GDI - Graphics Device Interface

- A graphical component of Windows
- It allows applications to use graphical devices without any knowledge of their drivers
- Applications create logical graphical objects which are realized during selecting on the device context
- GDI uses existing graphical possibilities of devices or can simulate them if missing
- Applications can use logical coordinates, devices use real coordinates

Drawing

- Coordinates on the screen in pixels
  - (0,0) at the left top corner
  - MapWindowPoints()
- Regions
  - update
  - visible
  - clipping
- Bounding rectangle - SetBoundsRect(), GetBoundsRect()
- Update lock - LockWindowUpdate()
- Drawing window's icon [up to NT 3.51]

Drawing Window's Content

- Background of the window
  - WM_ERASEBKGND
  - WNDCLASSEX.hbrBackground
- The client area
  - WM_PAINT - BeginPaint(), EndPaint()
  - GetDC(), GetDCEx(), ReleaseDC()
- The nonclient area
  - usually manage by the system
  - WM_NCPAINT, WM_NCACTIVATE
  - GetWindowDC(), GetDCEx(), ReleaseDC()

WM_PAINT

- BeginPaint()
  - gets the DC with a clipping region
  - sends WM_ERASEBKGND
  - sets window's update region to NULL
  - (call this function only in response for WM_PAINT)
- EndPaint()
- Update region
  - InvalidateRect(), InvalidateRgn()
  - ValidateRect(), ValidateRgn()
  - GetUpdateRect(), GetUpdateRgn(), ExcludeUpdateRgn()
- Immediate redrawing of the client area:
  - UpdateWindow(), RedrawWindow()

GDI Objects

- Pens, brushes, fonts, bitmaps, palettes, regions
- Creating and destroying
  - Create...(), e.g. CreatePen(), CreateSolidBrush(), CreateFont ()
  - all created objects must be destroyed
  - DeleteObject()
- SelectObject()
  - it realizes a logical object according to the capabilities of DC
  - objects selected as current must not be destroyed
- Stock objects
  - GetStockObject()

Using GDI Objects

```
hNewPen = CreatePen(PS_SOLID, 1, RGB(255, 0, 0));
if (hNewPen)
  hOldPen = SelectObject(hDC, hNewPen);
else
  hOldPen = NULL;                // no selection

.                                // all drawing operations
.                                // uses the new pen

if (hOldPen)
  SelectObject(hDC, hOldPen); // deselect hNewPen
if (hNewPen)
  DeleteObject(hDC, hNewPen); // delete the pen if created
```

Colours

- Values
  - COLORREF
  - RGB
  - GetRValue(), GetGValue(), GetBValue()
- Availability of colours
  - GetNearestColor()
- Combining pens and interiors with colors on the screen
  - SetROP2(), GetROP2()
- Palettes (useful, when only 256 colours can be used)
  - CreatePalette(), DeleteObject() SelectPalette(), RealizePalette(), UnrealizeObject(), ResizePalette()
  - GetPaletteEntries(), GetNearestPaletteIndex()
  - GetSystemPaletteEntries(), GetSystemPaletteUse()

Device Context (DC)

- Current objects used during drawing operations
  - SelectObject(), GetCurrentObject(), GetObject()
- Modes set on the DC:
  - background - SetBkMode(), GetBkMode()
  - drawing - SetROP2(), GetROP2()
  - mapping - SetMapMode(), GetMapMode()
  - polygon fill - SetPolyFillMode(), GetPolyFillMode()
  - stretching - SetStretchBltMode(), GetStretchBltMode()
- Types of device contexts:
  - screen - GetDC(), GetWindowDC(), BeginPaint()
  - printer - CreateDC()
  - memory - CreateCompatibleDC()
  - information - CreateIC()

Clipping

- Setting the clipping region
  - SelectClipRgn(), ExtSelectClipRgn(),
  - SelectClipPath()
- Checking visibility
  - PtVisible(), RectVisible()
- Modifying the clipping region
  - OffsetClipRgn(), ExcludeClipRect(), IntersectClipRect()

Pens

- Cosmetic pens
  - available properties: width, style, and colour
  - they always have fixed width (no scaling)
  - CreatePen(), CreatePenIndirect(), ExtCreatePen(), GetStockObject()
- Geometric pens
  - available properties: width, style, colour, pattern, optional hatch, end style and join style
  - can be scaled – width is set using logical coordinates
  - ExtCreatePen()

Lines and Curves

- Always drawn using the current pen
- Lines
  - MoveToEx(), LineTo()
  - PolylineTo(), Polyline(), PolyPolyline()
- Curves
  - Arc(), SetArcDirection(), GetArcDirection()
  - PolyBezier()
- Lines and curves
  - AngleArc(), PolyDraw()

Brushes

- Types of brushes:
  - solid - CreateSolidBrush()
  - hatch - CreateHatchBrush()
  - pattern - CreatePatternBrush(), CreateDIBPatternBrushPt()
- Stock brushes
  - GetStockObject()
- Patterns
  - PatBlt()
  - brush origin: SetBrushOrgEx(), GetBrushOrgEx()

Filled Figures

- The outline is drawn using the current pen
- The interior is filled using the current brush
- Rectangles:
  - Rectangle(), RoundRect()
  - FillRect(), FrameRect(), InvertRect()
- Other figures:
  - Ellipse()
  - Chord()
  - Pie()
  - Polygon()

Fonts

- Properties:
  - typeface, style, size
- Fonts families:
  - decorative, dontcare, modern, roman, script, swiss
- Types of fonts:
  - raster, vector, TrueType, OpenType
- Font's character set
  - Windows, Unicode, OEM, symbol

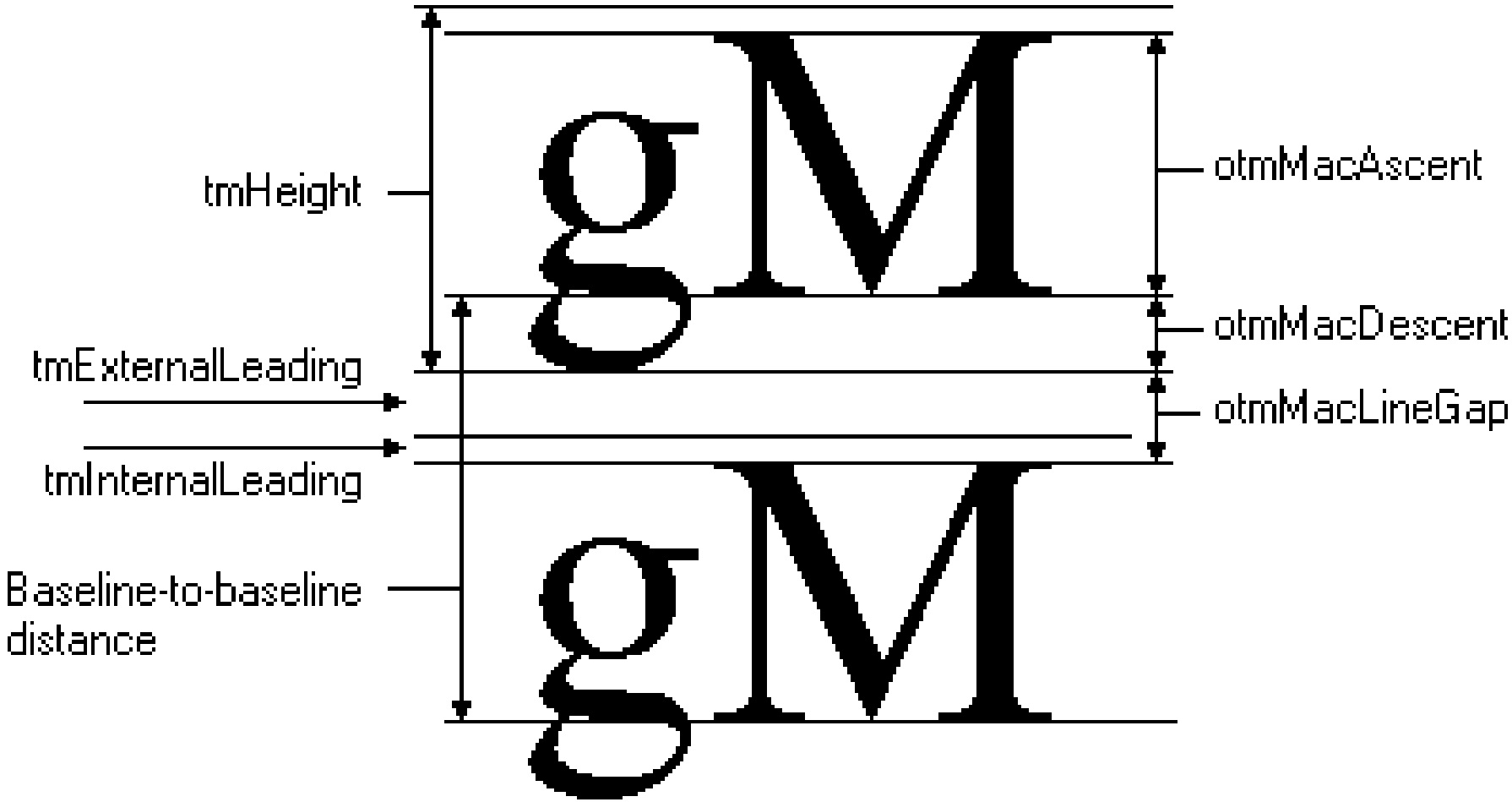
Using Fonts

- Creating a font
  - CreateFont(), CreateFontIndirect()
  - ChooseFont()
- Selecting a font on DC as current
  - SelectObject()
- Enumerating fonts installed in the system
  - EnumFonts(), EnumFontFamiliesEx()
- Getting information about a font
  - GetFontData(), GetOutlineTextMetrics(), GetGlyphOutline()
- Installing a font in the system
  - AddFontResource(), AddFontResourceEx()
  - RemoveFontResource()
  - WM_FONTCHANGE

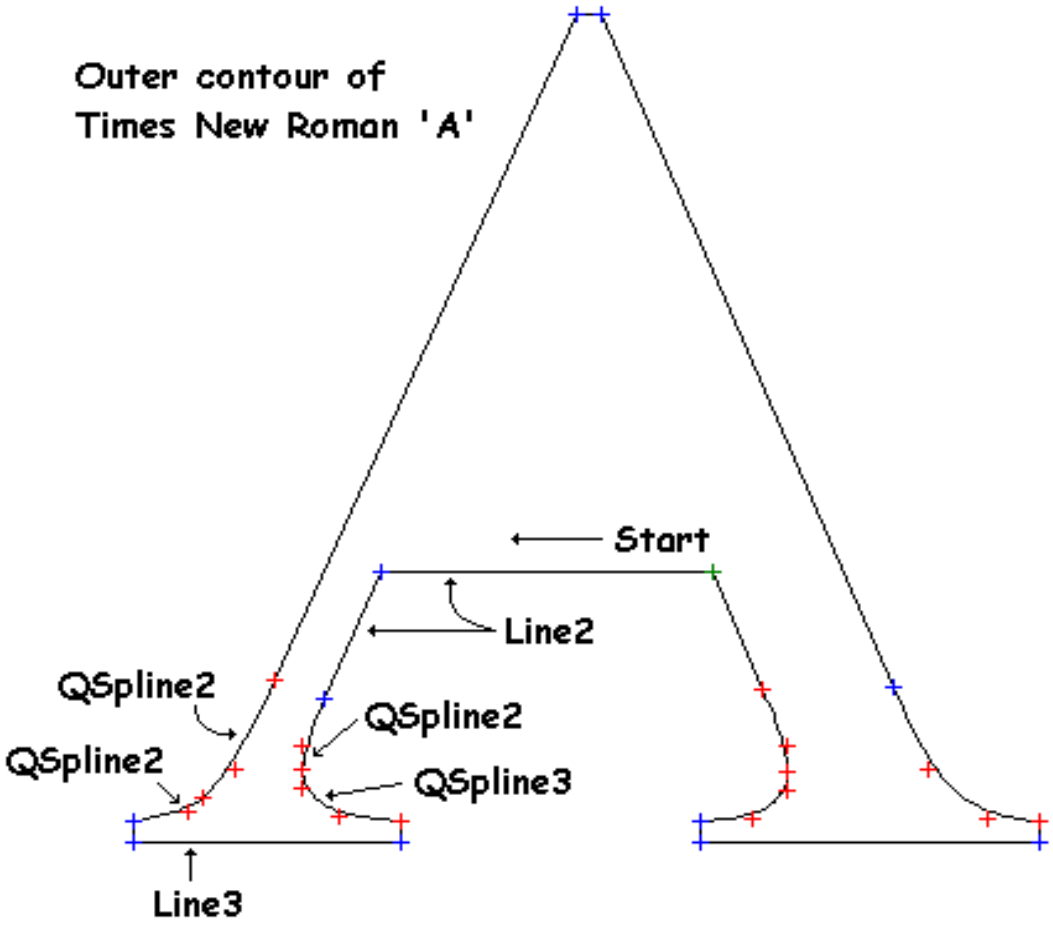
Texts

- Formatting
  - SetBkColor(), SetBkMode(), SetTextColor()
  - SetTextAlign(), SetTextCharacterExtra()
  - SetTextJustification()
- Getting size of the drawn text
  - GetTextExtendPoint32(), GetTabbedTextExtend()
  - GetCharWidth32(), GetCharWidthFloat()
  - GetCharABCWidths(), GetCharABCWidthsFloat()
  - GetTextMetrics(), GetOutlineTextMetrics()
- Drawing
  - DrawText(), DrawTextEx()
  - TextOut(), ExtTextOut(), PolyTextOut(), TabbedTextOut()

Font Size


tmHeight
otmMacAscent
otmMacDescent
tmExternalLeading
otmMacLineGap
tmInternalLeading
Baseline-to-baseline distance
gM
gM

Glyph Outline



http://support.microsoft.com/default.aspx?scid=kb;en-us;243285
http://my.execpc.com/~dg/tutorial/Glyph/Glyph.html

Rectangles

- RECT, *PRECT
- Operations
  - SetRect()
  - SetRectEmpty(), IsRectEmpty()
  - EqualRect(), CopyRect()
  - InflateRect(), OffsetRect()
  - PtInRect()
  - IntersectRect(), UnionRect()

Regions

- HRGN
- Creating
  - CreateRectRgn(), CreateRoundRectRgn(), CreateEllipticRgn(), CreatePolygonRgn()
- Selecting
  - SelectObject
- Filling
  - FillRgn()
  - SetPolyFillMode(), GetPolyFillMode()
- Drawing
  - PaintRgn()

Region Operations

- Combining
  - CombineRgn()
- Inversion
  - InvertRgn()
- Moving
  - OffsetRgn()
- Checking if a point is inside the region
  - PtInRegion()
- Getting the bounding rectangle
  - GetRgnBox()

Paths

- Creating
  1. BeginPath()
  2. drawing using GDI functions (not all functions are supported)
  3. EndPath()
- Stroking
  - StrokePath(), StrokeAndFillPath()
- Filling
  - FillPath(), SetPolyFillMode(), GetPolyFillMode()
- Clipping
  - SelectClipPath()
- Converting a path to a region
  - PathToRegion()

Metafiles

- Creating
  1. CreateEnhMetaFile()
  2. drawing using GDI functions
  3. CloseEnhMetaFile()
- Deleting
  - DeleteEnhMetaFile()
- Operations on metafiles
  - PlayEnhMetaFile()
  - CopyEnhMetaFile()
  - EnumEnhMetaFile()
  - GetEnhMetaFileHeader(), GetEnhMetaFileDescription()

Bitmaps

- Creating
  - CreateBitmap(), CreateBitmapIndirect(), CreateCompatibleBitmap()
  - DeleteObject()
- Rotating
  - PlgBlt() [NT]
- Scaling
  - StretchBlt(), SetStretchBltMode()
- Using a mask
  - MaskBlt() [NT]
- Getting and setting pixels
  - GetPixel(), SetPixel()

Bitmaps Special Effects

- Filling
  - PatBlt()
  - FloodFill()
- Transparency
  - AlphaBlend() [98/2000]
  - TransparentBlt() [98/2000]
- Gradient
  - GradientFill() [98/2000]

Types of Bitmaps

- Device dependent
  - contains a table of colours
  - BITMAPINFO, BITMAPINFOHEADER, RGBQUAD
  - GetDeviceCaps()
- Device Independent Bitmap - DIB
  - BITMAP – no table of colours
- .BMP files
  - there is no function to save a bitmap to a file, it can be saved manually using structures: BITMAPINFO, BITMAPINFOHEADER, RGBQUAD
  - LoadBitmap() – to read a bitmap from resources
  - LoadImage() – to read a bitmap from resources or a file

Flicker-Free Drawing

- Turn off drawing of the background
  - do nothing in response for the WM_ERASEBKGND message
- Use memory (offline) device context

```
HDC hMemDC = CreateCompatibleDC(hDC);
HBITMAP hBmp = CreateCompatibleBitmap(hDC,
                                      nWidth, nHeight);
HBITMAP hOldBmp = (HBITMAP)SelectObject(
                                      hMemDC, hBmp);

// drawing on hMemDC

BitBlt(hDC, 0, 0, nWidth, nHeight,
       hMemDC, 0, 0, SRCCOPY);

SelectObject(hMemDC, hOldBmp);
DeleteObject(hBmp);
DeleteDC(hMemDC);
```

Drawing Transparent Bitmaps

```
//HBITMAP hBmp = (HBITMAP)LoadImage(NULL, filePath,
//     IMAGE_BITMAP, 0, 0, LR_LOADFROMFILE);
HBITMAP hBmp = LoadBitmap(hInst,
    MAKEINTRESOURCE(IDB_SARG_MOVE_01));
BITMAP bmpInfo;
GetObject(hBmp, sizeof(BITMAP), &bmpInfo);
```

```
HDC hTmpDC = CreateCompatibleDC(hdc);
HBITMAP hOldBmp = (HBITMAP)SelectObject(
    hTmpDC, hBmp);

COLORREF transparentColor = GetPixel(hTmpDC, 0, 0);
TransparentBlt(hdc, 0, 0,
    bmpInfo.bmWidth, bmpInfo.bmHeight,
    hTmpDC, 0, 0, bmpInfo.bmWidth,
    bmpInfo.bmHeight, transparentColor);

SelectObject(hTmpDC, hOldBmp);
DeleteDC(hTmpDC);
```

Coordinate Space

- Transformations
  - SetWorldTransform() [NT3.1]
- Mapping modes
  - SetMapMode(), GetMapMode()
  - MM_TEXT, MM_TWIPS
  - MM_ANISOTROPIC, MM_ISOTROPIC,
  - MM_HIENGLISH, MM_LOENGLISH, MM_HIMETRIC, MM_LOMETRIC
- Custom coordinate space
  - SetWindowOrgEx(), SetWindowExtEx()
  - SetViewportOrgEx(), SetViewportExtEx()
- Converting device points to logical points and vice versa
  - DPtoLP(), LPtoDP()