



IBM ILOG Views V5.3

リリース・ノート

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.

目次

前書き	リリース・ノートについて.....	5
第1章	リリース・ノート — バージョン 5.3.....	7
	ILOG Views の以前のリリースをお使いの方へ.....	7
	新しいポート.....	7
	製品アイコンへの変更.....	8
	使用されなくなったパッケージの除去.....	8
第2章	リリース・ノート — バージョン 5.2.....	9
	ILOG Views の以前のリリースをお使いの方へ.....	9
	テクニカル・サポート.....	9
	新しい機能.....	9
	コンパイル.....	10
	本バージョンではサポートされなくなったプラットフォーム.....	13
	バージョン 5.1 との非互換性.....	13
	実行要件.....	14
	ライブラリ構築情報 (Unix プラットフォームのみ).....	14
	内容.....	15
	ILOG Views Studio.....	15
	ディレクトリ.....	15
	ILOG Views Data Access: バージョン 5.2 との既知の非互換性.....	16

	ライセンス.....	16
	異なるバージョン	16
	共有ライブラリの使用.....	20
	Unix ユーザの方へ.....	20
	Motif および共有ライブラリに関する重要事項.....	20
	Windows Server 2003/XP/Vista ユーザの方へ	21
	Microsoft Visual C++ ユーザの方へ.....	22
第 3 章	リリース・ノート - バージョン 5.1	23
	ILOG Views の以前のリリースをお使いの方へ.....	23
	テクニカル・サポート.....	23
	コンパイル.....	23
	本バージョンではサポートされなくなったプラットフォーム	25
	実行要件.....	26
	ライブラリ構築情報 (Unix プラットフォームのみ).....	27
	内容	27
	ILOG Views Studio.....	27
	ディレクトリ	27
	ILOG Views Data Access: バージョン 5.0 との既知の非互換性.....	28
	ライセンス.....	28
	異なるバージョン	29
	共有ライブラリの使用.....	32
	Unix ユーザの方へ.....	32
	Motif および共有ライブラリに関する重要事項.....	33
	Windows 95/98/Me および NT/2000/XP ユーザの方へ	33
	Windows 95 の既知の問題.....	34
	Microsoft Visual C++ ユーザの方へ.....	35
第 4 章	リリース・ノート - バージョン 5.0.1.....	37
	バージョン 5.0 以降行われた変更.....	37
	コンパイルの問題	38
	本バージョンではサポートされなくなったプラットフォーム	40

	実行要件.....	40
	ライブラリ構築情報 (Unix プラットフォームのみ)	41
	ILOG Views Studio に行われた変更.....	41
	ライセンス.....	42
	異なるバージョン	42
	共有ライブラリの使用.....	43
	バージョン 5.0 以降修正されたパッチ	49
第 5 章	リリース・ノート - バージョン 5.0.....	85
	新しい機能.....	85
	Foundation の機能	86
	Studio の機能	90
	Gadgets の機能	90
	IlvComboBox and IlvScrolledComboBox	91
	Application Framework の機能	91
	Manager の機能	91
	Data Access の機能	92
	Charts の機能	92
	Gantt の機能	92
	Maps の機能	93
	コードの変更.....	95
	ILOG Views ファンデーション・クラス	96
	ILOG Views Charts クラス	98
	ILOG Views Gadgets クラス	99
	ILOG Views Gantt Chart クラス	100
	ILOG Views Maps クラス	102
第 6 章	リリース・ノート - バージョン 4.0.2.....	105
	新しい機能.....	105
	Web 配布	105
	Prototypes	106
	システムの変更	108

プラットフォームの調整	108
Unix ユーザの方へ : Motif および共有ライブラリ	109
索引	111

リリース・ノートについて

このセクションには、次の IBM® ILOG® Views リリース・ノートが含まれています。

- ◆ リリース・ノート - バージョン 5.3
- ◆ リリース・ノート - バージョン 5.2
- ◆ リリース・ノート - バージョン 5.1
- ◆ リリース・ノート - バージョン 5.0.1
- ◆ バージョン 5.0 以降修正されたパッチ
- ◆ リリース・ノート - バージョン 5.0
- ◆ リリース・ノート - バージョン 4.0.2

リリース・ノート – バージョン 5.3

ILOG Views の以前のリリースをお使いの方へ

以前のバージョンと本 IBM® ILOG® Views バージョンの既知の非互換性については、インストール・フォルダ内にある COMPAT.TXT ファイルにすべて記述されています。

古いアプリケーションのコンパイルや実行の問題がある場合は、同ファイルを参照してください。

新しいポート

このバージョンでは、次のポートをサポートするようになりました。

- ◆ ia64_hpux11_6.17: このポートは、HP UX 11.23 またはそれ以降のバージョンで実行されている Itanium® プラットフォームで使用します。使用するコンパイラは、64 ビット・モードの C++ バージョン 6.17 です。
- ◆ ia64-32_hpux11_6.17: このポートは、HP UX 11.23 またはそれ以降のバージョンで実行されている Itanium プラットフォームで使用します。使用するコンパイラは、32 ビット・モードの C++ 6.17 です。

- ◆ x64_solaris10_11: このポートは、SunTM SolarisTM10 またはそれ以降のバージョンで実行されている x86-64 プラットフォーム (AMD 64 と呼ばれている) で使用します。使用するコンパイラは、64 ビット・モードの Sun Studio 11 です。
- ◆ x86_solaris10_11: このポートは、Sun Solaris 10 またはそれ以降のバージョンで実行されている x86 プラットフォームで使用します。使用するコンパイラは、32 ビット・モードの Sun Studio 11 です。
- ◆ power64_aix5.2_7.0: このポートは、AIX 5.2 で実行されている PowerPC プラットフォームで使用します。使用するコンパイラは、標準 C++ streams (-DIL_STD) を使用する 64 ビット・モードの Visual Age 7.0 (オプション -q64) で使用します。

これらの新しいポートでは、IBM® Informix® と Oracle® のみがサポートされていることに留意してください。

製品アイコンへの変更

ほとんどの製品のアイコンが、IBM® のルック・アンド・フィール・ガイドラインに準拠するために変更されました。

使用されなくなったパッケージの除去

Web 配布の IBM® ILOG® Views コンポーネント・スイート・パッケージが使用されなくなったため、本リリースから除去されました。

リリース・ノート – バージョン 5.2

ILOG Views の以前のリリースをお使いの方へ

以前のバージョンと ILOG Views 本バージョンの既知の非互換性については、すべて COMPAT.TXT ファイルに記述されています。これは、本 README ファイルと同じディレクトリにあります。

古いアプリケーションのコンパイルや実行の問題がある場合は、同ファイルを参照してください。

テクニカル・サポート

ILOG Views ライブラリ使用中に問題が生じたり、マニュアルを読んでも問題が解決できないという場合は、アイログのサポート・チームまでバグ・レポートをお送りください。その際に使うテンプレートは、ファイル BUGREPORT、Windows プラットフォームをご利用の場合は bugreprt.txt に配置されています。

新しい機能

- ◆ S57 地図形式のサポート

- ILOG Views Maps で、国際水路機関 (IHO) が発行した基準である S57 海里マップ用の地図形式がサポートされるようになりました。
- ◆ 半透明の PNG のサポート
 - 半透明のピクセルが含まれている PNG イメージがサポートされるようになりました。
- ◆ 容易になったスプライン・オブジェクトとのインタラクション
 - スプライン・オブジェクトの作成およびインタラクションを簡単にする新しいインタラクタが追加されました。
- ◆ グラフィック・オブジェクト上で異なるレベルの透明度をシミュレートするための新しいフィルタ効果
- ◆ CTRL + マウス・ホイールによる Studio でのズーム・イン、ズーム・アウトのサポート
- ◆ スピンボックスでのマウス・ホイールによる上下移動のサポート
- ◆ CTRL-Tab キーによる Tab- ペインの回転
- ◆ オブジェクトの作成インタラクタを中断するための ESC キーの一般使用
- ◆ Studio の GUI の再構成。新しいアイコンおよびダイアログ・ボックス
- ◆ Aero L&F による Windows Vista のより良いサポート
 - ILOG Views で、Windows Vista の新しいビジュアル効果とロック・アンド・フィール要素がサポートされるようになりました。
- ◆ 印刷用の新しいサンプルとマニュアル
- ◆ DataAccess での Oracle 10 および Oracle 11 のサポート
- ◆ DataAccess での DB2 9.5 のサポート
- ◆ DataAccess での OLE DB のサポート
- ◆ VS 2008 上でのポート
- ◆ バージョン 5.1 以降のすべてのバグ・フィックスの統合
- ◆ その他のフィックスと進化

コンパイル

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、ご利用になっているマシン、オペレーティング・システム、ご利用予定のコンパイラの種類によって異なります。

ディレクトリは、次に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)
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)
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

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
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)。

本バージョンではサポートされなくなったプラットフォーム

システム名	オペレーティング・システム	コンパイラ	置換システム
hp32_11_3.52	HP-UX 11	C++ A.03.52: -mt -AA オプション	hp32_11_3.73: HP ANSI C++ A.03.73 および -mt -AA オプション要
hp64_11_3.52	HP-UX 11	64 ビット・モード (+DA2.0W) / -mt -AA オプション使用の C++ A.03.52	64 ビット・モード (+DA2.0W) / -mt -AA オプション使用の hp64_11_3.73 HP ANSI C++ A.03.73 要
ia64_hpux11_5.52	HP-UX 11	HP C++/ANSI A.05.52	置換されていません
ia64_RHEL3.0_3.2	RedHat Enterprise Linux 3.0 (linux 2.4、glibc 2.3)	gcc3.2	置換されていません
x86_RHEL3.0_3.2	RedHat Enterprise Linux 3.0 (linux 2.4、glibc 2.3)	gcc3.2	x86_RHEL4.0_3.4: RedHat Enterprise Linux 4.0 (または互換機)、glibc2.3 および gcc3.4 要 x86_sles10.0_4.1: Suse 10.0 (または互換機)、glibc2.4 および gcc4.1 要
ppc_linux2_glibc2.3_gcc3.2	Linux 2.4、glibc 2.3	gcc3.2	置換されていません
msvc7	Windows 95/98/Me または NT/2000/XP	Microsoft Visual Studio C++ .NET 標準 2002 (7.0)	x86_.net2003_7.1 (または x86_.net2005_8.0/ x86_.net2008_9.0)、 Windows XP / Vista および Microsoft Visual C++ .NET 2003 (または 2005/2008) 要

バージョン 5.1 との非互換性

クラス IlvMakePolyPointsInteractor およびそのサブクラス:

メソッド

```
virtual void react(ILvPoint& p, ILUShort modifier = 0);
```

に、ポイントの追加を生成したイベントがトリガされたときにキーボード・モディファイアの設定を含む引数が1つ追加されました。この追加引数により、メソッドをオーバーライドしてモディファイアによってポイントを配置できるようになりました。

このメソッドがオーバーライドされることはほとんどありませんが、万一そうなった場合、コンパイル時間のエラーが発生します。この場合、派生メソッドのシグネチャを拡張してこの追加引数を含めます。

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
```

ILOG Views の正しい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。

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、**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っ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 **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)。

内容

ILOG Views Studio

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。ILOG Views Studio は、ILOG Views アプリケーション作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。

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、ILOG Views の評価版コピーになります。

ILOG Views アプリケーションを実行させるには、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
```

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

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

- ◆ winviews.lib: 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、"ILOG Views" または "ILOG Views Controls" のライセンスがある場合のみ、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`: ILOG スクリプト・ライブラリ
 - `ilvjs`: ILOG Views スクリプト実装
 - `iljsgide`: 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спек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
- ◆ Web 配布パッケージ (Advanced 2D)
 - `ilvweb`: Web 配布ライブラリ
- ◆ 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 (IlvGantChart 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

共有ライブラリの使用

ILOG Views ライブラリは、全プラットフォームで、スタティック・モード、共有モードの両方で提供されます (Unix ユーザ、Motif ベースのライブラリについては、次のセクションをご覧ください)。

Unix プラットフォームで共有ライブラリを使用する場合は、ダイナミック・ローダーで ILOG Views ライブラリが見つかるように設定する必要があります。これを行うには、環境変数 `LD_LIBRARY_PATH` (HP プラットフォームでは `SHLIB_PATH`、AIX プラットフォームでは `LIBPATH`) を `$ILVHOME/lib/<system>/<shareDir>` に設定します。

共有ライブラリを Windows プラットフォームで使用する場合は、システムが実際に ILOG Views DLL にアクセスするように設定する必要があります。詳細については、システムのマニュアルを参照してください。

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 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。

Motif および共有ライブラリに関する重要事項

`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、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 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しかできません。

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一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` (単一スレッド)。
- `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)。

注意:

1. これらのモードで *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 `wssock32.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` がリンクのコマンド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 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。

リリース・ノート — バージョン 5.1

ILOG Views の以前のリリースをお使いの方へ

以前のバージョンと ILOG Views 本バージョンの既知の非互換性については、すべて COMPAT.TXT ファイルに記述されています。これは、本 README ファイルと同じディレクトリにあります。

古いアプリケーションのコンパイルや実行の問題がある場合は、同ファイルを参照してください。

テクニカル・サポート

ILOG Views ライブラリ使用中に問題が生じたり、マニュアルを読んでも問題が解決できないという場合は、アイログのサポート・チームまでバグ・レポートをお送りください。その際に使うテンプレートは、ファイル BUGREPORT、Windows プラットフォームをご利用の場合は bugreprt.txt に配置されています。

コンパイル

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、ご利用になっているマシン、オペレーティング・システム、ご利用予定のコンパイラの種類によって異なります。

ディレクトリは、次に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_3.52	HP 9700	HP-UX 11 以降	-mt -AA オプション使用の HP HP ANSI C++ A.03.52 以降
hp64_11_3.52	HP 9700	HP-UX 11 以降	mt -AA オプション使用 /64 ビット・モード (+DA2.0W) の HP ANSI C++ A.03.52 以降
ia64_hpux11_5.52	ia64 (Itanium 2)	HP-UX 11 以降	HP aC++/ANSI A.05.52 以降
ia64_RHEL3.0_3.2	ia64 (Itanium 2)	RedHat Enterprise Linux 3.0 (linux 2.4, glibc 2.3)	gcc3.2
x86_RHEL3.0_3.2	x86	RedHat Enterprise Linux 3.0 (linux 2.4, glibc 2.3)	gcc3.2
ppc_linux2_glibc2.3_gcc3.2	PowerPC	Linux 2.4, glibc 2.3	gcc3.2
msvc7	x86	Windows 95/98/Me または Windows NT/2000/XP	Microsoft C++ 7.0
x86_.net2003_7.1	x86	Windows 95/98/Me または Windows NT/2000/XP	Microsoft Visual C++ .NET 2003
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
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)

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。

本バージョンではサポートされなくなったプラットフォーム

- ◆ 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)。

本バージョンではサポートされなくなったプラットフォーム

システム名	オペレーティング・システム	コンパイラ	置換システム
alpha_4_6.1	HP/Compaq/DEC Alpha、OSF1 V4	C++ 6.1	alpha_5.1_6.5 - Tru64 UNIX (OSF1) V5.1 および C++ 6.5 要
2_11_3.05	HP 9700、HP-UX 11	C++ 3.05	hp32_11_3.52 - HP ANSI C++ 3.52 および -mt -AA オプション要
hp32_11_3.30	HP 9700、HP-UX 11	C++ A.03.30 - -mt -AA オプション	hp32_11_3.52 - HP ANSI C++ 3.52 要
hp64_11_3.15	HP 9700、HP-UX 11	C++ A.03.15	hp32_11_3.52 - HP ANSI C++ A.03.52 および -mt -AA オプション要
hp64_11_3.39	HP 9700、HP-UX 11	C++ A.03.15 - -mt -AA オプション	hp32_11_3.52 - HP ANSI C++ A.03.52 要
i86_linux2_glibc2.2_gc c3.0	i86、Linux 2.4、glibc 2.2	gcc3.0.2	x86_RHEL3.0_3.2 - RedHat Enterprise Linux 3.0 (または互換機)、glibc2.3 および gcc3.2 要
i86_linux2_glibc2.3_gc c3.2	i86、Linux 2.4、glibc 2.3	gcc3.2	x86_RHEL3.0_3.2 - RedHat Enterprise Linux 3.0 (または互換機)、glibc2.3 および gcc3.2 要

システム名	オペレーティング・システム	コンパイラ	置換システム
rs6000_4.3_3.6	RS6000、AIX 4.3.2	CSet++ 3.6	rs6000_5.1_6.0 - AIX 5.1、Visual Age 6.0 および標準 C++ ストリーム (-DIL_STD) 要
sparc_5_5.0	Sparc、Solaris 2.4	Sun C++ 5.0	ultrasparc32_8_6.2 - Solaris 2.8 および Forte 6.2 (Sun C++ 5.3) 要
ultrasparc64_5_5.0	Ultra Sparc、Solaris 2.7	Sun C++ 5.0	ultrasparc64_8_6.2 - Solaris 2.8 および Forte 6.2 (Sun C++ 5.3) 要
msvc6	x86 Windows 95/98/Me または NT/2000/XP	Microsoft Visual C++ 6.x	x86_.net2003_7.1 - Microsoft Visual C++ .NET 2003 要

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 95/98/Me / Windows NT/2000/XP):

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

ILOG Views の正しい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。

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、**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っ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、`platform` は **ILOG Views** ライブラリのシステム識別子です (これは、`hp32_11_3.52`、`rs6000_5.1_6.0` など、いずれかの `<system>` になります)。

内容

ILOG Views Studio

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。**ILOG Views Studio** は、**ILOG Views** アプリケーション作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。

ILOG Views Data Access: バージョン 5.0 との既知の非互換性

Data Access 5.1 は、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、ヘッダー・ファイルが変更になっているためです(たとえば、Oracle 9i 対応の場合)。

ソース互換性は、文書化された API にのみ対応するものです。Data Access 5.1 では、Oracle 7.3 および Oracle 8.0 をサポートしていませんが、Oracle 9i には対応しています。Oracle 10.x のユーザは、Oracle 9i クライアントで、Oracle データベースにアクセスすることができます。

ppc_linux2_glibc2.3_gcc3.2 プラットフォーム用のデータベースについては、当社に同プラットフォームのデータベース・クライアントがないため、テストを実施していません。

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、ILOG Views の評価版コピーになります。

ILOG Views アプリケーションを実行させるには、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 95/98/Me / Windows NT/2000/XP):

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

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 (95/98/Me、および NT/2000/XP) 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: ILOG Views の純粋な Xlib コード
- ◆ libmviews: ILOG Views の Motif 依存コード

Windows 95/98/Me および NT/2000/XP プラットフォーム:

- ◆ winviews.lib: 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、"ILOG Views" または "ILOG Views Controls" のライセンスがある場合のみ、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
 - ilvjpg: 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: ILOG スクリプト・ライブラリ
 - ilvjs: ILOG Views スクリプト実装
 - iljsgide: 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
- ◆ **Web 配布パッケージ (Advanced 2D)**
 - **ilvweb**: Web 配布ライブラリ
- ◆ **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

共有ライブラリの使用

ILOG Views ライブラリは、全プラットフォームで、スタティック・モード、共有モードの両方で提供されます (Unix ユーザ、Motif ベースのライブラリについては、次のセクションをご覧ください)。

Unix プラットフォームで共有ライブラリを使用する場合は、ダイナミック・ローダーで ILOG Views ライブラリが見つかるように設定する必要があります。これを行うには、環境変数 `LD_LIBRARY_PATH` (HP プラットフォームでは `SHLIB_PATH`、AIX プラットフォームでは `LIBPATH`) を `$ILVHOME/lib/<system>/<shareDir>` に設定します。

共有ライブラリを Windows プラットフォームで使用する場合は、システムが実際に ILOG Views DLL にアクセスするように設定する必要があります。詳細については、システムのマニュアルを参照してください。

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 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。

Motif および共有ライブラリに関する重要事項

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、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 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しかできません。

Windows 95/98/Me および NT/2000/XP 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。

◆ msvc7 および 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)。

注意:

1. これらのモードで *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。

Windows 95 の既知の問題

Windows 95 の構成の中には、次のような問題が発生するものがあります。

- ◆ 上位ウィンドウを非表示にすると、その周りに跡が残る場合があります。この問題は、ビデオ・ドライバ構成に派生しているものと思われます。詳しくは、Windows 95 のオンライン・ヘルプで、KB Windows 95 についての記事「ATI Mach 64 Display Adapter Produces Garbled Screen」(1995 年 8 月、PSS-ID Q134487) を参照してください。

この問題を回避するには、次のような処理を行います。

1. [設定]メニューから[システム]を選択します。[システムのプロパティ]設定パネルが表示されます。
 2. [パフォーマンス]タブを選択します。
 3. [グラフィックス]ボタンをクリックします。[グラフィックスの詳細設定]パネルが表示されます。
 4. [ハードウェア アクセラレータ]のスライダをより低い値に設定します。このスライダは、ビデオ・ドライバのパフォーマンス・レベルを「なし」から「最大」の間で設定するものです。
- ◆ CD が挿入されていない CD-ROM ドライブを読み込もうとすると、プログラムがロックされます。システムに、他の Windows プラットフォーム同様、モーダル・エラー・ボックスが表示されるはずなのに、応答を待っても表示されません。この問題を回避するには、ドライブに CD を装填します。システムがロックされた状態でもかまいません。
 - ◆ Windows Plus をインストールすると、Borland C++ によるプログラムのビルドがクラッシュする (これは Borland 5.0 インストールの注意書きで指摘されています)。この問題は、入出力ストリームや浮動小数点処理のときに発生するようです。Windows Plus Agent を無効にして ILOG Views アプリケーションを実行するか、Microsoft 社から Windows Plus のパッチを入手する、または Microsoft 匿名サーバで ([ftp.microsoft.com](ftp://ftp.microsoft.com)) `softlib/mslfiles` から `plusupd1.exe` というファイルを入手します。

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` がリンクのコマンド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。

```
msvc7 or x86_.net2003_7.1:
%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 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。

```
msvc7 or x86_.net2003_7.1:
```

- 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。

リリース・ノート - バージョン 5.0.1

このリリース・ノートには、バージョン 5.0 以降の ILOG Views への変更について説明されています。本書は、以下の内容で構成されています。

- ◆ バージョン 5.0 以降行われた変更
- ◆ バージョン 5.0 以降修正されたパッチ

バージョン 5.0 以降行われた変更

メモ: 本 ILOG Views バージョンと以前のバージョンとの既知の非互換性についてはすべて、製品の README ファイルと同じ場所にある COMPAT.TXT ファイルに記述されています。

古いアプリケーションのコンパイルや実行の問題がある場合は、同ファイルを参照してください。

ILOG Views ライブラリ使用中に問題が生じたり、マニュアルを読んでも問題が解決できないという場合は、アイログのサポート・チームまでバグ・レポートをお送りください。その際に使うテンプレートは、ファイル BUGREPORT、Windows プラットフォームをご利用の場合は bugreprt.txt に配置されています。

コンパイルの問題

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、ご利用になっているマシン、オペレーティング・システム、ご利用予定のコンパイラの種類によって異なります。ディレクトリは、次にいずれかになります。

alpha_4_6.1	マシン : Compaq (DEC) Alpha オペレーティング・システム : OSF1 V4 以降 コンパイラ : C++ 6.1 以降
hp32_11_3.05	マシン : HP 9700 オペレーティング・システム : HP-UX 11 以降 コンパイラ : HP C++ A.03.05 以降
hp32_11_3.30	マシン : HP 9700 オペレーティング・システム : HP-UX 11 以降 コンパイラ : -mt -AA オプション使用の HP C++ A.03.30 以降
hp64_11_3.15	マシン : HP 9700 オペレーティング・システム : HP-UX 11 以降 コンパイラ : 64 ビット・モード (+DA2.0W) の HP C++ A.03.15 以降
hp32_11_3.39	M i86_linux2_glibc2.2_gcc3.0: マシン : PC オペレーティング・システム : Linux 2.4 以降 (glibc 2.2) コンパイラ : gcc3.0.2
i86_linux2_glibc2.3_gcc3.2	マシン : PC オペレーティング・システム : Linux 2.4 以降 (glibc 2.3) コンパイラ : gcc3.2

msvc6	マシン : PC オペレーティング・システム : Windows 95/98/Me または Windows NT/2000/XP コンパイラ : Microsoft Visual C++ 6.x
msvc7	マシン : PC オペレーティング・システム : Windows 95/98/Me または Windows NT/2000/XP コンパイラ : Microsoft Visual C++ 7.0
x86_.net2003_7.1	マシン : PC オペレーティング・システム : Windows 95/98/Me または Windows NT/2000/XP コンパイラ : Microsoft Visual C++ .NET 2003
rs6000_4.3_3.6	マシン : RS6000 オペレーティング・システム : AIX 4.3.2 以降 コンパイラ : CSet++ (3.6 以降) または Visual Age 5.0
rs6000_5.1_6.0	マシン : RS6000 オペレーティング・システム : AIX 5.1 以降 コンパイラ : 標準 C++ ストリーム (-DIL_STD) 使用の Visual Age 6.0
sparc_5_5.0	マシン : Sun Sparc Station オペレーティング・システム : Solaris 2.4 以降 コンパイラ : Sun C++ 5.0 (パッチ 1 M07311-05 以降)
ultrasparc64_5_5.0	マシン : Sun Ultra Sparc Station オペレーティング・システム : Solaris 2.7 以降 コンパイラ : Sun C++ 5.0 (パッチ 107311-05) 以降

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)。

本バージョンではサポートされなくなったプラットフォーム

次の表は、本製品ではサポートされなくなったプラットフォームをまとめたものです。

hp_10_1.0:	HP-UX 10、HP aC++ hp32_11_3.05 または hp32_11_3.30 で置換 (HP-UX 11 要)。
i86_linux2_glibc2.1_egcs1.1	Linux 2.0、Egcs 1.1。 i86_linux2_glibc2.2_gcc3.0 で置換
sgi32_6.5_7.2	IRIX プラットフォームについては、サポートを中止しました。
sparc_5_4.0	Solaris 2.4、Sun C++ 4.0。 Sun C++ 5.0 (以降) 要の sparc_5_5.0 または ultrasparc64_5_5.0 で置換
rs6000	AIX 3.2、CSet++ 2.1。 AIX 4.3 以降要の rs6000_4.3_3.6 で置換

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 95/98/Me / Windows NT/2000/XP):
    C:\> set ILVHOME=C:\ILOG\VIEWSXX
```

正しい製品インストール・ディレクトリについては、システム管理者にご確認ください。

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、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っ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、`platform` は ILOG Views ライブラリのシステム識別子です (これは、`hp32_11_3.30`、`rs6000_4.3_3.6`、`sparc_5_5.0` など、いずれかの `<system>` になります)。

ILOG Views Studio に行われた変更

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。

ILOG Views Studio は、ILOG Views アプリケーション作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」を構築するもので、それを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。
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。

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、ILOG Views の評価版コピーになります。

ILOG Views アプリケーションを実行させるには、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 95/98/Me / Windows NT/2000/XP):
    C:\> set ILOG_LICENSE_FILE=%ILVHOME%\access.ilm
```

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 (95/98/Me、および NT/2000/XP) 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: ILOG Views の純粋な Xlib コード
 - libmviews: ILOG Views の Motif 依存コード

◆ Windows 95/98/Me および NT/2000/XP プラットフォーム :

winviews.lib: 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):

ライブラリ名	説明
	ilog: ILOG ユーティリティ・クラス (配列、リストなど)
ILOG ユーティリティ・クラス (配列、リストなど)	ilvmllook: ilvwlook: ilvw95look: ilvwxplook:
Gadgets パッケージ (Controls)	
Application Framework パッケージ (Controls)	
Manager パッケージ (2D Standard)	
Grapher パッケージ (Advanced 2D)	
Prototypes パッケージ (Advanced 2D)	
Web 配布パッケージ (Advanced 2D)	
Data Access アドオン	
Charts アドオン	Charts ライブラリ (IlvChartGraphic および関連クラス)
Graph Layout アドオン	
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	

共有ライブラリの使用

ILOG Views ライブラリは、全プラットフォームで、スタティック・モード、共有モードの両方で提供されます (Unix ユーザ、Motif ベースのライブラリについては、次のセクションをご覧ください)。

Unix プラットフォームで共有ライブラリを使用する場合は、ダイナミック・ローダーで ILOG Views ライブラリが見つかるように設定する必要があります。これを行うには、環境変数 LD_LIBRARY_PATH (HP プラットフォームでは SHLIB_PATH、AIX プラットフォームでは LIBPATH) を \$ILVHOME/lib/<system>/<shareDir> に設定します。

共有ライブラリを Windows プラットフォームで使用する場合は、システムが実際に ILOG Views DLL にアクセスするように設定する必要があります。詳細については、システムのマニュアルを参照してください。

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 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。

Motif および共有ライブラリに関する重要事項

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、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 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しかできません。

Windows 95/98/Me および NT/2000/XP 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。

◆ msvc6 & msvc7 & 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)。
これらのモードで ILOG Script を使う場合は、フラグ /DILJSTDH も使用する必要があります。

メモ:

◆ msvc6 のみ:

- **stat_st**: 実行時スタティック・ライブラリがある単一スレッド内のスタティック・ライブラリ。
フラグ : /ML (単一スレッド)。
- **stat_mt**: 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 : /MT (マルチスレッド)。
- **stat_md**: 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 : /MD (マルチスレッド DLL)。
- **dll_md**: 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 : /MD /DILVDLL (マルチスレッド DLL)。
これらのモードで ILOG Script を使う場合は、フラグ /DILJSTDH も使用する必要があります。

メモ: 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。

Windows 95 の既知の問題

Windows 95 の構成の中には、次のような問題が発生するものがあります。

上位ウィンドウを非表示にすると、その周りに跡が残る場合があります。この問題は、ビデオ・ドライバ構成に派生しているものと思われます。詳しくは、Windows 95 のオンライン・ヘルプで、KB Windows 95 についての記事「ATI Mach 64 Display Adapter Produces Garbled Screen」(1995年8月、PSS-ID Q134487)を参照してください。

この問題を回避するには、次のような処理を行います。

1. [設定]メニューから[システム]を選択します。[システムのプロパティ]設定パネルが表示されます。

2. [パフォーマンス] タブを選択します。
3. [グラフィックス] ボタンをクリックします。[グラフィックスの詳細設定] パネルが表示されます。
4. [ハードウェアアクセラレータ] のスライダをより低い値に設定します。このスライダは、ビデオ・ドライバのパフォーマンス・レベルを「なし」から「最大」の間で設定するものです。

CD が挿入されていない CD-ROM ドライブを読み込もうとすると、プログラムがロックされます。システムに、他の Windows プラットフォーム同様、モーダル・エラー・ボックスが表示されるはずなのに、応答を待っても表示されません。この問題を回避するには、ドライブに CD を装填します。システムがロックされた状態でもかまいません。

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` がリンクのコマンド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。

```
msvc6:
```

```
msvc7 または x86_.net2003_7.1:
```

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 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。

- ◆ `msvc6:`
 1. 開発者版の Studio で、メニュー・オプション [プロジェクト] > [設定 ...] を選択します。
 2. [設定の対象] パネルですべてのデバッグ・プロジェクトを選択するようにしてください。
 3. [C/C++] ノートブック・タブを選択します。
 4. このページで、[カテゴリ] オプション・メニューから [プリプロセッサ] オプションを選択します。

5. [プリプロセッサの定義]フィールドで、"_DEBUG"を削除します。
 6. 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。
- ◆ msvc7 または x86_.net2003_7.1:
1. 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。
 2. 選択項目を右クリックして、[プロパティ]を選択します。
 3. [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。
 4. [プリプロセッサ定義]フィールドで、"_DEBUG"を削除します。
 5. 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。

バージョン 5.0 以降修正されたパッチ

PATCH 50.0001

[説明] : Views 4.02 forces the use of buffered drawing operations on gadget containers.

[ライブラリ] : views, ilvgadgt, ilvmgr

[パッケージ] : foundation, manager

PATCH 50.0002

[説明] : Menu Items of combo boxes may stay on the screen when using the 'fade item' feature of Windows.

[ライブラリ] : ilvgadgt

[パッケージ] : foundation

PATCH 50.0003

[説明] : On X11, IlvAbstractView::setSensitive doesn't propagate to its children views.

[ライブラリ] : xviews

[パッケージ] : foundation

PATCH 50.0004

[説明] : [dialogs] IlvFileBrowser::setFileName is no longer taken into account since patch 40.2045

[ライブラリ] : winviews

[パッケージ] : foundation

PATCH 50.0005

[説明] : Error with serialization of boolean values.

[ライブラリ] : ilvcharts, ilvgantt, ilvgrapher, ilvproto, ilvweb, views

[パッケージ] : charts, gantt, grapher, protos, web, foundation

PATCH 50.0006

[説明] : Last page may be previewed several times.

[ライブラリ] : ilvprint

[パッケージ] : foundation

PATCH 50.0007

[説明] : [layout] quadtree errors when using IlvBus Layout::setBus method.

[layout] no pragma which force the library link on PC

[ライブラリ]: ilvbus

その他のファイル:

include/ilviews/layout/macros.h, include/ilviews/layout/bus.h,
include/ilviews/layout/hierarchical.h, include/ilviews/layout/
orthlink.h, include/ilviews/layout/random.h, include/ilviews/
layout/tree.h

[パッケージ]: layout, appframe

PATCH 50.0008

[説明]: stretchBitmapData does not work in 8 bits.

[ライブラリ]: xviews, mviews, winviews

[パッケージ]: foundation

PATCH 50.0009

[説明]: IlvFilterFlow tries to load an empty filter using modules.

[ライブラリ]: xviews, mviews, winviews

[パッケージ]: foundation

PATCH 50.0010

[説明]: IlvIndexedBitmapData does not handle rectangular copy.

[ライブラリ]: xviews, mviews, winviews

[パッケージ]: foundation

PATCH 50.0011

[説明]: The tiff streamer cannot save tiled indexed bitmap datas Read/Write callbacks are not always called in the Tiff streamer. Tiff streamer does not handle correctly border tiles.

[ライブラリ]: ilvtiff

[パッケージ]: foundation

PATCH 50.0012

[説明]: The Data Access application crashes at exit.

[ライブラリ]: dataaccess

[パッケージ]: dataaccess

PATCH 50.0013

[説明]: It is impossible to hide the scrollbars in an IliDbStringList.

[ライブラリ]: dbgadget

[パッケージ]: dataaccess

PATCH 50.0014

[説明] : [appframe] no pragma which force the library link on PC.

[appframe] Some .h files in the distrib cannot be parsed with the samples/appframe/synedit example.

[Appframe] The document save modifications dialog box is not triggered under UNIX under specific circumstances

[ライブラリ]: ilvappframe

その他のファイル: samples/appframe/synedit/src/synparse.cpp,
include/ilviews/appframe/macros.h, include/ilviews/appframe/
manager/macros.h, include/ilviews/appframe/grapher/macros.h,
include/ilviews/appframe/wizard/macros.h

[パッケージ]: appframe

PATCH 50.0015

[説明] : [studio] The registered include files and libraries are not always properly taken into account during code generation.

[ライブラリ]: ivstudio

[パッケージ]: foundation

PATCH 50.0016

[説明] : Studio.appli doesn't register the ilvproto dependancies. There are missing dependencies for prototype library

[ライブラリ]: ivstudio

[パッケージ]: foundation

PATCH 50.0017

[説明] : IlvOutlinePolygon border is badly drawn with a large thickness

[ライブラリ]: winviews, views

[パッケージ]: foundation

PATCH 50.0018

[説明] : IlvArc:: contains may return wrong results under extreme conditions

[ライブラリ]: display

[パッケージ]: foundation

PATCH 50.0019

[説明] : [manager]: cannot undo/redo IlvManagerMakeListLabelInteractor object creations

[ライブラリ]: ilvmgr, ilvgadmgr, ilvadvgadmgr

[パッケージ]: manager, gadgets

PATCH 50.0020

[説明]: The `IlvParser` class doesn't delete its variables.

[ライブラリ]: `ilvvar31`

[パッケージ]: `compat`

PATCH 50.0021

[説明]: The old charts code was not updated after the studio code generation was modified in Views5.0

[ライブラリ]: `views31, ilvstoldcharts31`

[パッケージ]: `compat`

PATCH 50.0022

[説明]: When Caps Lock is on on Windows, pressing a key + shift yields an event where `_modifiers` is `IlvShiftModifier` instead of 0

[ライブラリ]: `winviews`

[パッケージ]: `foundation`

PATCH 50.0023

[説明]: `IlldbTreeGadget` is too slow when you have a lot of items

[ライブラリ]: `dbgadget`

[パッケージ]: `dataaccess`

PATCH 50.0024

[説明]: Studio crash at exit

[ライブラリ]: `ivstudio`

[パッケージ]: `foundation`

PATCH 50.0025

[説明]: Small defects with the dialog used to select the size of the printing in `ivstudio`.

[ライブラリ]: `ivstudio`

[パッケージ]: `foundation`

PATCH 50.0026

[説明]: [studio] The callbacks list in graphics inspector are not notified of the interactor changes.

[ライブラリ]: `ivstudio`

[パッケージ]: `foundation`

PATCH 50.0027

[説明]: [appframe] Closing one of the containers opened on the document will trigger the prompt for saving the document.

[ライブラリ]: ilvappframe

[パッケージ]: appframe

PATCH 50.0028

[説明]: zoomable label potentially corrupts the world transformation on NT platforms

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0030

[説明]: Bad management of submenus with Windows Multiple Monitors feature

[ライブラリ]: ilvgadgt, ilvwlook, ilvw95look

[パッケージ]: foundation

PATCH 50.0031

[説明]: bad update of the attachments when a panel is resized at init time

[foundation] attached graphics may be badly resized when setting a guide limit.

[ライブラリ]: views

[パッケージ]: foundation

PATCH 50.0032

[説明]: Crash after embedding a menu in a graphic handle. A tear off popup menu closes automatically after selection

[ライブラリ]: ilvgadgt, ilvmgr

[パッケージ]: foundation, manager

PATCH 50.0033

[説明]: IFC: XML bad encoding check for UTF-16 without signature and Wrong endian UTF-16 without signature.

[ライブラリ]: ilog

[パッケージ]: foundation

PATCH 50.0034

[説明]: Palette clip conflict when previewing before printing.

[ライブラリ]: ilvprint

[パッケージ]: foundation

PATCH 50.0035

[説明] : [Studio][Protos] Invisible graphics of the proto are made visible on edition
[protos] prstudio crashes when saving an .ilv file that contains a prototype that was previously removed from its library

[ライブラリ]:

[パッケージ]: protos

PATCH 50.0036

[説明] : [prstudio] crash when removing the name of a protoinstance
[protos] cannot set a name to a protographic copy under ivfstudio

[ライブラリ]: ilvproto

[パッケージ]: protos

PATCH 50.0037

[説明] : [protos][studio] save proto library as crashes

[ライブラリ]: prstudio

[パッケージ]: protos

PATCH 50.0038

[説明] : [Protos] 3.1 to 4.0 incompatibility: inherited values are hardcoded in 4.0 proto's description.

[ライブラリ]: ilvproto

[パッケージ]: protos

PATCH 50.0039

[説明] : IlvContainerGraphicHolder::setPropertyToGraphic leads to a free memory read.

[ライブラリ]: views, ilvadvgdt, ilvmgr

[パッケージ]: foundation, gadgets, manager

PATCH 50.0040

[説明] : PrStudio: cannot select an attribute name in an attribute parameter field.

[ライブラリ]: prstudio

[パッケージ]: protos

PATCH 50.0041

[説明] : Cannot close the combo box menu using the Escape key.

[ライブラリ]: ilvgadgt

[パッケージ]: foundation

PATCH 50.0042

[説明]: A pair of notify/watch accessor might not work as expected if the attribute attached to the watch accessor is untyped.

[ライブラリ]: views

[パッケージ]: foundation

PATCH 50.0043

[説明]: [Studio]In some circumstance the apply button does not work fine.

[ライブラリ]: ivstudio

[パッケージ]: foundation

PATCH 50.0044

[説明]: [windows] IlvWindowsDevice printing is affected by small/large fonts display settings.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0045

[説明]: [printing] studio crashes when setting a dummy preview page.

[printing] missing localized printing.dbm file in the distribution.

[ライブラリ]: ilvprtdlg

その他のファイル: data/res/printing.rc, data/printing/locale/fr_FR.windows-1252/printing.dbm, data/printing/locale/fr_FR.ISO-8859-1/printing.dbm

[パッケージ]: gadgets, foundation

PATCH 50.0046

[説明]: [appframe] Alt Keyboard accelerator key is mapped to the IlvAltModifier (should be IlvMetaModifier).

[ライブラリ]: ilvappframe

[パッケージ]: appframe

PATCH 50.0047

[説明]: On Windows, it is not possible to return to English locale if ivfstudio is started in Japanese or Chinese.

[ライブラリ]:

その他のファイル: bin/res/i2dblock.rc, data/res/winviews.rc, data/iljscript/locale/fr_FR.windows-1252/gide.dbm, data/iljscript/locale/fr_FR.windows-1252/messages.js,

studio/data/res/foundation.rc, studio/data/res/script.rc, studio/msvc6/ivfstudio.exe, studio/msvc7/ivfstudio.exe

[パッケージ]: foundation

PATCH 50.0048

[説明]: On Windows, it is not possible to return to English locale if `ivfstudio` is started in Japanese or Chinese.

[ライブラリ]:

その他のファイル: data/res/appframe.rc

[パッケージ]: appframe

PATCH 50.0049

[説明]: On Windows, it is not possible to return to English locale if `ivfstudio` is started in Japanese or Chinese.

[ライブラリ]:

その他のファイル: studio/data/res/gadgets.rc

[パッケージ]: gadgets

PATCH 50.0050

[説明]: [Windows] The printing of opaque bitmaps is black.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0051

[説明]: Problem with same messages defined in several dbm files with different languages.

[ライブラリ]: display

[パッケージ]: foundation

PATCH 50.0052

[説明]: `IlvMatrix` does not show its tooltip after having shown an `IlvGraphicMatrixItem` tooltip.

[ライブラリ]: `ilvadvgdt`

[パッケージ]: gadgets

PATCH 50.0053

[説明]: Crash when trying to export a table schema from `ivfstudio`.

[ライブラリ]: `dbstsql`

[パッケージ]: `dataaccess`

PATCH 50.0054

[説明] : `IlvDisplay::colorTable` doesn't properly handle mutable colors.

[ライブラリ] : `display`

[パッケージ] : `foundation`

PATCH 50.0055

[説明] : Keyboard shortcuts are not invoked with minimized MDI views.

[ライブラリ] : `views`, `ilvgadmgr`, `ilvadvgdt`

[パッケージ] : `foundation`, `manager`, `gadgets`

PATCH 50.0056

[説明] : The `IlvDisplayTestApi` notification mechanisms works partially on Windows.

[ライブラリ] : `winviews`

[パッケージ] : `foundation`

PATCH 50.0058

[説明] : Tooltips might not work on a transient view with no border (on Windows).

[ライブラリ] : `winviews`, `views`, `ilvgadgt`

[パッケージ] : `foundation`

PATCH 50.0060

[説明] : `[studio][printing]` Japanese locale causes `ivfstudio` to crash at print preview.

[ライブラリ] : `ilvprint`

[パッケージ] : `foundation`

PATCH 50.0061

[説明] : The `IlITableComboBox` does not support the multi-display.

[ライブラリ] : `dbgadget`

[パッケージ] : `dataaccess`

PATCH 50.0062

[説明] : `[Windows]` Labels are not displayed with correct font.

[ライブラリ] : `winviews`

[パッケージ] : `foundation`

PATCH 50.0063

[説明] : Bad management of keyboard events in `IlvViewFrame` since Patch 50.00055

[ライブラリ] : `ilvadvgdt`

[パッケージ]: gadgets

PATCH 50.0064

[説明]: Bad drawing of a notebook in Windows XP Look.

[ライブラリ]: ilvawxpllook

[パッケージ]: gadgets

PATCH 50.0065

[説明]: ivfstudio may crash when creating and IlvFilteredGraphic on UNIX 8 bits display

[ライブラリ]: views

[パッケージ]: foundation

PATCH 50.0066

[説明]: [IFC-WINDOWS]Test of write access on pathname with a blank always returns IlvFalse.

[ライブラリ]: ilog

[パッケージ]: foundation

PATCH 50.0067

[説明]: Several problems on Dockable Windows

[ライブラリ]: winviews, ilvadvgdt

[パッケージ]: foundation, gadgets

PATCH 50.0068

[説明]: Cannot trace the memory manager

[ライブラリ]: ilog

[パッケージ]: foundation

PATCH 50.0069

[説明]: IlvDisplay::putBitmapData may not work on Windows after some drawings have been made in the bitmap.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0070

[説明]: fr_FR.windows-1252/dvwizard.dbm file uses an incorrect encoding.

[appframe][dvwizard] The docking position of views is not taken into account.

[appframe] There are inconsistencies in the modality/transience of some dialog boxes.

[ライブラリ]: ilvappframe

その他のファイル: bin/data/appframe/dvwizard/locale/fr_FR.windows-1252/
dvwizard.dbm, bin/data/appframe/dvwizard/locale/fr_FR.ISO-8859-1/dvwizard.dbm

[パッケージ]: appframe

PATCH 50.0071

[説明]: Bitmap streamers are not correctly registered.

[ライブラリ]: xviews, mviews, winviews

[パッケージ]: foundation

PATCH 50.0072

[説明]: IlvWindowsDevice::setOrientation() has no effect, only the printer default configuration is used.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0073

[説明]: [DvWizard][AppFrame] Deleting the "RecentProjects" command crashes the wizard.

[ライブラリ]: ilvappwizard

[パッケージ]: appframe

PATCH 50.0074

[説明]: IlvRGBBitmapData::alphaCompose MMX code is not optimal.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0075

[説明]: [Protos] IlvGroupInputFile badly reads double quoted values when the decimal symbol is ','.

[ライブラリ]: ilvproto

[パッケージ]: protos

PATCH 50.0076

[説明]: Bug in IlvText::getSelectedText in a multibyte environment.

[ライブラリ]: ilvadvgdt

[パッケージ]: gadgets

PATCH 50.0077

[説明]: The generated Panel subclass constructor code (ivfstudio) is incorrect.

[ライブラリ]: ilvstappli

[パッケージ]: gadgets

PATCH 50.0078

[説明]: `IlvDisplay::colorTable` might be called on Windows (depth usually > 8) after a 4 or 8 bits deep bitmap has been read.

[ライブラリ]: `xviews`, `mviews`, `winviews`

[パッケージ]: foundation

PATCH 50.0079

[説明]: `IlvMoveInteractor` leaves a ghost upon click on a button different from `IlvLeftButton`.

[ライブラリ]: `views`

[パッケージ]: foundation

PATCH 50.0080

[説明]: [AppFrame] generated project application may crash on `IlvDvProjectDocument::activateItem`.

[ライブラリ]: `ilvappframe`

[パッケージ]: `appframe`

PATCH 50.0081

[説明]: The drawing of 1-bit deep transparent bitmaps is incorrect after patch 402065.

[ライブラリ]: `winviews`

[パッケージ]: foundation

PATCH 50.0082

[説明]: `IlvDisplay::colorTable` might be called on Windows (depth usually > 8) when saving a 4 or 8 bits deep bitmap.

[ライブラリ]: `winviews`

[パッケージ]: foundation

PATCH 50.0083

[説明]: Impossible to choose different initial docking configurations for each view of an `appframe` document.

[ライブラリ]: `ilvappframe`

[パッケージ]: `appframe`

PATCH 50.0084

[説明]: `IlvWritePBMBitmap` may try to access the display `colorTable` while in `TrueColors`.

[ライブラリ]: winviews, mviews, xviews

[パッケージ]: foundation

PATCH 50.0085

[説明]: Drawing a bitmap is drawn with a palette that has an alpha value causes the memory use to increase (dll).

[ライブラリ]: winviews, ilvgdiplus

[パッケージ]: foundation

PATCH 50.0086

[説明]: IlvMarkingMenu does not delete all the windows upon creation.

[ライブラリ]: ilvadvgdt

[パッケージ]: gadgets

PATCH 50.0087

[説明]: IlXmlParser does not parse correctly the <!DOCTYPE ... > tag

[ライブラリ]: illog

[パッケージ]: foundation

PATCH 50.0088

[説明]: [studio] Two successive commas are generated in the 'Panel Subclass' constructor.

[ライブラリ]: ilvstappli

[パッケージ]: gadgets

PATCH 50.0089

[説明]: [Event player] The player does not play correctly when we move the slider of the scrollbar.

[ライブラリ]: xviews, winviews, mviews

[パッケージ]: foundation

PATCH 50.0090

[説明]: IlvSystemPort::GetBitmapData may crash when dealing with a masked bitmap.

[ライブラリ]: xviews, mviews, winviews

[パッケージ]: foundation

PATCH 50.0091

[説明]: [AppFrame] Closing one of the two views associated to a document may not close the document.

[ライブラリ]: ilvappframe

[パッケージ]: appframe

PATCH 50.0092

[説明]: Cannot save custom events in the event player.

[ライブラリ]: winviews, mviews, xviews

[パッケージ]: foundation

PATCH 50.0093

[説明]: [Windows] IlvZoomableTransparentIcon cannot be rotated by 180 deg
IlvZoomableIcon can't be flipped.

IlvTransformer::isScale returns true for symmetries.

[ライブラリ]: display

[パッケージ]: foundation

PATCH 50.0094

[説明]: [appframe] The ConfirmSave dialog box is not transient wrt the main window.

[ライブラリ]: ilvappframe

[パッケージ]: appframe

PATCH 50.0095

[説明]: Bad redraw when zooming in an area with a scale using crossing values and chart cursor.

[ライブラリ]: ilvcharts

[パッケージ]: charts

PATCH 50.0096

[説明]: The dockable pane initiates a dragging operation on a ButtonDown event without checking which button is down.

[ライブラリ]: ilvadvgdt

[パッケージ]: gadgets

PATCH 50.0097

[説明]: When replacing the whole text in a matrix cell, the first character may not be visible.

[ライブラリ]: ilvgadgt

[パッケージ]: foundation

PATCH 50.0098

[説明]: [AppFrame] duplicate resource are generated when two actions share the same icon.

[ライブラリ]: ilvappframe

[パッケージ]: appframe

PATCH 50.0099

[説明]: [AppFrame] A crash may occur on UNIX when closing very fast frames from the 'More Windows' panel.

[AppFrame] Closing one of the two views associated to a document may not close the document.

[ライブラリ]: ilvappframe

[パッケージ]: appframe

PATCH 50.0100

[説明]: Deleting a menu item from the ivfstudio menu inspector turns the item into a separator.

[ライブラリ]: ilvstgadget

[パッケージ]: gadgets

PATCH 50.0101

[説明]: `IliErrorMessage::getMessage()` returns an empty string.

[ライブラリ]: dataaccess

[パッケージ]: dataaccess

PATCH 50.0102

[説明]: Values (2, 1) for the `stepsCount/subStepsCount` in the `IlvSingleScaleDisplayer` causes a crash.

[ライブラリ]: ilvcharts

[パッケージ]: charts

PATCH 50.0103

[説明]: `DvWizard` crashes when a new item is inserted in the menu bar of the generated application.

`DvWizard`: When the document (or view) class changes, the "derived from" field is reset to `IlvDvDocument` (or `IlvDvTextView`)

[ライブラリ]: ilvappwizard

その他のファイル: `bin/data/appframe/dvwizard/dvwizard.odv`

[パッケージ]: appframe

PATCH 50.0104

[説明]: The code generation does not work for `IlvChartGraphic`.

[ライブラリ]: `ilvcharts, ilvgadgraph`

[パッケージ]: `charts, grapher`

PATCH 50.0105

[説明]: Z Order of overlapping sibling windows is not portable across platforms

[ライブラリ]: `winviews`

[パッケージ]: `foundation`

PATCH 50.0106

[説明]: An `IlvChartGraphic` with an `IlvPieChartDisplayer` read from an ILV file might not be updated properly when a point is added to one of its dataset.

[ライブラリ]: `ilvcharts`

[パッケージ]: `charts`

PATCH 50.0107

[説明]: The header file of non `IlvGadgetContainer` base classes is not added to generated code.

[ライブラリ]: `views, ilvstappli`

[パッケージ]: `foundation, gadgets`

PATCH 50.0108

[説明]: [studio] [solaris] Delete key has no action when num lock is on.

[ライブラリ]: `ivstudio`

[パッケージ]: `foundation`

PATCH 50.0109

[説明]: [appframe] appframe prompts for document reloading everytime it is reactivated.

[ライブラリ]: `ilvappframe`

[パッケージ]: `appframe`

PATCH 50.0110

[説明]: `IlvCircularScale::draw` uses a clip based on `_drawrect` instead of the object's bounding box.

[ライブラリ]: `views`

[パッケージ]: `foundation`

PATCH 50.0111

[説明]: Studio doesn't generate the `#include <ilviews/graphics/selector.h>` line when an `IlvSelector` is used.

[ライブラリ]: `views`

[パッケージ]: foundation

PATCH 50.0112

[説明]: Call to varargs functions must have explicitly typed arguments (HP 64 bits).

[ライブラリ]: ilvappframe, ilvappwizard

[パッケージ]: appframe

PATCH 50.0113

[説明]: IlvTransformedGraphic::applyTransform doesn't perform well on every objects.

[ライブラリ]: views

[パッケージ]: foundation

PATCH 50.0114

[説明]: [gadgets] IlvMenuItem accelerators should ignore some modifiers (ex: numlock on solaris).

[ライブラリ]: ilvgadgt

[パッケージ]: foundation

PATCH 50.0115

[説明]: [Windows] IlvSystemPort::fillPolyline may raise an error when drawing only one point.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0116

[説明]: [Windows] it is not possible to print with the default printer without displaying the print Dialog.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0117

[説明]: [gadgets] IlvIFileSelector does not match the UNIX behavior regarding relative paths.

[ライブラリ]: ilvadvgdt

[パッケージ]: gadgets

PATCH 50.0118

[説明]: Problem with IlvPrintableComposite

[ライブラリ]: ilvprint

[パッケージ]: foundation

PATCH 50.0119

[説明]: Clicking on an `IliSimpleToggle` makes the application crash.

[ライブラリ]: dbgadget

[パッケージ]: dataaccess

PATCH 50.0120

[説明]: [studio] It is not possible to specify a baseclass without class information (no header in the generated file).

[ライブラリ]: ilvstappli

[パッケージ]: gadgets

PATCH 50.0121

[説明]: `IlvSwitchAccessor` uses "delete []" to free memory allocated with `new` or `malloc` (FMM).

[ライブラリ]: ilvproto

[パッケージ]: protos

PATCH 50.0122

[説明]: The "16 bits" clipping done by Views before calling X11 drawing primitive is not enough to prevent XServer drawing oddities

[ライブラリ]: xviews, mviews

[パッケージ]: foundation

PATCH 50.0123

[説明]: [gadgets] `IlvNumberField` may crash studio while reading back the min/max data (precision issue).

[ライブラリ]: ilvgadgt

[パッケージ]: foundation

PATCH 50.0124

[説明]: [charts] Removing an scale from an `IlvChartGraphic` causes a crash.

[ライブラリ]: ilvcharts

[パッケージ]: charts

PATCH 50.0125

[説明]: `IlvReliefGauge` doesn't register any "ClassCode" information.

[ライブラリ]: views

[パッケージ]: foundation

PATCH 50.0126

[説明] : IlvStringList scrollbars adjusting produces unmatched up and down scrolls.

[ライブラリ] : ilvadvgdt

[パッケージ] : gadgets

PATCH 50.0127

[説明] : Several sensitivity/focus problem on notebook and containers.

[ライブラリ] : ilvgadmgr, ilvgadgt

[パッケージ] : manager, foundation

PATCH 50.0128

[説明] : The "State" accessor in the "general" tab of the studio graphic inspector doesn't work properly.

[ライブラリ] : ivstudio

[パッケージ] : foundation

PATCH 50.0129

[説明] : The generated code for Chart can crash when the format is static.

[ライブラリ] : ilvcharts

[パッケージ] : charts

PATCH 50.0130

[説明] : The include for IlvChartLegend is not generated by studio.

[ライブラリ] : ilvcharts

[パッケージ] : charts

PATCH 50.0131

[説明] : When a legend connected to a chart is removed from a studio buffer, the chart keeps referring to the legend.

[ライブラリ] : ilvstcharts

[パッケージ] : charts

PATCH 50.0132

[説明] : IlvSpinBox::setValue does return IlFalse on success since patch 40.2020 (BR 2001.189).

[ライブラリ] : ilvadvgdt

[パッケージ] : gadgets

PATCH 50.0133

[説明] : The contents of the manager may be printed on the footer and header.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0134

[説明]: Moving up and down datasets in chart inspector causes studio to crash.

[ライブラリ]: ivstudio

[パッケージ]: foundation

PATCH 50.0135

[説明]: The Studio IlvToggleFilteredGraphicCommand does not consider an IlvGrapher buffer.

[ライブラリ]: ilvstgrapher

[パッケージ]: グラファ

PATCH 50.0136

[説明]: Studio messages editor crashes (IlvMessagePanel bug).

[ライブラリ]: ilvedit

[パッケージ]: gadgets

PATCH 50.0137

[説明]: [studio] Remove data from the dataSet chart inspector does not refresh the last removed row.

[ライブラリ]: ivstudio

[パッケージ]: foundation

PATCH 50.0138

[説明]: [studio] inspector crashes after having deleted a composite displayer.

[ライブラリ]: ilvstcharts

[パッケージ]: charts

PATCH 50.0139

[説明]: IlvChartScrollInteractor can get stuck when reaching the maximum and minimum values.

[ライブラリ]: ilvcharts

[パッケージ]: charts

PATCH 50.0140

[説明]: After using IlvSingleScaleDisplayer::setStepLabels(), cannot come back to fixed steps.

[ライブラリ]: ilvstcharts

[パッケージ]: charts

PATCH 50.0141

[説明]: IlvGadgetItem does not display strings correctly in a multibyte locale.

[ライブラリ]: winviews, ilvgadgt

[パッケージ]: foundation

PATCH 50.0142

[説明]: Tooltips are not displayed on Windows in popup menus since patches 40.2114 and 50.0058.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0143

[説明]: Studio should not allow to delete a panel class being used in an application notebook.

[ライブラリ]: ilvstappli, ivstudio

[パッケージ]: gadgets, foundation

PATCH 50.0144

[説明]: Wrong transparency computation on 16 bits displays.

[ライブラリ]: xviews, mviews

[パッケージ]: foundation

PATCH 50.0145

[説明]: Crash in studio when adding a displayer to a chart with no displayers.

[ライブラリ]: ilvstcharts

[パッケージ]: charts

PATCH 50.0146

[説明]: [pie] reading back a chart does not restore the pie slice labels font.

[ライブラリ]: ilvcharts

[パッケージ]: charts

PATCH 50.0147

[説明]: bad redraw when zooming in an area with a scale using crossing values and chart cursor.

[ライブラリ]: ilvcharts

[パッケージ]: charts

PATCH 50.0148

[説明] : (Unix) Paste error with `getClipboard()` through Exceed.

[ライブラリ] : `xviews, mviews`

[パッケージ] : `foundation`

PATCH 50.0150

[説明] : Grouping objects in a `IlvSmarSet` corrupts the focus chain.

[ライブラリ] : `views`

[パッケージ] : `foundation`

PATCH 50.0151

[説明] : [studio] `IlvStringList` selection mode inspector does not work correctly since patch 50.0026.

[ライブラリ] : `ilvadvgdt, ilvstgadget`

[パッケージ] : `gadgets`

PATCH 50.0152

[説明] : Visual Studio wizard crashes when creating a project with an `IlvGrapher`.

[ライブラリ] :

その他のファイル : `tools/viewswiz50/viewswiz50.awx, tools/viewswiz50/viewswiz50.hlp`

[パッケージ] : `foundation`

PATCH 50.0153

[説明] : Unsatisfied symbols from `libilvmgr` in `libiljsgide`.

[ライブラリ] : `iljsgide`

[パッケージ] : `foundation`

PATCH 50.0154

[説明] : Matrix in Motif look and feel pastes twice at `IlvButtonUp` of right button.

[ライブラリ] : `ilvadvgdt`

[パッケージ] : `gadgets`

PATCH 50.0155

[説明] : The sensitivity of gadgets added to `IlvGadgetMatrixItems` is not persistent.

[ライブラリ] : `ilvadvgdt`

[パッケージ] : `gadgets`

PATCH 50.0156

[説明] : Using tooltips on gadgetItems with Motif may lead to a crash.

[ライブラリ]: views

[パッケージ]: foundation

PATCH 50.0157

[説明] : [AppFrame] [windows] classinfo.odv is missing in the appframe.rc resource file.

[ライブラリ]:

その他のファイル: data/res/appframe.rc

[パッケージ]: appframe

PATCH 50.0158

[説明] : IlvModeNotOr and IlvModeNotAnd use incorrect logical functions of the GC (X11).

[ライブラリ]: xviews, mviews

[パッケージ]: foundation

PATCH 50.0159

[説明] : the tm_isdst field isn't initialized by the JvConverter.

[ライブラリ]: ilvjs

[パッケージ]: foundation

PATCH 50.0160

[説明] : [gadgets] [foundation] IlvMatrix crashes when deselecting a 5000 * 8 table.

[ライブラリ]: ilvadvgdt

[パッケージ]: gadgets

PATCH 50.0161

[説明] : [variable] Bad behavior of the parser when the decimal symbol is ',' (comma).

[ライブラリ]: ilvvar31

[パッケージ]: compat

PATCH 50.0162

[説明] : The IlvMakePolylineLinkInteractor hangs when one double-clicks outside a node.

[ライブラリ]: ilvgrapher

[パッケージ]: グラファ

PATCH 50.0163

[説明] : `IlvManager::allIntersects` only takes into account the visible layers.

[ライブラリ] : `ilvmgr`

[パッケージ] : マネージャ

PATCH 50.0164

[説明] : `[prstudio] IlvAccessorParameter::getChoices()` receives an (inconsistent) null `IlvValueTypeClass` object.

[ライブラリ] : `prstudio`

[パッケージ] : `protos`

PATCH 50.0165

[説明] : Crash reading prototypes containing prototypes in a multidisplay environment.

[ライブラリ] : `ilvproto`

[パッケージ] : `protos`

PATCH 50.0166

[説明] : `[matrix]` Crash when selecting all rows of a 5000 * 8 table (bottom-up selection).

[ライブラリ] : `ilvadvgdt`

[パッケージ] : `gadgets`

PATCH 50.0168

[説明] : `IlvChartDragPointInteractor` does not always take the validation constraint into account.

[ライブラリ] : `ilvcharts`

[パッケージ] : `charts`

PATCH 50.0169

[説明] : An additional " appears when the prototype is saved in Views 5.0.

[ライブラリ] : `ilvproto`

[パッケージ] : `protos`

PATCH 50.0170

[説明] : Prototypes may not be able to retrieve properly their holder since patch 401118.

[ライブラリ] : `prstudio`

[パッケージ] : `protos`

PATCH 50.0171

[説明] : A prototype edition buffer is not marked as modified when the values of the prototype are modified.

[ライブラリ]: prstudio

[パッケージ]: protos

PATCH 50.0172

[説明]: [gadgets] Wrong gadget container background color in motif L&F.

[ライブラリ]: views

[パッケージ]: foundation

PATCH 50.0173

[説明]: Portage on HP 64 bits

[ライブラリ]: dbgadget, dbaccess

[パッケージ]: dataaccess

PATCH 50.0174

[説明]: [printing] IlvPrintableGraphic does not print anything (bad transformer setting).

[ライブラリ]: ilvprint

[パッケージ]: foundation

PATCH 50.0175

[説明]: [studio] Application panel inspector may not be refreshed properly and may lead to a crash.

[ライブラリ]: ilvstappli

[パッケージ]: gadgets

PATCH 50.0176

[説明]: Guides/Splitters limits in IlvGanttChartForm do not depend on L&F.

[ライブラリ]: ilvgantt

[パッケージ]: gantt

PATCH 50.0177

[説明]: [protos] FMM in IlvGroupMediator::undoSubscription.

[ライブラリ]: ilvproto

[パッケージ]: protos

PATCH 50.0178

[説明]: Changing the values of a prototype that contains an IlvPrototypeAccessor might mess up the IlvGroupInspector.

[ライブラリ]: prstudio

[パッケージ]: protos

PATCH 50.0179

[説明]: In some circumstances, Studio can crash when you add or remove a class from the application buffer.

[ライブラリ]: ilvstappli

[パッケージ]: gadgets

PATCH 50.0180

[説明]: [Sutdio] When a panel instance is reset in the application buffer, possible subpanels are not taken into account.

[ライブラリ]: ilvstappli

[パッケージ]: gadgets

PATCH 50.0181

[説明]: Using a displayer with child displayers, clicking its representation in the "Data sets" tab makes studio crash.

[ライブラリ]: ivstudio

[パッケージ]: foundation

PATCH 50.0182

[説明]: New Feature: Disable the Marking Menu in Studio.

[ライブラリ]: ivstudio

[パッケージ]: foundation

PATCH 50.0183

[説明]: `IlvAbstractMatrix::scrollTo` may not work as expected if the matrix has fixed rows.

[ライブラリ]: ilvadvgdt

[パッケージ]: gadgets

PATCH 50.0184

[説明]: `IlvChartGraphic::removeOrdinateScales` doesn't empty the `IlvAxisElement` array.

[ライブラリ]: ilvcharts

[パッケージ]: charts

PATCH 50.0185

[説明]: [ps print] wrong scaling factor for Japanese.

[ライブラリ]: xvviews, mvviews, winviews

[パッケージ]: foundation

PATCH 50.0186

[説明]: `IlvSingleScaleDisplayer::getCursorByName` is not protected against unnamed cursors.

[ライブラリ]: `ilvcharts`

[パッケージ]: charts

PATCH 50.0187

[説明]: Possible crash in Windows XP look & feel.

[ライブラリ]: `ilvwxplook, ilvawxplook`

[パッケージ]: foundation, gadgets

PATCH 50.0188

[説明]: `FMR`: an `IlvCoordinateInfo` that no longer exists is used during the `IlvChartGraphic` deletion.

[ライブラリ]: `ilvcharts`

[パッケージ]: charts

PATCH 50.0189

[説明]: The data associated with the `_ilvSelected` symbol set on a chart graphic by the `IlvChartSelectInteractor` may be invalid.

[ライブラリ]: `ilvcharts`

[パッケージ]: charts

PATCH 50.0190

[説明]: [Cursor] Label may not appear under some circumstances after patches 500.095 and 147.

[ライブラリ]: `ilvcharts`

[パッケージ]: charts

PATCH 50.0191

[説明]: `IlvFileBrowser` should prompt a message if saving on an existing file.

[ライブラリ]: `ilvadvgdt`

[パッケージ]: gadgets

PATCH 50.0192

[説明]: `IlvGadget::computePalettes` changes the `IlvPalette` instance if antialiasing is not `IlvNoAntialiasingMode`.

[gadgets] It is no longer possible to change the `gadgetdefault` colors using resources.

[ライブラリ]: views, winviews

[パッケージ]: foundation

PATCH 50.0193

[説明]: In some circumstances, Studio can generate an empty #include for an unknown container class.

[ライブラリ]: ilvstappli

[パッケージ]: gadgets

PATCH 50.0194

[説明]: IliSimpleToggle edition does not work fine in the first column of a table gadget.

[ライブラリ]: dbgadget

[パッケージ]: dataaccess

PATCH 50.0195

[説明]: IlvPrinterPreviewPort crashes drawing clipped transformed strings.

[ライブラリ]: ilvprint

[パッケージ]: foundation

PATCH 50.0196

[説明]: [EventPlayer] Popup menus are not getting pointer moved events inside a modal window.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0197

[説明]: IliFormat rounds a 5 in the decimal part as +0, not as +1.

[ライブラリ]: dataaccess

[パッケージ]: dataaccess

PATCH 50.0198

[説明]: IlvPort::drawLabel with IlvCenter horizontal alignment might shift 1 pixel to the left.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0199

[説明]: [Proto] Studio Inspector: need to validate on exit cell attribute, not just when hitting Return.

[ライブラリ]: prstudio

[パッケージ]: protos

PATCH 50.0200

[説明]: Resized IlvZoomableLabels do not appear in the preview.

[ライブラリ]: ilvprint

[パッケージ]: foundation

PATCH 50.0201

[説明]: IlvStereographicProjection and IlvTransverseMercatorProjection are incorrectly serialized.

[ライブラリ]: ilvmaps

[パッケージ]: maps

PATCH 50.0202

[説明]: Tear-off menu doesn't get rid of the possible mnemonic in its title.

[ライブラリ]: ilvgadgt

[パッケージ]: foundation

PATCH 50.0203

[説明]: [Cursor] Cursor is clipped to the data area since latest patches.

[ライブラリ]: ilvcharts

[パッケージ]: charts

PATCH 50.0204

[説明]: [printing] Postscript page setup issue with color selector.

[ライブラリ]: ilvprtdlg

[パッケージ]: gadgets

PATCH 50.0205

[説明]: [printing] Wrong clipping when printing a manager view in keep-ratio mode (default).

[printing] IlvRectangularScale does not print correctly.

[ライブラリ]: ilvprint, ilvmgrprint

[パッケージ]: foundation, manager

PATCH 50.0206

[説明]: [printing] Postscript options do not affect the preview and final ps file.

[ライブラリ]: ilvprtdlg

[パッケージ]: gadgets

PATCH 50.0207

[説明]: [printing] IlvRectangularScale does not printed correctly.

[ライブラリ]: display, ilvprint

[パッケージ]: foundation

PATCH 50.0208

[説明]: Views does not handle Unicode clipboard strings correctly.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0209

[説明]: [Protos] CTRL+M inspector shortcut on Solaris causes problems with the Enter numpad key (numlock off).

[ライブラリ]:

その他のファイル: studio/data/ivstudio/protos/interface.ilv

[パッケージ]: protos

PATCH 50.0210

[説明]: [Printing] Preview is not updated when changing the page orientation.

[printing] Postscript options do not affect the preview and final ps file.

[ライブラリ]: ilvprtdlg

[パッケージ]: gadgets

PATCH 50.0211

[説明]: IlvSystemPort::GetBitmapData does bad if rect not at 0, 0.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0212

[説明]: [Printing] preview scrollbars do not support panel resizing.

[ライブラリ]: ilvprtdlg

[パッケージ]: gadgets

PATCH 50.0213

[説明]: Removing a data point may cause a crash at IlvChartGraphic::dataPointsRemoved.

[ライブラリ]: ilvcharts

[パッケージ]: charts

PATCH 50.0214

[説明]: Using a scale with number of steps == 1 may lead to a crash.

[ライブラリ]: ivstudio

[パッケージ]: foundation

PATCH 50.0215

[説明]: [Pie chart] Floating point exception with a unique value equals to zero.

[ライブラリ]: ilvcharts

[パッケージ]: charts

PATCH 50.0216

[説明]: IlvMatrix (with fixed rows)/IlvSheet does not scrolls to the first scrollable row with PageUp.

[ライブラリ]: ilvadvgdt

[パッケージ]: gadgets

PATCH 50.0217

[説明]: The two accelerators based on the F6 key and that allow to navigate through a set of IlvViewFrame are swapped.

[ライブラリ]: ilvadvgdt

[パッケージ]: gadgets

PATCH 50.0218

[説明]: Printing an IlvGanttChart to a printer produces an unclipped output.

[ライブラリ]: ilvgantt

[パッケージ]: gantt

PATCH 50.0219

[説明]: IliDbNavigator: The setSentine (IliTrue) method does not work for user buttons.

[ライブラリ]: dbgadget

[パッケージ]: dataaccess

PATCH 50.0220

[説明]: IlvManager::allContains doesn't check the visibility of the objects stored in the highest layer.

[ライブラリ]: ilvmgr

[パッケージ]: マネージャ

PATCH 50.0221

[説明] : Crash in studio when modifying a scale property and adding a new scale before applying it.

[ライブラリ] : ilvstcharts

[パッケージ] : charts

PATCH 50.0222

[説明] : `IlvMapCompass::computeNorths()` does not check that the view transformer is null.

[ライブラリ] : ilvmaps

[パッケージ] : maps

PATCH 50.0223

[説明] : `IlvMapTileLoader` ignore all features of tiles whose first featured is not be ignored.

[ライブラリ] : ilvmaps

[パッケージ] : maps

PATCH 50.0224

[説明] : On Windows, points coordinates of polylines and polygons are still clipped to `[SHRT_MIN/2, SHRT_MAX/2]`.

[ライブラリ] : winviews

[パッケージ] : foundation

PATCH 50.0225

[説明] : `[Windows]ZoomableLabels`: very different fonts after a rotation.

[ライブラリ] : winviews

[パッケージ] : foundation

PATCH 50.0226

[説明] : `IlvPrinterPreviewDialog::initData` doesn't look for `printing.dbm` in the resource file.

[ライブラリ] : ilvprtdlg, ivstudio

その他のファイル : `studio/data/res/ivfstudio.rc`, `studio/<pf>/ivfstudio<exeExt>`

[パッケージ] : gadgets, foundation

PATCH 50.0227

[説明] : `IlvPSColorDevice` doesn't transform correctly colors into shades of gray.

[ライブラリ]: display

[パッケージ]: foundation

PATCH 50.0228

[説明]: On Windows, `IlvDisplay::getClipboard` doesn't lock the pool of char allocated to store the data.

[ライブラリ]: ilvatext, ilvadvgdt, ilvmgr

[パッケージ]: gadgets, manager

PATCH 50.0229

[説明]: Printed out strings may appear slightly truncated.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0230

[説明]: [Views 5.0] `IlvZoomInteractor` does not take the into account the default cursor given in the constructor.

[ライブラリ]: ilvmgr

[パッケージ]: マネージャ

PATCH 50.0231

[説明]: [AnnoText] Memory leak seen in the `IlvATHtmlReader`.

[ライブラリ]: ilvatext

[パッケージ]: gadgets

PATCH 50.0232

[説明]: `IlvDvDocument::revert` checks `IlvDvMessageDialog::popup()` result incorrectly.

[ライブラリ]: ilvappframe, ilvappmgr

[パッケージ]: appframe

PATCH 50.0233

[説明]: [X11]Memory leaks in X11 intialization.

[ライブラリ]: xviews, mviews

[パッケージ]: foundation

PATCH 50.0234

[説明]: [X11]Patch 50.0233 may cause a crash when the locale is not C.

[ライブラリ]: xviews, mviews

[パッケージ]: foundation

PATCH 50.0235

[説明]: [Windows] The Printer property dialog is not transient in ivfstudio.

[ライブラリ]: ivstudio

[パッケージ]: foundation

PATCH 50.0236

[説明]: Bad performance when doing bulk appendRow/deleteRow updated connected to an IliChartGraphic or to a gadget with several data sources.

[ライブラリ]: dataaccess, dbgadget

[パッケージ]: dataaccess

PATCH 50.0237

[説明]: Ivstudio may crash when a prototype instance is saved as an ActiveX.

[ライブラリ]: views, ilvstdynact

[パッケージ]: foundation

PATCH 50.0238

[説明]: IlvView objects do not receive focus in/out events with Motif libraries.

[ライブラリ]: mviews

[パッケージ]: foundation

PATCH 50.0239

[説明]: [Protos][Regression] Studio crashes when deleting a prototype component from the Group Inspector panel.

[ライブラリ]: prstudio

[パッケージ]: protos

PATCH 50.0241

[説明]: Scrolling an IlvSheet by clicking in the scrollable area of the scrollbar misses a row.

[ライブラリ]: ilvadvgdt

[パッケージ]: gadgets

PATCH 50.0242

[説明]: Crash deleting the container from the "focus in" callback of a button.

[ライブラリ]: ilvgadgt

[パッケージ]: foundation

PATCH 50.0243

[説明] : The event player cannot replay events in subwindows of modal views.

[ライブラリ] : xviews, mviews

[パッケージ] : foundation

PATCH 50.0244

[説明] : Round rectangle right and lower edges are not printed in preview.

[ライブラリ] : display

[パッケージ] : foundation

PATCH 50.0245

[説明] : [Studio] Layer names start with "Layer 1", but layer indexes in "Draw -> Layer" start at 0.

[ライブラリ] : ivstudio

[パッケージ] : foundation

PATCH 50.0246

[説明] : SDO Object model access to database do not release IldRequest objects properly.

[ライブラリ] : ilvdbmaps

[パッケージ] : maps

PATCH 50.0247

[説明] : Popup menu timer for hiding child popups fails if child popup has a child itself.

[ライブラリ] : ilvgadgt

[パッケージ] : foundation

PATCH 50.0248

[説明] : [Gantt] [IlvRectangularScale] Scale ticks and grid do not match exactly.

[ライブラリ] : winviews

[パッケージ] : foundation

PATCH 50.0249

[説明] : [Windows] Wrong font size since patch50.0225.

[ライブラリ] : winviews

[パッケージ] : foundation

PATCH 50.0250

[説明] : [Windows] IlvZoomableIcon are not always printed correctly.

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0252

[説明]: Popup menu timer for hiding child popups fails if child popup has a child itself.

[ライブラリ]: ilvgadgt

[パッケージ]: foundation

PATCH 50.0253

[説明]: IlvWindowsDevice::drawBitmap(pal, src, rect, point) is incorrect if the rectangle origin is not (0, 0).

[ライブラリ]: winviews

[パッケージ]: foundation

PATCH 50.0254

[説明]: [X1] On the Sun French/Spanish keyboards, Circumflex and Diaresis are not handled in IlvTextField.

[ライブラリ]: xviews, mviews

[パッケージ]: foundation

リリース・ノート – バージョン 5.0

本章では、バージョン 4 以降 ILOG Views に行われた変更について説明します。本章は、以下のトピックで構成されています。

- ◆ 新しい機能
- ◆ コードの変更

新しい機能

ILOG Views 5.0 の新しい機能を、次のセクションに分けて説明します。

- ◆ *Foundation* の機能
- ◆ *Studio* の機能
- ◆ *Gadgets* の機能
- ◆ *Application Framework* の機能
- ◆ *Manager* の機能
- ◆ *Data Access* の機能
- ◆ *Charts* の機能
- ◆ *Gantt* の機能

◆ *Maps* の機能

Foundation の機能

ILOG Views 5.0 に以下の新しい機能が追加されました。

フィルタ

◆ ILOG Views Foundation に、SVG フィルタと同じ機能を提供する次のようなイメージ処理クラスが追加されました。

- IlvBlendFilter
- IlvColorMatrixFilter
- IlvSaturationFilter
- IlvLuminanceToAlphaFilter
- IlvHueRotateFilter
- IlvComponentTransferFilter
- IlvComposeFilter
- IlvConvolutionFilter
- IlvDisplaceFilter
- IlvFloodFilter
- IlvGaussianBlurFilter
- IlvImageFilter
- IlvLightingFilter
- IlvDiffuseLightingFilter
- IlvSpecularLightingFilter
- IlvMergeFilter
- IlvMorphologyFilter
- IlvOffsetFilter
- IlvTileFilter
- IlvTurbulenceFilter
- IlvFilterFlow

および対応クラス：

- IlvTransferFunction
- IlvBitmapDataKernel
- IlvLightSource
- IlvDistantLight

- IlvPointLight
- IlvSpotLight
- IlvPerlinNoise
- IlvIdentityTransfer
- IlvLinearTransfer
- IlvTableTransfer
- IlvDiscreteTransfer
- IlvGammaTransfer
- IlvIdentityTransfer

これらのフィルタを SVG 形式に似た XML 形式で記述し、IlvRGBBitmapData イメージに適用することができます。これらを連結して、複雑なイメージ処理コマンドを作成することができます。

- ◆ グラフィック・オブジェクト IlvFilteredGraphic では、任意の IlvGraphic にフィルタを適用する機能 (イメージ処理操作のフロー) が提供されるようになりました。

印刷

- ◆ ILOG Views Foundation に、ILOG Views アプリケーションでの印刷を容易にする印刷フレームワークが追加されました。このフレームワークは、以下のクラスで構成されています。
 - IlvPrintableDocument - ドキュメントを定義
 - IlvPrintable - 以下のサブクラスで、印刷可能なオブジェクトを記述
 - IlvPrintableContainer
 - IlvPrintableText
 - IlvPrintableFormattedText
 - IlvPrintableGraphic
 - IlvPrintableFrame
 - IlvPrintableManager、IlvPrintableMgrView、および IlvPrintableManagerLayer (Manager パッケージでのみ使用可能)
 - IlvPrintableComposite
 - IlvPrintableLayout - 以下の定義済みレイアウトで、ページ・レイアウトを記述
 - IlvPrintableLayoutOnePage
 - IlvPrintableLayoutMultiplePages

- IlvPrintableLayoutIdentity
- IlvPrintableLayoutFixedSize
- IlvPrinter - 以下のサブクラスで、プリンタを記述
 - IlvPSPrinter
 - IlvWindowsPrinter
- IlvPrintUnit - 定義済み単位
 - IlvPrintPointUnit
 - IlvPrintCMUnit
 - IlvPrintInchUnit
 - IlvPrintPicaUnit
- IlvPaperFormat - 登録済み用紙フォーマットを選択
- 定義済みユーザ・インターフェース・ダイアログ
IlvPostScriptPrinterDialog および IlvPrinterPreviewDialog - プリンタおよび機能を選択。Gadgets パッケージでは、以上のすべてのクラスで印刷プレビュー・ダイアログも使用できます。
- ◆ 下位レベル Windows 印刷管理のためのクラスが新たに2つ追加されました。
 - IlvWindowsPrinterDCFactory
 - IlvWindowsPrinterDCDialogFactory

ビットマップ・データ

- ◆ IlvPort::stretchBitmapData のピクセル当たりのアルファを使用して、IlvBitmapData を描画するメソッドが新たに追加されました。
- ◆ TIFF イメージ用のビットマップ・ストリーマ IlvTIFFStreamer が新たに追加されました。
- ◆ IlvRGBBitmapData では、非常に硬質なストレッチを実現する様々なフィルタを使用して、フィルタリングされたイメージのリサンプリングが可能になりました。IlvRGBBitmapData::stretchSmooth 新しいメソッドが追加されました。
IlvRGBBitmapData データでは、アルファ乗算ができるようになりました。
- ◆ IlvIndexedBitmapData 内でカラーマップを処理するクラス IlvColorMap が新たに追加されました。

量子化

24 ビット用の新しい量子化、およびインデックス付きカラー変換が追加されました。

- ◆ IlvFixedQuantizer - 固定カラーマップに変換
- ◆ IlvNetscapeQuantizer - Netscape カラーマップに変換

次の量子化用の階層が追加されました。

- ◆ IlvQuantizer
- ◆ IlvFixedQuantizer
- ◆ IlvQuickQuantizer
- ◆ IlvNetscapeQuantizer
- ◆ IlvWUQuantizer
- ◆ PostScript ÉfÉoÉCÉX

PostScript デバイスで、PostScript レベル 2 機能 (デフォルト・モード) を使用できるようになりました。

PostScript デバイスは、レベル 2 の場合、IlvPostScriptEncoder から派生したエンコーダを使用して、イメージを様々な形式でエンコードできるようになりました。次の 3 つのエンコーダが用意されています。

- ◆ IlvASCIHexEncoder - 入力バイトを 16 進法に変換して、1 入力バイトから 2 文字を生成
- ◆ IlvASCII85Encoder - 入力バイトを可視文字セットに変換して、4 入力バイトから 5 文字を生成
- ◆ IlvRunLengthEncoder - 入力バイトを RunLength エンコード形式のバイナリ記述に変換。圧縮は入力に依存。

デフォルトのエンコーダは、レベル 1 の場合 IlvASCIHexEncoder、レベル 2 の場合 IlvASCII85Encoder です。

Windows の透明度

Windows プラットフォーム 98、2000、および XP の場合、IlvView は、最上位ウィンドウのときに部分的に透明にでき、完全透明度をカラー値によって定義することができます。レイヤ・ウィンドウなどの Windows の新しい機能を使用できます。以下のメソッドが追加されました。

- ◆ IlvAbstractView::setOpacity
- ◆ IlvAbstractView::getOpacity
- ◆ IlvAbstractView::setTransparentColorKey
- ◆ IlvAbstractView::getTransparentColorKey
- ◆ IlvAbstractView::setLayered

これらのメソッドは、他のプラットフォームでは無効です。

グラフィック・レンダリング

ILOG Views の下位レベルの描画プリミティブ API は、透明度とアンチエイリアシングをサポートするように機能強化されました。これらの機能は、グローバル設定の場合はポート・(IlvPort) レベル、ローカル設定の場合はパレット (IlvPalette) レベルというように、異なるレベルでアクセス可能です。詳細については、『Foundation ユーザ・マニュアル』の「*IlvResource: グラフィック・リソース・クラス*」にあるアルファ値およびアンチエイリアシング・モードを参照してください。

これらの機能は、Windows プラットフォームで、GDI+ をインストールした状態でのみ使用できます。詳細については、『Foundation ユーザ・マニュアル』の「付録 B」にある ILOG Views での GDI+ 機能を使用するを参照してください。

Studio の機能

ILOG Studio では、以前使用されていた機構に取って代わる新しい印刷パッケージが導入されました。

Gadgets の機能

ルック・アンド・フィールドの管理

- ◆ Windows XP のルック・アンド・フィールドは、Windows XP を実行するプラットフォームでサポートされています。
- ◆ ルック・アンド・フィールドは、異なるレベル、IlvGadget、IlvGadgetContainer、および IlvDisplay。これによって、複数のルック・アンド・フィールドを同じアプリケーションで組み合わせることができます。詳細については、『Gadgets ユーザ・マニュアル』の「Gadgets のルック・アンド・フィールド」を参照してください。
- ◆ 独自のルック・アンド・フィールド・ハンドラを作成できるようになりました。詳細については、『Gadgets ユーザ・マニュアル』の「Gadgets のルック・アンド・フィールド」を参照してください。
- ◆ 既存のルック・アンド・フィールド (Motif、Windows 3.11、Windows 95、および Windows XP) は、別のライブラリに格納されています。アプリケーションをリンクさせるために、make ファイルの変更が必要な場合もあります。詳細については、コードの変更にあるルック・アンド・フィールドの管理を参照してください。

IlvViewFrame

IlvViewFrame オブジェクトは、Motif 用に変更され、外観が Motif コンポーネントのようになりました。

ドッキング・ペイン

ドッキング・ハンドルの外観が変更されました。ドッキング・ハンドルは、関連ペインのタイトルを表示するようになりました。ただし、スタティック関数メソッド `IlvDockingHandlePane::UseDefaultHandlePanes` を使用して、以前の外観を保持することができます。

IlvComboBox and IlvScrolledComboBox

`IlvComboBox` のリストが開いている場合、`OpenList` コールバックが呼び出されます。詳細については、`IlvComboBox::OpenListSymbol` 関数を参照してください。

`IlvComboBox` リストが開いている場合、キーボード・フォーカスは引き続きテキスト・フィールドに残り、リストが表示されている間に文字を入力することができます。リストの選択は、テキスト・フィールドに何があるかによって変わります。

IlvMatrix

任意の `IlvGraphic` インスタンスで、マトリックス・アイテムを編集できるようになりました。詳細については、`IlvMatrixItemEditor` クラスを参照してください。

Application Framework の機能

アプリケーション・フレームワーク・エディタに、アプリケーションにダイアログ・ボックスおよびウィンドウを追加するための新しいパレット項目「Dialogs」が装備されました。

エディタの詳細については、『*ILOG Views Application Framework ユーザ・マニュアル*』の「アプリケーション・フレームワーク・エディタの使用」を参照してください。

Manager の機能

マネージャ・レイヤ

アルファ値およびアンチエイリアシング・モードを指定することによって、マネージャ・レイヤのレンダリングを変更できるようになりました。

アルファ値は、レイヤの透明度を制御し、このレイヤの下に描画されたレイヤを通してオブジェクトが見えるようになります。詳細については、`IlvManagerLayer::setAlpha` メンバ・メソッドを参照してください。

`ÉAÉiÉ`ÉGÉCÉäÉAÉVÉiÉO` モードは、レイヤの各オブジェクトにアンチエイリアシング・モードを設定するより簡単な方法でレイヤ全体のアンチエイリアシング・モードを設定します。詳細については、`IlvManagerLayer::setAntialiasingMode` メソッドを参照してください。

詳細については、『*Manager ユーザ・マニュアル*』（2D Graphics / Manager の基本機能）にある *Manager レイヤ* を参照してください。

Data Access の機能

ILOG Views 5.0 に以下の新しい機能が追加されました。

- ◆ 新しいデータベース・アクセス、DB2 が追加されました。
 - ◆ Dbschema が DbStudio または Data Access SQL プラグインの SQL Tables パレットに組み込まれています。
 - ◆ DbStudio の multidb フォーマットは、Data Access プラグインおよび ivfstudio に置き換えられました。
-

Charts の機能

IlvBarChartDisplayer、IlvHiLoChartDisplayer

バーの幅を、チャートの全体幅に比例して設定できるようになりました。

- ◆ `IlvBarChartDisplayer::setWidthPercent`
- ◆ `IlvHiLoChartDisplayer::setWidthPercent`

IlvChartCyclicPointSet

サイクリック・データ・セット・クラス `IlvChartCyclicPointSet` が新たに追加されました。

IlvChartGradientPointInfo

色の階調を定義するクラス `IlvChartGradientPointInfo` が新たに追加されました。

IlvXMLChartData

XML ファイルからデータ・セットを読み込むためのクラス `IlvXMLChartData` が新たに追加されました。

Gantt の機能

次は、ILOG Views Gantt バージョン 5.0 用の新しい機能をまとめたものです。

- ◆ `IlvGanttChartForm` で時間尺をサポートします。Foundation モジュールが提供する時間尺を `IlvGanttChartForm` で使用できます。動的カレンダー・グリッド `IlvGanttCalendarGrid` も用意されています。詳細はサンプル `<ILVHOME>/samples/gantt/calendar` を参照してください。
- ◆ ガントチャートに、新しいグリッド・ファミリー・クラスが追加されました。

- ◆ グラファァー・ビューに表示されるグリッドの外観は、`IlvPalette` を指定することによってカスタマイズできます。
- ◆ グラファァー・ビューは、幅の狭いボーダーを備えています。
- ◆ ガントのノードでツールチップを使用可能にできます。
- ◆ `IlvGanttSelectInteractor` によって、ガントのノードやラインを選択したり移動したりできます。
- ◆ グラファァー・ビュー間のハンドルを `IlvFilledRectangle` から `IlvReliefDiamond` に変更することで外観が向上し、グラファァー・ビュー間の領域も利用しやすくなりました。
- ◆ 作業領域の変更を通知できるように、コールバック `ork Area` を新たに `IlvGanttChart` に追加しました。
- ◆ サンプルが新たに追加されました。既存のサンプルは、大幅に改訂されました。
- ◆ `ILOG Views Gantt ユーザ・マニュアル` を大幅に書き換えました。

Maps の機能

このセクションでは、`ILOG Views Maps` のバージョン 4 以降行われた主な変更について説明します。

ILOG Views Maps クラスのコード変更も参照してください。

ロード・オン・デマンド・フレームワーク

以下の機能が追加されました。

- ◆ タイル・ローダーの作成を簡素化するために、新しいクラス `IlvMapTileLoader` が追加されました。このクラスには、すべての標準レンダリング機能およびすべてのタイル・ローダー用のタイル・ローディング機能が含まれています。
- ◆ `IlvLookAheadFeatureIterator`: 次の機能を読み込む前にその ID を取得する機能を備える `IlvMapFeatureIterator` のインターフェースを定義することによって、必要がない場合は次の機能をスキップすることができます。
- ◆ 形状ファイルのロード・オン・デマンド: `ESRI` 形状ファイルにロード・オン・デマンドのサポートが追加されました。形状ファイルを要求に応じて読み込むために、`.idx` 拡張子を持つ新しいタイプのファイルを導入しました。これらのファイルには、形状ファイルに関連付けられた空間インデックスが含まれています。

ロード・オン・デマンド機能をサポートするために、以下のクラスを追加しました。

- `IlvShapeFileIndex`: 新しく導入された形状ファイル・インデックス・ファイル (`.shx` 拡張子) を読み込みます。
- `IlvShapeSpatialIndex`: 新しく導入された空間インデックス・ファイル (`.idx` 拡張子) を読み込みます。
- `IlvShapeFileTileLoader`: 形状ファイル用ローダーで、`IlvTiledLayer` と併用する場合にロード・オン・デマンド・レイヤを作成します。
- `IlvShapeFileLayer`: `.ilv` ファイルに保存できるタイル状の形状ファイル・レイヤ。
- `IlvShapeFileTiler` クラスを追加し、空間インデックス・ファイルを生成できるようになりました。

新たにサポートされるようになったフォーマット

単一 `GeoTIFF` ファイルまたはタイル状の `GeoTIFF` ファイルの読み込みサポートを追加しました。

`GeoTIFF` フォーマットをサポートするために、以下のクラスが追加されました。

- ◆ `IlvGeoTIFFReader`: `GeoTIFF` ファイルのリーダー。
- ◆ `IlvGeoTIFFTileLoader`: タイル状の `GeoTIFF` ファイルを読み込むためのタイル・ローダー。
- ◆ `IlvGeoTIFFLayer`: `.ilv` ファイルに保存できるタイル状の `GeoTIFF` レイヤ。
- ◆ `IlvGeoTIFFTiler`: タイル状の `GeoTIFF` ファイルを生成するためのクラス。

イメージ・ファイルを読み込む機能が追加されました。対応イメージ・フォーマットは、`ILOG Views` によってサポートされているものです。

- ◆ `IlvImageReader`: 単一のイメージ・ファイルを読み込むためのクラス。
- ◆ `IlvImageTileLoader`: 各イメージがタイル状データ・セットの1つのタイルを表すイメージ・セットを読み込むためのクラス。
- ◆ `IlvImageLayer`: `.ilv` ファイルに保存できるタイル状の形状イメージ・レイヤ。

ファイル名の解決

ファイル名のリゾルバ機構が追加されました。タイル・ローダーがこの機構を使用して、パス名を解決します。

以下のクラスは、ファイル名リゾルバ機構をサポートしています。

- ◆ `IlvMapDataPathManager`: パス・リゾルバを管理するクラス。
- ◆ `IlvDefaultDataPathResolver`: ディレクトリにあるファイルを検索する `IlvMapDataPathResolver` のデフォルト実装。

- ◆ `IlvMapDataPathResolver`: すべてのパス・リゾルバの抽象ベース・クラス。

Oracle Spatial

- ◆ `IlvObjectSDOWriter::setCheckingStringAttributes` は、文字列アトリビュートのチェックを有効/無効にします。
- ◆ `IlvDefaultObjectSDOTileLoader::setRequestParameters` は、`IlvTile` が使用する要求を調整するように、いくつかのパラメータを設定します。
- ◆ `IlvObjectSDOUtil::GetLayerExtent` は、**SDO** レイヤの範囲 (すべてのジオメトリのグローバル・バウンディング・ボックス) を計算します。

サンプル

- ◆ 形状ファイルをどのように読み込んで並べるかを示す新しいサンプルが追加されました。
- ◆ **GeoTIFF** ファイルをどのように読み込んで並べるかを示す新しいサンプルが追加されました。
- ◆ マルチ解像度を並べて表示したデータ・セット (形状ファイルおよびイメージ) をどのように読み込むかを示す新しいサンプルが追加されました。
- ◆ 形状サンプルによって、形状ファイルのロード・オン・デマンドを並べて表示できるようになりました。

マップ・ビルダ

以下の機能が新たに追加されました。

- ◆ マップ・ビルダでは、データ CD にあるマルチ解像度データを読み込むことができるようになりました。
- ◆ 概要が追加されました。
- ◆ 新規凡例によって、レイヤをドラッグ・アンド・ドロップして、外観の順序を変更できるようになりました。
- ◆ 新規メニューを凡例で有効にすることによって、各レイヤのレンダリング・パレットを動的に変更できるようになりました。
- ◆ **GeoTIFF** および形状ファイルのロード・オン・デマンドをサポートするようになりました。

コードの変更

このセクションでは、バージョン 4 以降行われた既存の **ILOG Views** および **ILOG Script for ILOG Views** クラスにおけるコード変更について説明します。

- ◆ *ILOG Views* ファンデーション・クラス
- ◆ *ILOG Views Charts* クラス
- ◆ *ILOG Views Gadgets* クラス
- ◆ *ILOG Views Gantt Chart* クラス
- ◆ *ILOG Views Maps* クラス

ILOG Views ファンデーション・クラス

ILOG Views ファンデーション・クラスでは、次のコード変更が行われました。

IlvColorMap

- ◆ `IlvIndexedBitmapData` では、`IlvColorMap` を使用するようになりました。
- ◆ 量子化はモジュール化され、`IlvColorMap` を使用するようになりました。

IlvDisplay

メソッド `IlvDisplay::getBitmap` に、2 番目のパラメータが加わりました。プロトタイプは次のようになります。

```
IlvBitmap* getBitmap(const char* name, IlvBoolean read = IlvFalse) const;
```

`IlvTrue` に設定すると、`read` パラメータは、ビットマップが内部表示キャッシュ・リストにない場合、`IlvDisplay::readBitmap` メソッドを使用して読み込む必要があることを示します。

IlvPort

Windows プラットフォームでは、`IlvPort::drawTransformedString` メソッドが表示バグに対処するよう振る舞いに変更になりました。これによって、短い文字列が表示される場合があります。二次作用として、このメソッドを使用する `IlvZoomableLabel` オブジェクトの描画が、同じように変更になることがあります。

IlvBitmap

Views 5.0 以降、Windows プラットフォームでは、ディスプレイの奥行きが > 8 の場合、`IlvBitmapData` を内部で受け入れる `IlvBitmap` コンストラクタは、`IlvBitmapData` の奥行き `DIBSection` を使用するようになりました。これによって、以下のようにになります。

- ◆ `IlvIndexedBitmapData` を使用する場合、割り当てられたメモリは、ILOG Views 5.0 に割り当てられたメモリの ? になります。

- ◆ 作成された `IlvBitmap` インスタンスのサイズおよび数は、使用可能なシステム・メモリによってのみ制限されます。以前の **ILOG Views** のバージョンでは、システムによる制限 (`IlvBitmap` では約 48 MB、および Windows NT のすべての `IlvBitmap` インスタンスでは 192 MB の制限) がありました。
- ◆ `IlvIndexedBitmapData` を使用すると、ツール・カラーへの変換が必要ないため、パフォーマンスがはるかに向上します。
- ◆ `IlvBitmap` の奥行きは、ディスプレイの奥行きと常に等しくなるとは限りません。そのため、奥行き 8 の `IlvBitmap` で **ILOG Views** を使用して描画することによって、基底となる `DIBSection` のカラーマップが、描画に使用するカラーに正確には一致しないため、予期せぬ結果を招くことがあります。

IlvView

Windows の場合、以下のメソッドで、タスク・バーを考慮しないようになりました。

- ◆ `IlvView::ensureInScreen`
- ◆ `IlvView::moveToScreen`

IlvWindowsDevice

`IlvWindowsDevice` コンストラクタは、タイプ `IlvWindowsPrinterFactory` の追加のオプション・パラメータを備えています。

IlvWindowsVirtualDevice

- ◆ `IlvWindowsVirtualDevice::getHDC` の戻り値タイプは、現在では `HDC` であり、`unsigned long` ではありません。
- ◆ `IlvWindowsVirtualDevice::setHDC` のパラメータのタイプは、現在では `HDC` であり、`unsigned long` ではありません。

Typedef の変更

`IlvWindowProc` の typedef は、将来的にポートを Windows 64 ビットで使用できるように変更されました。現在は以下のとおりです。

```
typedef IlvBoolean (* IlvWindowProc) (IlvAbstractView* view,
                                     IlvSystemView hWnd,
                                     IlvMsgParam iMessage,
                                     IlvWParam wParam,
                                     IlvLParam lParam,
                                     IlvAny& returned,
                                     IlvAny userArg);
```

`IlvMsgParam`、`IlvWParam`、および `IlvLParam` は新しい 3 つの typedef であり、Windows 32 ビット用ではそれぞれ `unsigned int`、`unsigned int`、and `long` となります。つまり、以前のバージョンの **ILOG Views** 用に書かれたコードは、まだコンパイル

でき、動作することを意味します。ただし、このタイプは、Windows 64 ビット用とは異なり、移植可能な新しいタイプを使用する必要があります。

ポートは Windows64 ビットではまだ使用できないことに注意してください。詳細は未定です。

ILOG Views Charts クラス

以前の ILOG Views Charts クラス

新しいチャート (ilvcharts ライブラリにあります) を、以前のチャート (互換性のあるパッケージに提供されている views31 ライブラリにあります) と併用することができます。古いチャートと新しいチャートを共存させるために、以下のクラスおよびマクロの名前が変更になりました。

- ◆ IlvChartDataConstructor -> Ilv31ChartDataConstructor
- ◆ IlvChartDataClassInfo -> Ilv31ChartDataClassInfo
- ◆ IlvPreRegisterChartDataClass -> Ilv31PreRegisterChartDataClass
- ◆ IlvPostRegisterChartDataClass -> Ilv31PostRegisterChartDataClass
- ◆ IlvPredefinedChartDataIOMembers -> Ilv31PredefinedChartDataIOMembers

古いチャートと新しいチャートを同時に使用するには、クラスやマクロの古いチャート・コードでの使用状態をチェックして、それらを変更します。

メモ: 同時に使用しない場合は、古いチャート・コードを変更する必要はありません。

使用されなくなったメソッド

以下のメソッドは、本バージョンから使用できなくなりました。

IlvAbstractChartDisplayer

- ◆ `IlvAbstractChartDisplayer::getOrdinateInfo()`
- ◆ `IlvAbstractScaleDisplayer::getCoordinateInfo(IIUInt)`

IlvSingleScaleDisplayer

- ◆ `IlvSingleScaleDisplayer::getCoordinateInfo(IIUInt)`
- ◆ `IlvSingleScaleDisplayer::getGridDisplayer()`
- ◆ `IlvSingleScaleDisplayer::setGridDisplayer(IlvAbstractGridDisplayer*)`

IlvMultiScaleDisplayer

- ◆ `IlvMultiScaleDisplayer::getCoordinateInfo (IUInt)`

IlvScaleStepsUpdater

- ◆ `IlvScaleStepsUpdater::getCoordinateInfo ()`

ILOG Views Gadgets クラス***IlvMessageLabel***

`null` 値以外の幅の `IlvMessageLabel` オブジェクトは、そのフレームを立体矩形で描画するようになりました。ラベルは、それに応じてシフトされます。以前の振る舞いを保つには、つまり、`IlvMessageLabel` のフレームを除去するには、パラメータとして `0` で `IlvGadget::setThickness` メソッドを呼び出すか、パラメータとして `ILFalse` で `IlvGadget::showFrame` メソッドを呼び出します。

ルック・アンド・フィールの管理

ILOG Views は、定義済みのルック・アンド・フィールを別ライブラリで提供するようになりました。つまり、アプリケーションが使用するルック・ライブラリを追加するために、`make` ファイルを変更する必要があることを意味します。共有ライブラリを使用している場合は、ルック・アンド・フィール・ライブラリへのリンクを避けるために、`ILVNODEFAULTLOOK` 記号でコンパイルする必要が生じる場合があります。

詳細については、*ユーザ・マニュアル(Controls/Gadgets/Part II/Gadgets の理解)* の *Gadgets* のルック・アンド・フィールを参照してください。以下の表は、新しいライブラリをまとめたものです。

表 5.1 *Windows* プラットフォーム用ルック・ライブラリ

ルック	標準ガジェット・ライブラリ	高度なガジェット・ライブラリ
Motif	<code>ilvmllook.lib</code>	<code>ilvamlook.lib</code>
Windows 3.11	<code>ilvwlook.lib</code>	<code>ilvawlook.lib</code>
Windows 95	<code>ilvw95look.lib</code> 、 <code>ilvwlook.lib</code>	<code>ilvaw95look.lib</code> 、 <code>ilvawlook.lib</code>
Windows XP	<code>ilvwxplook.lib</code> 、 <code>ilvw95look.lib</code> 、 <code>ilvwlook.lib</code>	<code>ilvawxplook.lib</code> 、 <code>ilvaw95look.lib</code> 、 <code>ilvawlook.lib</code>

表5.2 UNIX プラットフォーム用ルック・ライブラリ

ルック	標準ガジェット・ライブラリ	高度なガジェット・ライブラリ
Motif	libilvmlook	libilvamlook
Windows 3.11	libilvwlook	libilvawlook
Windows 95	libilvw95look、 libilvwlook	libilvaw95look、 libilvawlook

ルック・アンド・フィールドのコールバック

ルックが変更されると呼び出されるコールバック

(`IlvDisplay::addChangeLookCallback` を参照してください) は、ルック・アンド・フィールドを使用するアプリケーションが起動すると呼び出されるようになりました。

ドッキング・ペイン

ドッキング・バーの管理は、ILOG Views 4.0 に比べて若干変更になりました。新しい振る舞いによって、ドッキング可能なコンテナの最後の可視ドッキング・バーが必ず伸縮バーになります。これは、ドック可能なコンテナのサイズが変更になると、最後のドッキング・バーのみのサイズが変更されることを意味します。

ILOG Views Gantt Chart クラス

新しいクラスおよびメソッド

`IlvGanttChart` には、以下の新しいクラスがあります。

- ◆ `IlvGanttHorizontalGridImpl`
- ◆ `IlvGanttHorizontalGrid`
- ◆ `IlvGanttLineGrid`
- ◆ `IlvGanttLineReliefGrid`
- ◆ `IlvGanttResourceGrid`
- ◆ `IlvGanttRowGrid`
- ◆ `IlvGanttCalendarGrid`

`IlvGanttChart` には、以下の新しいパブリック・メソッドがあります。

- ◆ `IlvPalette* getGridPalette();`
- ◆ `void setGridPalette(IlvPalette*);`

- ◆ `IlBoolean isDrawingGrapherViewBorders();`
- ◆ `void setDrawingGrapherViewBorders(IlBoolean);`
- ◆ `IlSymbol* WorkAreaCallbackType();`
- ◆ `IlvRect& getWorkArea(IlvRect& area);`
- ◆ `IlvRect& getLineArea(IlvRect& area);`
- ◆ `IlvGraphic* setLineGrid(IlvGraphic* grid, IlvBoolean redraw, IlInt layer);`
- ◆ `IlvGraphic* getLineGrid();`
- ◆ `IlvGraphic* setRowGrid(IlvGraphic* grid, IlvBoolean redraw, IlInt layer);`
- ◆ `IlvGraphic* getRowGrid() const;`
- ◆ `void setDoubleBuffering(IlBoolean);`
- ◆ `IlBoolean isDoubleBuffering() const;`

非互換性

`IlvGanttChartForm` の以下のメソッドの署名が変更になりました。

- ◆ `IlvGraphic* setCompositeScale(IlvGraphic*, IlBoolean, IlBoolean);`
- ◆ `IlvGraphic* getCompositeScale() const;`
- ◆ `IlvGraphic* setCompositeGrid(IlvGraphic*, IlvBoolean);`
- ◆ `IlvGraphic* getCompositeGrid() const;`

振る舞いの変更

`IlvGanttChart::rowResized(IlvUShort, IlvPos)` は、新しい「Work Area」コールバックを呼び出します。

`Gantt` ラインのデフォルトのレイヤは、0 から 1 に変更されます。

- ◆ `void addLine(IlvGanttLine* line, IlBoolean redraw, int layer=1);`
- ◆ `void insertLine(IlvGanttLine* line, IlUShort row, IlBoolean redraw, int layer=1);`

使用されなくなったメソッド

`IlvGanttChartForm::setCompositeGrid()` は、本バージョンから使用できなくなりました。代わりに、`IlvGanttChart::setRowGrid()` を使用してください。

サンプル

`Gantt` モジュールのサンプル・フォルダでは、`<ILVHOME>/samples/gantt`、`load` および `month` サンプルが大幅に改定されています。2 つの新しいサンプル `simple` および `calendar` が追加されました。

バグの修正

ILOG Views Gantt には以下のバグがありましたが、現在は修正されています。

- ◆ `IlvGanttChart::psDump` は、グラファー・ビューのグリッドをダンプしてしました。
- ◆ `IlvGanttChart` は、コンテナへの追加前に、適切にサイズ変更できませんでした。
- ◆ `IlvGanttChart` は、グラファー・ビューの間でハンドルをドラッグして、対話式にモノ表示に切り替えることができませんでした。

ILOG Views Maps クラス

IlvShapeFileReader

コンストラクタは、`.shx` ファイルを考慮するように変更されました。

- ◆ `shxFileName` パラメータを `IlvShapeFileReader` コンストラクタに追加しました。
- ◆ `shxIndex` パラメータを `IlvShapeFileReader` コンストラクタに追加しました。

IlvObjectSDOWriter

非互換性を解決するために、文書化した保護メソッドを `IlvObjectSDOWriter` から除去しました。`IlvObjectSDOWriter::getSDOGeometryString` メソッドは削除しました。このクラスでは、Oracle Spatial にジオメトリを保存するために、文字列は使用できなくなりました。

バグの修正

ILOG Views Maps には以下のバグがありましたが、現在は修正されています。

- ◆ デフォルトの `IlvObjectSDOKeyExtractor` は、Number Oracle タイプをサポートしていませんでした。
- ◆ `IlvObjectSDOReader` は、65536 ポイントを上回ると読み込めませんでした。
- ◆ `IlvObjectSDOFeatureIterator` は、誤ったマルチ・エリア・ジオメトリを戻していました。
- ◆ `IlvMapGeneralPath`: 特定の環境下で誤ったバウンディング・ボックス計算を行っていました。
- ◆ Oracle 9i Spatial を使用できませんでした。
- ◆ いくつかの `IlvSDOLayer/IlvObjectSDOLayer` レイヤの読み込み / 書き込みによって、クラッシュを引き起こすことがありました。
- ◆ SDO サンプルは、ILOG DB Link ライセンスが必要でした。

- ◆ ファイルが値のない値付きジオメトリを含んでいる場合、形状リーダーがクラッシュしていました。
- ◆ `IlvMapLoader` は、挿入レイヤを考慮していませんでした。
- ◆ `IlvMapsRegisterLicense` は、ライセンスを正しく登録できませんでした。
- ◆ `DoubleAttributeReader::read` (形状ファイル・リーダー) が、C ロケールに切り替わりませんでした。
- ◆ `IlvDefaultObjectSDOTileLoader` をカスタマイズできませんでした。
- ◆ コンストラクタ `IlvObjectSDOLayer` (`IlvSDOTileLoader*`, `IlvTileCache*`, `const char*`) が動作しませんでした。
- ◆ `IlvMapCoordinateViewer` における FMM。
- ◆ 複数のバッファを交換 / 削除 / 作成するとクラッシュしていました。
- ◆ **ILOG Maps GUI** コンポーネントにアタッチされたビューを削除する際に問題が生じていました。
- ◆ `IlvObjectSDOWriter::populateIndexes` の振る舞いが不正でした。
- ◆ ロード・オン・デマンドの使用時に、アプリケーションの CPU 処理時間がかかり過ぎていました。
- ◆ ライブラリ `ilvdbmaps` で、実行時ライセンスが適切に初期化できませんでした。
- ◆ `IlvObjectSDOWriter` では、いくつかのエラーが起こっていました。
- ◆ `IlvObjectSDOUtil::IsUsing816` が、適切に動作しませんでした。
- ◆ `IlvMapCoordViewer` のデストラクタは、**MLK** を生成していました。
- ◆ `IlvTileController::updateViews` でクラッシュが生じていました。
- ◆ `IlvMapGeneralPath` は、パレットを不正に処理していました (二次作用)。

リリース・ノート - バージョン 4.0.2

本リリース・ノートでは、バージョン 4.0.1 以降 ILOG Views に加えられた変更について説明します。本書は、以下の内容で構成されています。

- ◆ *新しい機能*
- ◆ *システムの変更*

新しい機能

ILOG Views コンポーネント・スイートのバージョン 4.0.2 で以下の新しい機能が追加されました。

- ◆ *Web 配布*
- ◆ *Prototypes*

Web 配布

新しい ILOG Views Web 配布パッケージを使うと、ILOG Views アプリケーションから Web アプリケーションを作成することができます。

詳細については、Web 配布マニュアルを参照してください。

Prototypes

Prototypes パッケージに新しい機能がいくつか追加されました。

- ◆ スイッチ・アクセサ
- ◆ 切り取り/コピー/貼り付け
- ◆ アニメーション・タイマーのオン/オフ

スイッチ・アクセサ

制御アクセサが新たに追加されました。

スイッチ・アクセサ (クラス `IlvSwitchAccessor`) は、`Switch` ステートメントを実装しています。

パラメータ

- ◆ **Switch:** 整数を返す式。その結果に応じて、アトリビュート `0...N` に、パラメータの値が割り当てられます。
- ◆ **case 0:** プロトタイプまたは値 "" のアトリビュート。Switch が 0 に評価した場合、このパラメータで指定されたアトリビュートの振る舞いが実行されます。
- ◆ **case 1:** プロトタイプまたは値 "" のアトリビュート。Switch が 1 に評価した場合、このパラメータで指定されたアトリビュートの振る舞いが実行されます。
- ◆ ...
- ◆ **case N:** プロトタイプまたは値 "" のアトリビュート。Switch が `N` 以上の値に評価した場合、このパラメータで指定されたアトリビュートの振る舞いが実行されます。

例

信号機を、設定に応じて以下のように実装できます。

```
il   êæêi
do
    Switch Value
    case 0 doRed
    case 1 doOrange
    case 2 doGreen
    case 3 Anomaly

doRed
do
    greenEllipse.visible=False
    orangeellipse.visible=False
    redEllipse.visible=True
    doBlink=False

doOrange
do
    greenEllipse.visible=False
```

```

orangeellipse.visible=True
redEllipse.visible=False
doBlink=False

doGreen
do
  greenEllipse.visible=True
  orangeellipse.visible=False
  redEllipse.visible=False
  doBlink=False

Anomaly
do
  greenEllipse.visible=False
  orangeellipse.visible=True
  redEllipse.visible=False
  doBlink=true

doBlink Boolean
do
  blink orangeEllipse.visible 150

```

追加制御と他のアクセサの詳細については、『*ILOG Views 2D Graphics ユーザ・マニュアル*』の「定義済みアクセサ」を参照してください。

切り取り / コピー / 貼り付け

ILOG Studio のプロトタイプ拡張機能を使用する場合、編集機能で切り取り / コピー / 貼り付けが新たに利用できるようになりました。

[グループ・インスペクタ] パネルで、[アトリビュート] および [振る舞い] のページの [編集] メニューに、切り取り / コピー / 貼り付けの各コマンドがあります。

- ◆ アトリビュート・ツリーの一行目を選択して、続いて [編集] > [コピー] または [編集] > [切り取り] を選択すると、アトリビュート全体とその振る舞いをコピーしたり、切り取ることができます。
- ◆ [アトリビュート] ページで、振る舞いの行を選択して、続いて [編集] > [コピー] または [編集] > [切り取り] を選択すると、1 つの振る舞いをコピーしたり、切り取ることができます。
- ◆ まず、アトリビュートを挿入したい行を選択して、続いて [編集] > [貼り付け] を選択すると、アトリビュート・クリップボードの内容を貼り付けることができます。

編集機能の詳細については、『*ILOG Views 2D Graphics ユーザ・マニュアル*』の「プロトタイプの作成および使用」を参照してください。

アニメーション・タイマーのオン / オフ

アニメーション・タイマーにオン / オフ・トグルが追加されました。この追加制御は、プロトタイプ拡張機能の付いた ILOG Studio メニュー・バーの [表示] メニューから使用できます。

この追加コマンドは、以下のとおりです。

ToggleTimers

ラベル	トグル・アニメーション・タイマー
パス	[表示]メニュー
カテゴリ	プロトタイプ
アクション	プロトタイプのアニメーション・アクセサのアニメーション・タイマーをオンまたはオフして、プロトタイプを編集し、その振る舞いをテストします。

他のプロトタイプ・コマンドについては、『*ILOG Views 2D Graphics ユーザ・マニュアル*』の「プロトタイプ拡張コマンド」を参照してください。

システムの変更

ILOG Views コンポーネント・スイートのバージョン 4.0.2 では、以下のシステム変更が行われました。

- ◆ プラットフォームの調整
- ◆ Unix ユーザの方へ : Motif および共有ライブラリ

プラットフォームの調整

ユーザの方に最適な開発環境を提供できるように、アイログでは、対応プラットフォームの継続的な更新を行っています。コンパイラおよびオペレーティング・システムの進展に伴い、プラットフォームのアップグレードを追加したり、以前のプラットフォームへの対応を随時取り止めるようにしています。ILOG Views Component Suite の次リリース、バージョン 4.0.2 で変更になるポートを以下にまとめます。

新しいポート

このリリースでは、以下が含まれます。

- ◆ Sun UltraSparcStation ユーザ用の 64 ビット・ポート。このポートは、ultrasparc64_5_5.0 と呼ばれます。
- ◆ -AA コンパイル・フラグをサポートする HP 11 ユーザ用の新しいライブラリ・フォーマット。このポートは、hp32_11_3.30 と呼ばれ、以下のライブラリ・フォーマットを含みます。static_pic_mt_AA および shared_mt_AA。

警告 : SGI ユーザの方へ

今回のリリース (バージョン 4.0.2) が、SGI をサポートする ILOG Views Component Suite の最終リリースとなります。2002 年に発売になる ILOG Views Component Suite のリリースでは、SGI 用のポートは装備されません。

ILOG Views 4.0.2 は、2003 年 6 月まで引き続き SGI プラットフォーム上で使用できます。SGI プラットフォーム上で ILOG Views Component Suite をご利用いただいているユーザの方は、ILOG Views Component Suite 対応プラットフォームへの移行計画の準備をするようお勧めします。また保守期間の延長が必要な場合は、アイログの担当者までご連絡ください。

プラットフォームのアップグレード

以下のポートの提供は、ILOG Views Component Suite の今回のリリース (バージョン 4.0.2) が最後になります。次回リリースでも、これらのオペレーティング・システム用のポートを引き続き提供しますが、最新のコンパイラおよびオペレーティング・システム用のものではなくなります。

- ◆ msvc5
- ◆ hp_10_1.0
- ◆ sparc_5_4.2
- ◆ rs6000_4.3_3.6
- ◆ i86_linux2_glibc2.1_egcs1.1
- ◆ alpha_4_6.1

Unix ユーザの方へ : 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。Version 4.0 以降、ILOG Views が提供するすべての共有ライブラリは、libxviews を使用して構築されており、libmviews との互換性はありません。お近くのアイログのサポート・サイトにご連絡いただければ、ILOG は、お使いのプラットフォーム用の libmviews で構築される共有ライブラリを一式提供いたします。将来のバージョンでは、libmviews は、スタティック・ライブラリとしてのみ提供され、共有フォーマットではサポートされません。

索引

A

Application Framework

新しい機能 **91****C**

Charts

新しい機能 **92**コードの変更 **98**使用されなくなったメソッド **98****D**

Data Access

新しい機能 **92****G**

Gadget

新しい機能 **90**コードの変更 **99**

Gantt

機能 **92**コードの変更 **100**使用されなくなったメソッド **101****I**

ILOG Studio

新しい機能 **90**

ILOG Views 4.0.2

Motif および共有ライブラリ **109**プラットフォームの調整 **108**IlvSwitchAccessor クラス **106****M**

Manager

機能 **91**

Maps

Oracle Spatial **95**新しいサンプル **95**

新たにサポートされるようになったフォーマット

94機能 **93**コードの変更 **102**ファイル名の解決 **94**ロード・オン・デマンド **93****P**postscript デバイス **89****T**ToggleTimers コマンド **108**

W

Web 配布 **105**

Windows

透明度 **89**

あ

アニメーション・タイマーのオン/オフ **107**

い

イメージ処理クラス **86**

印刷フレームワーク **87**

き

切り取り/コピー/貼り付け
プロトタイプ編集 **107**

く

グラフィック・レンダリング **90**

す

スイッチ・アクセサ **106**

ひ

ビットマップ・データ **88**

ふ

ファンデーション

新しい機能 **86**

コードの変更 **96**

フィルタ

ILOG Views における **86**

プラットフォーム

バージョン 4.0.2 での新しいポート **108**

プロトタイプ

バージョン 4.0.2 での新しい **106**

ほ

ポート

バージョン 4.0.2 での新しい **108**

ま

マップ・ビルダ

新しい機能 **95**

り

量子化 **88**

リリース・ノート V5.3

Web 配布 **8**

新しいポート **7**

製品アイコン **8**

ろ

ロード・オン・デマンド **93**