

I have needed something to do this and it ended as quite nice small utility and in examples folder...

BTW, the conversion routine is surprisingly simple in U++:

```
struct TextToSvg : FontGlyphConsumer {
    String t;

    void Put(Pointf p);

    virtual void Move(Pointf p);
    virtual void Line(Pointf p);
    virtual void Quadratic(Pointf p1, Pointf p2);
    virtual void Cubic(Pointf p1, Pointf p2, Pointf p3);
    virtual void Close();
};

void TextToSvg::Put(Pointf p)
{
    t << Format("%.2f %.2f ", p.x, p.y);
}

void TextToSvg::Move(Pointf p)
{
    t << 'M';
    Put(p);
}

void TextToSvg::Line(Pointf p)
{
    t << 'L';
    Put(p);
}

void TextToSvg::Quadratic(Pointf p1, Pointf p)
{
    t << 'Q';
    Put(p1);
    Put(p);
}
```

```
void TextToSvg::Cubic(Pointf p1, Pointf p2, Pointf p)
{
    t << 'C';
    Put(p1);
    Put(p2);
    Put(p);
}
```

```
void TextToSvg::Close()
{
    t << 'Z';
}
```

```
String TextToSvgPath(double x, double y, const char *text, Font fnt, bool singleline)
{
    WString ws = ToUnicode(text, CHARSET_DEFAULT);
    TextToSvg t;
    for(const wchar *s = ~ws; *s; s++) {
        fnt.Render(t, x, y, *s);
        x += fnt[*s];
        if(!singleline)
            t.t << "\n";
    }
    return t.t;
}
```

File Attachments

1) [TextToSvg.png](#), downloaded 597 times
