



AutoCAD カスタマイズ&クラウドセミナー

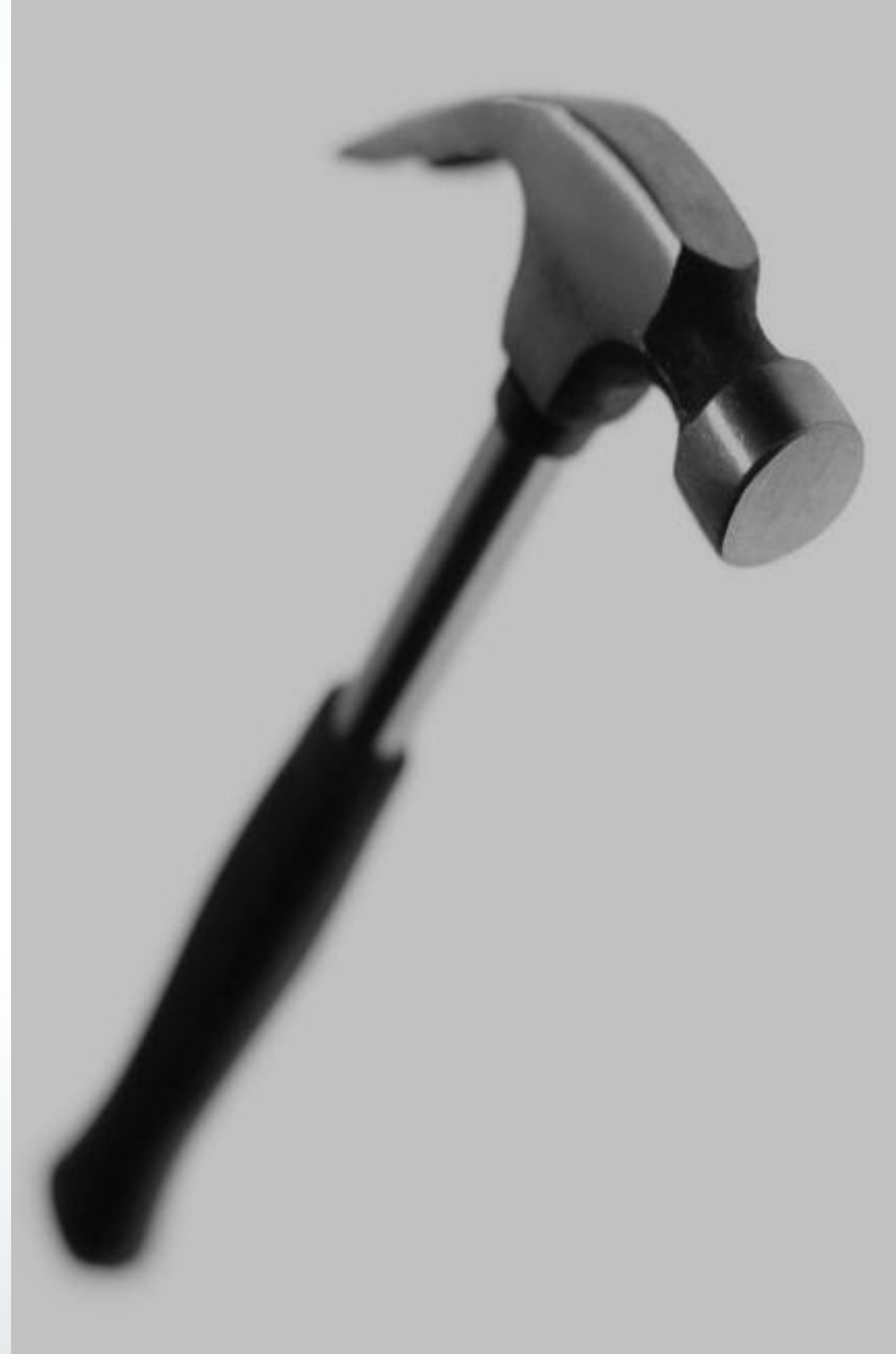
最適な AutoCAD API の選択と移植性

伊勢崎 俊明

Developer Technical Services

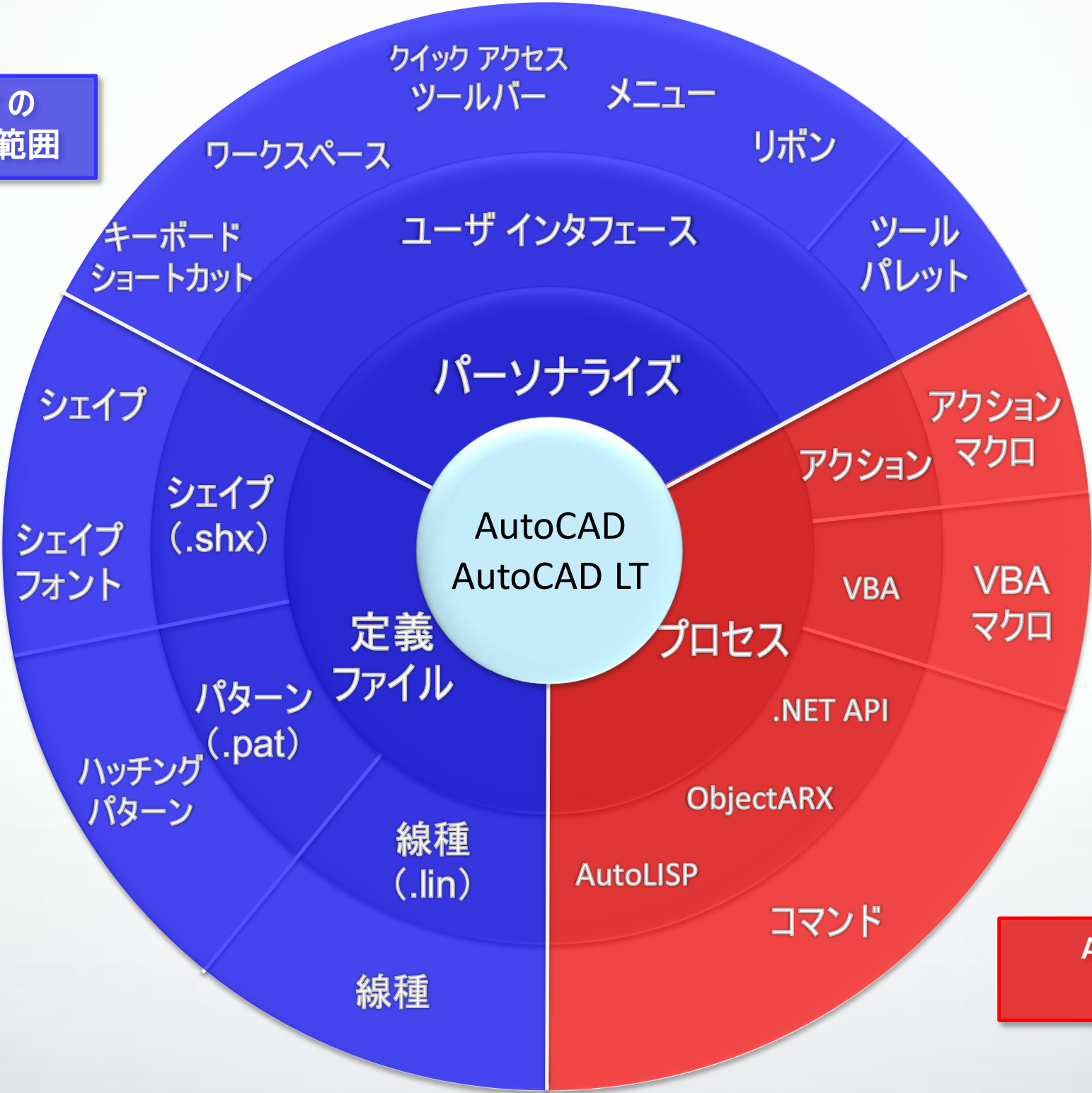


AutoCAD のカスタマイズとは？



カスタマイズ可能な項目

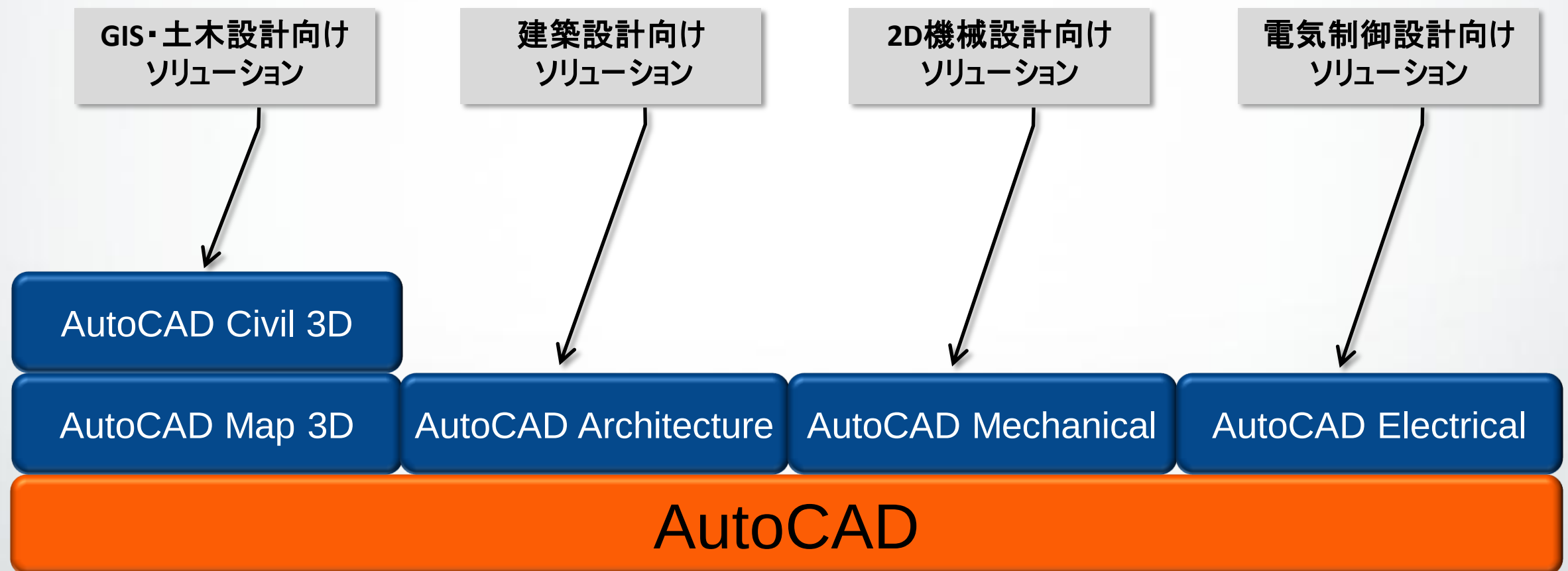
AutoCAD と AutoCAD LT の
両方で可能なカスタマイズ範囲



AutoCAD でのみ可能な
カスタマイズ範囲

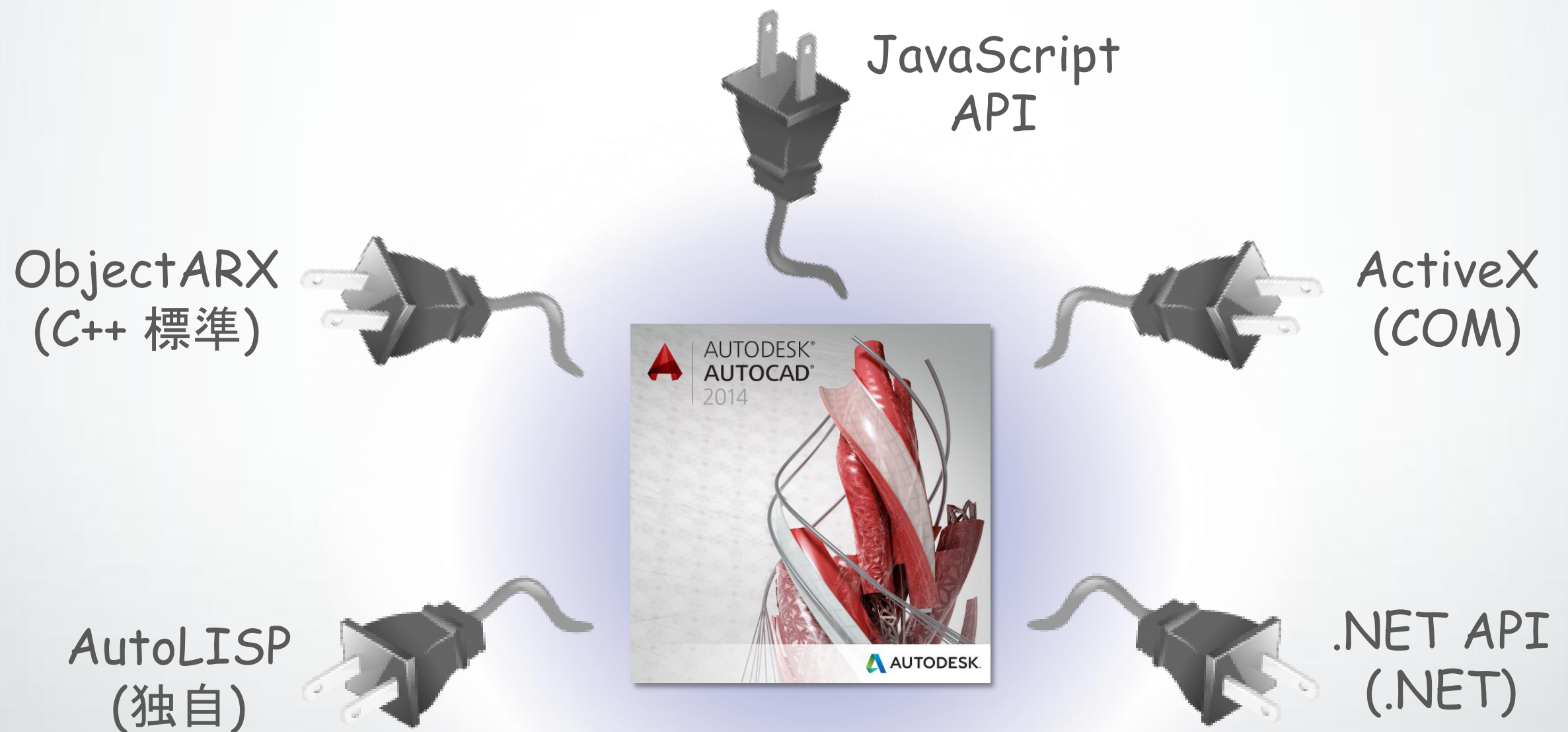
カスタマイズにより強力な設計環境を実現

- 業界別オートデスク製品も AutoCAD API で拡張
 - アドオン アプリケーションによる機能拡張が可能

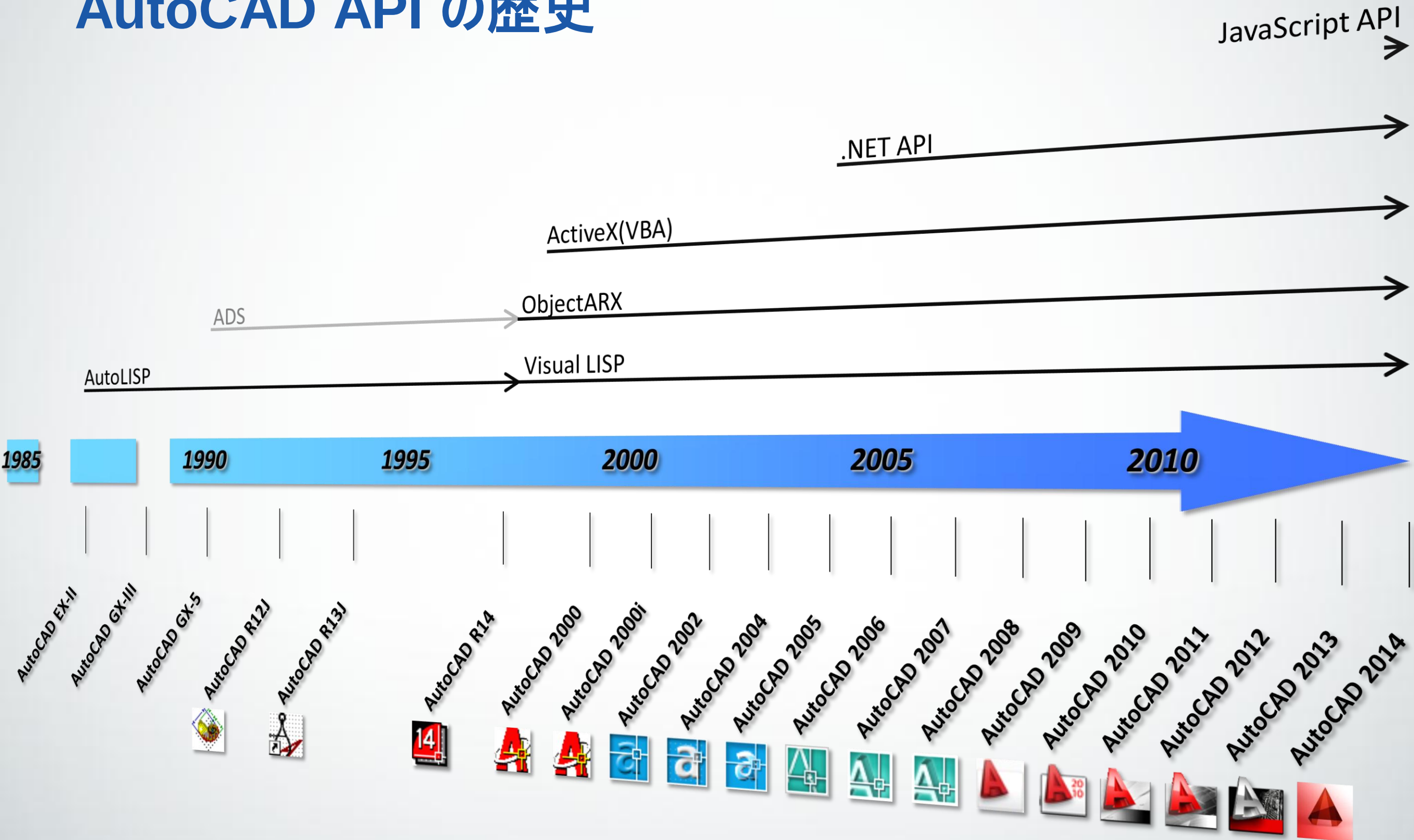


AutoCAD API

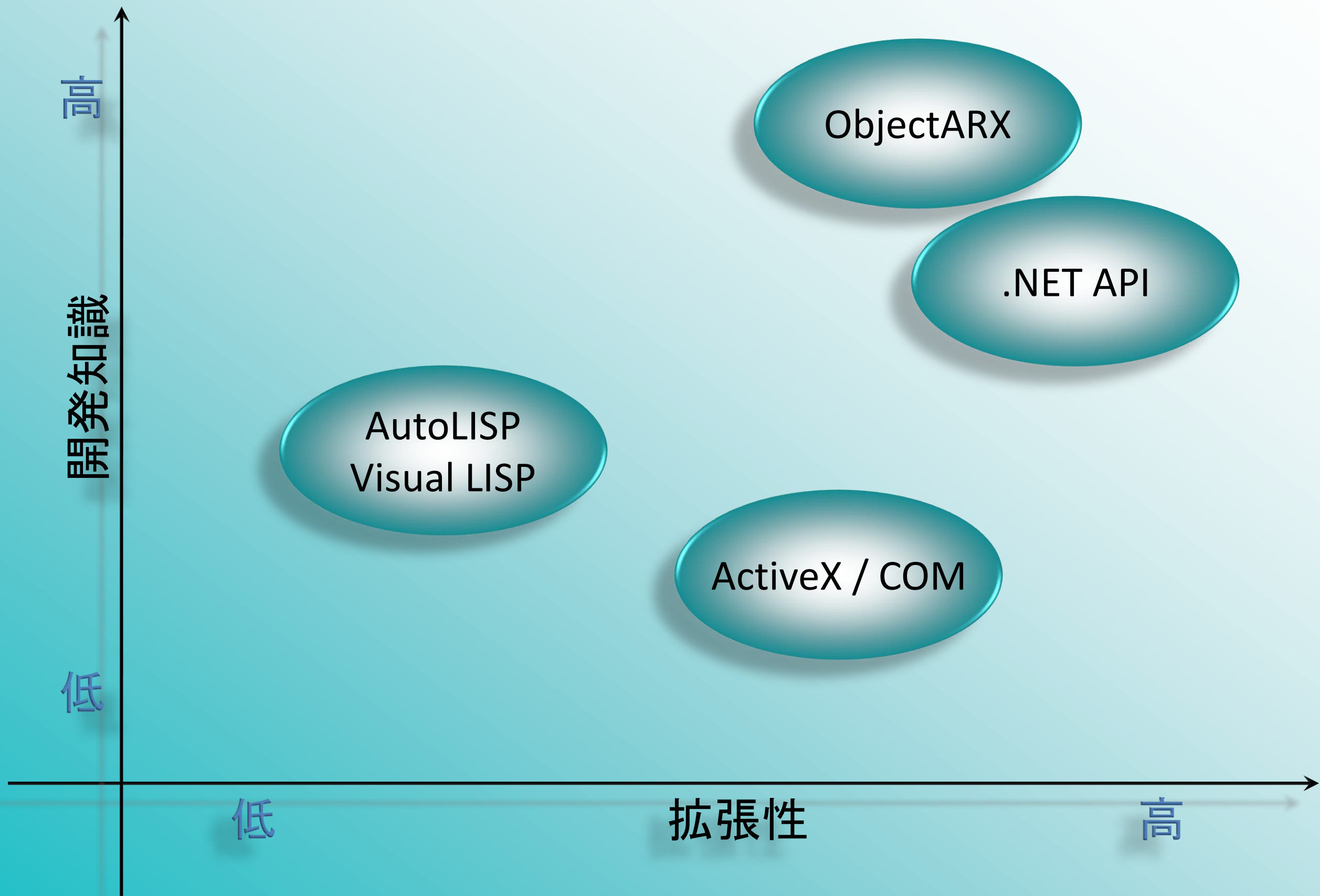
- 柔軟で強力な Application Programming Interface



AutoCAD API の歴史



AutoCAD API の位置付け



AutoCAD API の特徴と機能



AutoLISP の特徴

- 独自のプログラム言語
 - AutoLISP 言語
 - インタプリタ言語
- 独自の開発エディタ
 - Visual LISP エディタ
- 開発時に利用可能なドキュメント
 - AutoCAD 製品内に同梱
- 利用できる機能は AutoCAD によって左右される
 - AutoCAD 内の作図処理用途が中心
 - ダイアログ定義言語 (DCL) でダイアログボックスの作成も可能

```
(defun C:GetAttributes ()  
  (setq blklns (car (entsel "\n属性値を取得するブロック参照を指示してください:")))  
  (if blklns  
    (progn  
      (setq theBlockInsert (entget blklns))  
      (if (equal (cdr (assoc 0 theBlockInsert)) "INSERT")  
        (progn  
          (setq blkEnt (entnext (cdr (assoc -1 theBlockInsert))))  
          (while blkEnt  
            (setq aBlkEnt (entget blkEnt))  
            (if (equal (cdr (assoc 0 aBlkEnt)) "ATTRIB")  
              (progn  
                (setq aAttributName (cdr (assoc 2 aBlkEnt)))  
                (setq aAttributData (cdr (assoc 1 aBlkEnt)))  
                (princ (strcat "\n属性値[" aAttributName "] = " aAttributData))  
              )  
            )  
            (if (equal (cdr (assoc 0 aBlkEnt)) "SEQEND")  
              (setq blkEnt nil)  
              (setq blkEnt (entnext (cdr (assoc -1 aBlkEnt))))  
            )  
          )  
        )  
      )  
    )  
  )
```

AutoLISP とは ...



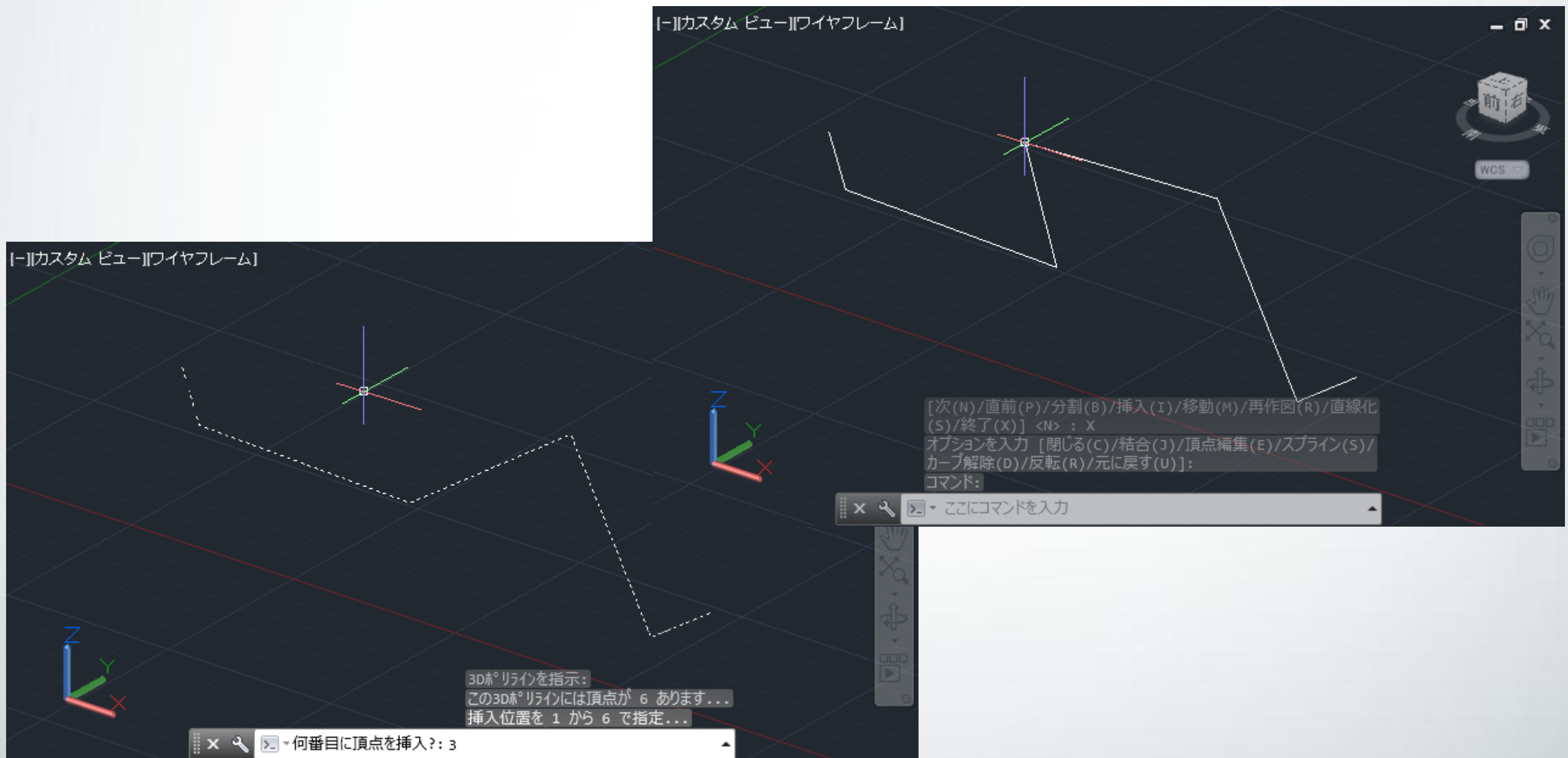
- 最古の AutoCAD API で**独自の AutoLISP言語** を使います



- ダイアログ ボックスを作成するために **DCL 言語**も利用することができます
- AutoCAD 間で **プログラム互換** を持ち、**カスタム コマンド**の定義も可能です
- 独自実装なので AutoLISP はいまも **高い移植性** を誇っています
- **ActiveX オートメーション (COM)** を利用することもできるようになっています
- 手軽さや多くのプログラム資産もあり、今後も **サポートを継続** します

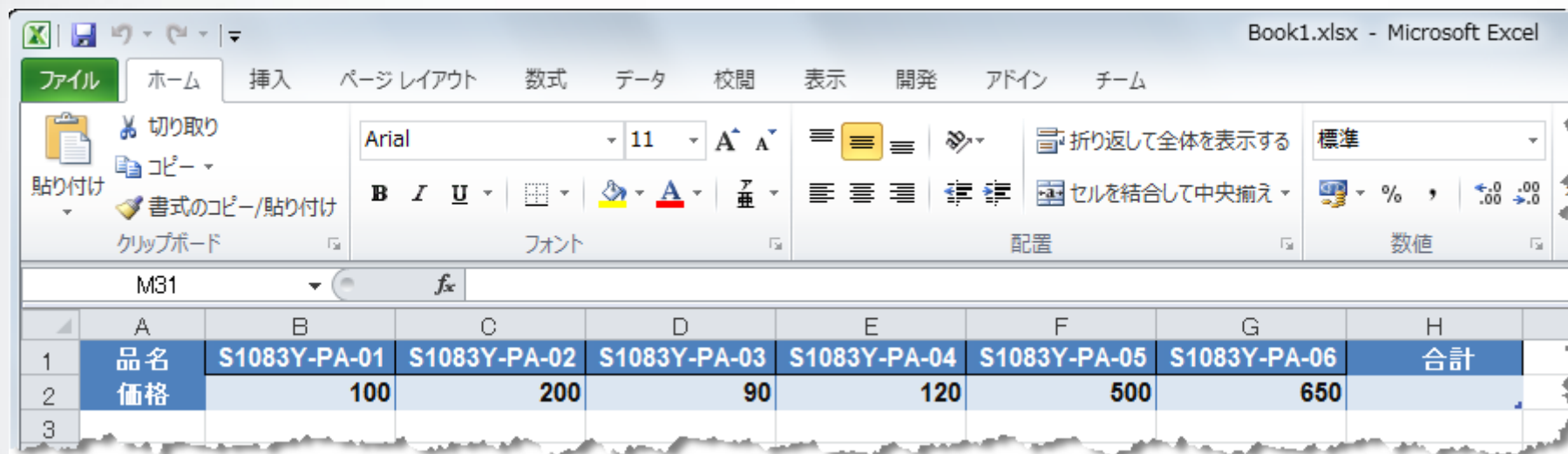
リスト操作による 3D ポリラインの編集

■ 3D ポリラインの頂点の追加と削除



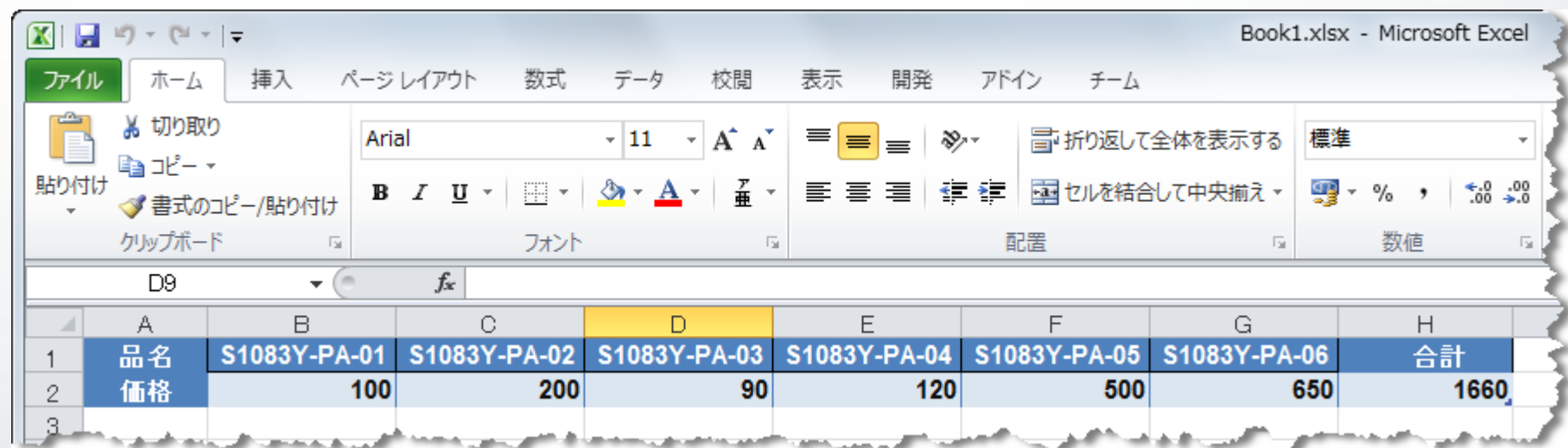
ActiveX オートメーションによる Excel の操作

- Excel の起動とセル値の読み取り、書き込み



Book1.xlsx - Microsoft Excel

	A	B	C	D	E	F	G	H
1	品名	S1083Y-PA-01	S1083Y-PA-02	S1083Y-PA-03	S1083Y-PA-04	S1083Y-PA-05	S1083Y-PA-06	合計
2	価格	100	200	90	120	500	650	
3								



Book1.xlsx - Microsoft Excel

	A	B	C	D	E	F	G	H
1	品名	S1083Y-PA-01	S1083Y-PA-02	S1083Y-PA-03	S1083Y-PA-04	S1083Y-PA-05	S1083Y-PA-06	合計
2	価格	100	200	90	120	500	650	1660
3								

ObjectARX の特徴

```
// トランザクション境界の開始
actrTransactionManager->startTransaction();

// マーカー配列内のマーカー形状&サイズを更新
AcDbObject* pMarker;
Adesk::Boolean bBool;
for( int nCnt=0 ; nCnt<DocVars.docData( pDoc ).gMarkerIds.length() ; nCnt++ ){
    actrTransactionManager->getObject( pMarker, DocVars.docData( pDoc ).gMarkerIds[nCnt], AcDb:
    AsdkMarker::cast( pMarker )->setType( DocVars.docData( pDoc ).gnType );
    AsdkMarker::cast( pMarker )->setSize( DocVars.docData( pDoc ).gdSize );
}
actrTransactionManager->flushGraphics();

// トランザクション境界の終了
actrTransactionManager->endTransaction();

// マーカー点滅
if ( m_Check1.GetCheck() == 1 ){
    DocVars.docData( pDoc ).gbBlink = true;
    DocVars.docData( pDoc ).gnInterval = m_Slider2.GetPos();
    acedGetAcadFrame()->SetTimer( 12345, DocVars.docData( pDoc ).gnInterval, TimerProc );
    acutPrintf( _T("\nマーカーの点滅を開始しました[タイマー更新]...\n") );
}
```

- ANSI 標準のプログラム言語
 - C++ 言語
 - コンパイラ言語
- Microsoft のエディタ(統合開発環境)
 - Visual C++ (Visual Studio 2010 SP1) : 別売
 - ObjectARX SDK が必須(無償ダウンロード:
<http://www.autodesk.com/objectarx>)
- 開発時に利用可能なドキュメント
 - ObjectARX SDK 内に同梱(英語)
- 標準的なC++言語なので他の技術と融合しやすい
 - 作図処理から UI 定義、イベント処理など多彩な処理で利用
 - オートデスクの開発者と同じレベルのカスタマイズが可能

ObjectARX とは ...

- **最も強力な** AutoCAD API で、開発には **ObjectARX SDK** が必要です



- 作業は**Visual Studio**で**ANSI標準**の **C++ 言語** を使ってプログラムします



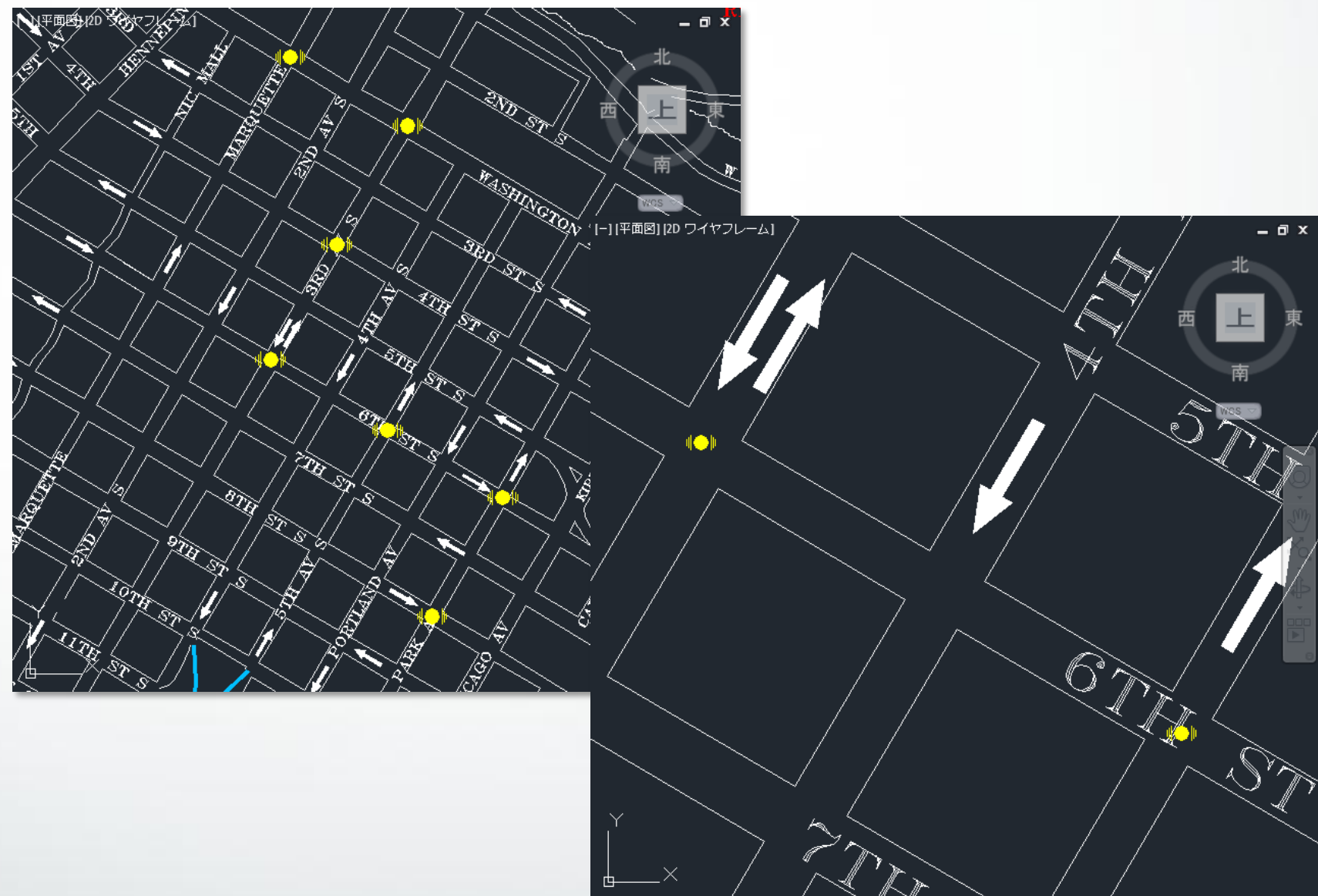
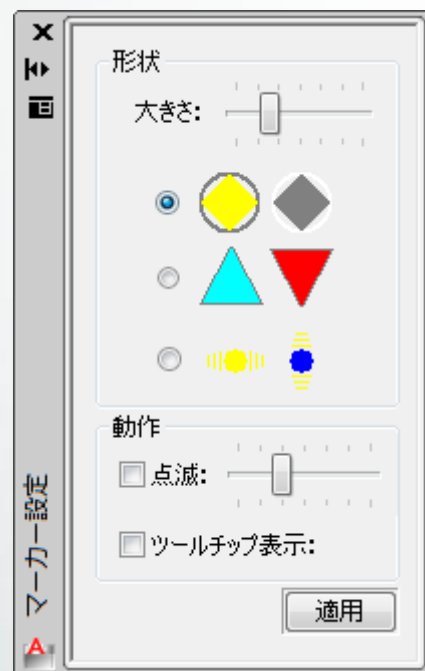
- **カスタム コマンド**の定義の他に **カスタム オブジェクト** の定義もできます

- ObjectARX SDK は開発者に **クラス ライブラリ** を提供します

- バージョン間やプラットフォーム間の **移植作業** が必須な API でもあります

AutoCAD カスタマイズの具体例 — ObjectARX カスタム オブジェクト

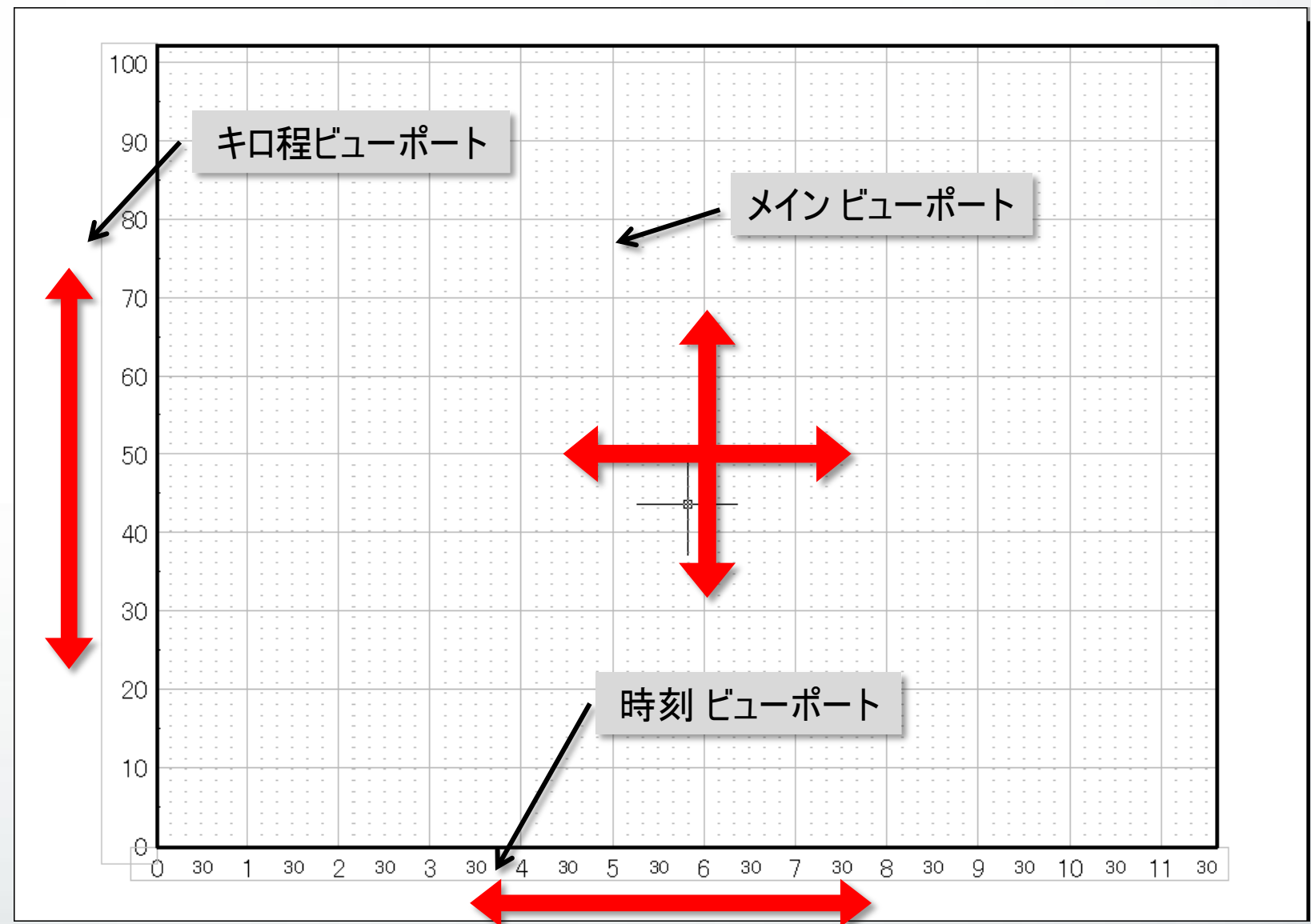
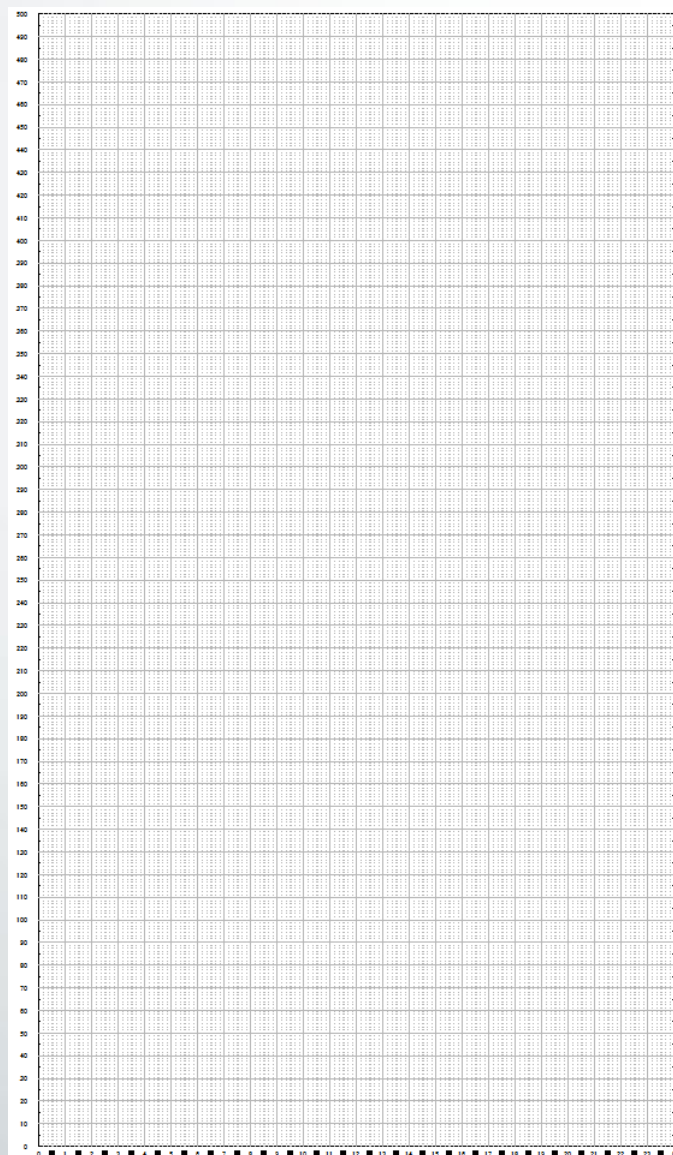
- 独自の振る舞いを持つ マーカー オブジェクト
 - 同じ表示サイズを維持、点滅、入力文字をツールチップ表示



AutoCAD カスタマイズの具体例 — ObjectARX

マウス フックと画面制御

- 帳票やシステム端末への応用
 - マウス操作の通知と抑止で一定の手続きを自然に強制



ActiveX オートメーションの特徴

- Microsoft の技術（業界標準）
 - ActiveX オートメーション（COM）という標準技術
- Microsoft のプログラム言語
 - 親しみやすい Visual Basic 言語
- Microsoft のエディタ（統合開発環境）
 - VBA、Visual Studio（Visua Basic、Visual C++ ...）
- 開発時に利用可能なドキュメント
 - AutoCAD 製品内に一部同梱
- Windows では標準的な技術と言語なので他と融合しやすい
 - AutoCAD 内では作図中心の用途で利用
 - COM を採用する他のアプリケーションと連携が可能

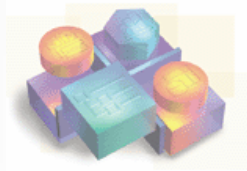
```
Dim objUCS As AcadUCS
Dim ptBase(0 To 2) As Double
Dim ptXAxis(0 To 2) As Double
Dim ptYAxis(0 To 2) As Double
ptBase(0) = 0#: ptBase(1) = 0#: ptBase(2) = 0#
ptXAxis(0) = 1#: ptXAxis(1) = 0#: ptXAxis(2) = 0#
ptYAxis(0) = 0#: ptYAxis(1) = 0#: ptYAxis(2) = 1#
Set objUCS = ThisDrawing.UserCoordinateSystems.Add(ptBase, ptXAxis, ptYAxis, "XZ_Plane")
ThisDrawing.ActiveUCS = objUCS
```

```
Dim objCircle As AcadCircle
Dim ptCenter(0 To 2) As Double
Dim dRadius As Double
ptCenter(0) = 0#: ptCenter(1) = 0#: ptCenter(2) = 0#
dRadius = 10#
Set objCircle = ThisDrawing.ModelSpace.AddCircle(ptCenter, dRadius)
```

```
Dim objRegion As Variant
Dim objCurves(0 To 1) As Object
Set objCurves(0) = objCircle
```

ActiveX オートメーションとは ...

- **ActiveX オートメーション** は Microsoft の API 公開/再利用テクノロジーです



- 最近では **COM** (Component Object Model) と呼ばれる事があります



- COMは多くの開発ツールでサポートされ、**AutoCAD VBA** もその1つです



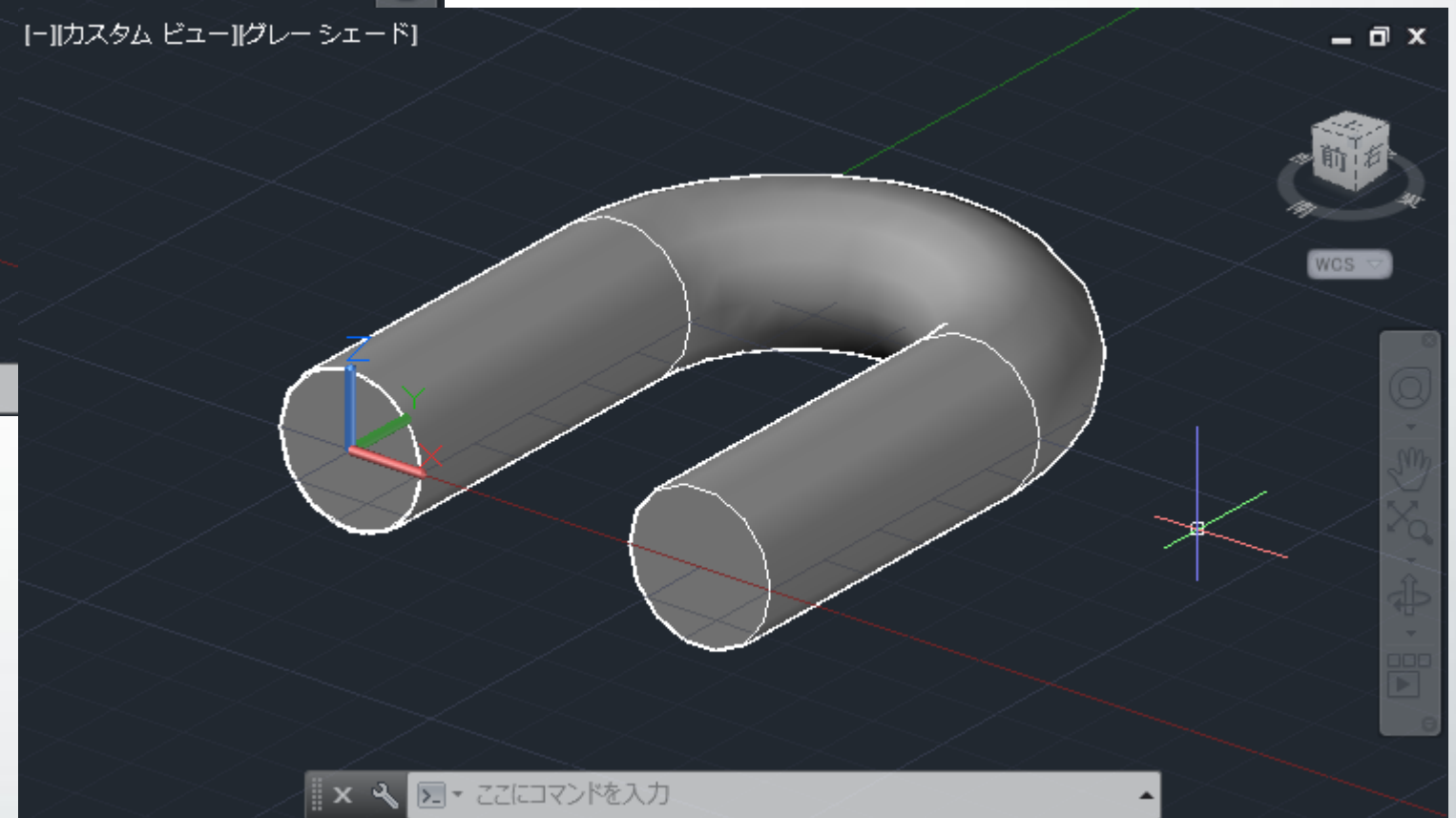
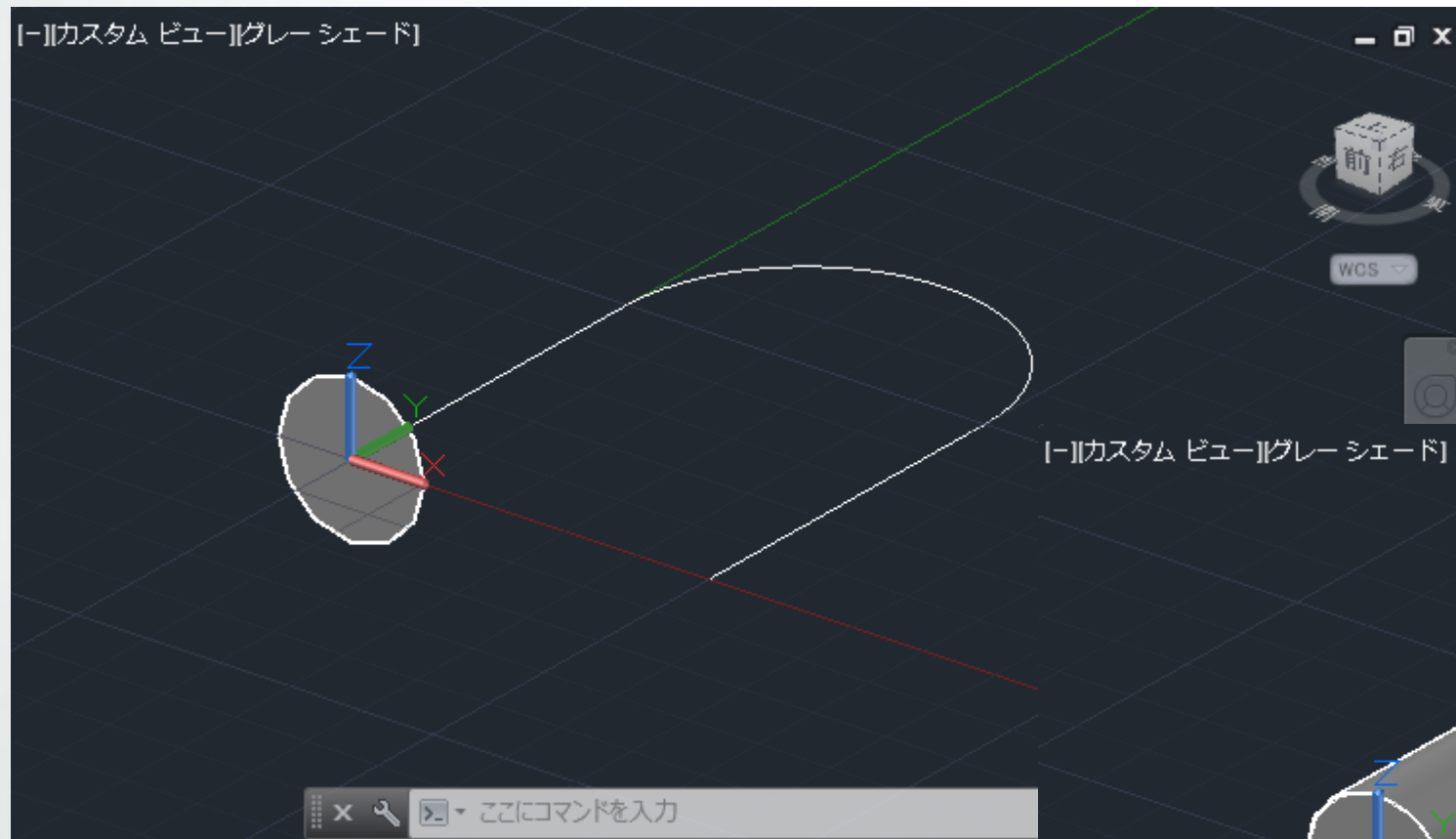
- VBA 上では **Visual Basic 言語** で開発することができます



- Visual Studioや **他のアプリケーション VBA** でのプログラム作成も可能です

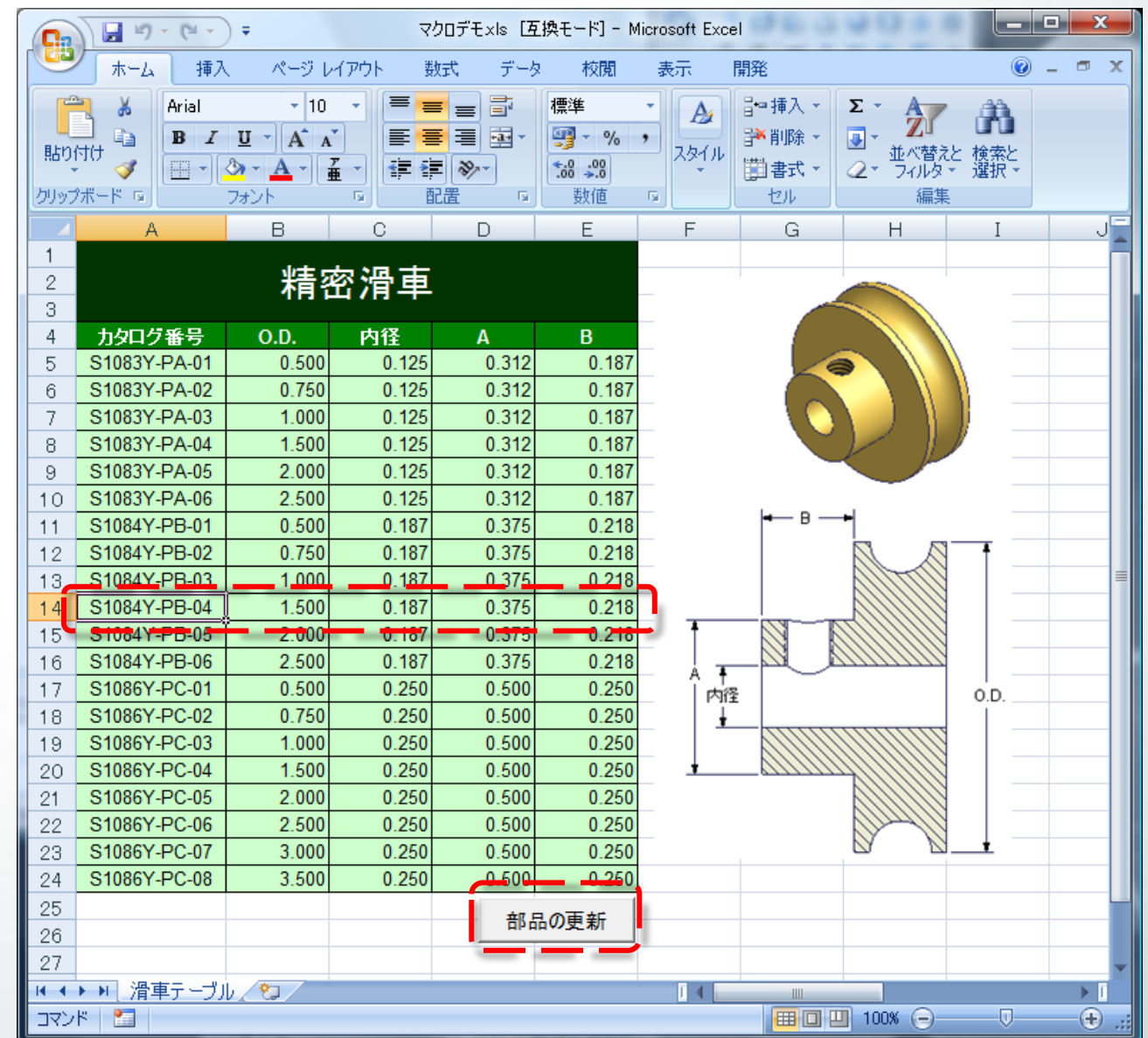
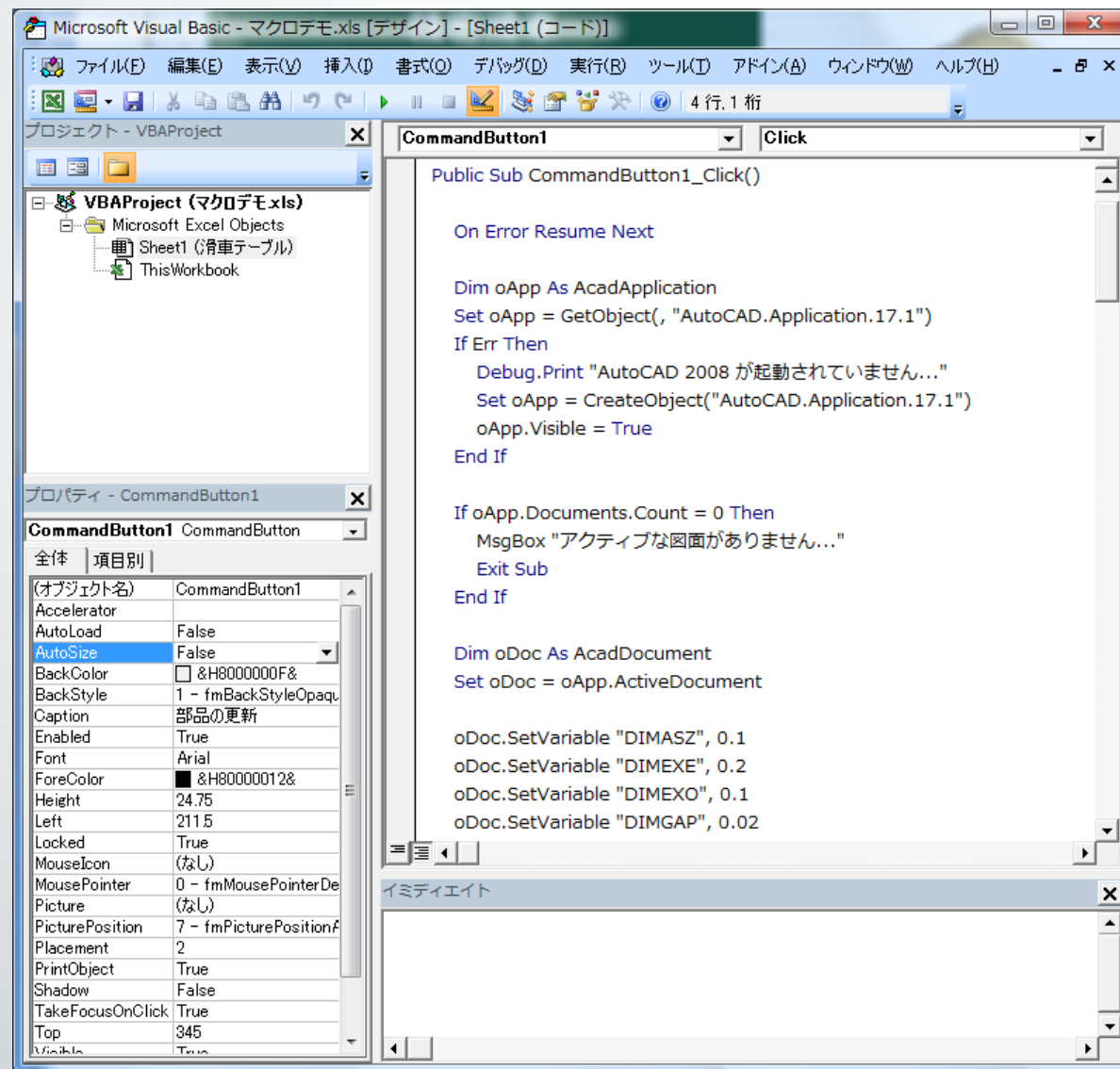
AutoCAD カスタマイズの具体例 — ActiveX オートメーション 3D ソリッドの作成

- リージョンをパス曲線に沿って押し出し



AutoCAD カスタマイズの具体例 — ActiveX オートメーション 他のアプリケーションとの連携

■ Excel のセルの値を利用した自動作図



AutoCAD .NET API の特徴

```
・ドキュメントのロック
Dim oDocLock As DocumentLock = Autodesk.AutoCAD.ApplicationServices.Application

・トップノードの判別
Dim objId As ObjectId
Dim strNode As String = e.Node.Text
If strNode = "モデル空間" Then

    ・全オブジェクトのハイライト&ハイライト解除
    Dim oBt As BlockTable = oTr.GetObject(oDb.BlockTableId, OpenMode.ForRead)
    Dim oBtr As BlockTableRecord = oTr.GetObject(oBt.Item("*MODEL_SPACE"), OpenMode.ForRead)
    Dim oId As ObjectId
    Dim oEnt As Entity
    For Each oId In oBtr
        oEnt = oTr.GetObject(oId, OpenMode.ForRead)
        Dim oSubPath As FullSubentityPath = New FullSubentityPath(New ObjectId(oId))
        If e.Node.Checked Then
            oEnt.Highlight(oSubPath, True)
        Else
            oEnt.Unhighlight(oSubPath, False)
        End If
    Next
```

- Microsoft の新技術(次世代標準)
 - Windows の .NET Framework という標準技術上で動作
- 選べる プログラム言語 (Microsoft、ANSI)
 - Visual Basic 言語、C# 言語、C++ 言語
- Microsoft のエディタ(統合開発環境)
 - Visual Studio 2010(SP1): 別売
- 開発時に利用可能なドキュメント
 - ObjectARX SDK 内に同梱(英語)
- 標準的な技術と言語なので他の技術と融合しやすい
 - .NET を利用して Web 越しの連携も可能で拡張性が高い !!

AutoCAD .NET API とは ...



- 最新の AutoCAD .NET API は .NET テクノロジ を使っています



- AutoCAD .NET API は ObjectARX 相当の強力な機能があります



- .NET テクノロジ では複数の 開発言語 を選択することができます



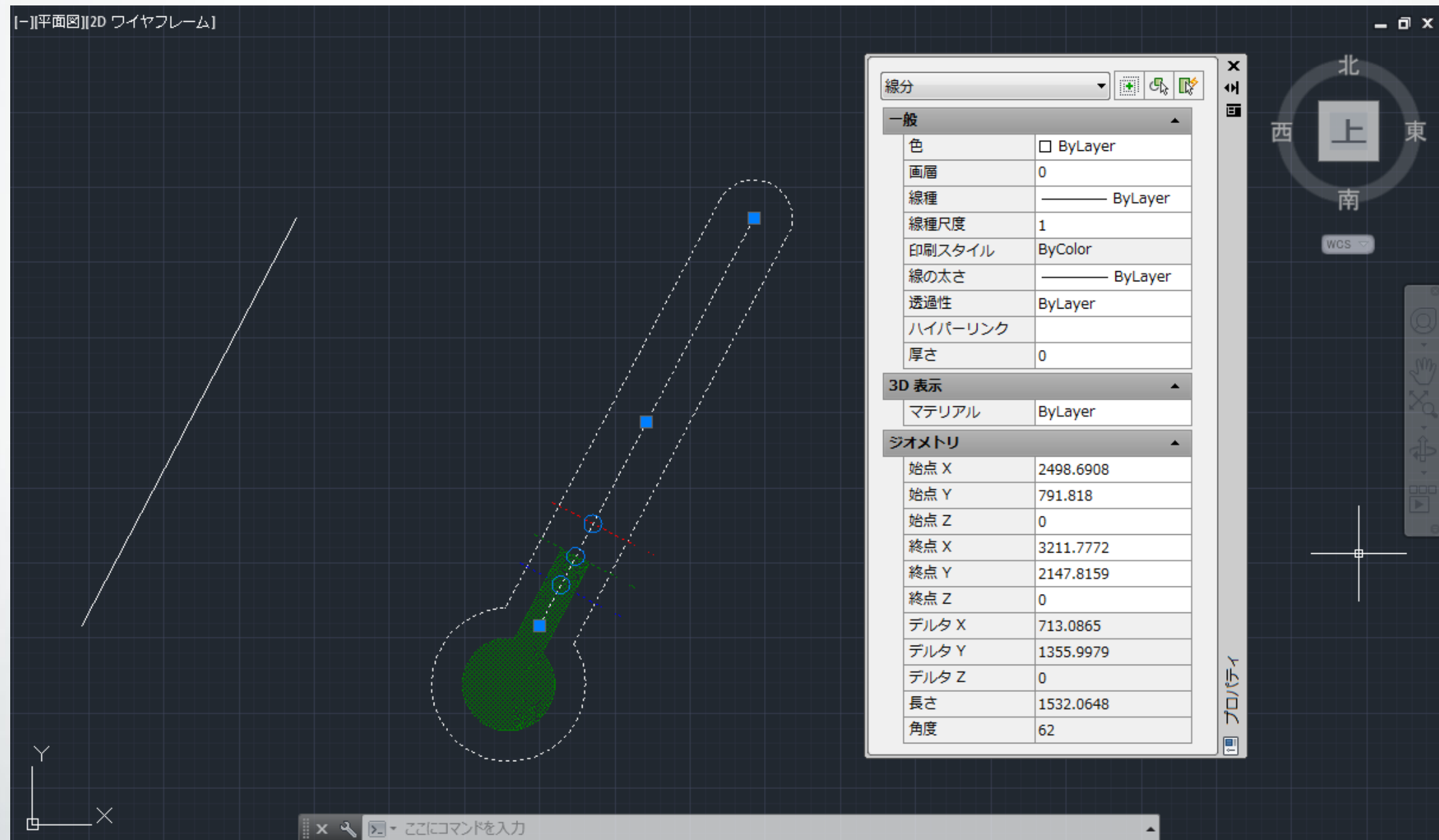
- Visual Basic 言語 で ObjectARX 的な強力なカスタマイズができます



- .NET API の開発には Visual Studio を利用します

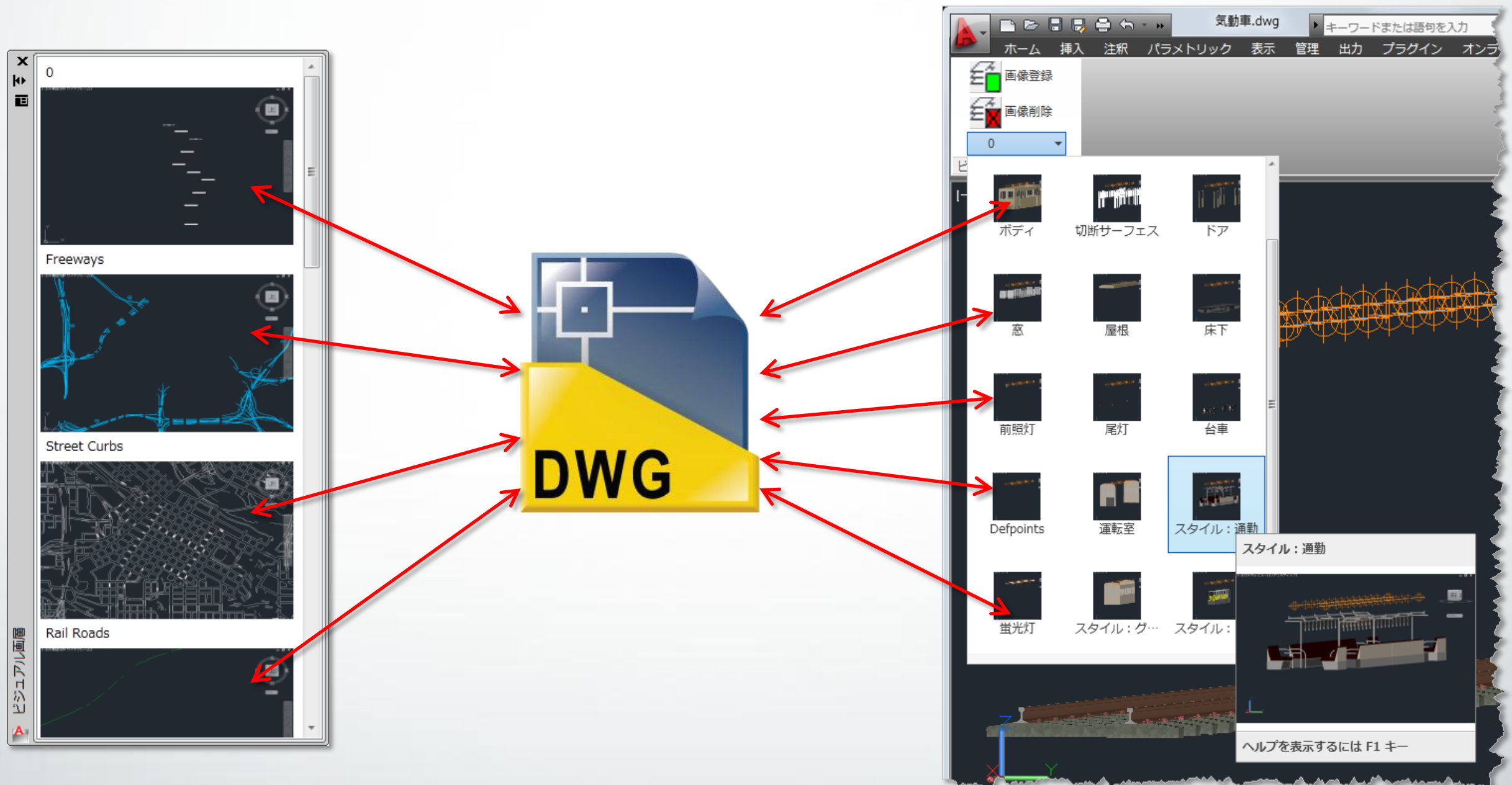
オーバールールによる標準オブジェクトの表現変更

■ 線分 (LINE) を温度計に変更



独自情報を図面ファイルへ埋め込み DWG を拡張

- 画層毎の状態をキャプチャして埋め込み UI に再利用



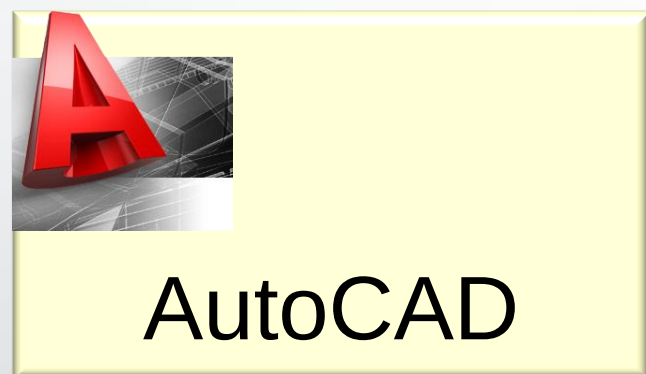
AutoCAD API を使ったアプリケーションの種類

- AutoCAD の内側で実行するタイプ
 - AutoCAD にファイルをロードして実行



AutoLISP
ObjectARX
Active (COM)
.NET API

- AutoCAD の外側から実行するタイプ
 - 独自の実行ファイルや Excel などから AutoCAD をコントロール



Active (COM)

AutoCAD API 別の能力

	AutoLISP	ActiveX/VBA	ObjectARX	.NET API
一般 グラフィカル オブジェクトの作成	○	○	○	○
一般 グラフィカル オブジェクトの編集	○	○	○	○
一般 非グラフィカル オブジェクトの作成	○	○	○	○
一般 非グラフィカル オブジェクトの編集	○	○	○	○
カスタム オブジェクトの定義	—	—	○	—
カスタム オブジェクトの作成	○	○	○	○
カスタム オブジェクトの編集	○	○	○	○
ファイルへの図面の保存	○	○	○	○
既存図面を AutoCAD 上に開く	○	○	○	○
既存図面をメモリ上に開く	—	—	○	○
AutoCAD 上での新規図面の作成	○	○	○	○
メモリ上での新規図面の作成	—	—	○	○
オブジェクトの選択	○	○	○	○
データ リスト操作	○	—	○	—
コマンドの定義	○	○	○	○
コマンドの発行	○	○	○	○
リアクタ / イベント処理	○	○	○	○
メニュー UI の実装	○	○	○	○
ダイアログ UI の実装	○	○	○	○
データベース アプリケーションとの連携	○	○	○	○
他の SDK との連携	○	○	○	○
幾何演算ライブラリ	—	—	○	○
境界表示解析(B-Rep)	—	—	○	○
COM インタフェースの実行	△	○	○	○
オブジェクト指向プログラミング	○	○	○	○

AutoCAD カスタマイズ事例

- 少し古いですが ... <http://www.autodesk.co.jp/developcase>



会社概要
株式会社ノーリツは、1951年創業。ガス風呂釜、ボイラ、ガス給湯器の開発・製造を営む。1980年代から燃費技術、システム化の発展に入り、現在では、ガス温水機器、石油温水機器、温水暖房システム、空調機器、ソーラーシステム、システムバス、システムキッチン、炭化水素ガス温水炉など、業務用装置製造設備などを幅広く開発・製造・販売。四社（営業本部、営業開発部では、ろ過機器、給湯機器、コ・ジェネレーション機器などの業務用機器の営業・販売を行っている）。

<http://www.noritz.co.jp/>

● APIを利用したAutoCADのカスタマイズにより、Microsoft Excelをポータルとした見積書作成アプリケーションから、ろ過システムの系統図・機器表を自動的に作成

- AutoCAD 2006
- 利用したオートデスクAPI
- ActiveX オートメーション (COM)

●株式会社大塚商会
<http://www.otsuka-shokai.co.jp/>

●豊富な商品知識を持つ熟練者しかた、見慣れたシステム図作成の作業をこにより、誰でも容易にシステム図の作

- 平均的な12〜13パーツ程度のろ過システムにおいて、系統図作成に要する時間を、従来の1時間程度から15〜30分程度に、大幅に削減
- スピーディな顧客対応が可能となり、ケアレスミスも削減

“業務用ろ過システム系統図・
機器表自動作図システム”を構築
AutoCADのカスタマイズにより、
系統図作図に要する時間を従来の1/2以下に短縮

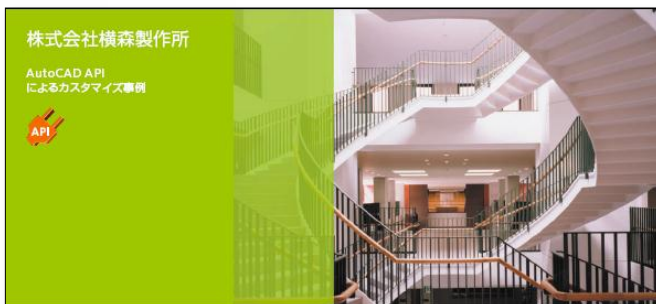
株式会社バーズ(以下、バーズ)は、事業本部 事業開発部は、共有設備などの設備構築、動画編集、ウェブ制作・プログラミングなどの業務用設備の販賣・販売を行っている。見学者用システムについては、顧客企業である大手ゼネコンや設計事務所にも協力し、見学者向けにブロック化されたバーズのCloudデータにアクセスし、系統を作成する業務用設備が必要不可欠になっている。従来は、Microsoft ExcelとAutoCAD LTを5台5台に使用して、見学者の設備名から探すブロック型をAutoCAD LTに導き入るという工数のかかる系統を作製。2006年1月、パートナーの株式会社大森商会から、いっそうの自動化のために、AutoCAD 2006/AE2を使用したシステムデザインの有償であるという提案があり、AutoCAD 2006へ移行を決意。見学者から系統図の作成を行う時間を大幅に短縮している。

AutoCADのカスタマイズにより、各種図作の完全な自動化を実現

1990年 日本電産 建築電気図では、1996年ごろから、2Dシステムの系統図の作成にCADを導入していった。系統図の作成は、これまで経験者・熟練者のみにしか行われていた作業であった。当初は、汎用機に対するパーツ図面をシステムに結び付けるといった手段は、他のメーカーのCADを待つ必要が5年程度あった。1997年からは、AutoCAD Lを併用して使用することになった。

営業所課 主事 鎌田賢治氏は、AutoCAD Lの導入に関して、「当初、取引先でもCADの導入がまだ少ない。契約のデータ交換やサポートであるDWGを見ることもできなかった。弊社は、AutoCAD-Jw.cadが主流だったので、パートナーである大倉からアドバイスが得たため、汎用性の高いAutoCAD Lを選択しました」。

AutoCAD Lの導入後、2Dシステムの系統図に使用される機器をブロック化して作業を楽にした。



会社概要
株式会社横森製作所は、1951年(昭和26年)創業。当初は、建築金物を製造していたが、高度成長期に入り、ある解体工事が組み立て式階段の着想を得る。1970年代、超高層ビルが数多く建てられ、鉄骨階段の専業メーカーとして実績を築き、現在では、従来からのノコギリの需要、大規模な階段を製造する能力を高く評価され、日本の超高層ビルの約90%にヨコモリの階段が使用されている。
<http://www.yokomori.co.jp/>

超高層ビル階段の3次元設計に、AutoCADを導入
APIを使用したAutoCADのカスタマイズにより、
設計の精度・効率が大幅に向上

株式会社森田製作所(以下、COMO)は、ランドマークテクノ、六本木ビルズ、森田テクノによる総出資率100%の約90%の施設設計・製造を行っている。同社では、従来のUNIXベースの2次元CADから3次元CADへの移行を図るにあたり、取引先とのデータ交換にDWGファイル形式が一般的になっていることから、AutoCADを選択、株式会社構造設計研究所(以下、構造設計研究所)をパートナーとして、新システム「Cadsy21」の開発に着手した。同工程の最先端・施工システムとしては、最も最初にAutoCADをシステムライズできるObjectARXを使って、ウィザードスタイルによる基本機能の入り付けで、3次元データ作成から、設計連携から上棟工程の工程四作図まで同時に行なうことが可能なため、設計の精度向上、設計連携の大幅な増進を実現している。

3次元化により、プログラムの開発と設計を自動化

コンピュータは従来、もとのデータの外形を自動で認識し、標準的にCADを導入しては、1998年当時適用していた、UNIXベースのDEWS (Engineering Workstation) に搭載した独自開発の2次元CADだったが、プログラムの2次元データが膨れ上がり、書き込みの「ジョブファイル」が膨れ上がり、制御不良、新製品の開発遅延、製品の品質低下、ソフトウェア構築の遅延、コスト増大、といった弊害を生じ、これを解消してはじめて、汎用の3次元CADを導入するを絶対に目標とすることになった。

同社社長は「機械技術部長（兼）CAD開発部長 門田昌雄氏は、3次元化への移行を画した原点に立ち戻りたところから要するに、自社の製品の構造形式は、アークが基本であり、従来の2次元型システムではうまく表現できない、すなわち、前工程で設計された図を、同じような加工工程で生産、NC加工とつながる、リアルタイム設計システムです。しかも、従来のような2次元CAD開発システムでは、加工した製品の丸角3次元CADで再現できない」と

AutoCAD でパラメトリック 3D を実現 階段設計に特化したカスタマイズ



会社概要
株式会社 セレコホールディングスは、1993年設立。新築着工・プレハブ工法によるアパート・建群・郊外に特化した土地活用活用を促進し、年間1,000戸以上のアパート・建群の実績を重ねている。財団法人日本建築センターの厳しい基準をクリアした戸建の安心と安価。

・近所型住宅設計 (建築設計事務所)
・住宅型住宅設計 (建築設計事務所)
<http://www.cel-co.com/>

システム概要

- APIを利用したカスタマイズにより、AutoCAD 2006、Autodesk Revit Building Bを利用した設計・構築発注統合システムを構築

導入製品

- AutoCAD 2006
- Autodesk Revit Building B

利用したオートデスクAPI

- ObjectARX
- ActiveX オートメーション (COM)

パートナー
 ・株式会社大塚商会
<http://www.o-tsuka-shokai.co.jp/>

●営業図面から受発注システムまでデータを流通させることで、社内・社外が強化され、コミュニケーションが

●設計・構築用統合システムの導入により、従
当前に関係者の欠けが生れ、半年後、1年後
を見据えたCADの普及を策定
●使用したCADの普及を策定することによ

社内システムに AutoCAD を組み込み データ流通で jw cad を置き換え



**AutoCADのカスタマイズ機能を活用し、
駅・車両基地構内作業計画作成支援システムを構築
駅構内ダイヤ図の調整・帳票印刷を強力にサポート**

[illegible][illegible]

CAD を意識しないシステム組み込み グラフィック端末としてダイヤ図を作成

バージョン差による移植作業の壁



バージョンアップ等によるアプリケーション移植の負荷

- **AutoLISP/Visual LISP**

- 影響は**小**
- バージョン差、プラットフォーム差※は AutoCAD が吸収

- **ActiveX オートメーション/COM(VBA)**

- 影響は**中**
- 新しいバージョンの参照設定で対応可能、ただし、VBA は移植を推奨

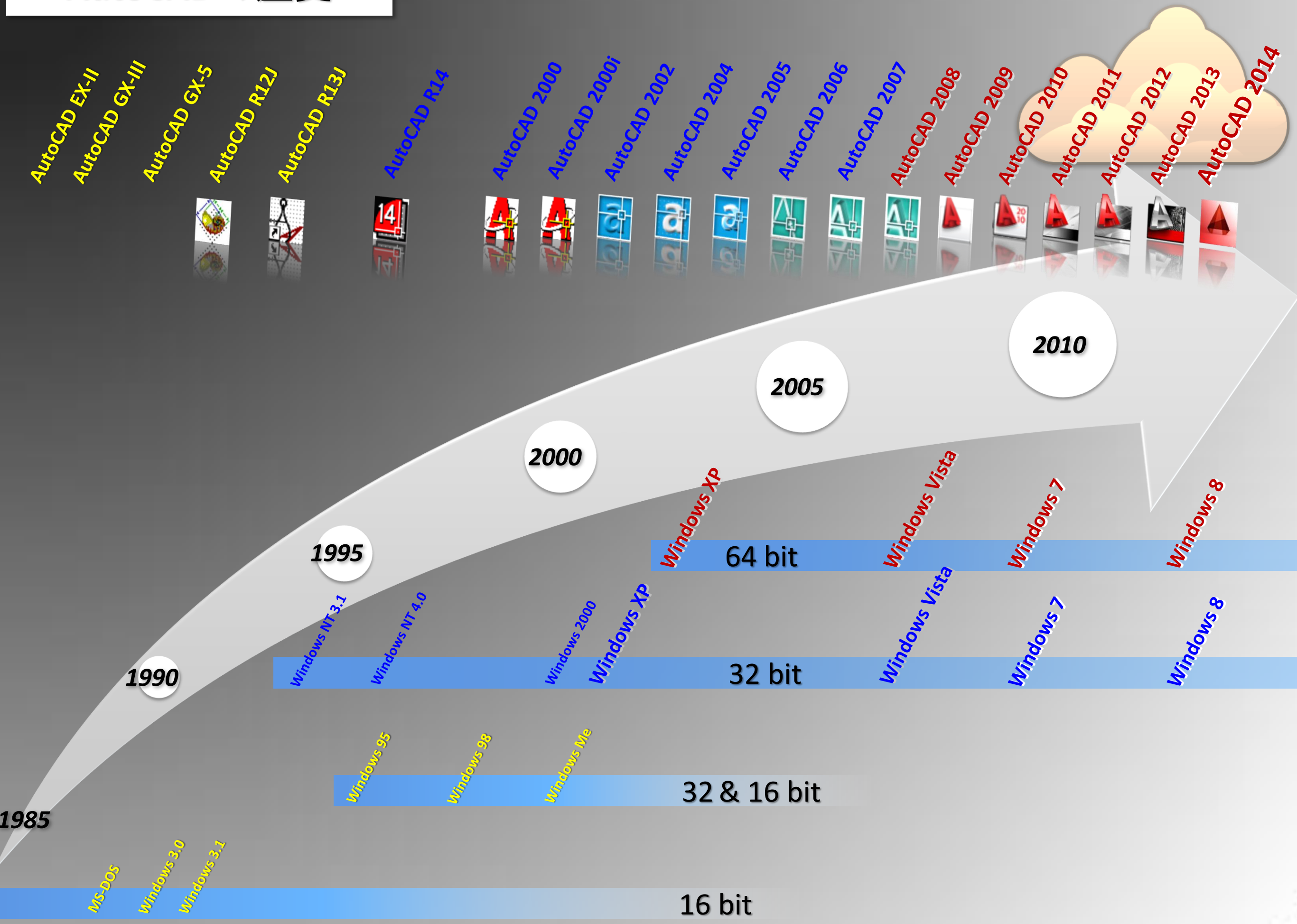
- **ObjectARX**

- 影響は**大**
- バージョン差、プラットフォーム差によりプログラムレベルの変更が必須

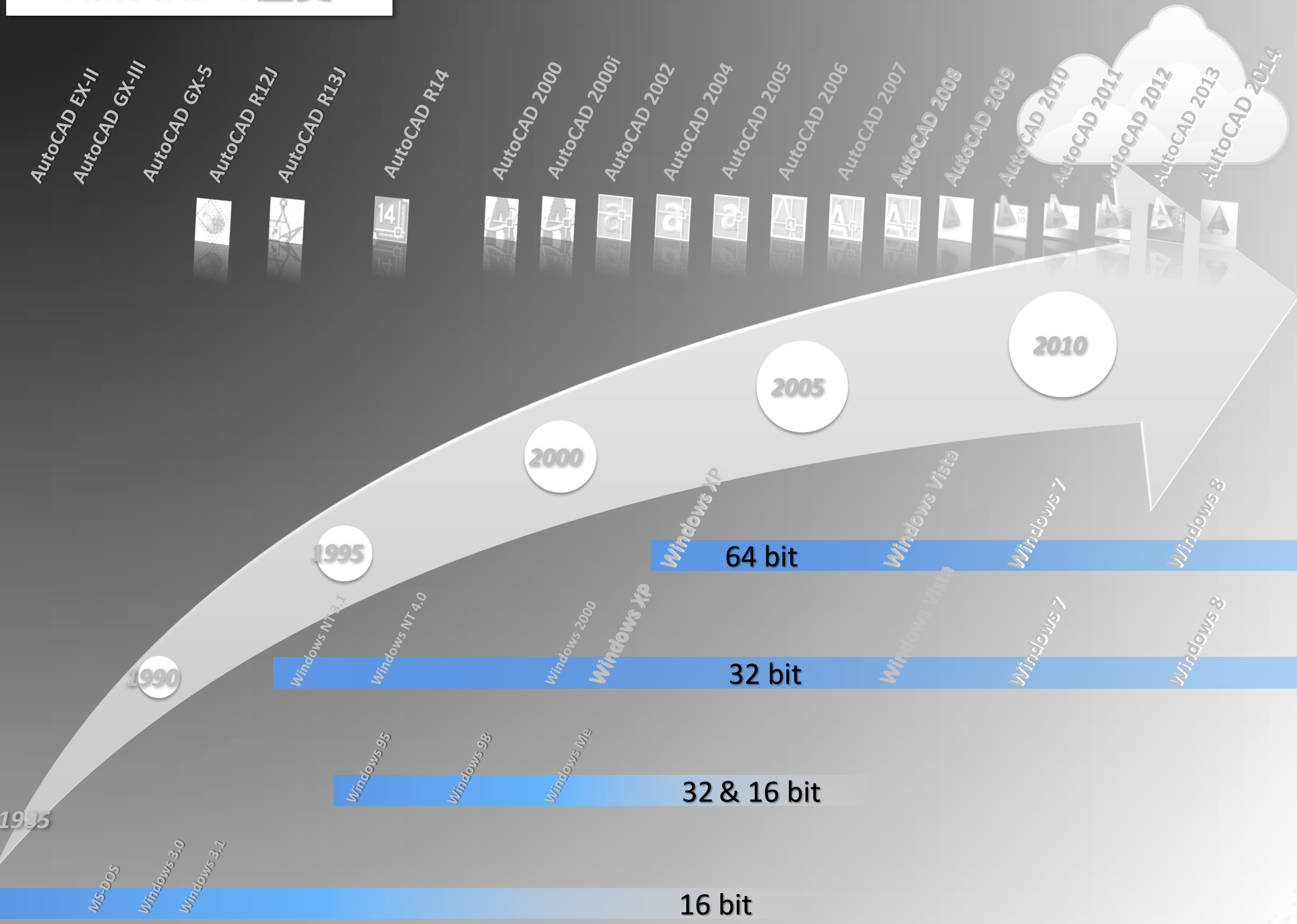
- **.NET API**

- 影響は**中**
- プラットフォーム差※は .NET Framework が吸収

AutoCAD の歴史



AutoCAD の歴史



ObjectARX アプリケーション
64 ビット版 Windows への対応

- 利用可能なメモリ領域の拡大



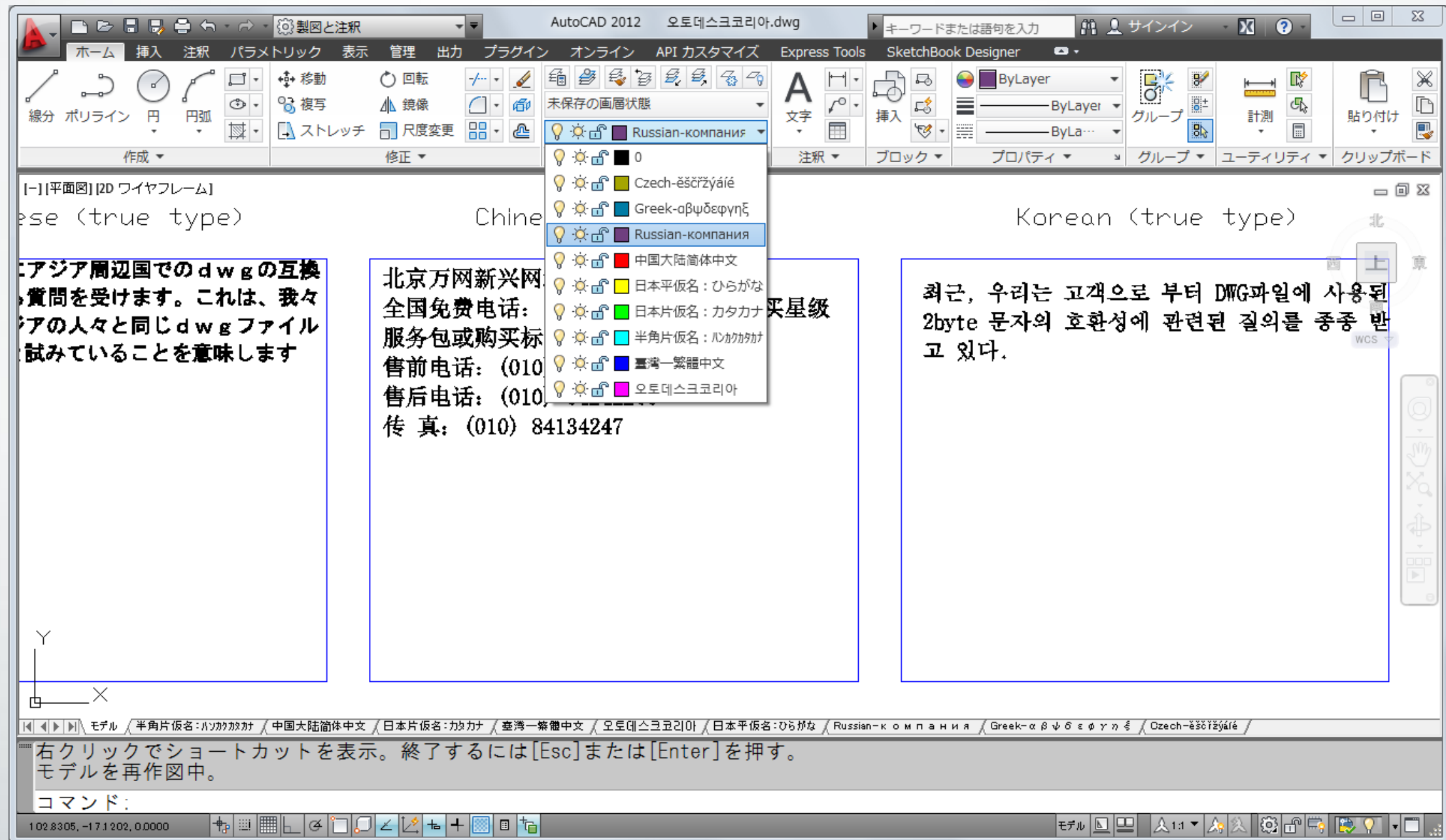
... 2^{32} vs. 2^{64}

アーキテクチャ コンポーネント	Windows 32 ビット	Windows 64 ビット
論理仮想メモリ	4 ギガバイト(GB)	16 テラバイト(TB)
ユーザ モード (1アプリケーションが利用可能な領域)	2 ギガバイト(GB)	8 テラバイト(TB)

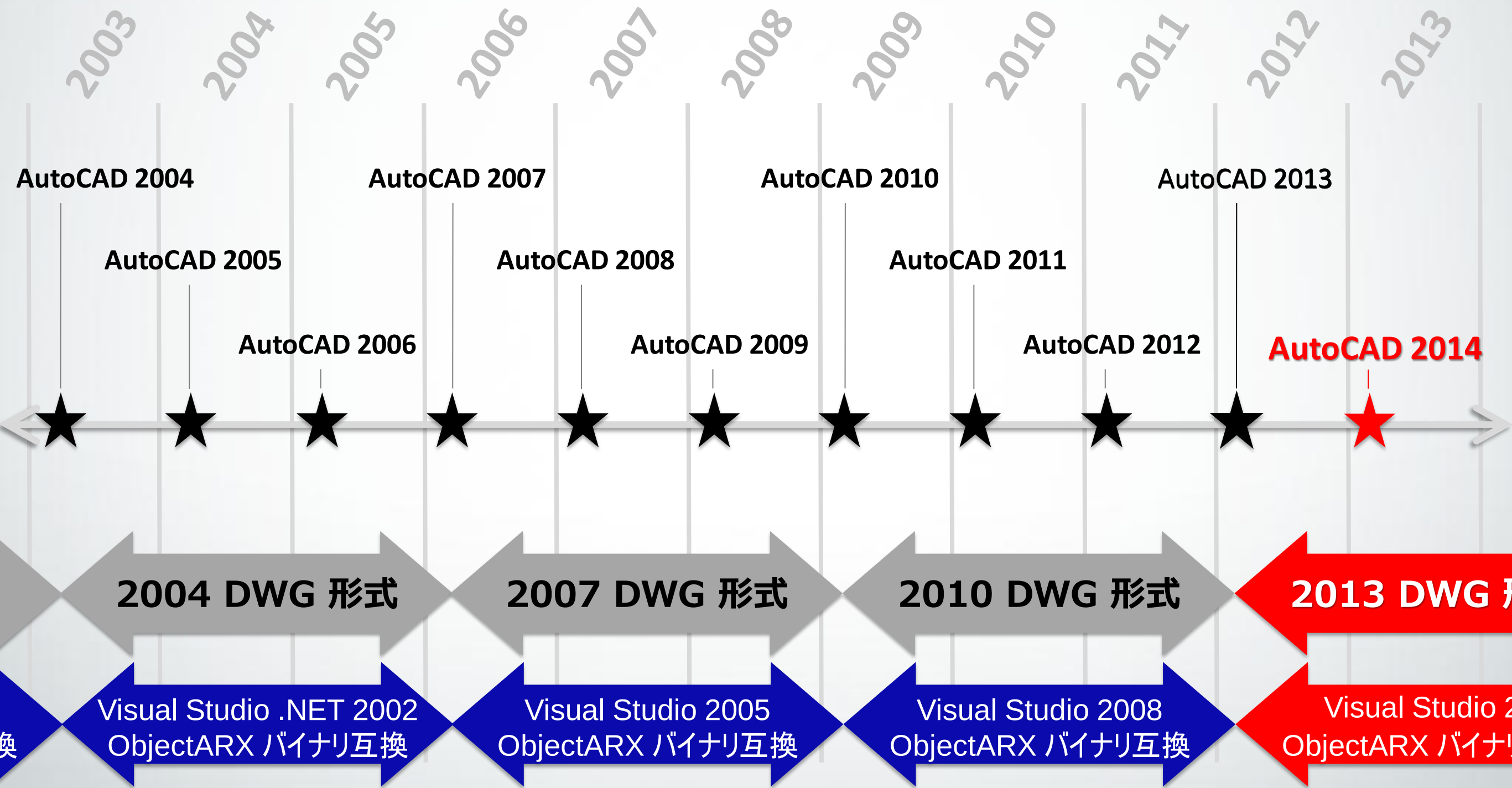
AutoCAD 2008 以降、32 ビット版 AutoCAD と 64 ビット版 AutoCAD を同梱しています

ObjectARX アプリケーション 国際化のための UNICODE 対応

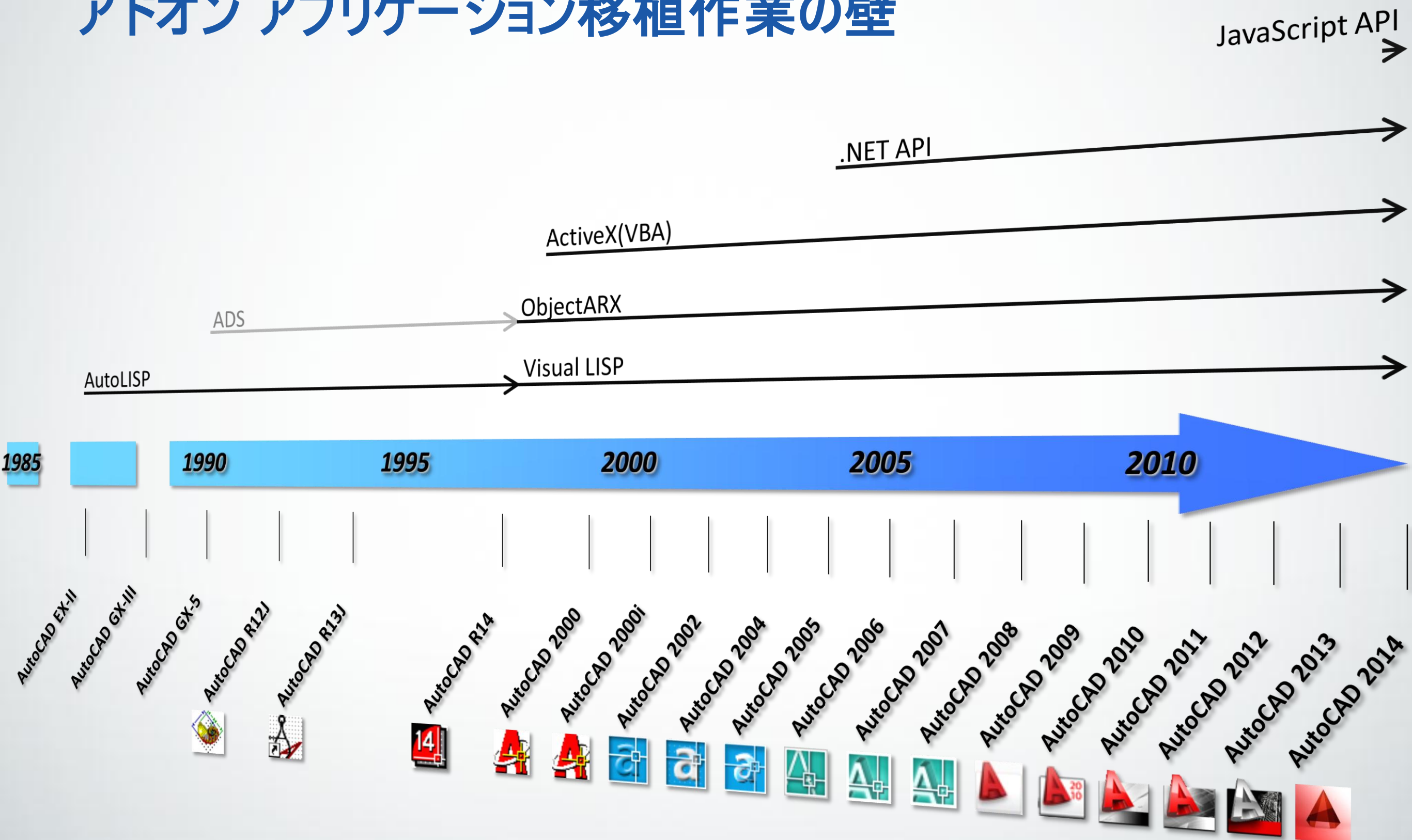
- AutoCAD エディタ上では UNICODE 文字コードで表示、編集



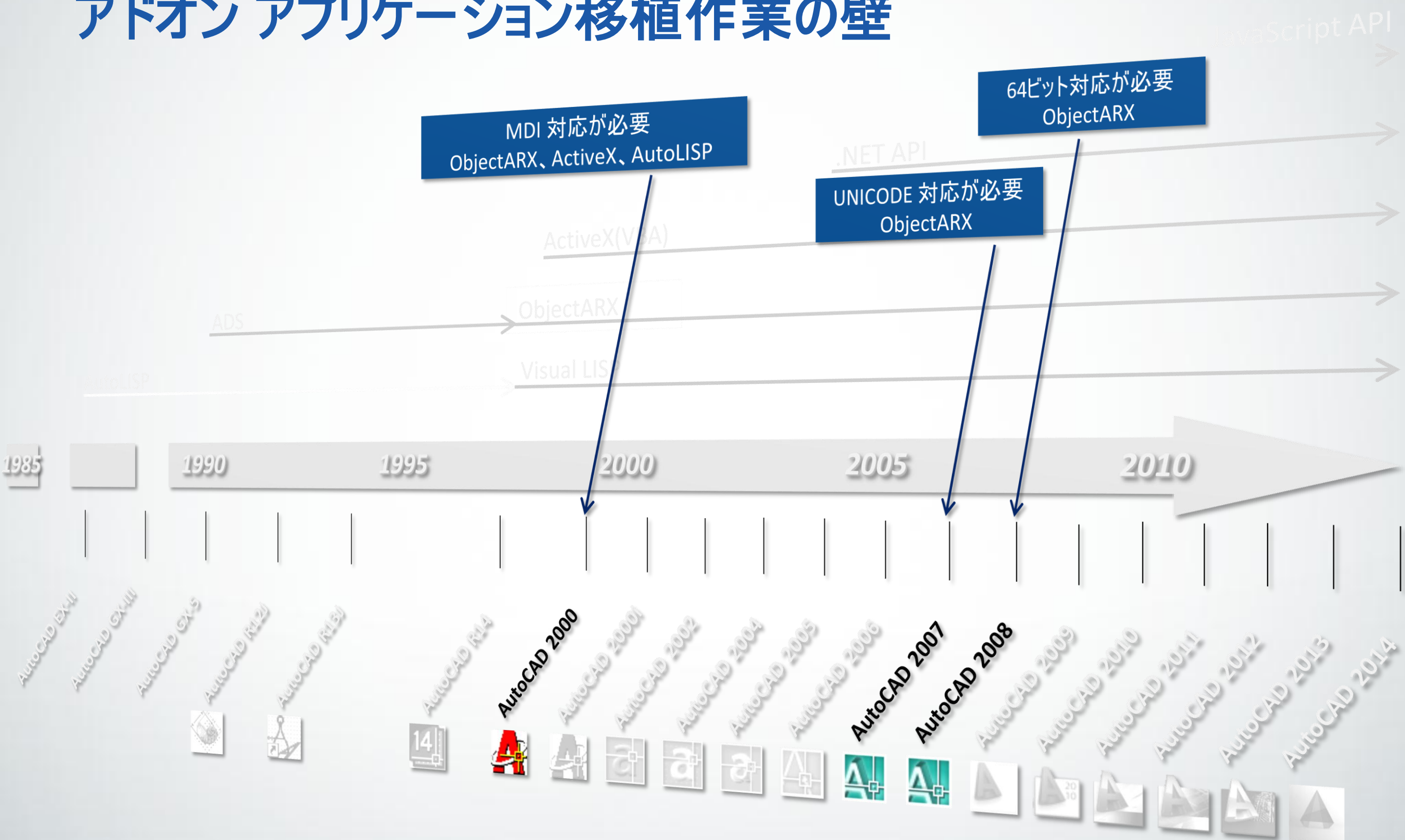
図面ファイルとアドオン アプリケーションの互換性



アドオン アプリケーション移植作業の壁

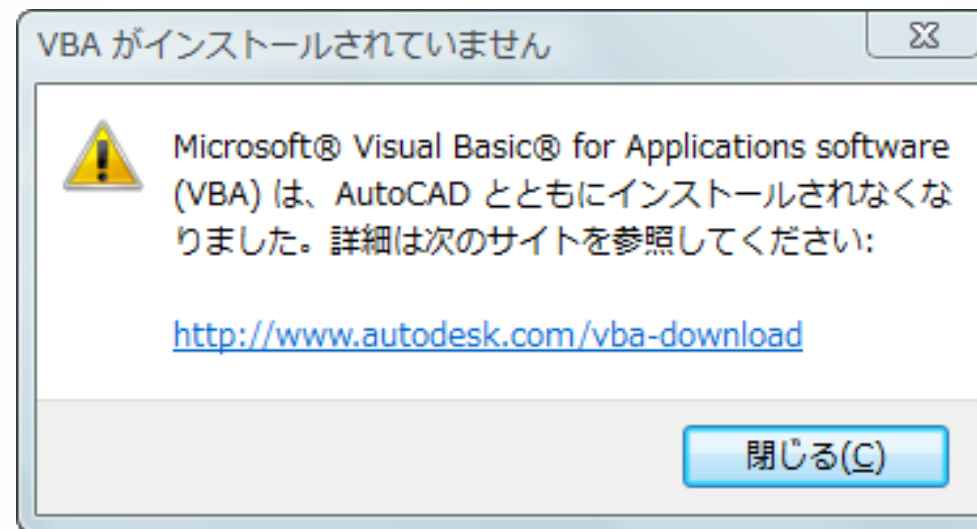


アドオン アプリケーション移植作業の壁



AutoCAD VBA について

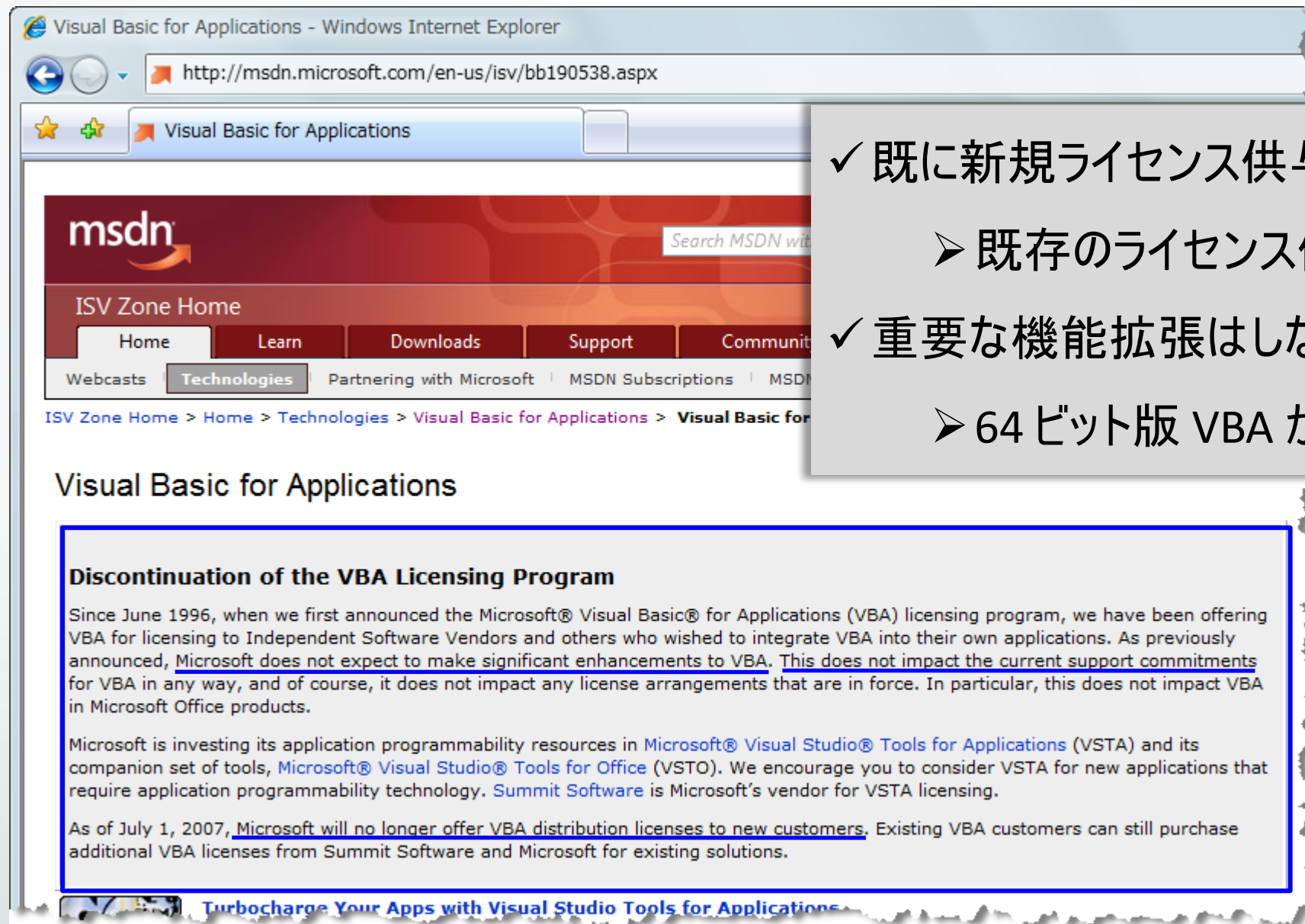
- AutoCAD 2010～2013 用に Web からの無償ダウンロードで提供
 - <http://www.autodesk.com/vba-download>



- 64 ビット版 VBA が利用可能ですが .NET API への移植を推奨を継続
 - .NET API では COM と .NET のテクノロジーの併用が可能

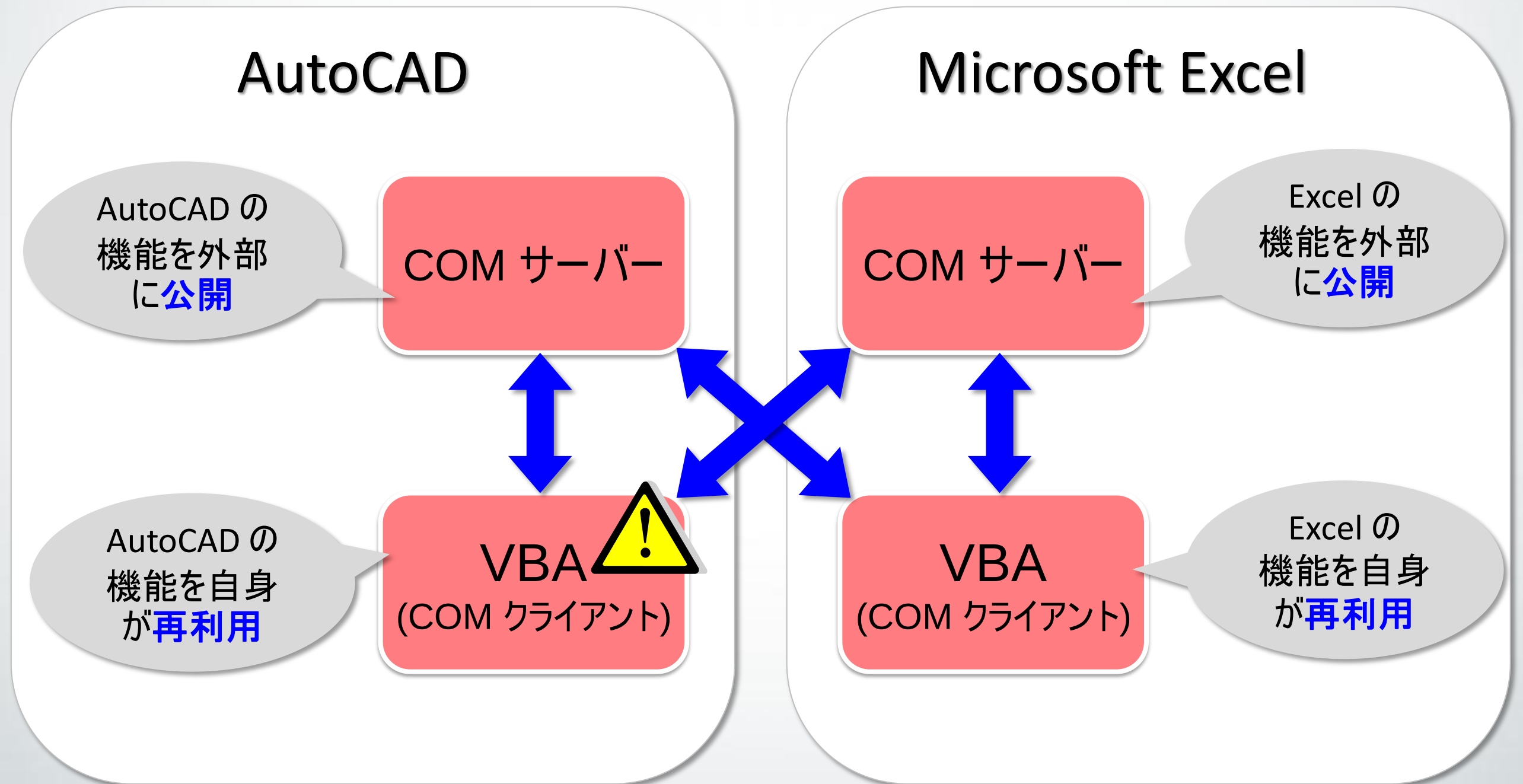
AutoCAD VBA について

■ 2007 年 7 月に発表された Microsoft 社のアナウンス

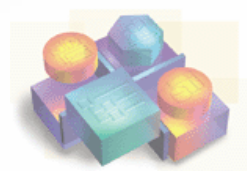


- ✓ 既に新規ライセンス供与を停止
 - 既存のライセンス供与には影響せず
- ✓ 重要な機能拡張はしない
 - 64 ビット版 VBA がない

製品と VBA の動作メカニズム: ActiveXオートメーション



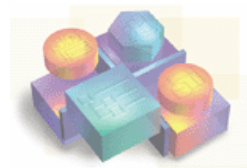
VBA プロジェクトの移植の考え方



- VBA プログラム は COM テクノロジ を使っています



- 最新の AutoCAD .NET API は .NET テクノロジ を使っています



- .NET テクノロジ から COM テクノロジを利用できます



- AutoCAD .NET API 用のモジュールに VBA プログラム を移植します

AutoCAD ベースの 業種別製品の API



AutoCAD ベース製品固有の API

- 固有オブジェクト、固有操作等が可能な API の有無

製品	AutoLISP	ObjectARX	ActiveX/COM	.NET API
AutoCAD	○	○	○	○
AutoCAD Mechanical	×	△	△	×
AutoCAD Electrical	○	×	×	×
AutoCAD Map 3D	×	△	△	○
AutoCAD Civil 3D	×	×	○	○
AutoCAD Architecture	×	○	△	○
AutoCAD P&ID	×	○	×	○
AutoCAD Plant 3D	×	○	×	○
Raster Design	×	×	×	×

△：一部の固有オブジェクト、固有操作のみの API 制御のみ可能

※ AutoCAD が提供する 標準の API 5 種はどの業界別製品でも動作します



Autodesk is a registered trademark of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and/or other countries. All other brand names, product names, or trademarks belong to their respective holders. Autodesk reserves the right to alter product and services offerings, and specifications and pricing at any time without notice, and is not responsible for typographical or graphical errors that may appear in this document.