# StartPage窗口代码分析报告

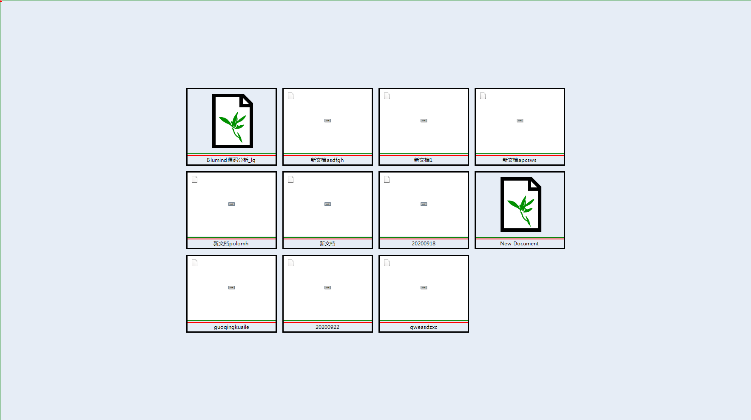
## 一、功能描述

Blumind的startPage页主框架均采用手动绘制方式实现。

本报告主要分析右startPage窗口页的布局方式，绘制方式（包括各种特效的实现）。

## 二、效果预览

本次代码梳理共计耗时1个工作日，梳理完代码后目前可以自定义了startPage界面视觉特效，达到预期分析目标。自定义界面如下：

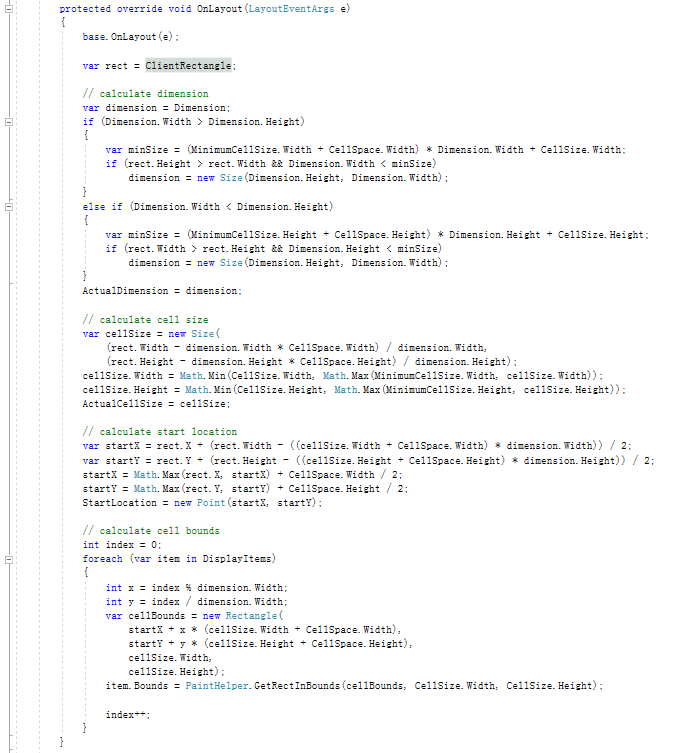


## 三、业务功能实现

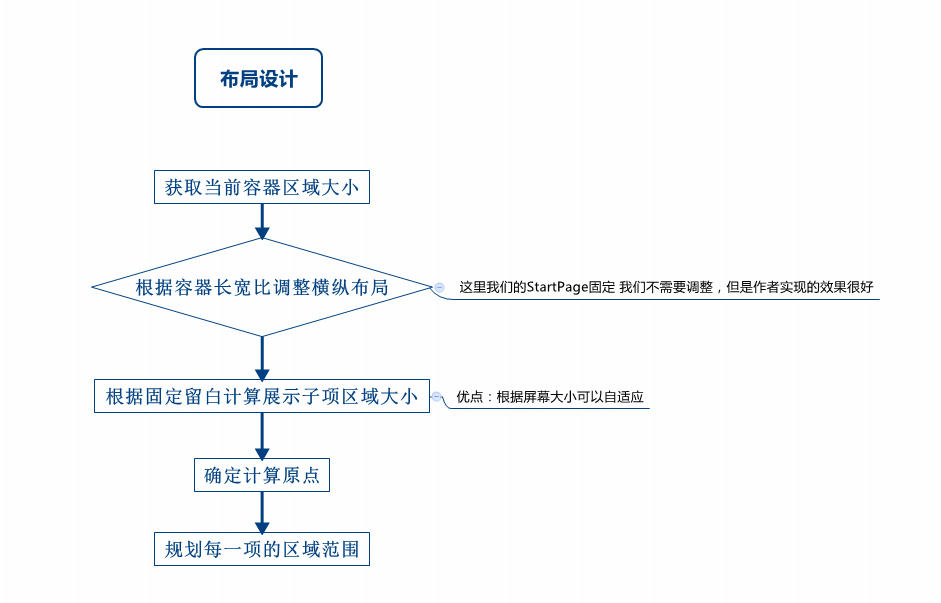
startpage页的关键点在于两部分，布局函数中计算好展示项的位置区域，paint中分块绘制 图片，文字及相关的选中特效。

### 3.1 布局函数分析

布局功能位于ThumbView 的OnLayout函数。代码概览如下：

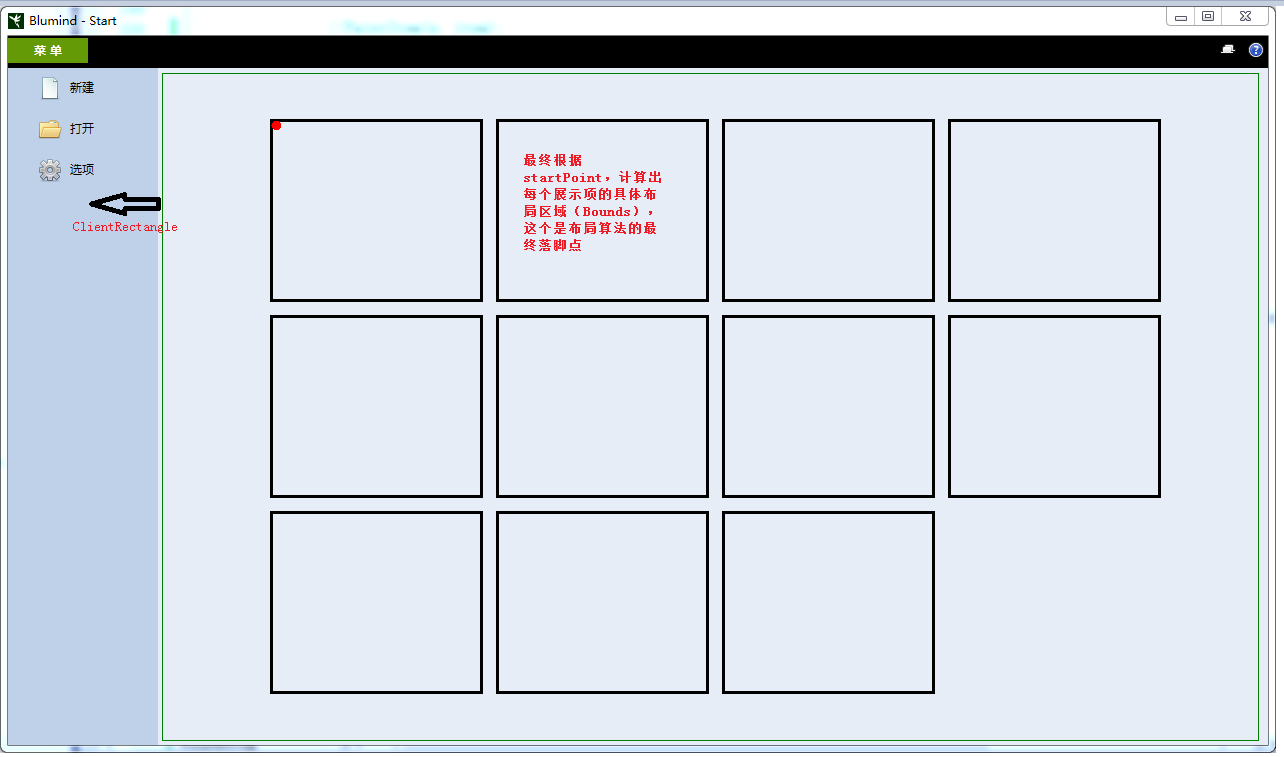


#### 3.1.1 实现思路



#### 3.1.2 落脚

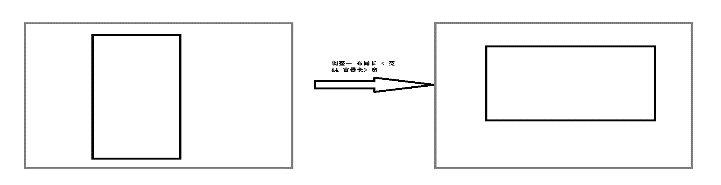
Onlayout 布局将最近打开文档的缩略图所放置的区域划分好，即每项对应的Bounds。注：红色圆点即为布局过程中计算的StartLocation



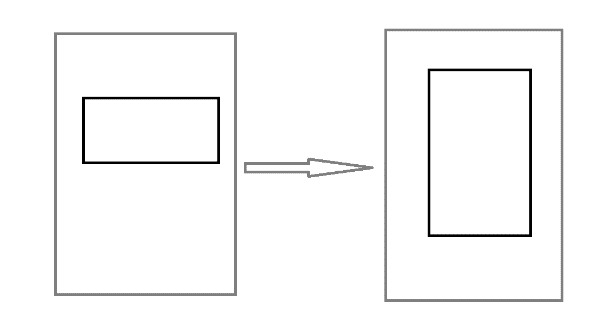
#### 3.1.3 容器长宽调整

1. 实现思路

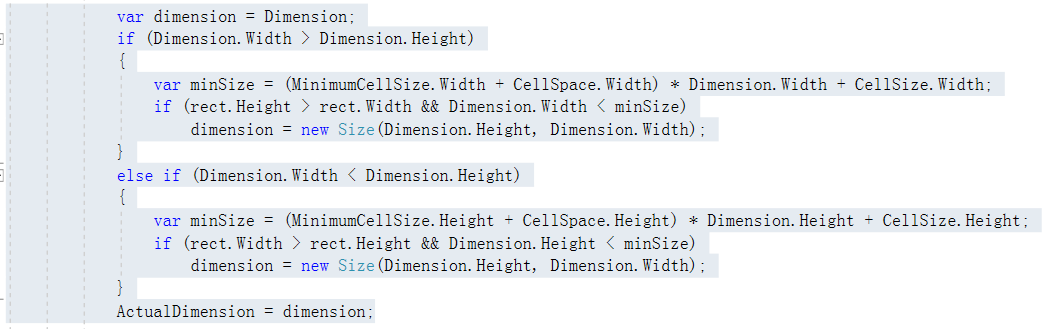
情况一：布局长< 布局宽 && 容器长 > 容器宽



情况二：布局长 > 布局宽 && 容器长 < 容器宽



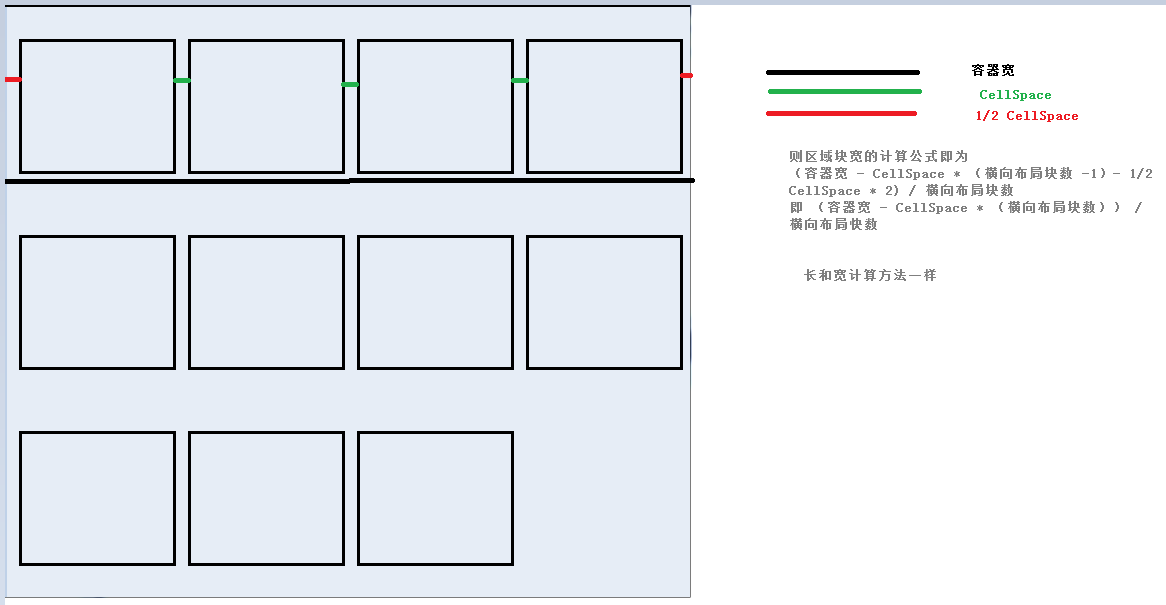
1. 实现代码



#### 3.1.4 展示子项区域大小确定

1. 计算方法

Step1：先计算出左右两侧各留1/2 cellsapce大小空间，且各项之间留cellsapce大小空间的极限区域块大小cellSzie。

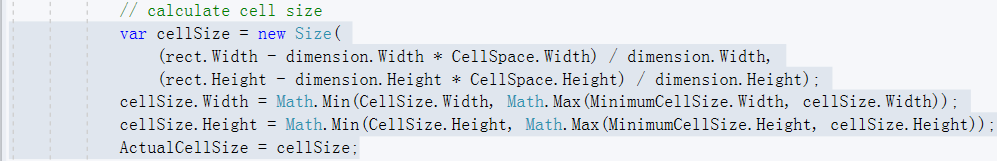


Step2:

与阈值做判断，不能过大也不能过小，界即设为对应的阈值(将三值的中位数作为区域块大小)

MinimumCellSize[默认] =< cellSzie <= CellSize[默认]

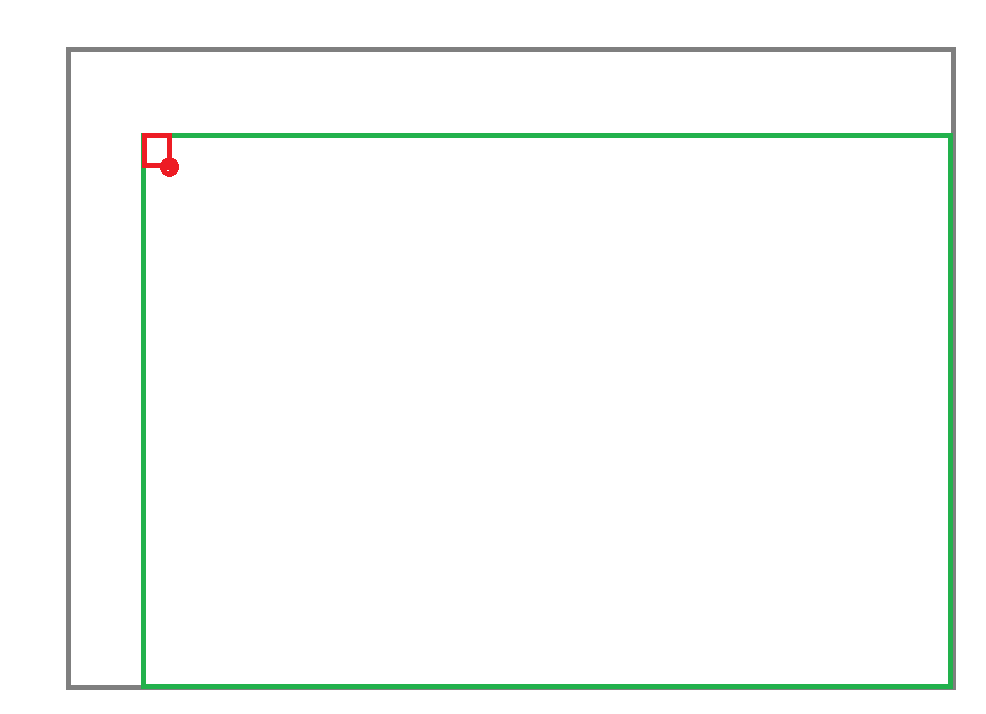
1. 具体代码



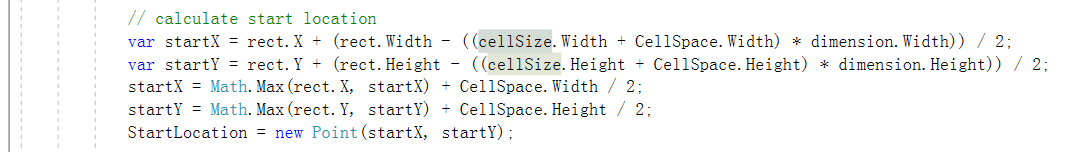
#### 3.1.5 布局起点确定

（1）思路

大小确定，起点就很好定了。以容器区域右下角开始，减去规划项区域大小，确立坐标点，并向右向下便宜1/2 留白。如果发现减去后的坐标点小于容器的左上角点，则调整为容器左上角点内收1/2留白。



1. 代码

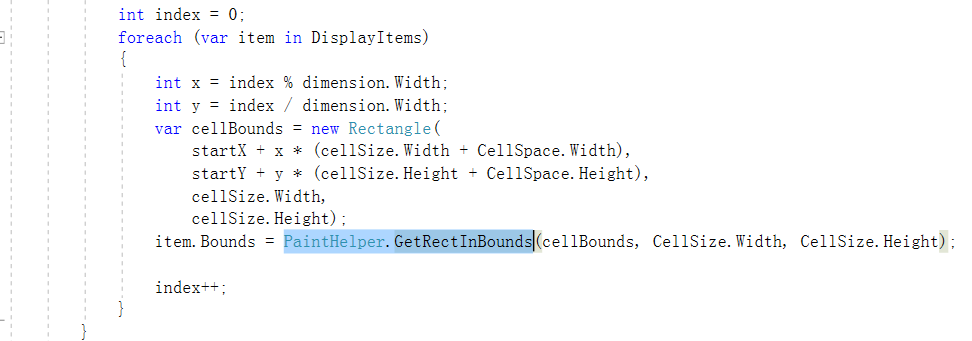


#### 3.1.6 各项位置具体规划

1. 思路

主要是通过对每行放置窗口数相除取证确定第几行，除余确定第几列，加上对应的空白及自身大小，PaintHelper.GetRectInBounds中 对计算值进行微调。

1. 代码



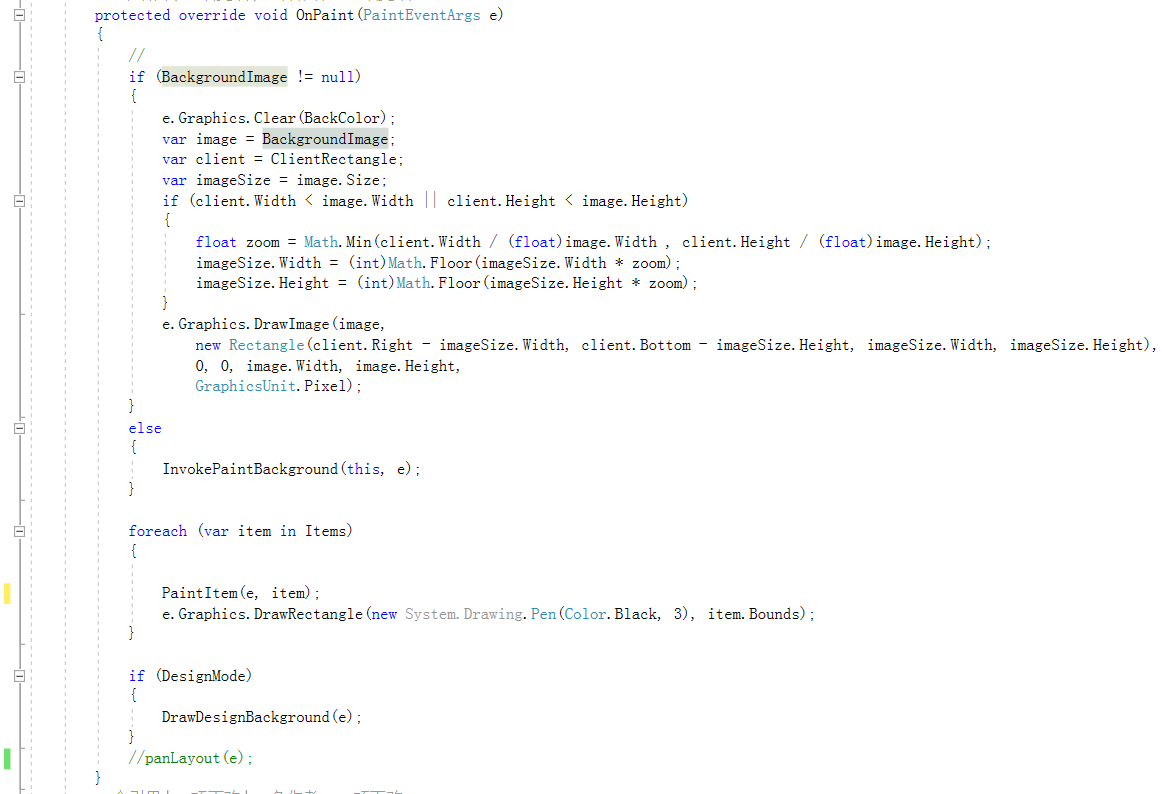
#### 3.1.7 函数调用

布局函数和重绘函数一样，有属于自己的激活函数，当需要指定情况下重新布局时，即可调用

PerformLayout();

### 3.2 Paint函数分析

Paint 里面的核心点在于图片，字体均为作者绘制上去，且模拟了悬浮特效，未采用任何控件，因此会发现在startpage 自适应界面调整布局的时候，非常流畅。入口函数Onpaint()代码如下：



#### 3.2.1 绘制思路



#### 3.2.2 细节分析

作者循环绘制小窗口的时候，先于红色区域绘制文档名 【Graphics.DrawString()函数】，然后于图片区域(绿色线框部分)绘制图片【DrawImage()函数】。

