この製品の概要には、以下の内容が記載されています。

- ◆ *IBM ILOG Views* コンポーネント・スイート
  - *内容*
  - *基本製品*
  - *オプション製品*
- ◆ *IBM ILOG Views* コンポーネント・スイートを選択する理由
- ◆ *IBM ILOG Views* コンポーネント・スイートからメリットを受けるアプリケーション
- ◆ *IBM ILOG Views* コンポーネント・スイート・パッケージ
  - *Foundation*
  - *Studio*
  - *Gadget*
  - *Applicaton Framework*
  - *Manager*

- *Grapher*
- *Prototypes*
- *Gantt Chart*
- *Charts*
- *Maps*
- *Data Access*
- *Graph Layout*
- ◆ *IBM ILOG Views* のセットアップについて、次のセクションに分けて説明します。
  - ディスク・スペースおよびメモリ
  - ディレクトリ構造
  - UNIX 上でのセットアップ
  - Windows 上でのセットアップ
  - Windows でのコンパイルおよびリンクのオプション
  - ライブラリの従属性
- ◆ 実行要件
- ◆ コンパイル
- ◆ ライブラリ構築情報 (Unix プラットフォームのみ)
- ◆ ライセンス
- ◆ Unix ユーザの方へ
- ◆ Windows Server 2003/XP/Vista ユーザの方へ
- ◆ Microsoft Visual C++ ユーザの方へ
- ◆ 異なるバージョン

---

## IBM ILOG Views コンポーネント・スイート

IBM® ILOG® Views コンポーネント・スイートは複数の製品からなり、各製品にさらに複数のテクニカル・パッケージが含まれています。ご使用になるアプリケーション要件に応じて、これらの製品からいずれかを選択します。さらに、適宜オプションも選択できるようになっています。

基本製品では、各製品の概要を説明します。各製品の主要なサービス、含まれているテクニカル・パッケージ、および選択可能なオプションについての説明になっています。

	パッケージ												
	Foundation	Studio	Gadgets	Application Framework	Manager	Grapher	Prototypes	Web Deployment	Gantt	Charts	Maps	Data Access	Graph Layout
<b>基本製品の定義</b>													
ILOG Views Controls	X	X	X	X									
ILOG Views 2D Graphics Standard	X	X			X								
ILOG Views 2D Graphics Professional	X	X			X	X	X	X					
ILOG Views	X	X	X	X	X	X	X	X	X				
<b>オプション製品の定義</b>													
ILOG Views Charts										X			
ILOG Views Maps											X		
ILOG Views Data Access												X	
ILOG Views Graph Layout													X

図1 IBM ILOG Views コンポーネント・スイート製品

## 内容

### IBM ILOG Views Studio

IBM® ILOG® Views Studio のすぐに利用可能なバージョンが、studio/<system> ディレクトリにある ivfstudio というバイナリ・ファイルで提供されています。IBM ILOG Views Studio は、IBM ILOG Views アプリケーション作成および生成のお手伝いをする GUI ビルダです。

#### ディレクトリ

- ◆ data/ilviews: ライブラリで使用するデータ・ファイルが含まれています (インスペクタ・パネル、メッセージ・データベースなど)。
- ◆ data/images: イメージ・データ・ファイルが含まれます。
- ◆ data/icon: いくつかのアイコンが含まれます。
- ◆ data/DCW: いくつかの DCW 生成ファイルが含まれます。



- ◆ **bin:** バイナリ・ファイルとソース・コードが含まれます。同ディレクトリにある README ファイルには、これらのバイナリ・ファイルの構築方法に関する説明があります。<ILVHOME>/bin/<system> で、2つの有用なプログラムを構築することができます。ilv2data は「リソース・ファイル」を構築でき、アプリケーションとリンクすれば環境から独立させることができます。splitdbm は、バージョン 3.0 以前のメッセージ・データベースを新しい言語定義やエンコードなども含めて、新しいフォーマットに変更するものです。
- ◆ **samples:** サンプル・ファイルを含みます。ここをクリックして、README ファイルの内容を参照することができます。各サンプルは、プラットフォーム・ディレクトリに移動した上で、make ユーティリティを使って構築する必要があります。
- ◆ **tools:** よくある問題に対する特定の解決法が含まれます。

---

### IBM ILOG Views Data Access: バージョン 5.2 との既知の非互換性

Data Access 5.2 は、Open Ingres database を除いて Data Access 5.0 とソース互換性があります。ただし、バイナリ互換性は保証されていません。C++ コードを再コンパイルしてプログラムを再リンクする必要があります。これは、新規データベースをサポートするなどの理由で、ヘッダ・ファイルが変更になっているためです。

ソース互換性は、サポートされなくなったデータベースの API を除いた文書化された API のみに対応するものです。Data Access 5.2 では Oracle 8.x および Open Ingres をサポートしていませんが、Oracle 9i、10g、および 11g には対応しています。

---

## 基本製品

**基本製品**は、Controls、2D Graphics (Standard と Professional) および IBM® ILOG® Views の 4 つから構成されています。これらのすべての製品には、それぞれの専門領域以外に、Foundation クラスおよび基本 IBM ILOG Views Studio も組み込まれています。

- ◆ *IBM ILOG Views Controls*
- ◆ *IBM ILOG Views 2D Graphics Standard*
- ◆ *IBM ILOG Views 2D Graphics Professional*
- ◆ *IBM ILOG Views*

---

### IBM ILOG Views Controls

IBM ILOG Views Controls は、ポータブル・コントロールでアプリケーション GUI を開発するためのものです。これらのコントロールは、Microsoft® Windows® およ

び Motif® の規格に準拠し、アプリケーションのローカライゼーションを容易にします。パネルおよび関連スクリプトや C++ コードの作成をサポートする開発ツールも用意されています。アプリケーションでのドキュメント/ビュー・アーキテクチャ実行や、対応するユーザ・インターフェース生成を容易にするウィザードとフレームワークも備わっています。

#### 含まれるパッケージ：

- ◆ *Foundation*
- ◆ *Studio*
- ◆ *Gadget*
- ◆ *Applicaton Framework*

#### オプション：

- ◆ *IBM ILOG Views Charts*
- ◆ *IBM ILOG Views Data Access*

---

### IBM ILOG Views 2D Graphics Standard

IBM ILOG Views 2D Graphics Standard 製品は 2D Graphics の基本製品です。

IBM ILOG Views 2D Graphics Standard を使用することにより、ズーム、パン、ドラッグ・アンド・ドロップ、オブジェクトの描画に必要なインタラクティブ機能を備えた非常に効率的な 2D ベクトル・グラフィックスを開発することができます。オブジェクトは複数の階層で表示でき、それぞれのオブジェクトを表示したり、非表示にすることができます。ズームやパンも瞬時に行え、グラフィカル・オブジェクト上で異なる拡大率で複数ビューを同時表示することもできます。

#### 含まれるパッケージ：

- ◆ *Foundation*
- ◆ *Studio*
- ◆ *Manager*

#### オプション：

- ◆ *IBM ILOG Views Charts*
- ◆ *IBM ILOG Views Maps*

---

### IBM ILOG Views 2D Graphics Professional

IBM ILOG Views 2D Graphics Professional は、IBM ILOG Views 2D Graphics Standard にグラフィャー管理 (ノードとリンク) とポイント・アンド・クリックによるグラ

フィック・オブジェクト生成 (Prototypes) を追加したものです。アプリケーションでグラフやネットワーク表示が必要になったときに、真価を発揮するのが同製品です。IBM ILOG Views Studio を使用して、ネットワークを編集を行ったり、さらに、新しいタイプの動的なグラフィック・オブジェクトを作成することもできます。

**含まれるパッケージ：**

- ◆ *Foundation*
- ◆ *Studio*
- ◆ *Manager*
- ◆ *Grapher*
- ◆ *Prototypes*

**オプション：**

- ◆ *IBM ILOG Views Charts*
- ◆ *IBM ILOG Views Maps*
- ◆ *IBM ILOG Views Graph Layout*

---

**IBM ILOG Views**

IBM ILOG Views には、基本製品のすべてのパッケージが含まれています。この製品には、IBM ILOG Views Controls と IBM ILOG Views 2D Professional の機能が含まれます。さらに、Gantt Chart パッケージも含まれます。

**含まれるパッケージ：**

- ◆ *Foundation*
- ◆ *Studio*
- ◆ *Gadget*
- ◆ *Applicaton Framework*
- ◆ *Manager*
- ◆ *Grapher*
- ◆ *Prototypes*
- ◆ *Gantt Chart*

**オプション：**

- ◆ *IBM ILOG Views Charts*

- ◆ *IBM ILOG Views Maps*
- ◆ *IBM ILOG Views Data Access*
- ◆ *IBM ILOG Views Graph Layout*

## オプション製品

次のオプション製品を使うと、IBM® ILOG® Views 製品に特定の機能を追加することができます。

表1 IBM ILOG Views コンポーネント・スイート・オプション製品

パッケージ	説明
<i>IBM ILOG Views Charts</i>	すべての基本製品に対応。データ視覚化のためのパワフルな製品です。極座標、デカルト座標チャート、固定チャートまたは任意の方向へスクロールできるチャートなどが備わり、向きを選択や高レベルのカスタマイズが可能です。すべての <b>IBM ILOG Views</b> コンポーネント・スイート製品で機能します。
<i>IBM ILOG Views Maps</i>	地図背景を必要とするアプリケーション向けの地図作成パッケージ。投影図法やロード・オン・デマンドなどの高度な地図作成機能を備えています。 <b>IBM ILOG Views Maps</b> を使用するには <b>IBM ILOG Views 2D Standard</b> 製品が必要です。
<i>IBM ILOG Views Data Access</i>	データ（特にリレーショナル・データベース）に接続機能を付加します。 <b>IBM ILOG Views Studio</b> エディタの拡張機能を使うと、図表による <b>SQL</b> ステートメントの作成や <b>GUI</b> オブジェクトと結果の結合が可能です。同製品を使用するには、 <b>IBM ILOG Views Controls</b> 製品が必要です。
<i>IBM ILOG Views Graph Layout</i>	ノードやリンクからなる階層的ネットワークを表示するためのレイアウト機能を提供します。同製品を使用するには <b>IBM ILOG Views 2D Graphics Professional</b> 製品が必要です。

---

## IBM ILOG Views コンポーネント・スイートを選択する理由

IBM® ILOG® Views は、2D グラフィックの威力と GUI ビルダの簡便さを、わかりやすい 1 つの配列にまとめた **クロス・プラットフォーム C++ ライブラリ** です。同製品は、単純なフォーム・ベースのグラフィカル・ユーザ・インターフェースから数十万のオブジェクトを同時に管理する複雑なリアルタイムのベクトル・アプリケーションに至るまで、幅広いグラフィック・タスクを処理します。

IBM ILOG Views には、振る舞いとグラフィック面の分離や、多くのオブジェクトを簡単に管理できるハイレベルな抽出機能を重視する高度なアーキテクチャが備わっているため、このようなタスクを実行することができるのです。このアーキテクチャは、PostScript ダンプ、イベントの記録と再生、ファイル内でのオブジェクトの永続性など、すべてのインターフェースにまたがる共通のサービスにも対応しています。

**IBM ILOG Views はパワフルです。** IBM ILOG Views は、Microsoft® Windows® システムや Open Software Foundation の Motif® の限界を補うものです。Microsoft Windows や Motif でもウィンドウや単純な仕掛けを作成することができますが、グラフの作成、地図の描画、スプレッドシートの管理、数千のオブジェクトの動画化はできません。IBM ILOG Views を使えば、数千もの動画化されたオブジェクトを同時に表示できるのです。

**IBM ILOG Views は、効率的です。** 低いメモリ使用率と最適なデータ構造により高速処理が可能です。クワッドツリーは、数十万のオブジェクトについて表示の更新を管理します。2重バッファリングでちらつきのない動画を実現します。最適なメモリ割り当てと共有コンポーネントを用いたオブジェクトの最小化によって、大型アプリケーションでもスワップすることがありません。

**IBM ILOG Views はシンプルです。** ハイレベルなグラフィック・オブジェクトにより、コーディングも簡単です。グラフィカーが別途、必要になることはありません。IBM ILOG Views 2D Graphics Professional にはグラフィカー・クラスが備わっているからです。棒グラフ、散布プロット、ゲージについても、IBM ILOG Views には、チャート作成の専用製品があります。スプレッドシートも、IBM ILOG Views Controls Standard 製品にある IBM ILOG Views のマトリックス・クラスを使用すれば簡単です。これ以外にも多くのハイレベルなクラスが備わり、インターフェースの迅速な作成に必要な諸機能が提供されます。

**IBM ILOG Views は拡張可能です。** IBM ILOG Views は模範的な C++ スタイルで作成されています。クラス階層や内部プロトコルの文書化も行き届いているため、文書化された IBM ILOG Views クラスから派生する新しいクラスを作成することができます。オブジェクト指向の開発ではコードの再利用が重要になります。IBM ILOG Views を使用すれば、コード再利用による利点をフルに活かすことができます。

**IBM ILOG Views は移植可能です。** IBM ILOG Views は特定の環境やプラットフォームに制限されません。多くの C++ 開発環境でテストされているため、いく

## IBM ILOG Views コンポーネント・スイートからメリットを受けるアプリケーション

つものグラフィカル環境にも統合可能です。IBM ILOG Views アプリケーションは、Windows、Motif、Xlib で透過的に動作します。グラフィック・プリミティブの使用により、基盤となるウィンドウ・システムから独立してコードを作成することができます。たとえば、単一のアプリケーションを Windows (PC) と Motif (UNIX) の双方に統合化できます。IBM ILOG Views 関数を呼び出すだけで、これら 2 つの環境間で表示のルック・アンド・フィールを変更することができます。

**IBM ILOG Views はすぐに使用できます。** IBM ILOG Views には、簡単に使用できる GUI エディタとガジェット、グラフィック・オブジェクト、振る舞い、チャート、ゲージを含む多数の定義済みオブジェクトが備わっています。さらに、スプレッドシート機能としっかりとしたグラフ、描画、数式エディタも用意されています。IBM ILOG Views パッケージには、ガジェットのクラス階層がすべて含まれています。ガジェットとは、柔軟なグラフィック表示や定義済み振る舞いでの特定の用途を想定した対話式オブジェクトのことです。ガジェットにはスクロールバー、メニューバー、ボタンなどが含まれます。IBM ILOG Views を使用すれば、基本の描画プリミティブと対話式ガジェットを結合して、直感的で見栄えのよいインターフェースを作り出すことができます。

**IBM ILOG Views はモジュール方式です。** IBM ILOG Views は複数の製品から構成され、各製品は、GUI コントロール、高度な 2D グラフィック、グラフ・レイアウト、地図、チャートなど特定の分野に対応しています。各製品には、ライブラリ、Studio 拡張機能、サンプル、および包括的な資料が付属しています。必要な IBM ILOG Views 機能だけを組み合わせ使用できます。また将来的に、アプリケーションの基準が変わった時点で、IBM ILOG Views 製品を追加、拡張することも可能です。

---

## IBM ILOG Views コンポーネント・スイートからメリットを受けるアプリケーション

グラフィカル・ユーザ・インターフェース (GUI) を必要とするすべてのアプリケーションは、C++ コードの自動生成機能などを含む IBM® ILOG® Views に備わった時間節約の GUI 構築機能からメリットを受けます。IBM ILOG Views があれば、標準の GUI であれ、高度な 2 次元グラフィックの備わる GUI であれ、思いのままに GUI を構築できます。さらに、IBM ILOG Views のコードは移植可能です。特定のプラットフォーム上で作成したインターフェースを、簡単なコンパイルで別のプラットフォームに移植することができます。

数多くの IBM ILOG Views の用途のうちから、数例をここでご紹介します。

- ◆ 地図アプリケーション
- ◆ 専用エディタ
- ◆ 制御・処理アプリケーション

- ◆ ネットワーク監視アプリケーション
- ◆ 金融アプリケーション
- ◆ 交通・運輸アプリケーション

## 地図アプリケーション

IBM® ILOG® Views は、マルチビュー、レイヤ機能、大量データの処理機能、および IBM ILOG Views (投影図法、ロード・オン・デマンドなど) の特殊な地図作成機能を備えており、地図作成の分野で必要な威力を発揮します。

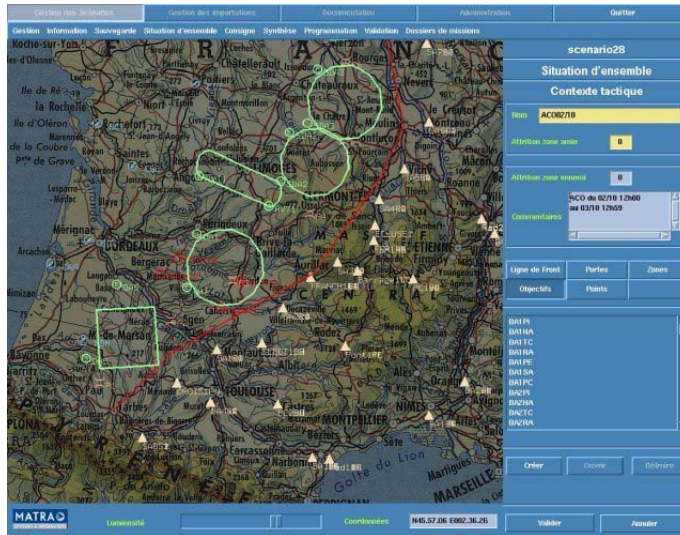


図2 IBM ILOG Views Controls、IBM ILOG Views 2D Graphics Professional、IBM ILOG Views Maps を使用した地図アプリケーション

## 専用エディタ

IBM® ILOG® Views 2D Graphics Standard には定義済みの振る舞いまたは「インタラクタ」が備わっています。これは、ライブラリの拡張性と組み合わせることで、大量の操作を編集用アプリケーションへ簡単に統合できるようにします。

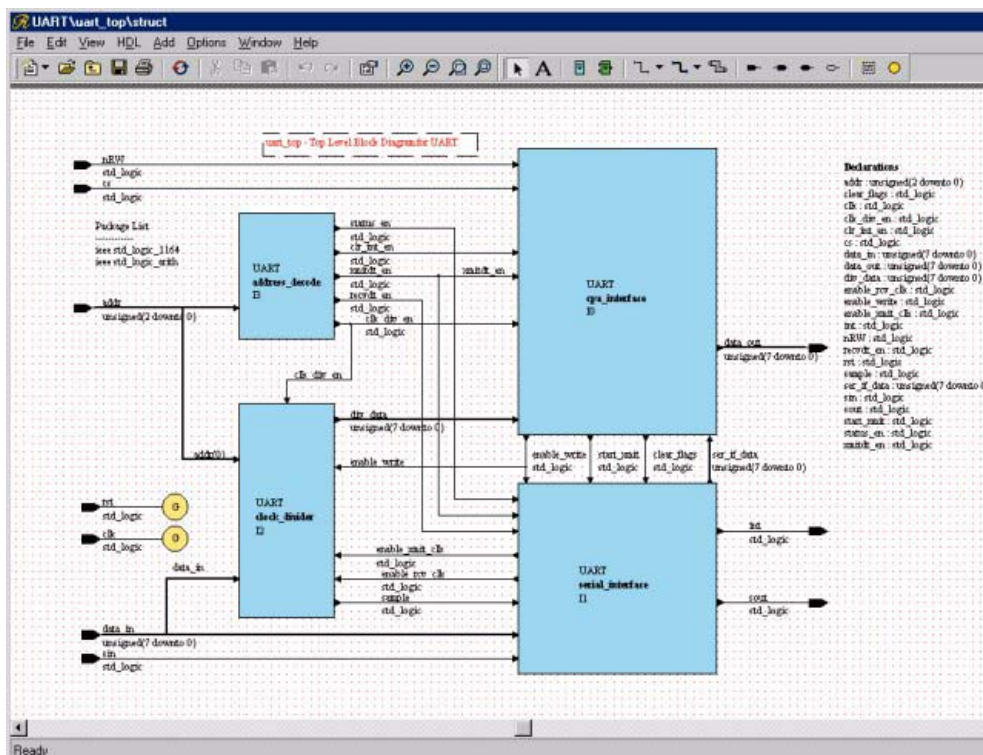


図3 ダイアグラム・エディタ

プラットフォーム間の移植を必要とするエディタの場合は、IBM ILOG Views Controls Standard と IBM ILOG Views 2D Graphics Professional を併用することで、移植コントロール、アプリケーション・フレームワーク、編集機能が提供されます。

### 制御・処理アプリケーション

システム監視や電気ネットワーク監視などのアプリケーションでは、画面上でオブジェクトを動画化する際に、リアルタイム・アプリケーションとリンクした IBM® ILOG® Views 機能を利用すると便利です。IBM ILOG Views 2D Graphics Professional は Prototypes (ポイント・アンド・クリック・エディタを使用して、設計や動画化ができるビジネス・グラフィック・オブジェクト)を提供しています。



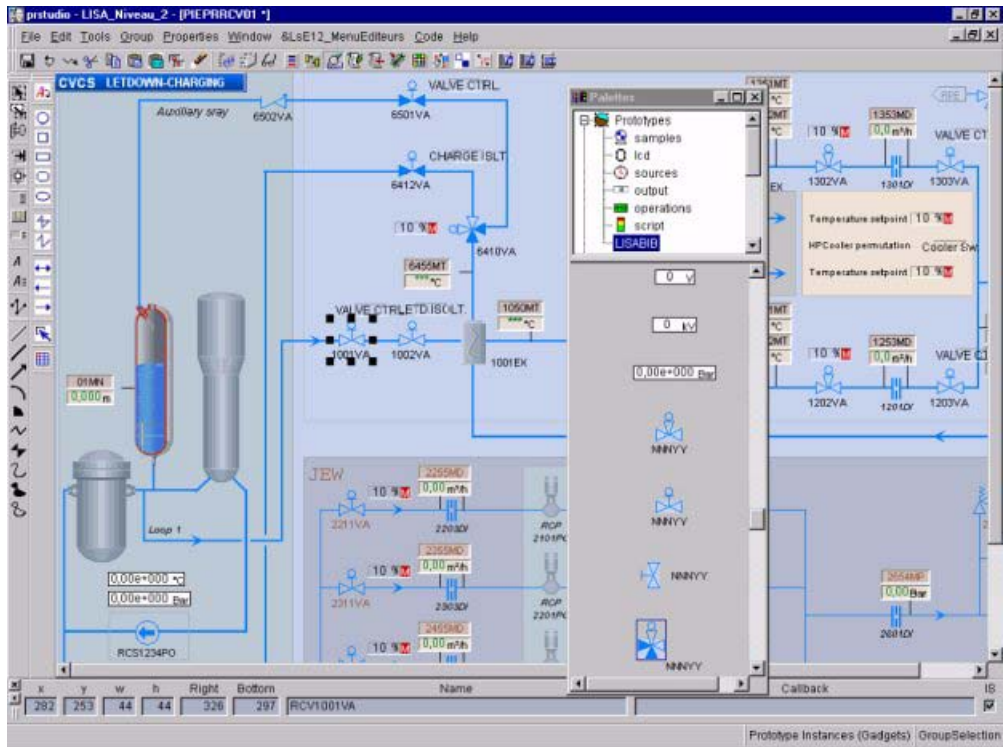


図4 グラフィカル・モデリングおよびシミュレーション・システム

## ネットワーク監視アプリケーション

IBM® ILOG® Views 2D Graphics Professional にはグラファァが含まれています。これはノードやリンクを作成するパワフルな方法で、各要素はテキスト、幾何形状、ボタン、複雑なチャートなどの個別オブジェクトとして定義されています。これ以外にも数多くの可能性があります。IBM ILOG Views Graph Layout パッケージでは、複数の定義済みレイアウト・アルゴリズムを使用してノードとリンクの自動配置を行うことができます。

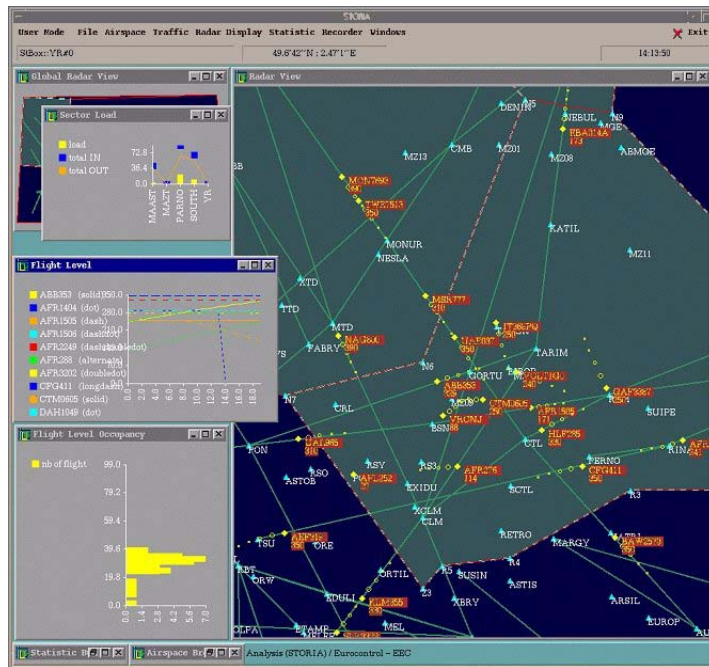


図5 航空交通管理

IBM ILOG Views Charts はパワフルなデータ視覚化機能を提供します。極座標やデカルト座標による投影図法、複数のスクロール・モード、向きを選択機能など、よく用いられるあらゆる種類のチャート（線、棒、円）が含まれます。

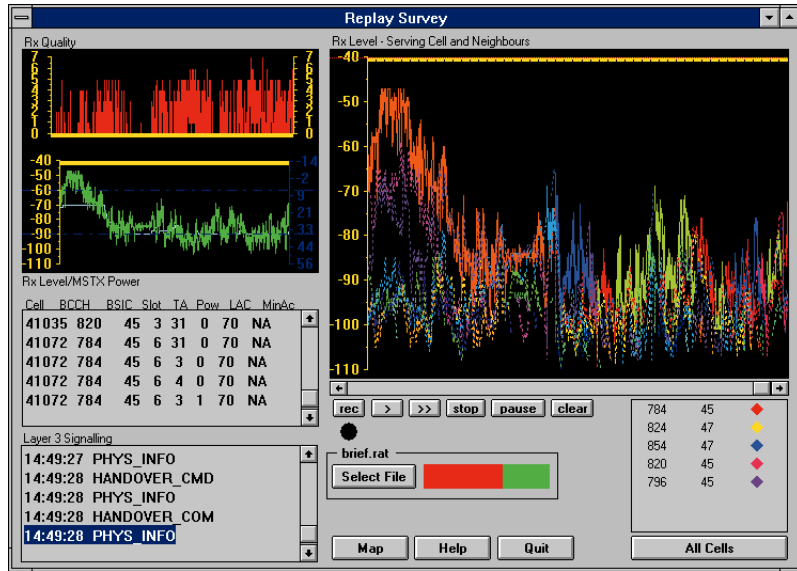


図6 GSM ネットワーク・パフォーマンス監視

## 金融アプリケーション

IBM® ILOG® Views Views Controls を IBM ILOG Views Charts と併用すると、金融アプリケーションに必要なすべてのツールが提供されます。これらのツールには、パワフルなチャート、定義済みゲージ、およびリアルタイムで大量のデータを表示できるスプレッドシートがあります。



図7 商取引用アプリケーション

## 交通・運輸アプリケーション

IBM® ILOG® Views 2D Graphics Standard には、マルチビューでグラフィックの高速アップデートを処理できるマネージャがあります。このマネージャは特に、さまざまな地図上で迅速に機体の位置を更新することが重要な航空交通管制に適用されます。また、移動車両の監視が必要なすべてのアプリケーション用に汎用化することもできます。



---

## IBM ILOG Views コンポーネント・スイート・パッケージ

特定の IBM® ILOG® Views テクニカル・パッケージについては、次の項目を参照してください。

- ◆ *IBM ILOG Views Foundation*
- ◆ *IBM ILOG Views Studio*
- ◆ *IBM ILOG Views Gadgets*
- ◆ *IBM ILOG Views Application Framework*
- ◆ *IBM ILOG Views Manager*
- ◆ *IBM ILOG Views Grapher*
- ◆ *IBM ILOG Views Prototypes*
- ◆ *IBM ILOG Views Gantt*
- ◆ *IBM ILOG Views Charts*
- ◆ *IBM ILOG Views Maps*
- ◆ *IBM ILOG Views Data Access*
- ◆ *IBM ILOG Views Graph Layout*

---

### IBM ILOG Views Foundation

IBM® ILOG® Views Foundation は、グラフィック・アプリケーションを実装するためのすべての基本クラスが入ったテクニカル・パッケージです。これらの基本クラスには、表示、基本グラフィック・オブジェクト、リソース管理、コンテナ、および基本インタラクタが含まれます。その他の IBM ILOG Views パッケージはすべて基本パッケージ上に構築されます。

#### 特徴

IBM ILOG Views Foundation の機能は、次の各セクションで説明します。

- ◆ *構造化 2D グラフィック*
- ◆ *グラフィック・プリミティブ*
- ◆ *その他のサービス*

#### 構造化 2D グラフィック

以下の機能が基本グラフィック・オブジェクトを提供します。

- ◆ **定義済みグラフィック・オブジェクト。**グラフィック・エンジンは、IBM ILOG Views プリミティブ上に構築され、グラフィック・オブジェクトを定義します。四角形、ラベル、多角形、スプラインなどの多数のオブジェクトは、拡張可能な基本グラフィック・オブジェクトの代表例です。

IBM ILOG Views ではグラフィック・オブジェクトと振る舞いが明確に区別されているため、特定の振る舞いをオブジェクトに適用することができます。

- ◆ **定義済みの振る舞いオブジェクト。**IBM ILOG Views では、定義済みの振る舞いをインタラクタと呼びます。インタラクタは、すべてのグラフィック・オブジェクトに適用して特定の振る舞いを所持せ、グラフィック・オブジェクトの機能を定義することができます。
- ◆ **コンテナ。**IBM ILOG Views は、コンテナまたはマネージャのいずれかの記憶域データ構造にオブジェクトを保持します。この2つは基本タイプの記憶域データ構造です。ビューは、コンテナまたはマネージャに格納されているグラフィック・オブジェクト一式に関連付けられています。

コンテナは一定数のグラフィック・オブジェクトを格納しており、コンテナ内に格納されているオブジェクトを表すビュー1つのみと関連付けられています。各オブジェクトは特定の振る舞いに関連付けることができます。また、定義済み関数をすぐ呼び出すキーボード・イベントであるアクセラレータをコンテナ自体に付加できます。

IBM ILOG Views には、コンテナ以外に、マネージャと呼ばれるより効率的なデータ構造があります。マネージャは *IBM ILOG Views Manager* パッケージ (IBM ILOG Views 2D Graphics Standard に含まれている) にあります。マネージャには、レイヤ、マルチビュー、高速再描画、永続性、および編集の機能が備わっています。i2 は、これらの違いを説明したものです。

## グラフィック・プリミティブ

次の基本 Foundation 機能を使用することで、高度な描画・表示操作が可能になります。

- ◆ **ウィンドウおよびビュー。**IBM ILOG Views では、ビューは、基本サービスを付け加えることができるオブジェクトです。オブジェクトは、UNIX での X Window? などの基礎となる表示システムのウィンドウと関連付けられます。すべての描画は、オブジェクトまたはオブジェクトのサブセットのイメージを表すビュー内で行われます。このイメージは、オブジェクト自体に影響を与えることなく、移動、拡大、または回転によって幾何学的に変換できます。
- ◆ **描画プリミティブ。**IBM ILOG Views は、2次元ベクトル・グラフィック・エンジンを使用して、描画ポート (メモリ、画面、ダンプ・ファイル)、および基本幾何学形状を作成する描画プリミティブの大規模セットを提供します。円弧、曲線、四角形、ラベルなどの基本幾何学形状を描画できます。メモリ内の画面上での描画、PostScript などのダンプ・ファイルの生成が可能です。白黒およびカラーのイメージを作成できます。

- ◆ **リソース。** IBM ILOG Views には定義済みの幾何学オブジェクトがありますが、数多くの描画ツールを使用して、グラフィック・イメージを作成することもできます。塗りつぶしパターンや線種、色、フォントの属性など、多くのリソースから自由に選択できます。グラフィック・オブジェクトやテキストにこれらのリソースを適用する場合は、パレットを使用します。
- ◆ **イベント。** キーボード、マウス、タイマー・イベントを管理します。
- ◆ **トランスフォーマ。** グラフィック・オブジェクトの描画を行うために幾何学変換を計算し、移動、回転、拡大などの振る舞いを利用するオブジェクトであるトランスフォーマを操作します。

#### その他のサービス

- ◆ **ActiveX。** ActiveX の生成と統合。このサービスでは、簡単なボタンクリック操作で ActiveX を作成できる上、既存のあらゆる ActiveX をユーザの GUI に統合できます。
- ◆ **スクリプト。** IBM ILOG Script は、グラフィック・オブジェクトにアクセスできるように、JavaScript™ スクリプト言語を実装したものです。
- ◆ **国際化。** POSIX ロケール・モデルを使用した国際化アプリケーション。
- ◆ **XML。** XML パーサーを使用することにより、たとえば、XML ファイルを読み込んでコンテナ内にオブジェクトを作成したり、IBM ILOG Views データから XML ファイルを作成したりできます。

---

### IBM ILOG Views Studio

IBM® ILOG® Views は、IBM ILOG Views オブジェクトの作成、編集、保存を行うためのパワフルなエディタです。IBM ILOG Views Studio を使用すると、パネルの設計、ガジェットの編集、グラフィック・オブジェクトの描画とインポート、編集のためのオブジェクトのグループ化、オブジェクトへの属性や多言語ラベルの割り当てを行うことができます。一般的には、独自の GUI が作成できます。

この実行可能プログラムは、特定の各用途のための動的モジュール (Gadets 用 Studio、Manager 用 Studio など) で補完するように設計されています。

#### Studio の主な機能

IBM ILOG Views Studio の主要機能は次のとおりです。

- ◆ **ドラッグ・アンド・ドロップとテスト・モードを含む WYSIWYG エディタ。**
- ◆ **ドラッグ・アンド・ドロップ。**



- ◆ **マーキング・メニュー**。IBM ILOG Views Studio には新しい種類のポップアップ・メニューがあります。これはマーキング・メニューと呼ばれるもので特定方向へマウスを移動して1つ以上のアイテムを選択できる方向性のあるポップアップ・メニューです。使い方に慣れれば非常に簡単に使用できる上、効率的なメニューです。

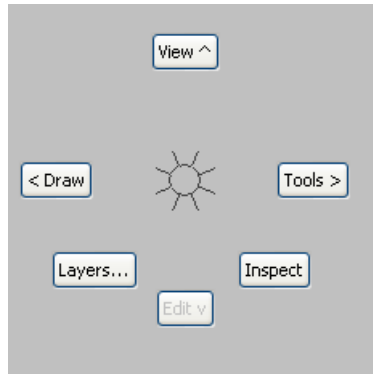


図10 IBM ILOG Views Studio マーキング・メニュー

- ◆ **MDI (Multiple Dockable Interface)、ドッキング可能パネルとツールバー**。
- ◆ **プラットフォームのロック・アンド・フィール**。IBM ILOG Views Studio を使用すると、簡単なメニュー選択によってプラットフォーム上でロック・アンド・フィールを指定することができます。

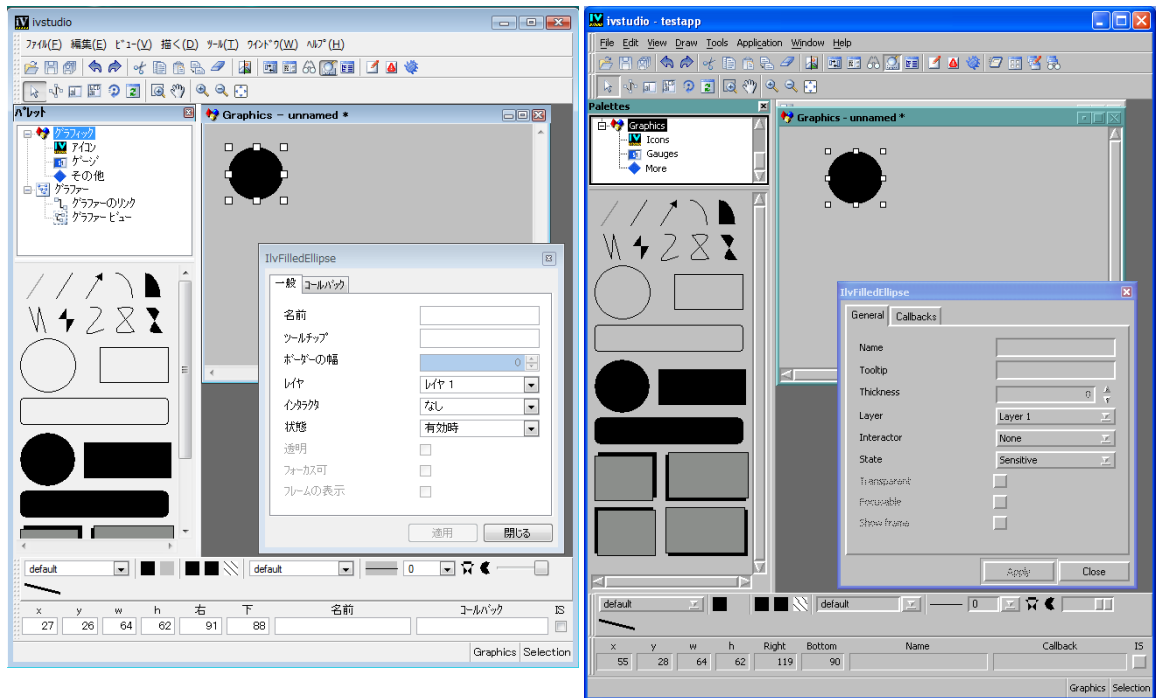


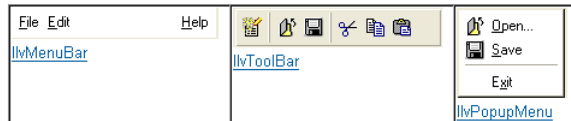
図11 さまざまなプラットフォーム上でのIBM ILOG Views Studio のルック・アンド・フィール: Windows® XP (左) およびMotif® (右)

- ◆ IBM ILOG Script へのアクセス
- ◆ C++ コードの生成 (必要な場合)。
- ◆ カスタマイズ・ファイル。
- ◆ プラグインによる拡張機能。プラグインを追加して IBM ILOG Views Studio を補完することができます。ほとんどの IBM ILOG Views パッケージには、新しいグラフィック・オブジェクトとそのインスペクタ、データまたはコード生成、描画ツールなどの拡張機能を持ったプラグインがあります。
- ◆ 再コンパイル可能機能。IBM ILOG Views Studio には、プラグイン以外にユーザ・コードとリンクするライブラリがあります。このため、IBM ILOG Views Studio をユーザ独自のアプリケーション・エディタとして使用できます。この機能を使うには、Gadget、Manager、Grapher 各パッケージへのアクセスが必要です。

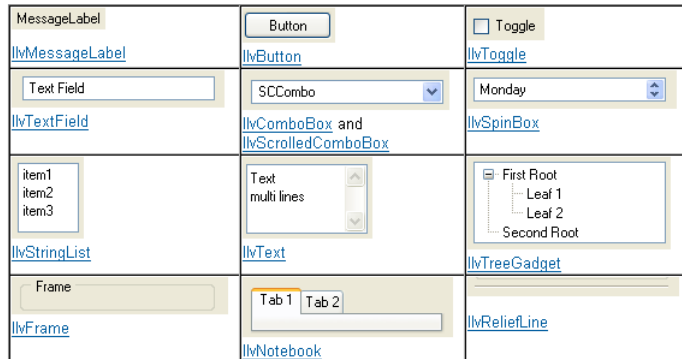
## IBM ILOG Views Gadgets

IBM® ILOG® Views Gadgets があると、メニュー、ボタン、スクロール・バー、その他多くの「ガジェット」など、いくつものインターフェース・オブジェクトにアクセスできます。これらのオブジェクトは、IBM ILOG Views Studio で GUI パネルを構築して、簡単に表示したり、操作できるようになっています。Gadgets の IBM ILOG Views Studio を使うと、独自のアプリケーションのプログラミングに使う C++ コードを生成すると同時に、簡単なドラッグ・アンド・ドロップ式の操作で定義済み「ガジェット」から独自のインターフェースを構築することもできます。

メニュー



共通ガジェット



マトリクス

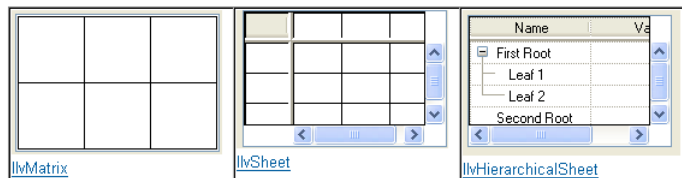


図12 共通ガジェットの一部分

## Gadgets の主な機能

以下に Gadgets の機能の一部を示します。

- ◆ Gadgets は Microsoft® Windows® システム、および Motif® のルック・アンド・フィールに準拠しています。
- ◆ IBM ILOG Views は特定のウィンドウ・システムに依存することなくガジェットを実行するため、ドッキング可能ツールバーや MDI (Multiple Document Interface) などの Windows 固有機能を Linux® と UNIX® 上で使用できます。
- ◆ ガジェットのアクティブ化の際に実行されるコールバックは、IBM ILOG Script または C++ で定義できます。
- ◆ ガジェットには、テーブル、ツリー、リスト、ダイアログなどが含まれます。
- ◆ C++ コードを IBM ILOG Views Studio から生成して、独自の GUI アプリケーションを作成します。
- ◆ ローカライゼーションのサポート。ガジェットは、オペレーティング・システムのロケール・サポートと表示システム入力方式によって日本語、中国語、韓国語をサポートし、メッセージの記憶域を必要に応じてマルチバイトの文字列として実行します。アラビア語などの言語をサポートするために双方向バージョンも用意されています。(Bidi をサポートしている AIX® とアラビア語 Windows® に限ります)。

前提となるパッケージ: *Foundation, Studio*.

---

## IBM ILOG Views Application Framework

IBM® ILOG® Views Application Framework は、ドキュメント/ビュー・アーキテクチャを使用して完全なアプリケーションを非常に効率的に作成できる新しいライブラリです。このアーキテクチャを使用すると、複数のビューで同一のドキュメントにアクセスできます。ビュー内で変更があった場合は、必要に応じて他のすべてのビューも更新されます。また、このアーキテクチャでは、ビューとドキュメント間でのメッセージ送信の複雑さを排除できます。

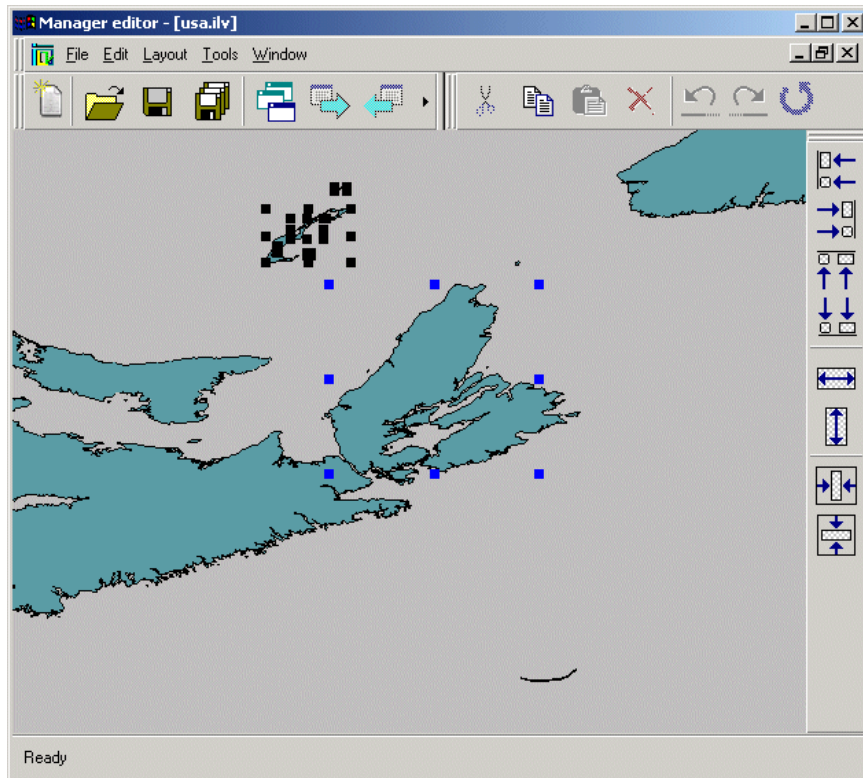


図13 標準アプリケーション

ドキュメント/ビュー・アーキテクチャに加えて、Application Framework で簡易化されるのが、ユーザ・インターフェースの開発です。これは Application Framework が MDI やドッキング可能ウィンドウの実装に必要なすべての厄介なコーディング作業を処理するためです。また、Microsoft Windows? でのファイルのドラッグ・アンド・ドロップや最近使用したファイルの管理機能も提供しています。さらに、Application Framework を使用すると、メニューやツールバーなどのアプリケーション・ユーザ・インターフェースの構成に加え、その内容や配置も変更できます。この構成はユーザ・ベースで保存され、アプリケーションの再起動時に復元されません。

本モジュールは、Application Framework Editor ( 図 14 参照 ) というツールを提供しています。このツールを使用すると、アプリケーションをグラフィカルに編集できます。メニュー、ツールバー、アクション、動的メニューなどはすべて、このツールを使って指定します。

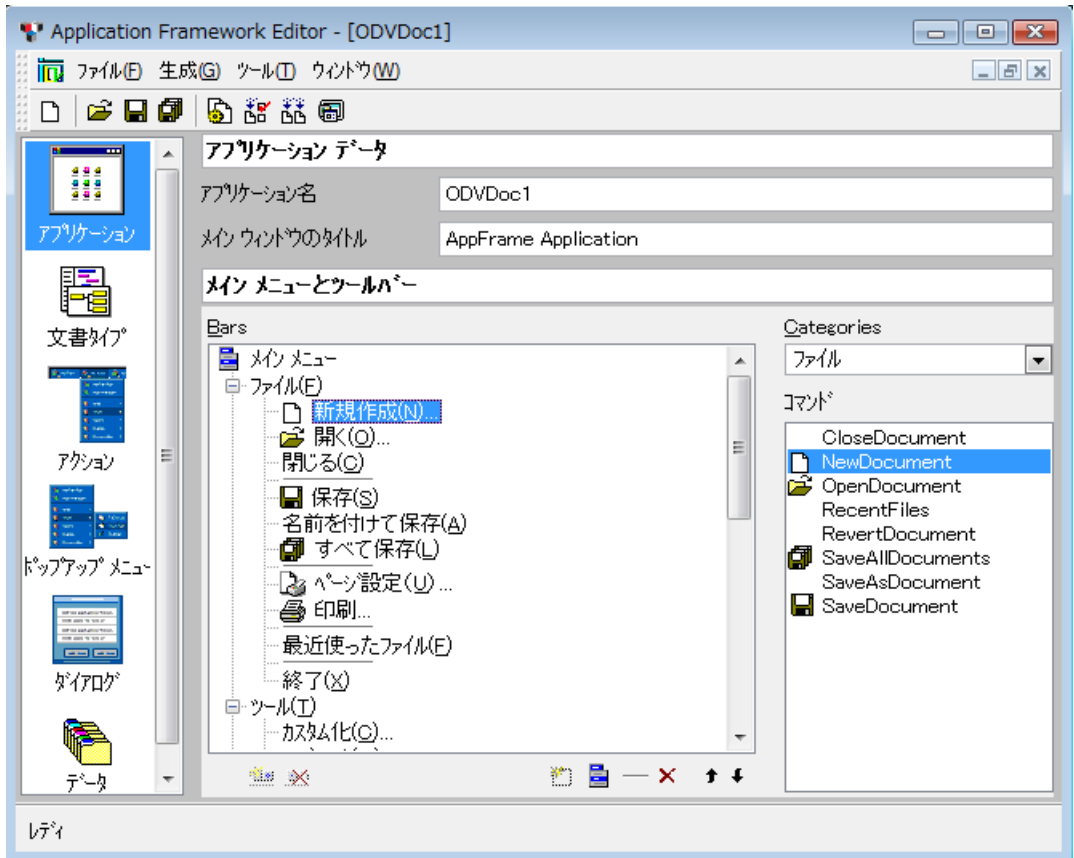


図 14 Application Framework Editor

### Application Framework の主な機能

- ◆ Model-View-Controller (MVC) アーキテクチャのドキュメント/ビュー実行機能を提供します。このアーキテクチャは、MFC プログラマや最新の設計パターンと Java に精通しているプログラマが容易にアクセスできる環境を提供しています。
- ◆ 複数ドキュメントの同時編集をサポートします。
- ◆ ドッキング可能メカニズムをカプセル化し、すべてのドッキング可能バーの位置を格納（および復元）します。
- ◆ 実行時にメニューやツールバーの内容を再編成できます。すべての変更内容が保存復元されます。
- ◆ 最近使用したファイル・リストを保持します。

- ◆ 独自のアプリケーション・オプションを追加できる拡張オプション言語があります。
- ◆ 実行時の変更内容は、同じフォーマットでユーザのホーム・ファイルに保存されます。
- ◆ **Application Framework** コマンドについて、元に戻す、繰り返しの自動機能を提供します。
- ◆ ツールバーやメニューからのアクションは直接アクティブなビューやドキュメントに送られます。コールバックは必要ありません。
- ◆ インターフェース・メカニズムが実装されています。このメカニズムは、ユーザ名と C++ メソッド間の対応関係を確立するものです。このマップを使用して、イントロスペクション § スクリプト、COM 用にユーザ定義の C++ メソッドを自動的にエクスポートできます。

前提となるパッケージ: *Foundation, Studio, Gadget.*

---

## IBM ILOG Views Manager

マネージャで強化されたハイレベルな機能は、コンテナで可能な操作にとどまらず、さまざまな状況下のオブジェクトの相互作用や表示をアクティブに制御します。

マネージャは、グラフィック・オブジェクト・セットにパフォーマンス、機能、およびグローバル・コントロールを追加します。たとえば、マネージャを使用して、さまざまなビューにオブジェクトを付加したり、表示の特定領域を拡大または縮小することができます。

詳細については、次のセクションを参照してください。

- ◆ マネージャの基本機能
- ◆ *Manager* のサービス機能
- ◆ マネージャとコンテナの比較

### マネージャの基本機能

マネージャを使用して、次の処理を行えます。

- ◆ オブジェクト・セットとビュー・セット間の関連性の管理。
- ◆ オブジェクトのレイヤ化の管理。

マネージャを使うと、オブジェクト・セットを別々のビューに表示できます。マネージャはレイヤ化を制御するため、表示や振る舞いの目的でオブジェクトを特定のレイヤに割り当てることができます。各レイヤは、マネージャに制御されている各ビューで表示または非表示に設定できます。

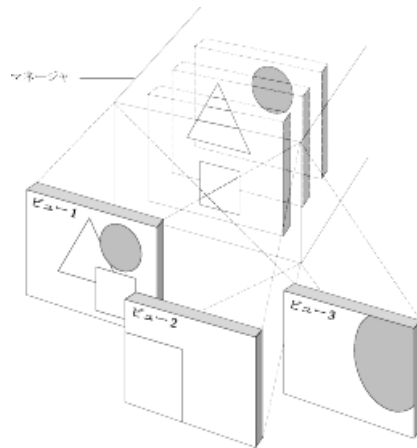


図15 マネージャを使うと、異なるビューを画面上に表示できる

- ◆ オブジェクト・セットの異なるビューの管理。
- ◆ ビューに特定の振る舞いを割り当て。

マネージャ使用時は、インタラクタが、オブジェクトに依存しないビュー全体に振る舞いのコンテキストを提供することにより、オブジェクト・レベルを超えた振る舞いを実現します。ビュー・インタラクタをアクティブにして、編集モードに入ります。このモードでは、オブジェクトの選択、ビュー・オブジェクトの描画、ドラッグ・アンド・ドロップなどを行うことができます。

### Manager のサービス機能

マネージャは次のようなサービスも提供します。

- ◆ ファイルへの保存
- ◆ 元に戻す、繰り返しイベントのコマンド・ログを最小コーディングで保持
- ◆ クワッドツリーなどの特殊データ構造を使用したパフォーマンスの改善
- ◆ 高度なりフレッシュ機能

### マネージャとコンテナの比較

次の表は、コンテナとマネージャの相違点をまとめたものです。

表2 マネージャとコンテナの機能比較

特徴	マネージャ	コンテナ
マルチビュー	●	
レイヤ化	●	



表2 マネージャとコンテナの機能比較

特徴	マネージャ	コンテナ
保存	●	
オブジェクト・インタラクタ	●	●
ビュー・インタラクタ	●	
アクセラレータ	●	●
トランスフォーマ (拡大/移動/回転)	●	●
ダブル・バッファリング	●	●
領域バッファリング	●	●
多数オブジェクト	●	1
ガジェット指向	1	●
元に戻す	●	

<sup>1</sup> 可能ですが、最善の方法ではありません。

前提となるパッケージ: *Foundation, Studio.*

## IBM ILOG Views Grapher

IBM® ILOG® Views Grapher は、階層情報や相互関連情報のグラフィック表示を専門に行います。Grapher はカスタマイズ可能なネットワーク・エディタで、グラフィック・オブジェクトを使用してノードやリンクを表示します。Grapher を IBM ILOG Views Manager パッケージ上に構築することで、さらに機能を拡張することができます。IBM ILOG Views Studio Grapher 拡張機能によって定義済みリンクが提供され、これにより、新しい種類のリンクをグラフィカルに作成できます。

### Grapher の主な機能

IBM ILOG Views Grapher の機能例を次に示します。

- ◆ ノード操作に関するリンク管理。
- ◆ リンクのビジュアル定義。
- ◆ 接続ポイントのビジュアル定義: ノード上またはリンク上の接続ポイントのビジュアル定義。

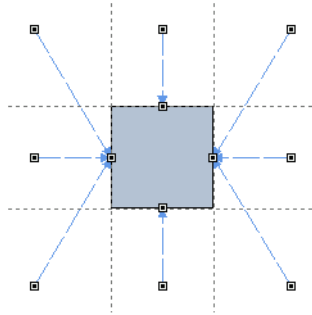


図16 接続ピンが定義されていない場合の最終ポイント位置

- ◆ **リンク・タイプの豊富な選択肢**：カスタマイズされたリンク、リンク・ラベル、ストレート、スプライン、円弧、およびフリー・フォームのポリライン。自動または固定の向きを有した単一（一方向）、二重（双方向）リンク。
- ◆ **グラフィャー・ピン編集**：グラフィャー・ピンを定義すると、オブジェクトを拡大/縮小または回転しても一貫性が保たれます。（ピン・ポイントはオブジェクトの重心に関連付けられます）。

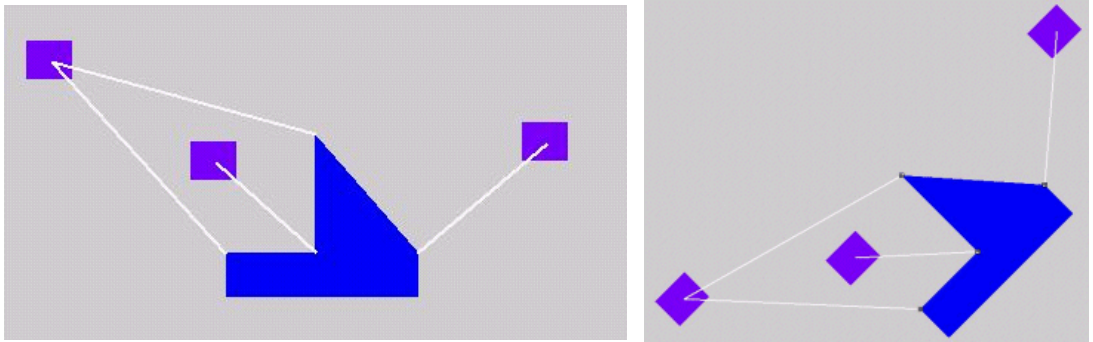


図17 リンクはグラフィャー・ピンでオブジェクトに付加され、オブジェクトが回転しても不変のままです。

前提となるパッケージ：Foundation, Studio, Manager.

## IBM ILOG Views Prototypes

アプリケーション開発者は、ユーザが対話的に使用できるようなドメイン固有アプリケーション・オブジェクトを表現するために、グラフィック・オブジェクトをカスタマイズする必要に迫られることがよくあります。そのようなカスタマイズされたドメイン固有グラフィック・オブジェクト作成に役立つのが IBM® ILOG® Views Prototypes パッケージです。これらのビジネス・グラフィッ

ク・オブジェクトは、IBM ILOG Views Studio の Prototypes 拡張機能を使用して対話式に作成されます。Prototypes を使用すると、相互に対話する精巧なオブジェクトを作成できます。また、スクリプト言語を使用して複雑な振る舞いをこれらのオブジェクトに付加することができます。

IBM ILOG Views Studio でプロトタイプを作成する際には、オブジェクトのドメイン値と外観を定義してから、振る舞いを付加します。次に、プロトタイプのインスタンスを作成し、基本 IBM ILOG Views グラフィック・オブジェクトで行う場合と同様、それらのインスタンスをマネージャやコンテナ内で使用します。アプリケーション・オブジェクトはそれらに対応するプロトタイプにリンクできます。表示、同期化、およびユーザ・インタラクションは、Prototypes パッケージによって処理されます。プロトタイプはいつでも編集または変更できます。プロトタイプのインスタンスは自動的に更新されます。

ドメイン固有オブジェクト用にパワフルで直接操作できるインターフェースを作成するのは、フォーム・ベースのインターフェースを作成するのと同じ位簡単ですが、直接操作できるインターフェースの方がより魅力的でわかりやすいものとなります。

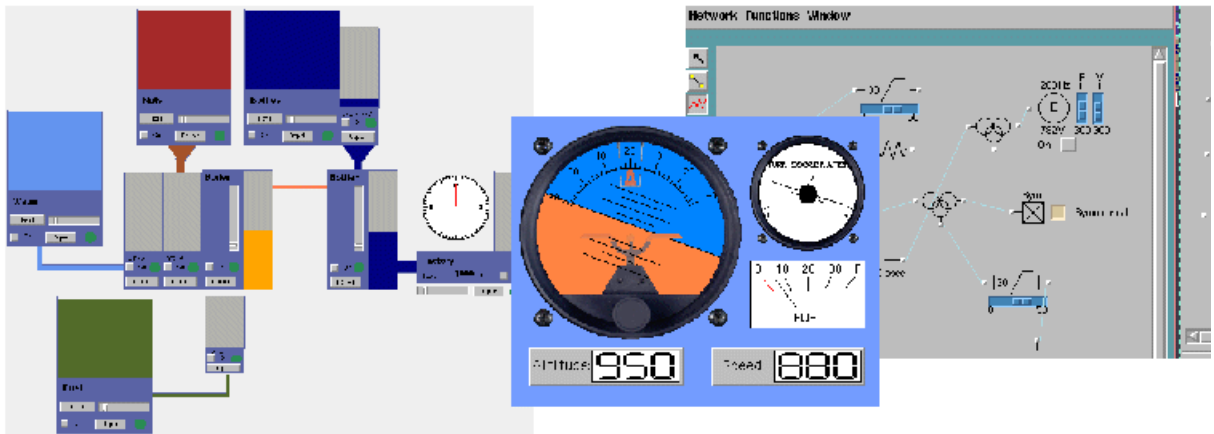


図18 Prototypes で構築したアプリケーション・パネルの例  
前提となるパッケージ: Foundation, Studio, Manager, Grapher.

## IBM ILOG Views Gantt

Gantt Chart パッケージは IBM® ILOG® Views 基本製品の中に含まれています。Gantt Chart はスケジューリング・データの表示と編集を管理します。スケジューリングは、リソースを一定期間でアクティビティに割り当てるプロセスです。

ガントチャート(タイム・ライン)は通常、プロジェクト内のタスク間の時間関係を示す水平のバー・チャートを指します。

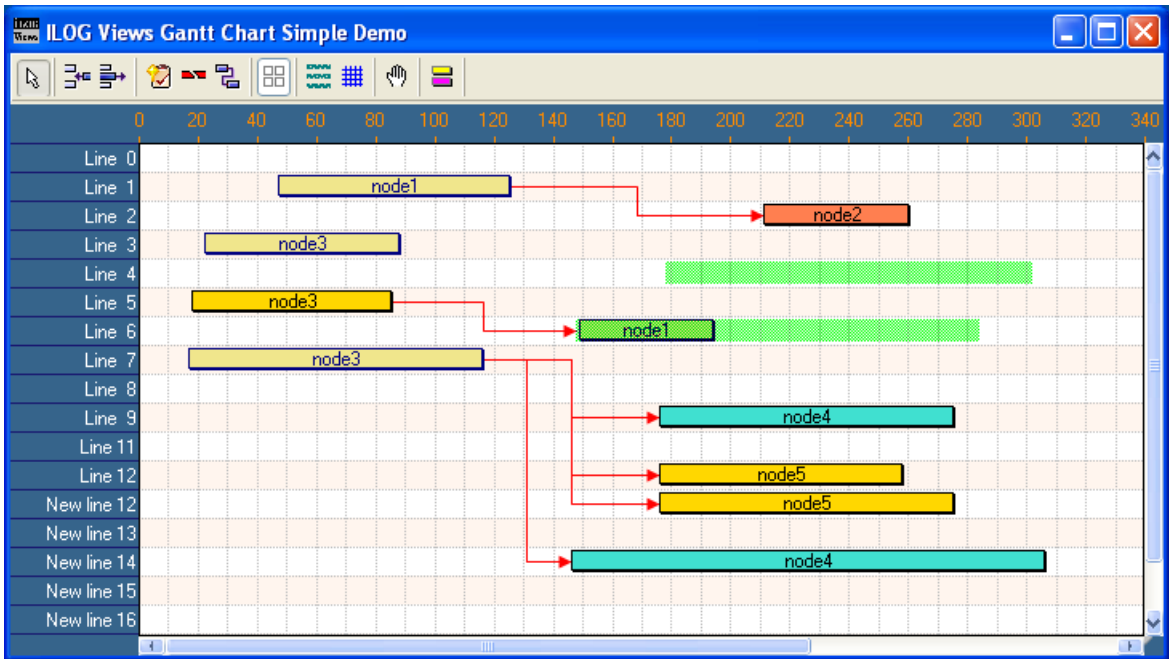


図19 ガントチャートの例

IBM ILOG Views Gantt チャートを使用すると、次のような広範なデータ・セットの作成、編集、管理ができます。

- ◆ リソースおよびタスク。スケジューリング問題の一般入力。
- ◆ アクティビティおよびサブアクティビティ。開始時刻、終了時刻、およびリソース・キャパシティ。
- ◆ 優先順位の制約。タスクの依存性を表示。これらは、開始アクティビティ、終了アクティビティ、および遅延を含むガント・リンクによって定義されます。

次に、カレンダー・アプリケーションの例を示します。

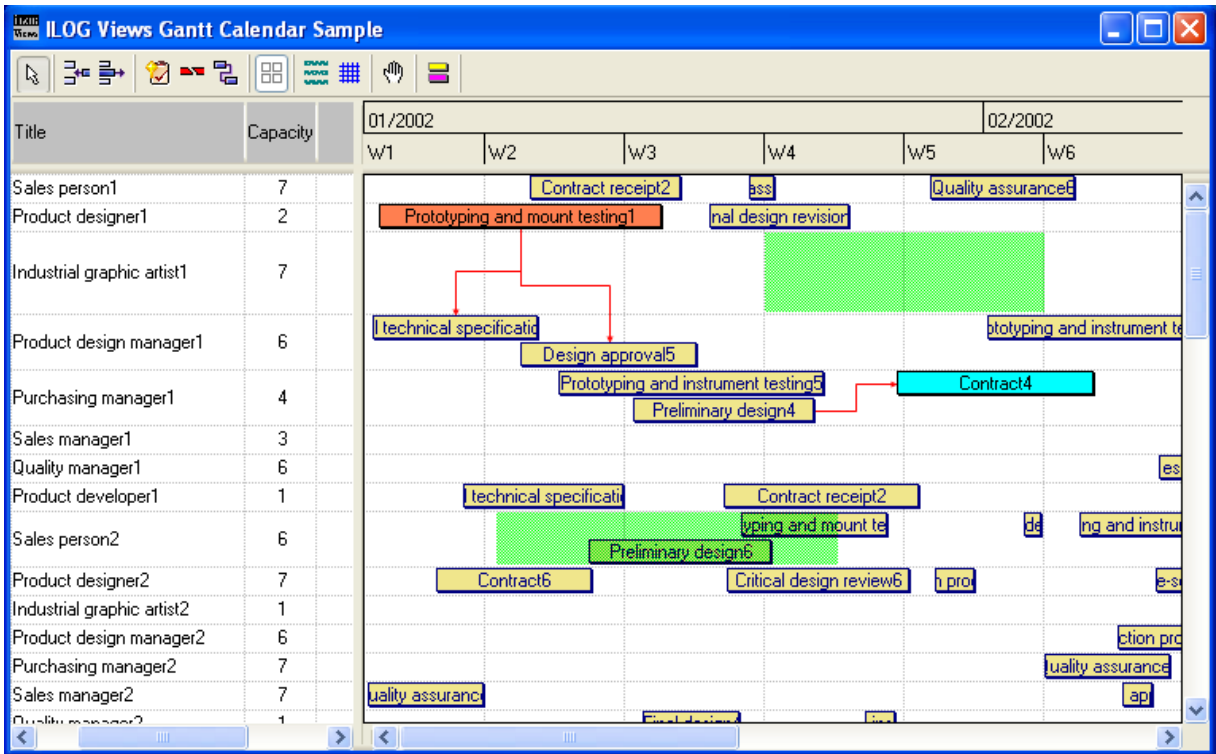


図20 ガントチャートを使ったカレンダー・アプリケーション

詳細については、*IBM ILOG Views Gantt ユーザ・マニュアル*を参照してください。

前提となるパッケージ：*IBM ILOG Views*

## IBM ILOG Views Charts

IBM® ILOG® Views Charts は、データ視覚化で威力を発揮するパッケージです。このパッケージでは、高度なカスタマイズが可能なデカルト座標や極座標を使って、定義済みデータやリアルタイム・データを図表化することができます。

### Charts の主な機能

Charts パッケージには次の機能が含まれています。

- ◆ **豊富なチャート表現。** 次のようなチャート表現が可能です。散布、折れ線、多角形、マーク付き折れ線、棒グラフ、3D 棒グラフ、ステップ、階段、高低表示、パイ、平行棒グラフ、積み重ね、積み重ね棒、積み重ね 3D 棒、および積み重ね多角形。

- ◆ **デカルト座標または極座標。**データをそれぞれ標準方式または円形方式で表示します。また表示の切り替えも簡単に行えます。
- ◆ **データ認識チャート。**リアルタイム監視に最適です。
- ◆ **スクロールの自動管理。**複数のモードがあります(停止、シフト、サイクリック)。
- ◆ **定義済みインタラクタ。**拡大、データ・ポイントの選択、その他のアクション用。
- ◆ **各主軸の対数目盛り。**
- ◆ **複数の縦座標軸目盛り。**任意の数の縦座標軸目盛りを追加できます(横座標軸は1つのみです)。
- ◆ **選択可能な目盛りの向き。**デカルト座標チャートの横座標スケールと縦座標スケールは、水平または垂直のいずれかに向きを指定できます。極座標では、円形横座標軸目盛りは時計方向または反時計方向に指定でき、放射状縦座標軸目盛りは任意の角度に向きを指定できます。
- ◆ **チャート凡例とグリッド管理。**
- ◆ **関数定義用のスクリプトの統合。**たとえば、IBM ILOG Script を使用して、チャート上に表示される関数を定義できます。
- ◆ **IBM ILOG Views Studio への完全統合。**この機能により、コーディングなしでチャートの作成とカスタマイズができます。

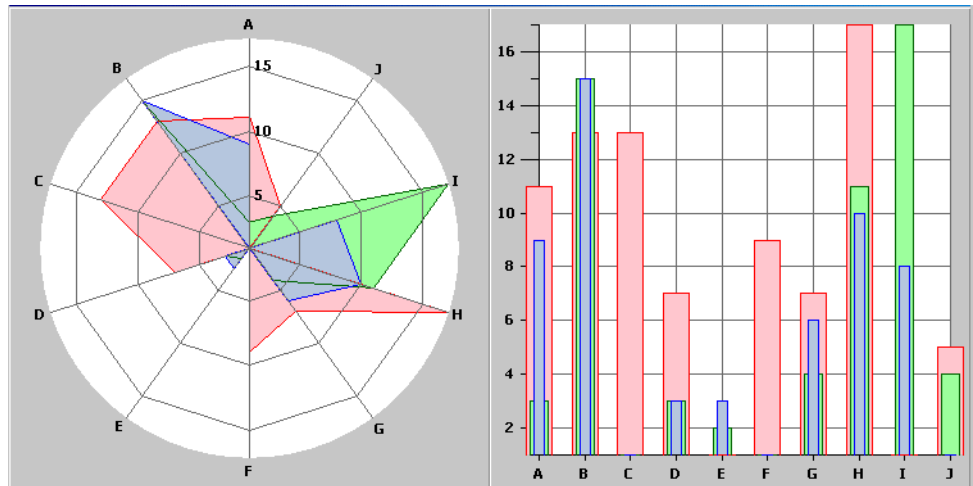


図21 例：極座標およびデカルト座標

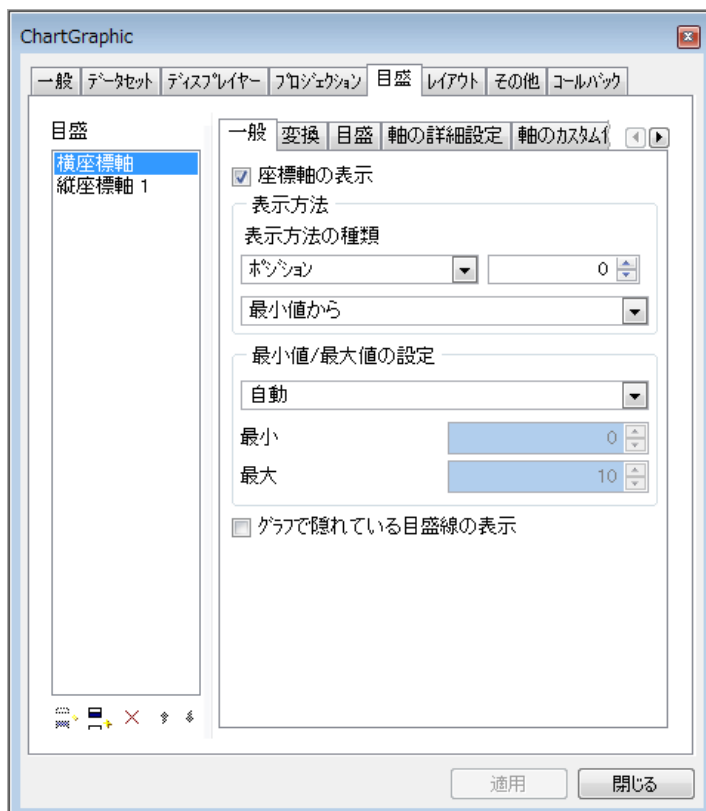


図22 Studio のチャート・インスペクタを使うと、チャート上のすべての項目を変更可能前提となるパッケージ: *Foundation, Studio*

## IBM ILOG Views Maps

IBM ILOG Views Maps は、複数のデータ形式、投影図法、および空間データの視覚化と管理を必要とするアプリケーション向けに開発された高度な地図作成機能をサポートする地図作成用パッケージです。

IBM ILOG Views Maps には、次のような機能が備わっています。

- ◆ **リーダー**。Oracle 8i、Shape、DTED、および CADRG は次のパラグラフで説明するリーダー・フレームワーク・アーキテクチャを使用します。**Shape** リーダーは属性を、**CADRG** リーダーはロード・オン・デマンドを管理し、さらに **DTED** リーダーはロード・オン・デマンドとカラー・モデルを管理します。

- ◆ **リーダー・フレームワーク。** IBM ILOG Views Maps アーキテクチャにより、新しいリーダーの開発が容易になります。このアーキテクチャには、**地図機能** (OpenGIS のシンプル機能に準拠した、グラフィック表現のないオブジェクトの地図作成) のコンセプトと **レンダラ** (地図機能をビュー・グラフィック・オブジェクトに結合するオブジェクト) のコンセプトが含まれています。あらかじめ定義されたレンダラがいくつか用意されています。
- ◆ **投影図法。** 非常に正確なレンダリングと歪み補正のための地図作成要件を満たす、円柱、円錐、方位角系列など広範な投影図法。測地データによって、より正確な位置決めが可能になります。
- ◆ **ロード・オン・デマンド。** この機能では、キャッシュ管理により、表示の必要がある場合のみデータをロードすることによって、特大サイズの地図の管理が可能になります。
- ◆ **定義済みコントロール。** 地図ベース・アプリケーション専用ガジェットが用意されています。この機能には、地図コントロール用の特化されたツールバー (拡大/縮小、パン・ツール、選択ツール)、コンパス、目盛りなどが含まれます。これらの機能を使うには、IBM ILOG Views Gadgets パッケージが必要です。
- ◆ **マップ・ビルダ。** このアプリケーションでは、データファイル、レイヤ可視フィルタ、ロード・オン・デマンド用タイリングなどの機能を選択することによって、対話式に地図を作成できます。保存した地図は、アプリケーションで使用できます。また、すべてのロード・オン・デマンドとレイヤ可視機能は自動的に処理されます。



図23 地図投影図法のサンプル



## IBM ILOG Views Data Access

IBM ILOG Views Data Access パッケージ (旧名は InForm) は、グラフィックを多用するアプリケーション開発用のビジュアル環境です。Data Access は、IBM ILOG Views で作成されたグラフィック・オブジェクトをデータ・ソースにリンクすることによって、それらのオブジェクトの機能を強化します。同パッケージは、SQL データ・ソースを提供し、新しいデータ・ソースを作成することで、情報ストリーム・データ認識グラフィック・オブジェクトに接続できるようにします。

詳細については、次のセクションを参照してください。

- ◆ *Data Access* の主な機能
- ◆ *IBM ILOG Views Studio* 用の *Data Access* 拡張機能

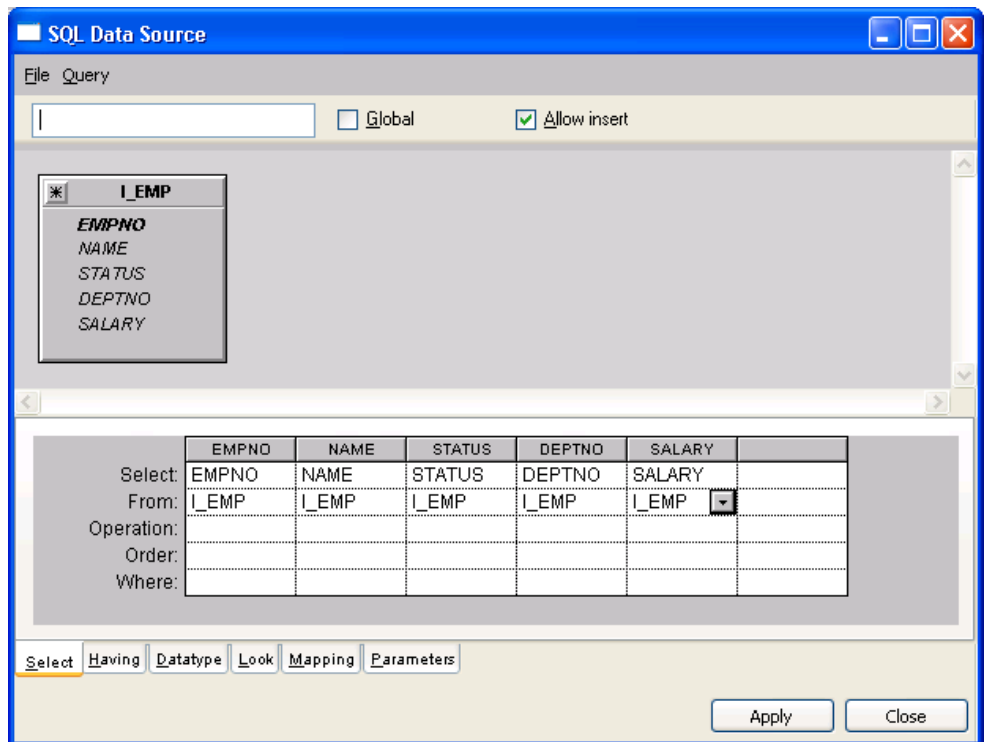


図 24 サンプル: SQL データ・ソース

## Data Access の主な機能

- ◆ Data Access を使用すると、クライアント / サーバ・アプリケーション用に、直感的で使いやすいインターフェースを迅速に作成できます。これらのアプリケーションは通常、フィールド・セット (テキスト・フィールド、チェック・ボックスなど) を含むフォームから構成されます。これらのフィールドに示される値は、フィールドと外部データ・システムのデータとの間のマッピングによるものです。
- ◆ グラフィカル・ユーザ・インターフェースのフィールドと外部データ・システムとの間のマッピングは双方向で行われます。つまり、データをデータ・ベースから取得してフィールドに表示できる一方、ユーザはこのデータを変更して長期記憶装置の外部データ・ソースで更新できます。
- ◆ アプリケーションの作成は簡単です。スタジオのパレットから GUI オブジェクトをドラッグし、そのオブジェクトを新しいパネルにドロップするだけです。データ・アクセスはデータ・ソース・オブジェクトによって管理されます。いったんデータベースに接続されると § ビューを構成する表や列が表示されます。コーディングをまったく必要としない、シンプルなポイント・アンド・クリック・アクションで典型的な SQL 操作が定義されます。

## IBM ILOG Views Studio 用の Data Access 拡張機能

Data Access は IBM ILOG Views Studio にプラグインを提供します。Studio は、アプリケーション・インターフェース・パネルの作成と視覚的なデータベースへのアクセスの両方に使用されます。シンプルかつグラフィックな方法で外部データ・ソースとの接続を可能にする特別なインターフェースもあります。

前提となるパッケージ: *Foundation, Studio, Gadget*

---

## IBM ILOG Views Graph Layout

IBM ILOG Views Graph Layout パッケージは、ノードやリンクから構成されたグラフィック表示を必要とするアプリケーションに、自動ネットワーク・レイアウトを提供します。グラフィック・オブジェクトであれば、ノードとして動作するように定義することができ、リンクを介して他のノードと接続させることができます。リンク自体は多くの異なるフォームを持つことができます。レイアウトは、簡単に使用できるように直接 IBM ILOG Views Studio で利用できるようになっており、また完全な API も提供されています。

Graph Layout には、次のレイアウト・アルゴリズムが含まれています。

- ◆ ツリー
- ◆ 階層
- ◆ 直交リンク型
- ◆ ランダム

◆ バス

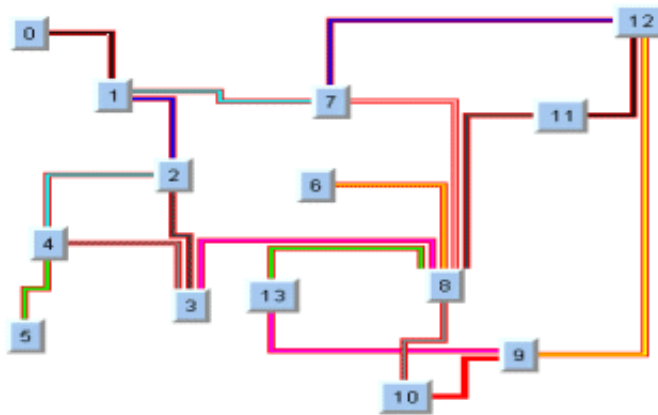


図25 直交リンク型レイアウトの例

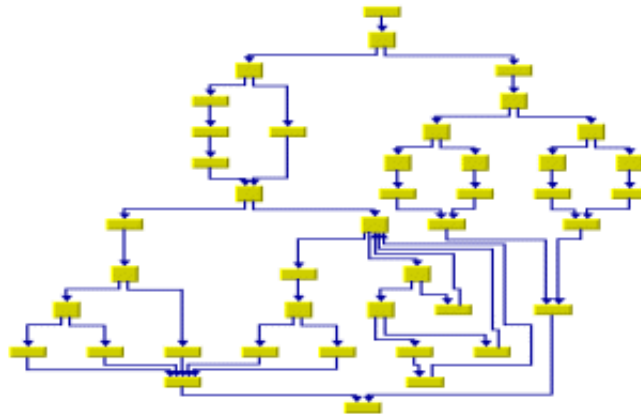


図26 階層レイアウトの例

前提となるパッケージ: *Foundation, Studio, Manager, Grapher*

---

## IBM ILOG Views のセットアップ

IBM® ILOG® Views を起動する前に適切なディレクトリにインストールして、マニュアルで説明されているように製品に付属の IBM ILOG License Manager (ILM) を設定する必要があります。

- ◆ 参照：IBM ILOG Views ディスク・スペースおよびメモリの必須要件
- ◆ インストール時に作成されるディレクトリは、ディレクトリ構造に一覧化されています。

### システム環境

IBM ILOG Views を使用する前に、プラットフォームに応じていくつかのセットアップの段取りを行う必要があります。各セクションは独立しているため、該当するセクションに直接進んでください。

- ◆ UNIX 上でのセットアップ
- ◆ Windows 上でのセットアップ

Microsoft® Windows® システムをご利用の場合、以下の補足のセットアップ情報も参照するようにしてください。

- ◆ Windows でのコンパイルおよびリンクのオプション

IBM ILOG Views を使用できる最新のプラットフォームのリストについては、この製品に同梱されている README ファイルを参照してください。

---

### ディスク・スペースおよびメモリ

IBM ILOG Views および IBM ILOG Views アプリケーション実行に必要なディスク・スペースとメモリは次のとおりです。

- ◆ **ディスク・スペース**：要件はインストールするコンポーネントによって異なります。単一プラットフォームにすべてのオプションを含めて完全インストールを行うには、Microsoft® Windows® システムで約 450 MB、UNIX® で約 350 MB 必要です。

クライアントの完全なアプリケーションに必要なディスク・スペースの要件は、通常、約 10 MB です。

- ◆ **メモリ**：IBM ILOG Views アプリケーションの実行に必要なメモリは、通常 16 MB 未満です。

システムには、オペレーティング・システム、IBM ILOG Views アプリケーション、およびそのデータ、さらにコンパイラ（使用する場合）をサポートする十分なメモリが必要です。IBM ILOG Views を実行するために追加メモリも必要です。

**メモ**：IBM ILOG Views ライブラリは、コンパイルされて x486 Intel プロセッサ上で最適化されます。これらのライブラリは、Ix486 以降のプロセッサで動作します。

---

## ディレクトリ構造

IBM ILOG Views をマシンにインストールすると、複数のディレクトリが作成されます。これらの一部には専用の README ファイルが付属していますので、目を通すようにしてください。作成される主なディレクトリは次のとおりです。

- ◆ **bin** およびそのサブディレクトリ。特定のエディタにソースとデータを提供します。内容はインストールするパッケージによって異なります。
- ◆ **data** およびそのサブディレクトリ。パネル説明ファイル(サフィックス: .ilv)とアイコンを提供します。これらは、提供される IBM ILOG Views のサンプルとエディタで使用されます。変更はできません。
- ◆ **doc** およびそのサブディレクトリ。IBM ILOG Views 起動ソースを提供します。
- ◆ **include** およびそのサブディレクトリ。すべての IBM ILOG Views クラスのヘッダ・ファイルを提供します。
- ◆ **lib** およびそのサブディレクトリ。IBM ILOG Views ライブラリが含まれます。
- ◆ **samples** およびそのサブディレクトリ。IBM ILOG Views が提供するクラスの特典部分を表示できるようにするサンプル・コーディングを提供します。
- ◆ **src** いくつかの IBM ILOG Views ソース・ファイルを含み、その一部は本書で説明されています。
- ◆ **studio** IBM ILOG Views エディタの構築と起動に必要なファイルを含みます。
- ◆ **tools** とそのサブディレクトリ。  
IBM ILOG Views クラスの拡張であるツールにソース・ファイルを提供しますが、ライブラリには含まれません(詳細については、README を参照してください)。これらのツールは、すべてのプラットフォーム上で実行できます。nbm は Microsoft? Windows? システム専用です。

すべての主要なディレクトリ階層で、特定のターゲット・システムと実行可能ファイル用の <system> サブディレクトリがあります。この構成によって、編成全体の共用が保証され、同じマシン上に異なるコンパイラ・システムやオペレーティング・システムをインストールできます。

---

## UNIX 上でのセットアップ

ここでは、UNIX® 上で IBM ILOG Views を構成する方法を説明します。次のインストール情報を用意しておいてください。

- ◆ **インストール・ディレクトリ**。IBM ILOG Views コンポーネントは通常、/usr/ilog/ ディレクトリにインストールします。これには、/usr/ilog/viewsXX (XX はバージョン番号を示す)が含まれます。IBM ILOG Views を別のディレクトリにインストールする場合は、後続のデフォルト・ディレクトリへの参照を実際のインストール・ディレクトリと置き換えてください。

- ◆ **システム名。**<system> ディレクトリ名は、製品に同梱されている README ファイルに記載されています。たとえば、x86\_RHEL4\_3.4 や hp64\_11\_3.73 です。以下の手順では、<system> を実際のシステム名と置き換えてください。

UNIX 上で IBM ILOG Views を構成するには、次の手順に従います。

1. ILVHOME 変数を設定します。

シェル変数 ILVHOME は IBM ILOG Views インストール・ディレクトリに設定する必要があります。

- ◆ sh またはそれから派生した ksh や bash を使用する場合は、次のように入力します。

```
$ ILVHOME=/usr/ilog/viewsXX
$ export ILVHOME
```

- ◆ csh や tcsh を使用する場合は、次のように入力します。

```
% setenv ILVHOME /usr/ilog/viewsXX
```

2. 共有ライブラリ・パス変数を設定します。

ほとんどのオペレーティング・システムでは、ダイナミック・ライブラリを含んだディレクトリは環境変数に追加する必要があります。HP-UX 上では SHLIB\_PATH、AIX® 上では LIBPATH、その他のシステム上では LD\_LIBRARY\_PATH を追加します。

- ◆ sh またはそれから派生したものを使用する場合は、次のように入力します。

```
$ LD_LIBRARY_PATH=$ILVHOME/lib/<system>:$LD_LIBRARY_PATH
$ export LD_LIBRARY_PATH
```

- ◆ csh または tcsh を使用する場で、変数 LD\_LIBRARY\_PATH をまだ定義していない場合は、次のように入力します。

```
% setenv LD_LIBRARY_PATH $ILVHOME/lib/<system>
```

- ◆ csh または tcsh を使用する場で、変数 LD\_LIBRARY\_PATH をすでに定義している場合は、次のように入力します。

```
% setenv LD_LIBRARY_PATH $ILVHOME/lib/<system>:$LD_LIBRARY_PATH
```

この手順は、上記コマンドをシェル始動ファイルへ格納すると自動化できます (sh を使用する場合は .profile、csh を使用する場合は .login)。

---

## Windows 上でのセットアップ

ここでは、Microsoft® Windows® システム上で IBM ILOG Views を構成する方法を説明します。次のインストール情報を用意しておいてください。

- ◆ **インストール・ディレクトリ。**IBM ILOG Views コンポーネントは通常、C:\ILOG\ディレクトリにインストールします。これには、C:\ILOG\VIEWSEXX (XX はバージョン番号を示す) が含まれます。IBM ILOG Views を別のディレクトリにインストールする場合は、後続のデフォルト・ディレクトリへの参照を実際のインストール・ディレクトリと置き換えてください。
- ◆ **システム名。**<SYSTEM> ディレクトリ名は、製品に同梱されている README ファイルに記載されています。この名前は使用するポートの名前です。たとえば、Windows XP または Vista を実行している x86 プロセッサ・マシン上で Visual Studio 2008 を使用する場合は、システム名は x86\_.net2008\_9.0 となります。
- ◆ **サブシステム名。**IBM ILOG Views ライブラリは、Windows のバージョンによって異なります。これは、IBM ILOG Views ライブラリをコンパイルする際に使用したコンパイル・オプションが異なるためです。たとえば、ライブラリが静的 (.lib) か動的 (.dll) かによる相違などです。これらの異なるライブラリ・タイプは LIB\<<SYSTEM> のサブディレクトリ内にあり、<SUBSYSTEM> の名前で参照されます。使用可能なそれぞれのバージョン (x86\_.net2003\_7.1, x86\_.net2005\_8.0 および x86\_.net2008\_9.0) には、特別なコンパイラ・フラグが必要です。コンパイラ・フラグの詳細については、*Microsoft Visual C++ .NET 2003, 2005 または 2008* によるコンパイルとリンクを参照してください。

以下の手順では、コマンド名およびディレクトリ名での文字列 <SYSTEM> および <SUBSYSTEM> を実際のシステム名およびサブシステム名と置き換えてください。

Windows 上で IBM ILOG Views を構成するには、次の手順に従います。

1. ILVHOME 変数を IBM ILOG Views インストール・ディレクトリに設定します。

- ◆ Windows XP または Vista の場合：

[コントロール・パネル]を開き、[パフォーマンスとメンテナンス] (Vista の場合は [システムとメンテナンス]) > [システム] > [詳細設定] タブの順に選択して [環境変数] をクリックします。ダイアログボックスが表示されるため、ここで新しい変数を定義することができます。このダイアログボックスを使用して、次のように ILVHOME 変数を定義します。

```
C:\ILOG\VIEWSEXX
```

ダイアログボックスは再使用するため、閉じないでください。

あるいは、次の行が VIEWS.INI ファイルに自動的に追加されます。

```
[Ilog Views]
home=C:\ILOG\VIEWSEXX
```

このファイルは、Windows NT、2000、または Me がインストールされたディレクトリに格納されています。

2. PATH 変数を設定します。

IBM ILOG Views DLL を含むディレクトリを PATH 変数に追加する必要があります。そのためには、DLL をパスのディレクトリへコピーするか、次のようにします。

- ◆ Windows XP、または Vista の場合は、[ システム ] ダイアログボックスを使用して、PATH 変数を次のように定義します。

```
C:\ILOG\VIEWSXX\LIB\<system>\<subsystem>;%PATH%
```

[ 適用 ]、次に [OK] をクリックして変更内容を有効にします。

3. 変更内容を有効にするために、コンピュータを再起動します。

---

## 実行要件

製品がインストールされたディレクトリに環境変数 ILVHOME を設定し、提供されたバイナリがすべて正しく作動するようにしてください。通常は、以下のように設定します。

### (csh/tcsh):

```
% setenv ILVHOME /usr/ilog/viewsXX
```

### (sh/bash):

```
$ ILVHOME=/usr/ilog/viewsXX  
$ export ILVHOME
```

### (Windows Server 2003/XP/Vista):

```
C:\> set ILVHOME=C:\ILOG\VIEWSXX
```

IBM® ILOG® Views のインストール場所については、システム管理者にご確認ください。



## コンパイラ

IBM ILOG Views には、ディレクトリに格納されるライブラリのセットが同梱されていますが、ディレクトリは、ご使用のマシン、オペレーティング・システム、使用するコンパイラの種類によって異なります。

ディレクトリは、次のいずれかになります。

システム名	ハードウェア	オペレーティング・システム	コンパイラ
alpha_5.1_6.5	HP/Compaq (DEC) Alpha	Tru64 UNIX (OSF1) V5.1 以降	C++ 6.5 以降
hp32_11_3.73	HP 9700	HP-UX 11.11 以降	-mt -AA オプション使用の HP ANSI C++ A.03.73 以降
hp64_11_3.73	HP 9700	HP-UX 11.11 以降	64 ビット・モード (+DA2.0W) / -mt -AA オプション使用の HP ANSI C++ A.03.73 以降
ia64_hpux11_6.17	IA-64	HP-UX 11i v3	HP C++/ANSI A.06.17 (64 bit モード)
ia64-32_hpux11_6.17	IA-64	HP-UX 11i v3	HP C++/ANSI A.06.17
x86_RHEL4.0_3.4	x86	RedHat Enterprise Linux 4.0 (linux 2.4、glibc 2.3)	gcc3.4
x86-64_RHEL4.0_3.4	x86-64	RedHat Enterprise Linux 4.0 (linux 2.4、glibc 2.3)	gcc3.4
x86_sles10.0_4.1	x86	SUSE Linux Enterprise Server 10 (linux 2.6、glibc 2.4)	gcc4.1
x64_solaris10_11	x86-64	Solaris 10	Sun C++ 5.8 (aka Sun One Studio 11) (64 bit モード)
x86_solaris10_11	x86	Solaris 10	Sun C++ 5.8 (aka Sun One Studio 11)
x86_.net2003_7.1	x86	Windows Server 2003/XP/Vista	Microsoft Visual C++ .NET 2003 (7.1)
x86_.net2005_8.0	x86	Windows Server 2003/XP/Vista	Microsoft Visual C++ .NET 2005 (8.0)

システム名	ハードウェア	オペレーティング・システム	コンパイラ
x86_.net2008_9.0	x86	Windows 2000/Server 2003/XP/Vista	Microsoft Visual C++ .NET 2008 (9.0)
x64_.net2008_9.0	x64	Windows (64) Server 2003/XP/Vista	Microsoft Visual C++ .NET 2008 (9.0 - 64 bits)
rs6000_5.1_6.0	RS6000	AIX 5.1 以降	標準 C++ ストリーム (-DIL_STD) 使用の Visual Age 6.0
power32_aix5.2_7.0	PowerPC	AIX 5.2 以降	標準 C++ ストリーム (-DIL_STD) 使用の Visual Age 7.0
ultrasparc32_8_6.2	Sun Ultra Sparc	Solaris 2.8 以降	Sun C++ 5.3 (Forte 6.2) 以降
ultrasparc64_8_6.2	Sun Ultra Sparc	Solaris 2.8 以降	64 ビット・モードの Sun C++ 5.3 (Forte 6.2) 以降 (-xtarget=ultra -xarch=v9)
ultrasparc32_10_11	Sun Ultra Sparc	Solaris 10	Sun C++ 5.8 (aka Sun One Studio 11)
ultrasparc64_10_11	Sun Ultra Sparc	Solaris 10	64bit モードの Sun C++ 5.8 (aka Sun One Studio 11) (-xtarget=ultra -xarch=v9)

以下のテキストでは、OS を <system> と呼びます。独自のインストール要件に合わせるため、提供された **make** ファイルを変更する必要があるかもしれません。

- ◆ VIEWSDIR は、製品の開梱を行ったディレクトリ (通常、UNIX の場合は /usr/ilog/viewsXX、Windows の場合は c:\ilog\viewsxx になり、XX は ILOG Views の現在のバージョンを表します) に設定します。
- ◆ XINC および XLIBS は、x11 があり、ライブラリのディレクトリが配置されているディレクトリに設定します (Unix プラットフォームの場合のみ)。
- ◆ MINC および MLIBS は、Motif があり、ライブラリのディレクトリが配置されているディレクトリに設定します (Unix プラットフォームの場合のみ)。

---

### Windows でのコンパイルおよびリンクのオプション

ライブラリが提供する実例のコンパイルは、次のいずれかを使用して行います。

- ◆ Visual Studio

Visual Studio 用の複数のソリューション (.sln) とプロジェクト (.vcproj) ファイルがコンパイル用に提供されています。

◆ コマンド・ライン・ツール

nmake (Microsoft Visual C++) コマンドで使用できる多くの Makefile が用意されています。

デフォルト定義ファイル VIEWS.DEF は各 <SUBSYSTEM> ディレクトリで提供されています。

**注記:** <SUBSYSTEM> ディレクトリは、システム固有のディレクトリに配置されています(x86\_.net2003\_7.1, x86\_.net2005\_8.0 または x86\_.net2008\_9.0)。  
<SUBSYSTEM> の値については、使用するコンパイラに該当するセクションを参照してください。

ライブラリ作成に使う構造体とユニオン・メンバのパック・アライメント値は、*Microsoft Visual C++ .NET 2003*、*2005* または *2008* によるコンパイルとリンクで説明されているコンパイラのデフォルト値です。

**Microsoft Visual C++ .NET 2003、2005 または 2008 によるコンパイルとリンク**

このセクションでは、Microsoft Visual C++ 2003 またはそれ以降でソースファイルをコンパイルするための要件を説明します。

ほとんどのバイナリ・ファイルは IBM ILOG Views のライブラリで提供されているデータ・ファイルに基づいているため、プログラムを別のプラットフォームにインストールする際には、これらのデータ・ファイルと .EXE ファイルをコピーする必要があります。また、実行可能ファイルになるリソースの中にこれらのデータ・ファイルを配置することもできます。

構造体とユニオン・メンバのパック・アライメントは 8 バイトです。

ライブラリは標準 C++ ライブラリで、次の形式で提供されています。

- ◆ DLL\_MDA: サブシステム DLL\_MDA がある動的マルチスレッド実行時ライブラリ
- ◆ STAT\_MDA: サブシステム STAT\_MDA がある静的マルチスレッド実行時ライブラリ
- ◆ STAT\_MTA: サブシステム STAT\_MTA がある静的マルチスレッド実行時ライブラリ
- ◆ STAT\_STA: サブシステム STAT\_STA (Microsoft Visual C++ .NET 2003 のみ) がある静的単一スレッド実行時ライブラリ

使用するコンパイル・フラッグは次のとおりです。

表3 コンパイル・フラッグ

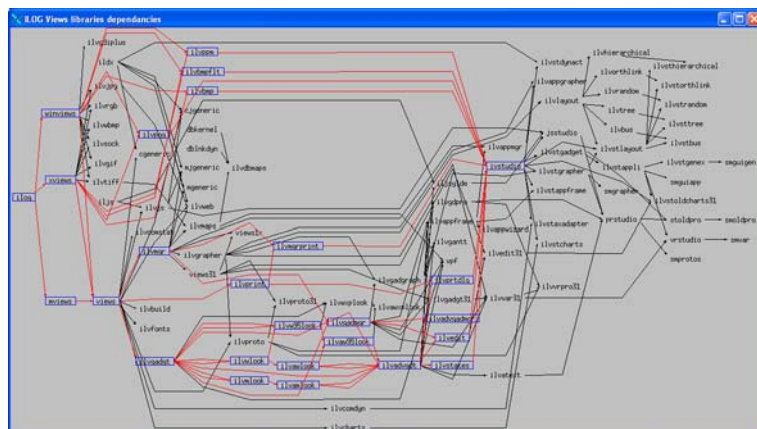
	Visual Studio .NET 2003	Visual Studio .NET 2005 および 2008
DLL_MDA	/MD /GX /GR /DILVSTD /DILVDLL	/MD /EHsc /GR /D_CRT_SECURE_NO_DEPRECATED / DILVSTD /DILVDLL
STAT_MDA	/MD /GX /GR /DILVSTD	/MD /EHsc /GR /D_CRT_SECURE_NO_DEPRECATED / DILVSTD
STAT_MTA	/MT /GX /GR /DILVSTD	/MT /EHsc /GR /D_CRT_SECURE_NO_DEPRECATED / DILVSTD
STAT_STA	/ML /GX /GR /DILVSTD	[ 利用不可 ]

**メモ:** ファイルをデバッグ・モードでコンパイルする場合は、実行時ライブラリの非デバッグ・バージョンを使用する必要があります。さらにマクロ `_DEBUG` も削除する必要があります (IBM ILOG Views は非デバッグ実行時ライブラリのみを使用するためです)。

### ライブラリの従属性

アプリケーションに含まれているライブラリは、他のライブラリに依存する場合があります。それは宣言にも含まれていなければなりません。

`<installdir>\samples\layout\libdeps\index.html` にある `libdeps` サンプルは、これらの従属性を図示するとともに、**Graph Layout** パッケージを使用して、リンクされたデータの解釈を単純化する方法例を示したものです。



**警告:** このサンプルは、*Graph Layout* パッケージのライセンスを有している場合にのみ実行できます。

また、次の表は、アプリケーションにリンクするライブラリの従属性を識別する簡単な方法を示しています (この表は、第 1 レベルの従属性のみを表しています。使用しているライブラリが依存するライブラリ自体が、他のライブラリに依存していないことを確認してください)。

表 4 ライブラリの従属性

ライブラリ	従属性
$\langle displaylib \rangle^a$	ilog
views	$\langle displaylib \rangle$
cgeneric	ildx views
cjgeneric	ilvjs ildx views
mgeneric	ildx ilvmgr
mjgeneric	ilvjs ildx ilvmgr
iljsgide	ilvadvgdgdt iljs
ilvadvgdgdt	ilvgadgdt
ilvadvgdgdt	ilvgadgdt ilvamlook ilvawlook ilvaw95look
ilvadvgadmgr	ilvadvgdgdt ilvgadmgr
ilvamlook	ilvadvgdgdt ilvmlook

表4 ライブラリの従属性(続き)

ライブラリ	従属性
ilvappframe	ilvadvgdt ilvgadmgr ilvgadgraph
ilvappgrapher	ilvappmgr ilvgrapher
ilvappmgr	ilvappframe ilvmgr
ilvappwizard	ilvappframe
ilvatext	ilvadvgdt
ilvaw95look	ilvawlook ilvw95look
ilvawlook	ilvadvgdt ilvwlook
ilvawxplook	ilvaw95look ilvwxplook
ilvbmp	xviews
ilvbmp	winviews
ilvbmpflt	xviews
ilvbmpflt	winviews
ilvbuild	views
ilvcomstat	views
ilvcomdyn	views
ilvedit	ilvadvgdt ilvgadmgr
ilvfonts	views
ilvgadgt	views
ilvgadgt	views ilvmllook ilvwlook ilvw95look
ilvgadgraph	ilvgrapher ilvgadmgr
ilvgadmgr	ilvgadgt ilvmgr
ilvgantt	ilvgadgraph ilvadvgdt
ilvgdiplus	winviews
ilvgdpro	ilvproto ilvgadgraph
ilvgif	xviews

表4 ライブラリの従属性(続き)

ライブラリ	従属性
ilvgif	winviews
ilvgrapher	ilvmgr
ilvjpg	xviews
ilvjpg	winviews
ilvjs	iljs views
ilvmgr	views
ilvmgrprint	ilvprint ilvmgr
ilvmlook	ilvgadgt
ilvpng	xviews
ilvpng	winviews
ilvppm	xviews
ilvppm	winviews
ilvprint	views
ilvproto	views ilvgrapher
ilvprtdlg	ilvprint ilvadvgdt
ilvrgb	xviews
ilvrgb	winviews
ilvstappframe	ilvappframe ivstudio
ilvstappli	ilvstgadget
ilvstates	ilvadvgdt
ilvstaxadapter	ivstudio ilvcomdyn
ilvstbus	ilvstlayout ilvbus
ilvstcharts	ivstudio ilvcharts
ilvstdynact	ildx ivstudio
ilvstgadget	ivstudio

表4 ライブラリの従属性(続き)

ライブラリ	従属性
ilvstgenex	ilvstappli
ilvstgrapher	ivstudio ilvgadgraph
ilvsthierarchical	ilvstlayout ilvhierarchical
ilvstlayout	ilvlayout ivstudio ilvstgrapher
ilvstorthlink	ilvstlayout ilvorthlink
ilvstrandom	ilvstlayout ilvrandom
ilvsttree	ilvstlayout ilvtree
ilvtiff	xviews
ilvtiff	winviews
ilvw95look	ilvgadgt ilvwlook
ilwbitmap	xviews
ilwbitmap	winviews
ilvwlook	ilvgadgt
ilvwxplook	ilvgadgt ilvw95look
ivstudio	ilvedit ilvstates ilvadvgadmgr ilvpng ilvbitmap ilvppm ilvbmpflt
jsstudio	ivstudio ilvjs iljsgide
mviews	ilog
prstudio	ilvgdpro jsstudio ilvatext ilvstgrapher
smgrapher	ilvstgrapher
smguiapp	ilvstappli
smguigen	ilvstgenex
smoldpro	stoldpro
smprotos	prstudio
smvar	vrstudio
stoldpro	prstudio ilvstgadget



表4 ライブラリの従属性(続き)

ライブラリ	従属性
views	winviews
views	xviews
vrstudio	ilvstappli ilvvar31
winviews	ilog
xviews	ilog
ilvcharts	views
ilvbus	ilvlayout
ilvhierarchical	ilvlayout
ilvlayout	ilvgrapher
ilvorthlink	ilvlayout
ilvrandom	ilvlayout
ilvtree	ilvlayout
iljs	(なし)
views31	ilvmgr
ilvgadgt31	ilvgadmgr ilvadvgdt views31
ilvedit31	ilvedit views31
ilvproto31	ilvproto views31
ilvvar31	ilvgadmgr ilvgrapher ilvadvgdt ilvgadgt31
ilvvrpro31	ilvgdpro ilvproto ilvvar31
ilvstoldcharts31	ilvstappli ivstudio ilvedit31
views1x	ilvmgr views31
ilvmaps	ilvmgr ilvtiff
ilvdbmaps	dblnkdyn dbkernel ilvmaps
views	mviews

<sup>a</sup> <displaylib> は、ライブラリに対応して、ウィンドウ・システムとの連動を確実にします。Windows の場合は winviews、UNIX/X11 の場合は xviews、UNIX/X11/Motif の場合は mviews になります。

---

## Motif および共有ライブラリ

libmviews (Motif ベース) は、共有ライブラリ・フォーマットでは使用できなくなりました。

バージョン 4.0 以降、IBM® ILOG® Views で提供されるすべての共有ライブラリは libxviews を使用するようになり、libmviews とは互換性がなくなりました。libmviews はスタティック・ライブラリとしてのみ提供されており、その他の IBM® ILOG® Views ライブラリのスタティック・バージョンとして使用することはできません。

---

## ライブラリ構築情報 (Unix プラットフォームのみ)

アプリケーション実行中に問題が生じた場合は、IBM ILOG Views ルート・ディレクトリにある ilvversion.sh と呼ばれるシェル・スクリプトを使って、ライブラリ構築に使用したプラットフォームに関する情報を取得することができます。

lib の特定スタティック・サブプラットフォーム・サブディレクトリに移動し、以下を入力します。

```
../../../../ilvversion.sh
```

以下のフォームをもつ 2 行が印刷されます (これらの行は位置が変わることもあります)。

**IlvVersion: X.X**

```
IlvBuild: OS - platform - compiler - display revision
```

シェル・スクリプトによって文字列 'IlvBuild' および 'IlvVersion' がライブラリ・ファイル libxviews.a で特定され、結果が印刷されます。

ディレクトリ ILVHOME に複数のプラットフォームをインストールしている場合は、各プラットフォームを次のようにしてチェックすることができます。

```
$ cd $ILVHOME
$ ./ilvversion.sh platform
```

ここで、プラットフォームは IBM ILOG Views ライブラリのシステム識別子です (これは、hp32\_11\_3.73、power32\_aix5.2\_7.0 など、いずれかの <system> になります)。

(ただし、Oracle 8.x および Open Ingres はサポートしません。)

---

## ライセンス

Unix にある `$ILVHOME/access.ilm` ファイルや Windows にある `%ILVHOME%\access.ilm` ファイルは、IBM® ILOG® Views の評価版コピーになります。

IBM ILOG Views アプリケーションを実行するには、次のようにして、このパス名に環境変数 `ILOG_LICENSE_FILE` を設定する必要があります。

### (csh/tcsh):

```
% setenv ILOG_LICENSE_FILE $ILVHOME/access.ilm
```

### (sh/bash):

```
$ ILOG_LICENSE_FILE=$ILVHOME/access.ilm  
$ export ILOG_LICENSE_FILE
```

### (Windows Server 2003/XP/Vista):

```
C:\> set ILOG_LICENSE_FILE=%ILVHOME%\access.ilm
```

---

## Unix ユーザの方へ

Unix® プラットフォームでアプリケーションを IBM® ILOG® Views とリンクする場合は、純粋な Motif? コードを含めるかどうかを決定する必要があります。その場合は、`libmviews` とリンクさせる必要があります。

アプリケーションを純粋な Xlib アプリケーションとする場合は、`libxviews` とリンクさせます。

この両方とリンクすることはできません。必ず、どちらか一方とリンクするようにしてください。ライブラリ "views" は、すべてのアプリケーションで必要になる可能性があります。

---

## Windows Server 2003/XP/Vista ユーザの方へ

ライブラリは複数の Windows® バージョンで提供されており、すべて `lib\<system>` のサブディレクトリに配置されています。これは、実行時システム・ライブラリの異なるバージョンと DLL によるものです。各バージョンには、次に述べるように特定のコンパイラ・フラグが必要です。

◆ x86\_.net2003\_7.1:

- `stat_sta`: 新しい入出力ストリームを使った実行時スタティック・ライブラリがある単スレッド内のスタティック・ライブラリ。フラグ: `/GX /GR /ML /DIL_STD` (単スレッド)。
  - `stat_mta`: 新しい入出力ストリームを使った実行時スタティック・ライブラリがあるマルチスレッド内のスタティック・ライブラリ。フラグ: `/GX /GR /MT /DIL_STD` (マルチスレッド)。
  - `stat_mda`: 新しい入出力ストリームを使った実行時ダイナミック・ライブラリがあるマルチスレッド内のスタティック・ライブラリ。フラグ: `/GX /GR /MD /DIL_STD` (マルチスレッド DLL)。
  - `dll_mda`: 新しい入出力ストリームを使った実行時ダイナミック・ライブラリがあるマルチスレッド内のダイナミック・ライブラリ。フラグ: `/GX /GR /MD /DIL_STD /DILVDLL` (マルチスレッド DLL)。
- ◆ `x86_.net2005_8.0` `x86_.net2008_9.0` `x64_.net2008_9.0`:
- `stat_mta`: 新しい入出力ストリームを使った実行時スタティック・ライブラリがあるマルチスレッド内のスタティック・ライブラリ。フラグ: `/EHsc /GR /MT /DIL_STD` (マルチスレッド)。
  - `stat_mda`: 新しい入出力ストリームを使った実行時ダイナミック・ライブラリがあるマルチスレッド内のスタティック・ライブラリ。フラグ: `/EHsc /GR /MD /DIL_STD` (マルチスレッド DLL)。
  - `dll_mda`: 新しい入出力ストリームを使った実行時ダイナミック・ライブラリがあるマルチスレッド内のダイナミック・ライブラリ。フラグ: `/EHsc /GR /MD /DIL_STD /DILVDLL` (マルチスレッド DLL)。

**Notes:**

1. これらのモードで *IBM ILOG Script* を使う場合は、フラグ `/DILJSTDH` も使用する必要があります。
2. さらに、システム・ライブラリ `wsock32.lib` と `imm32.lib` にリンクする必要があります。

**Microsoft Visual C++ ユーザの方へ**

すべてのプロジェクトについて、`wsock32.lib` および `imm32.lib` がリンクのコメントラインで指定されていることを確認してください。

プログラムのデバッグをより簡単に行うには、同ファイルの [AutoExpand] セクションに次の行を追加します。

```
x86_.net2003_7.1, x86_.net2005_8.0, x86_.net2008_9.0 or x64_.net2008_9.0:
```

```

%INSTALLDIR%\Common7\Packages\Debugger\autoexp.dat

; from Ilog Views
IlvRect =x = <_orig._x,d> y = <_orig._y,d> width = <_w,u> height = <_h,u>
IlvPoint =x = <_x,d> y = <_y,d>
IlvTransformer =x11 = <_x11,g> x12 = <_x12,g> x21 = <_x21,g> x22 = <_x22,g> x0
= <_x0,g> y0 = <_y0,g>

```

以上の行によって、マウスを変数の上に置いた時に表示されるツールチップで(アドレスの代わりに)これらのクラスが定義するオブジェクトの値を表示できるようになります。

Visual C++ 6.0 を統合した開発環境を使って、デバッグ・モードでプロジェクトを構築すると、一連のリンク・エラーが発生することがあります。

デバッグ・モードで、アプリケーションを Visual C++ および IBM® ILOG® Views で提供されたライブラリとリンクできるようにするには、以下の処理を行います。

x86\_.net2003\_7.1, x86\_.net2005\_8.0, x86\_.net2008\_9.0 or x64\_.net2008\_9.0:

- Microsoft Visual Studio.NET で、[ソリューションエクスプローラ] ウィンドウからプロジェクトを選択します。
- 選択項目を右クリックして、[プロパティ] を選択します。
- [プロパティ ページ] ウィンドウで、[構成プロパティ]>[C/C++]>[プリプロセッサ] を選択します。
- [プリプロセッサ定義] フィールドで、"\_DEBUG" を削除します。
- 再コンパイルを行い、アプリケーションを再リンクします。

---

## 異なるバージョン

ライブラリは、Unix? の場合はディレクトリ lib/<system> のサブディレクトリに、Windows? (Server 2003/XP/Vista) の場合は lib\<system> に格納されています。

次のようなライブラリ・ファイル(スタティックおよび共有)があります:

Unix プラットフォーム:

- ◆ libxviews: IBM® ILOG® Views の純粋な Xlib コード
- ◆ libmviews: IBM ILOG Views の Motif 依存コード

Windows Server 2003/XP/Vista プラットフォーム:

- ◆ winviews.lib: IBM ILOG Views の Windows 依存コード

すべてのプラットフォーム (名前は、UNIX プラットフォームの場合は接頭辞 `lib` が、Windows プラットフォームの場合は拡張子 `.lib` が付いています):

◆ Foundation パッケージ

- `ilog`: ILOG ユーティリティ・クラス (配列、リストなど)
- `views`: コア・ライブラリ。Standard 2D グラフィックスを含む。
- `ilvgadgt`: コア Gadgets ライブラリ  
このライブラリは Foundation パッケージに同梱されていますが、"IBM ILOG Views" または "IBM ILOG Views Controls" のライセンスがある場合にのみ、使用、配布することができます。このライブラリには、次のクラスが含まれます。IlvMessageLabel、IlvFrame、IlvButton、IlvToggle、IlvColoredToggle、IlvComboBox、IlvTextField、IlvNumberField、IlvPasswordField、IlvPopupMenu、IlvToolBar、IlvMenuBar、IlvScrollBar、IlvSlider、IlvGadgetItem、IlvMenuItem、IlvGadgetContainer、IlvScrolledView、IlvDialog、IlvIMessageDialog、IlvIInformationDialog、IlvIErrorDialog、IlvIWarner、IlvIQuestionDialog、IlvGadgetContainerRectangle、IlvSCGadgetContainerRectangle、IlvSCViewRectangle。
- `ilvmllook`: Motif? ルック・アンド・フィール・ライブラリ。
- `ilvwlook`: Windows 3.11 ルック・アンド・フィール・ライブラリ
- `ilvw95look`: Windows 95 ルック・アンド・フィール・ライブラリ
- `ilvwxplook`: Windows XP ルック・アンド・フィール・ライブラリ (Windows のみ)
- `ilvprint`: 印刷サポート・ライブラリ
- `ilvbmpflt`: ビットマップ・フィルタ・ライブラリ
- `ilvbmp`: BMP ビットマップ・ストリーマ
- `ilvwbmp`: WBMP ビットマップ・ストリーマ
- `ilvpng`: PNG ビットマップ・ストリーマ
- `ilvjjpg`: JPG ビットマップ・ストリーマ
- `ilvppm`: PBM-PPM ビットマップ・ストリーマ
- `ilvtiff`: TIFF ビットマップ・ストリーマ
- `ilvrgb`: SGI RGB ビットマップ・ストリーマ
- `iljs`: IBM ILOG スクリプト・ライブラリ。
- `ilvjs`: IBM ILOG Views スクリプト実装。

- iljsgide: IBM ILOG スクリプト・デバッガ・インターフェース。
- ilvbuild: バイナリ用のリソース処理クラス
- ◆ **Gadgets パッケージ (Controls)**
  - ilvadvgdt: 高度な Gadgets ライブラリ (IlvMatrix、IlvTreeGadget など)
  - ilvadvgadmgr: 高度なガジェットを使った Manager クラス
  - ilvatext: IlvAnnotext およびその関連クラス
  - ilvedit: インスペクタ・クラス (色チューザ、フォント・チューザなど)
  - ilvstates: ステート・ライブラリ
  - ilvamlook: 高度な Motif ルック・アンド・フィール・ライブラリ
  - ilvawlook: 高度な Windows 3.11 ルック・アンド・フィール・ライブラリ
  - ilvaw95look: 高度な Windows 95 ルック・アンド・フィール・ライブラリ
  - ilvawxplook: 高度な Windows XP ルック・アンド・フィール・ライブラリ (Windows のみ)
  - ilvprtdlg: 印刷ダイアログ・ライブラリ
- ◆ **Application Framework パッケージ (Controls)**
  - ilvappframe: Application Framework ライブラリ
  - ilvappmgr: Manager パッケージを使用した Application Framework クラス (IlvDvManagerDocument および関連クラス)。
  - ilvappgrapher: Grapher パッケージを使用した Application Framework クラス (IlvDvGrapherDocument および関連クラス)。
  - ilvappwizard: ウィザード用の Application Framework ライブラリ
- ◆ **Manager パッケージ (2D Standard)**
  - ilvmgr: Manager ライブラリ (IlvManager および関連クラス)。
  - ilvgadmgr: ガジェットを使った Manager クラス (IlvGadgetManager および関連クラス)。
  - ilvmgrprint: Manager 印刷サポート・ライブラリ
- ◆ **Data Access アドオン**
  - dataaccess: コア Data Access ライブラリ (ILOG InForm 3.0 では 'inform')。
  - dbaccess: リレーショナル・データベース・ライブラリ (ILOG InForm 3.0 では 'dbinform')。
  - dbchart: Charts ベースの Data Access クラス・ライブラリ

- dbgadget: Gadgets ベースの Data Access クラス・ライブラリ
- dbgantt: Gantt Chart ベースの Data Access クラス・ライブラリ
- dbgraphe: Grapher ベースの Data Access クラス・ライブラリ
- dbsqlgad: リレーショナル・データベースの Gadgets クラス・ライブラリ
- ◆ Grapher パッケージ (Advanced 2D)
  - ilvgrapher: Grapher ライブラリ (IlvGrapher および関連クラス)。
  - ilvgadgraph: ガジェットを使った Grapher クラス (IlvSCGrapherRectangle)。
- ◆ Prototypes パッケージ (Advanced 2D)
  - ilvproto: Prototypes ベース・ライブラリ
  - ilvgdpro: ガジェットを使った Prototype クラス
- ◆ Charts アドオン
  - ilvcharts: Charts ライブラリ (IlvChartGraphic および関連クラス)。
- ◆ Graph Layout アドオン
  - ilvlayout: コア Graph Layout ライブラリ
  - ilvbus: バス・レイアウト・ライブラリ
  - ilvhierarchical: 階層レイアウト・ライブラリ
  - ilvorthlink: 直行リンク・レイアウト
  - ilvrandom: ランダム・レイアウト・ライブラリ
  - ilvtree: ツリー・レイアウト・ライブラリ
- ◆ Gantt アドオン
  - ilvgantt: Gantt Chart ライブラリ (IlvGanttChart および関連クラス)。
- ◆ Maps アドオン
  - ilvmaps: コア Maps ライブラリ
  - ilvdbmaps: データベース・ベースの Maps ライブラリ





## 索引

**A**

Application Framework パッケージの概要 **36**

**C**

C++

前提条件 **8**

Charts パッケージの概要 **45**

**D**

Data Access パッケージの概要 **49**

**F**

Foundation パッケージの概要 **30**

**G**

Gadgets パッケージの概要 **35**

Gantt Chart パッケージの概要 **43**

Graph Layout パッケージの概要 **50**

Grapher パッケージの概要 **41**

**I**

IBM ILOG Views

2D Graphics Professional 製品 **18**

2D Graphics Standard 製品 **18**

Application Framework パッケージの概要 **36**

Charts パッケージの概要 **45**

Controls 製品 **17**

Data Access パッケージの概要 **49**

Foundation パッケージの概要 **30**

Gadgets パッケージの概要 **35**

Gantt Chart パッケージの概要 **43**

Graph Layout パッケージの概要 **50**

Grapher パッケージの概要 **41**

Manager パッケージの概要 **39**

Maps パッケージの概要 **47**

Prototypes パッケージの概要 **42**

Studio パッケージの概要 **32**

オプション製品 **20**

基本製品 **17**

ご使用になる前に **52**

セットアップ **51**

ディレクトリ構造 **53**

プラットフォーム **53**

IBM ILOG Views 2D Graphics Professional **18**

IBM ILOG Views 2D Graphics Standard **18**

IBM ILOG Views Component Suite 製品 **15**

IBM ILOG Views Controls **17**

IBM ILOG Views Studio パッケージの概要 **32**

IBM ILOG Views のセットアップ **51**

UNIX 環境 **53**

Windows 環境 **54**

## M

Manager パッケージの概要 **39**

Maps パッケージの概要 **47**

## P

Prototypes パッケージの概要 **42**

## U

UNIX

IBM ILOG Views インストール **53**

## W

Windows

IBM ILOG Views インストール **54**

コンパイルおよびリンクのオプション **58**

## い

インストール

UNIX 環境 **53**

Windows 環境 **54**

インストール・ディレクトリ

UNIX **53**

Windows **55**

## さ

サブシステム名

Windows **55**

## し

システム名

UNIX **54**

Windows **55**

## せ

製品の概要 **14**

## て

ディスク・スペースの要件 **52**

ディレクトリ構造 **53**

## な

内容 **16**

## ひ

表記法 **8**

## ふ

プラットフォーム **53**

## ま

マニュアル

表記法 **8**

命名規則 **9**

## め

命名規則 **9**

メモリの要件 **52**