

基于压缩技术的影像管理组织和发布



顾学明

esri中国 专业服务部

内容

- 影像成为GIS核心价值
- 影像处理工作流程和组织方式
- eYalimage影像压缩技术
- eYalimage产品
- eYalimage和ArcGIS
- eYalimage On Internet
- 总结

影像成为GIS核心价值的一部分

- 应用
 - 土地调查和监督
 - 环境监测
 - 灾害评估
 - 军事侦查
 - 影像制图
 - ...
- 获取成本/处理和使用成本
 - 存储
 - 管理
 - 处理
 - 发布
 - 访问

影像处理 workflows 和组织方式



eYalimage影像压缩

- eYalimage小波压缩(DWT)

- 原理

- 高低分频

- 高频相当于边缘提取

- 低频相当于平滑操作

- 局部算法，运算连续运算量少，不用分段

- 较好的高低频特性

- 特点

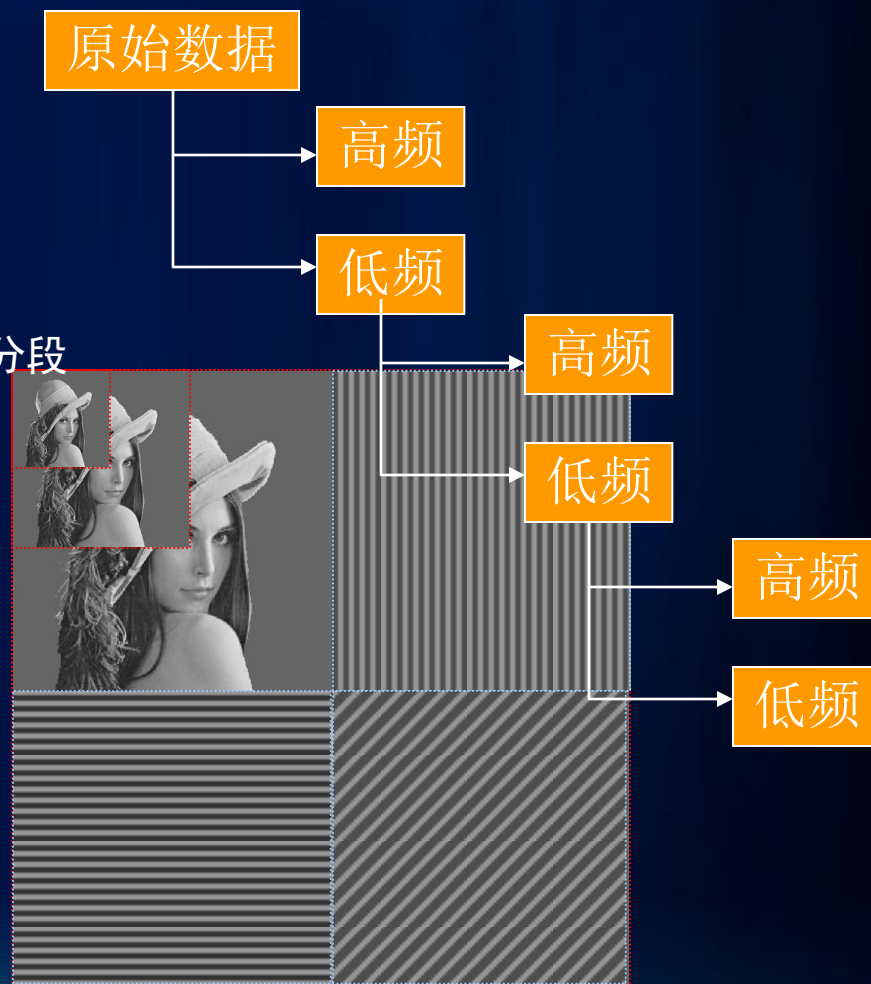
- 整体计算,无边缘损失

- 压缩质量好,质量高

- 内置金字塔

- 分块存储,局部解压

- 快速提取和显示



eYalimage影像压缩

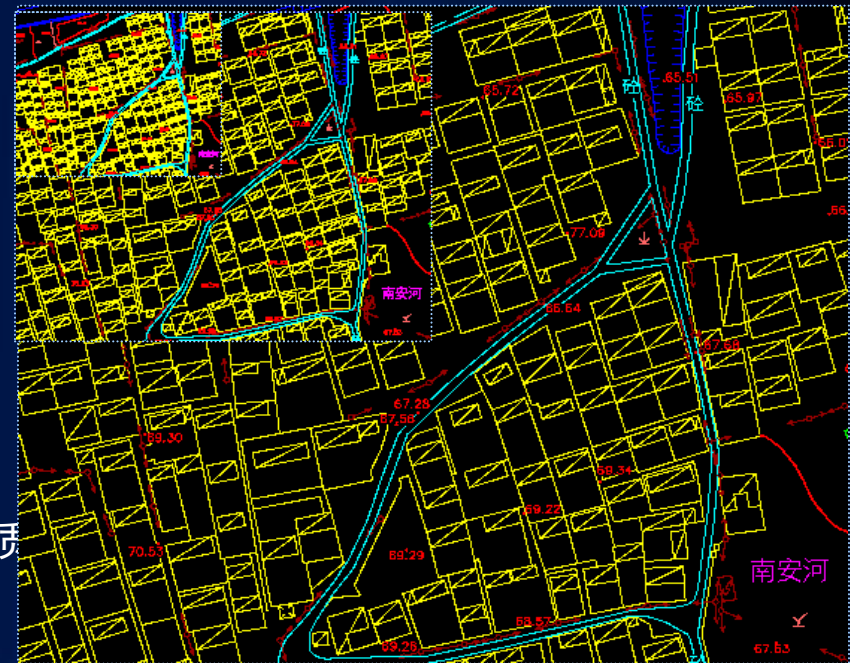
• Quadplex压缩

特点

- 可以支持索引色或黑白图像压缩
- 无损,采用二维数学压缩
- 比GIF,LZW,PNG等要小两倍以上。
- 支持海量数据压缩
- 速度比小波压缩快
 - 6G/hr
- 1:25万DRG, 100x以上

功能

- 内置金字塔, 局部数据实时数据解压
- 前景色提升, 当缩小时, 可以保留线划质量
- 相当于预先生成的矢量显示
- 用户可以选择颜色的优先顺序。
- 透明背景。
- 线划和色块可以象矢量数据一样叠加在其他图上



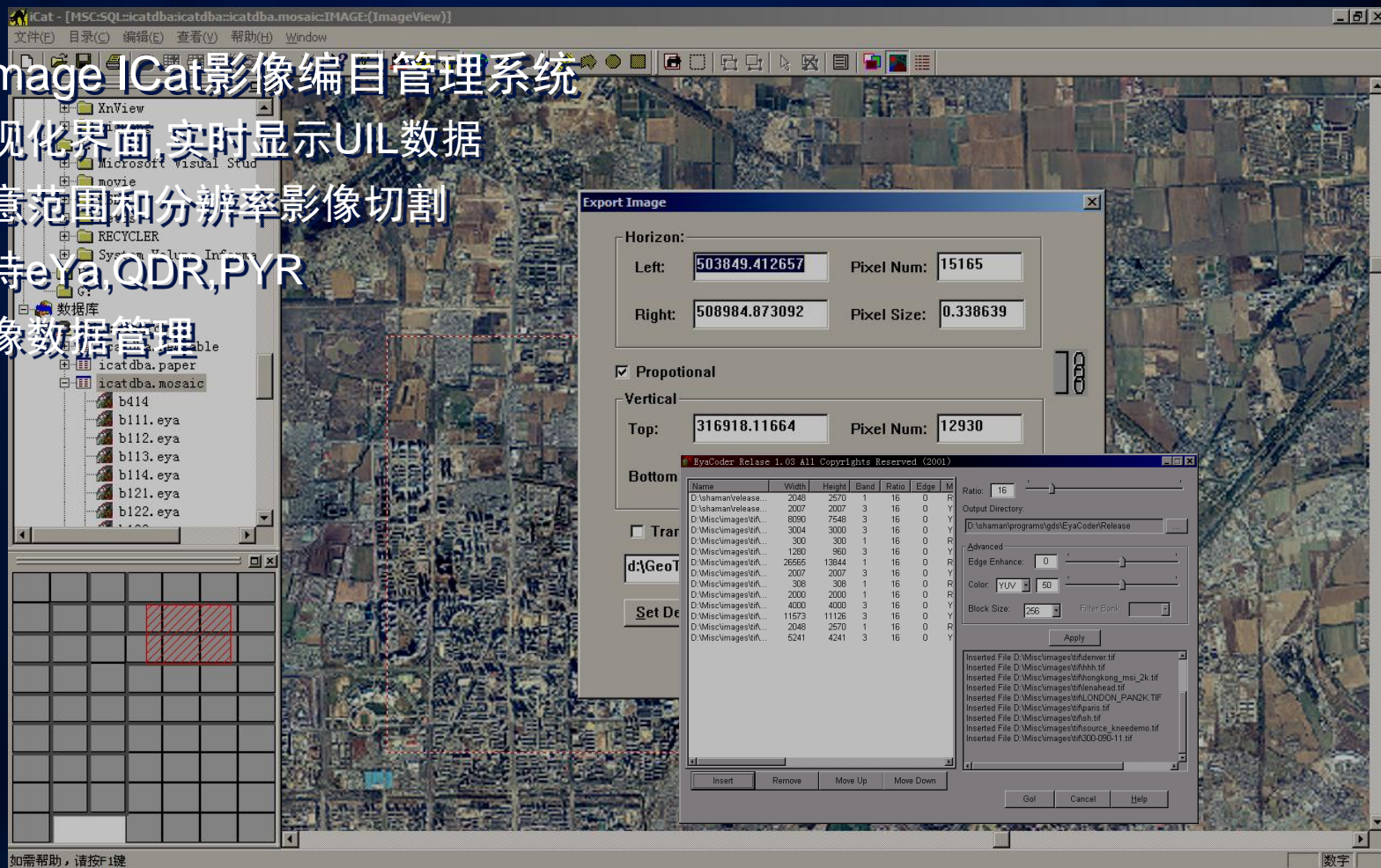
eYalimage影像压缩

- 影像一体化定位(UIL)
 - 统一的格式(文件,数据库,HTTP)
 - 支持不同格式和存储方式影像集成
 - 可以利用数据库的优势
 - 与其他系统集成的桥梁.(*.eil文件)

eYalimage产品

• eYalimage Icat影像编目管理系统

- 可视化界面,实时显示UIL数据
- 任意范围和分辨率影像切割
- 支持eYa,QDR,PYR
- 影像数据管理



如需帮助, 请按F1键

数字

eYalimage产品

- eYalimage Server (3.0.1)

- 平台

Windows Server Fabric, Azure.

集群缓存技术

- 不用切片，数据直接发布。
 - 速度快
 - 适应任意切片Schema
 - 应用迁移代价小（文件拷贝，文件数量少）
 - 通过AGS API (CustomLayer)和已有应用无缝结合

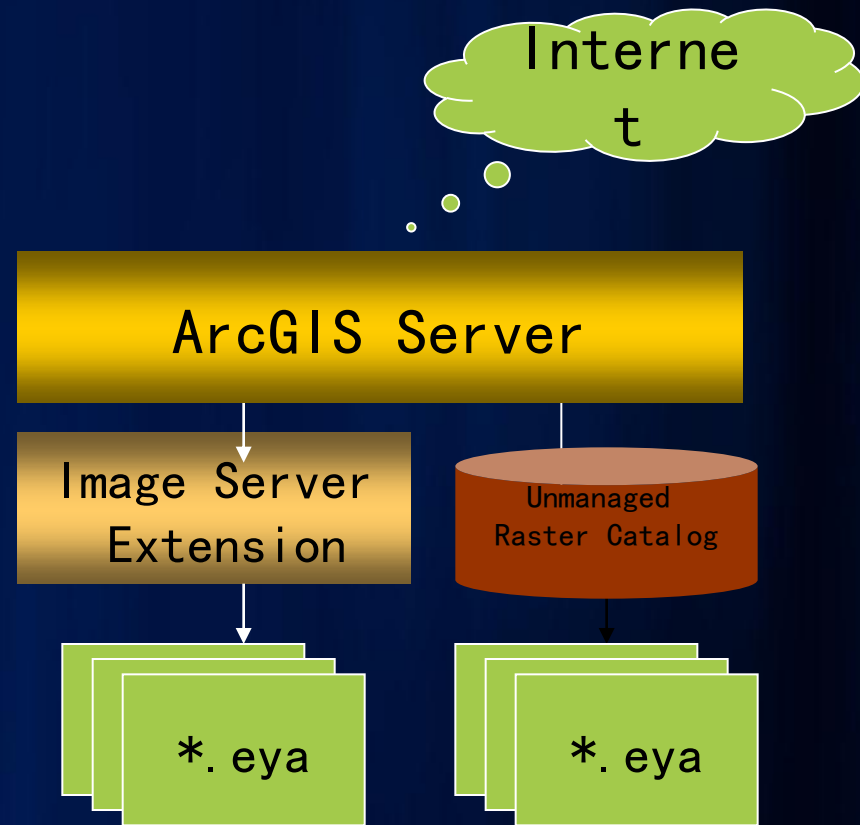


eYalimage产品

- 插件：
 - ArcGIS:
 - Desktop, ArcGIS Server, Engine, ArcIMS
 - Erdas Imagine
 - MapObjects
 - ArcView 3x
 - AutoCAD*
 - IE*
 - ArcPAD*
- eYalimage 3.0.1
 - ArcGIS 10.0
 - AutoCAD 2010

eYalimage和ArcGIS

- 和ArcGIS Server一起使用
 - 更大的Tile
 - 更少的文件
 - 更大缩放范围
 - 更容易更新和备份
 - 不需要切片，每个Tile单独处理



eYalimage和ArcGIS

- eYalimage和ArcGIS Desktop/Engine
 - 直接访问/不需要数据库或服务器
 - 理论上目标文件可以达到32G
 - 实际中每个文件原始数据达到200G
 - 通过UIL可以实时拼接逻辑上无限制

eYalimage On Internet

- 影像服务需求
- Windows Application Server Fabric/Cache Cluster
- 压缩格式逻辑切片服务与实现
- eYalimage Server和ArcGIS Server

影像服务需求

- 影像是地图服务的重要内容
- 比较
 - Mosaic Dataset
 - 支持实时投影、正射、融合等功能，适合C/S下影像管理
 - Web上发布如不切片速度很慢，无法满足发布要求
 - 切片需花费大量时间和存储空间
 - Schema固定
 - eYalimage Image server
 - 需要投影好的数据
 - 不需要切片，Cache On Demand
 - 实时逻辑拼接和镶嵌
 - 高性能Web Cache
 - Schema可以不固定

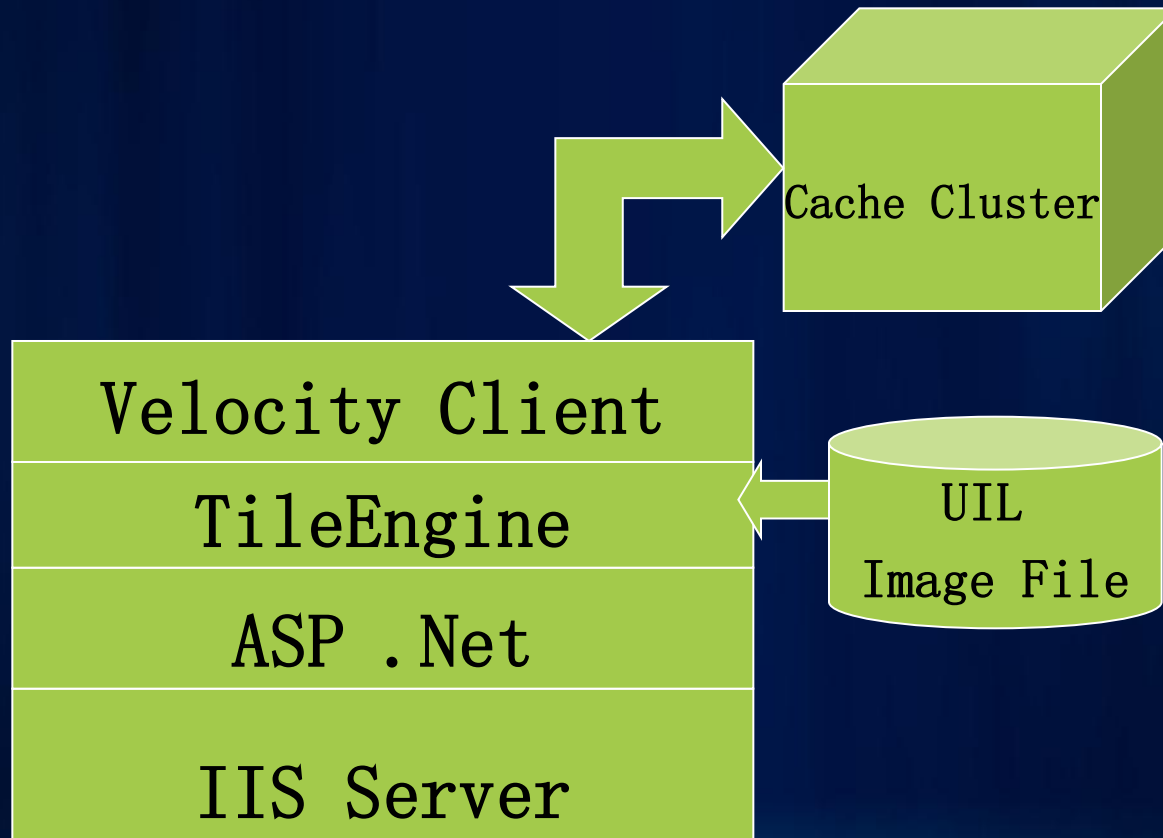
影像服务需求

- 切片花费大量时间
 - eYalmage 100G/8CPU小时
 - ArcGIS Server 100G/(14*8CPU小时)
- 切片可移植性
 - 逻辑 Schema(eYalmage)
 - N个文件
 - 固定的Schema(AGS)
 - 上亿个文件
 - Compact Tile
- 实时Mosaic
 - Mosaic Dataset(AGS)
 - 支持实时投影，和坐标变化
 - 各种格式支持
 - UIL(eYalmage)
 - 不支持投影变换
 - 将支持更多带金字塔的格式

影像服务需求

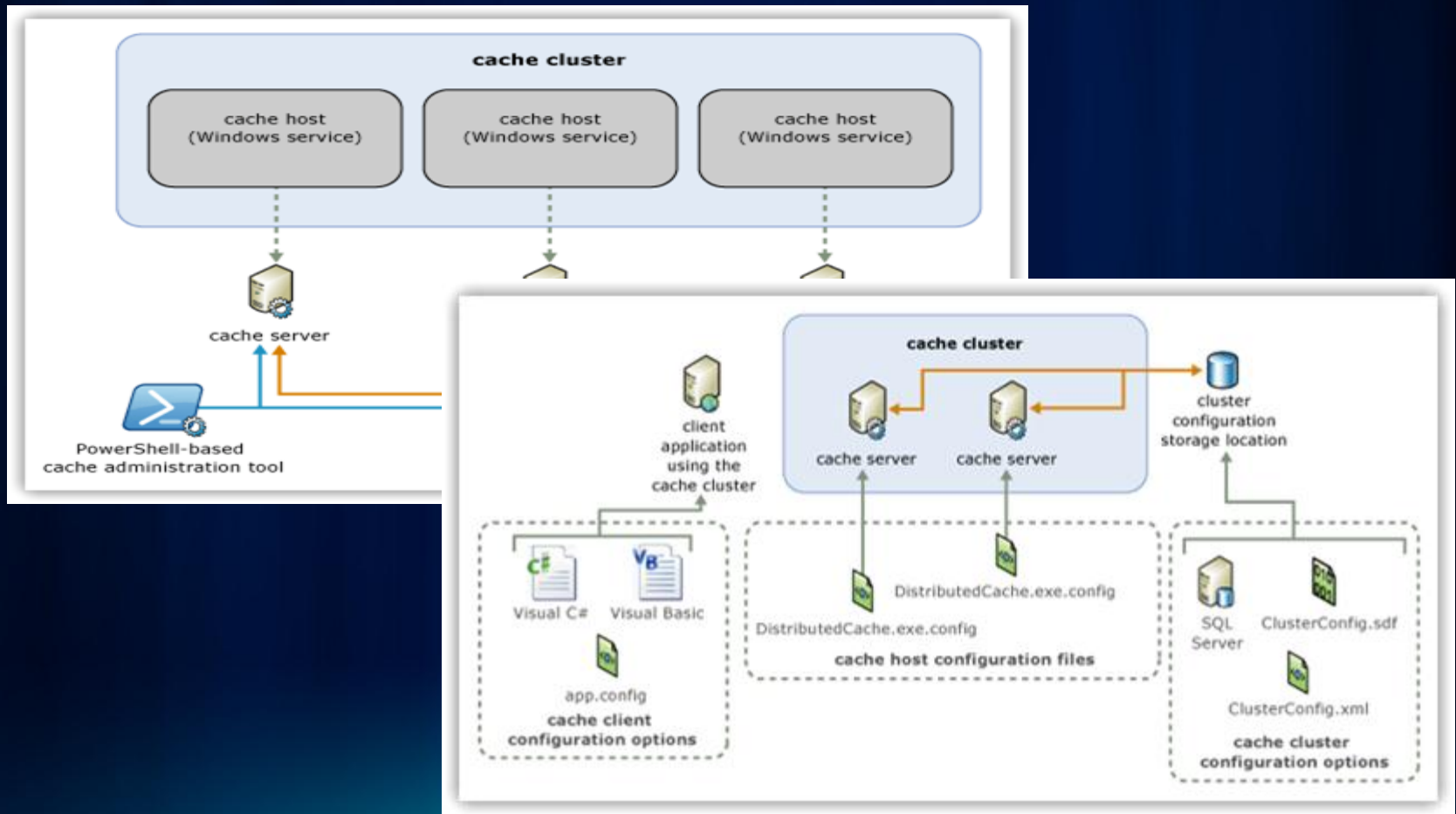
- 大数据量，正射数据需要发布(▶▶▶▶▶)
 - 影像经常需要经常更新(plus ▶▶▶▶▶)
 - 没有专业的数据维护队伍(plus ▶▶▶▶▶)
 - 需要高性能的稳定的服务(plus ▶▶▶▶▶)
 - 已经有了eya格式数据(plus ▶▶▶▶▶)

逻辑架构



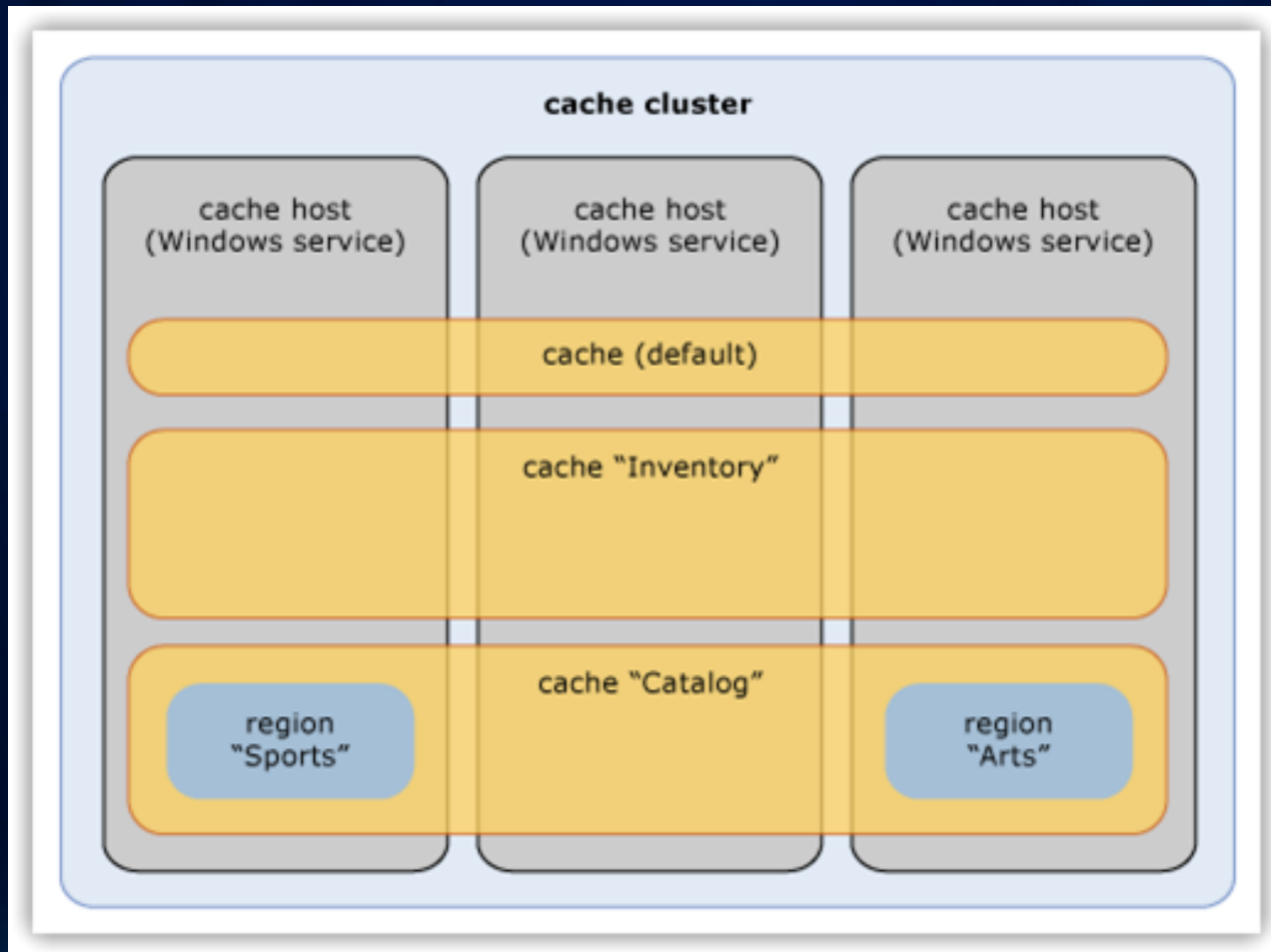
Cache Cluster

- Windows Azure平台
- Windows Application Server Fabric的一部分



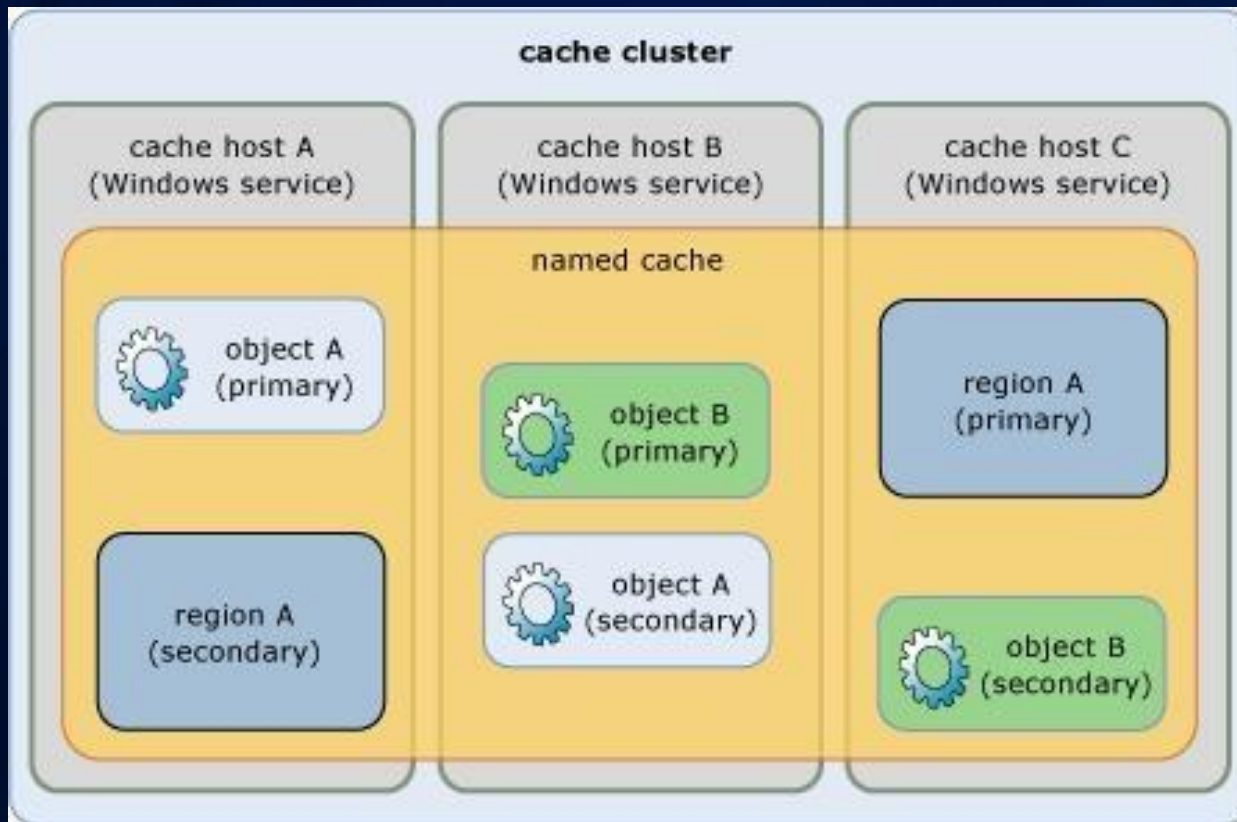
Cache Cluster

- 逻辑模型



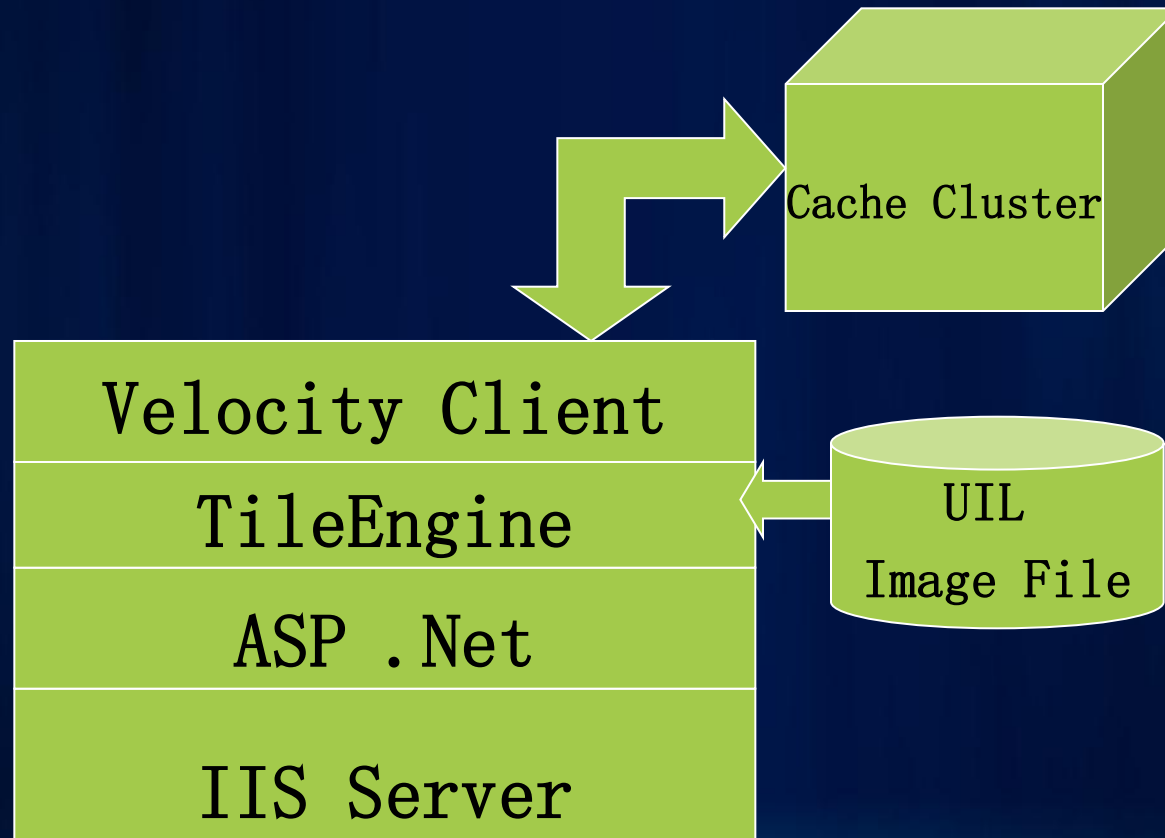
Cache Cluster

- 架构设计上的 HA (High Availability)



若其中一台机器或缓存服务故障，缓存的数据及客户端的 Session 不会遗失

逻辑架构



Eyalmage影像逻辑拼接

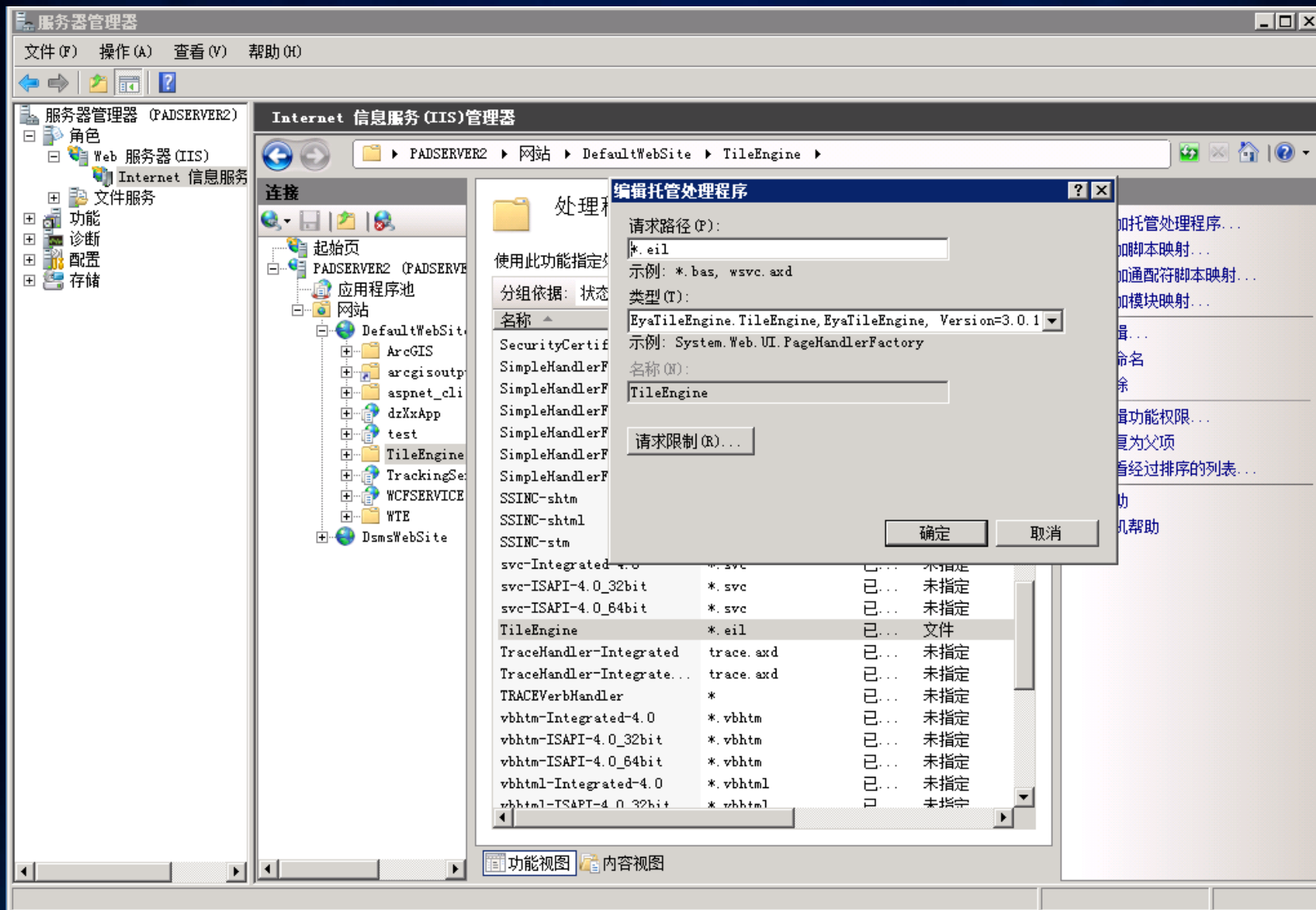
- 多分辨率压缩
- 金字塔内置
- UIL(Unified Image Locater)
 - DIR:G:/CompressedImages
 - ORA:Imageserver:imagebda:imagedba:imagemosaic:image:Resolution>0.5
- ❖ 实时采样，不受影像本身分辨率和尺寸限制

Cache On Demand

- <http://192.168.200.84/eyatileengine/911.eya?getimage&width=512&height=512&pngoption=0>



在IIS中的配置

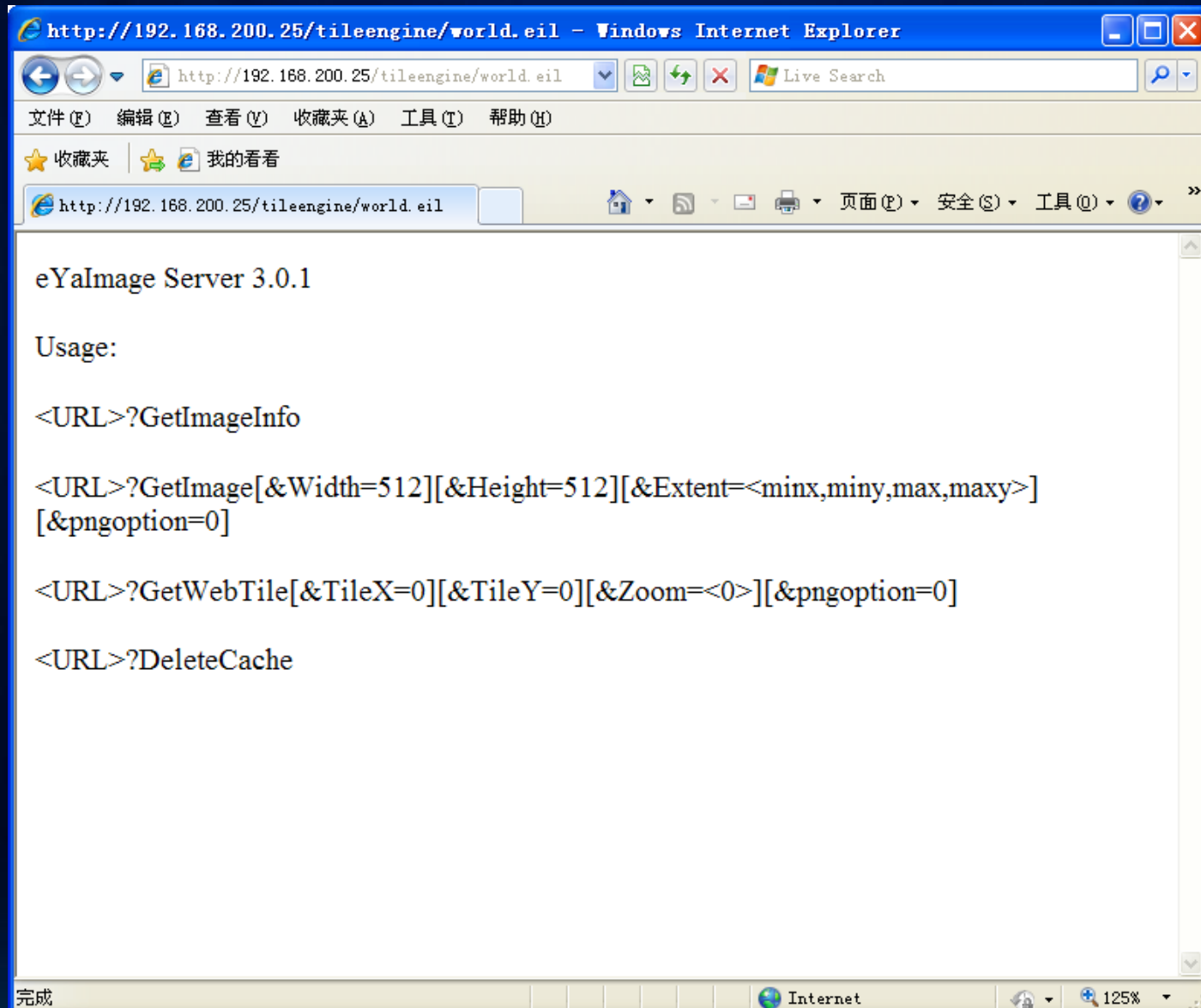


在IIS中的配置

Web.Config文件内容

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <system.webServer>
    <handlers accessPolicy="Read, Execute, Script">
      <add name="TileEngine" path="*.eil" verb="GET"
        type="EyaTileEngine.TileEngine,EyaTileEngine, Version=3.0.1.0, Culture=neutral,
        PublicKeyToken=0fff7cbddeb4f9c3" resourceType="File" requireAccess="Execute"
        preCondition="integratedMode" />
    </handlers>
    <directoryBrowse enabled="true" />
  </system.webServer>
</configuration>
```


Usage



Web Sessions

#	Result	Protocol	Host
1	200	HTTP	www.fiddler2.com
2	200	HTTP	192.168.200.25
3	404	HTTP	192.168.200.25
4	200	HTTP	192.168.200.25

Statistics

Inspectors

AutoResponder

Request Builder

Log

Filters

Timeline

Headers

TextView

WebForms

HexView

Auth

Raw

XML

Request Headers

[\[Raw\]](#) [\[Header Definitions\]](#)

GET /tileengine/world.eil?getimage HTTP/1.1

Client

User-Agent: Fiddler

Transport

Host: 192.168.200.25

Transformer

Headers

TextView

ImageView

HexView

WebView

Auth

Caching

Privacy

Raw

XML

Response Headers

[\[Raw\]](#) [\[Header Definitions\]](#)

HTTP/1.1 200 OK

Cache

Cache-Control: public,max-age=7200

Date: Tue, 01 Mar 2011 09:24:37 GMT

Entity

Content-Length: 85575

Content-Type: image/png

Last-Modified: Tue, 01 Mar 2011 09:24:36 GMT

Miscellaneous

Server: Microsoft-IIS/7.5

X-AspNet-Version: 2.0.50727

X-Powered-By: ASP.NET

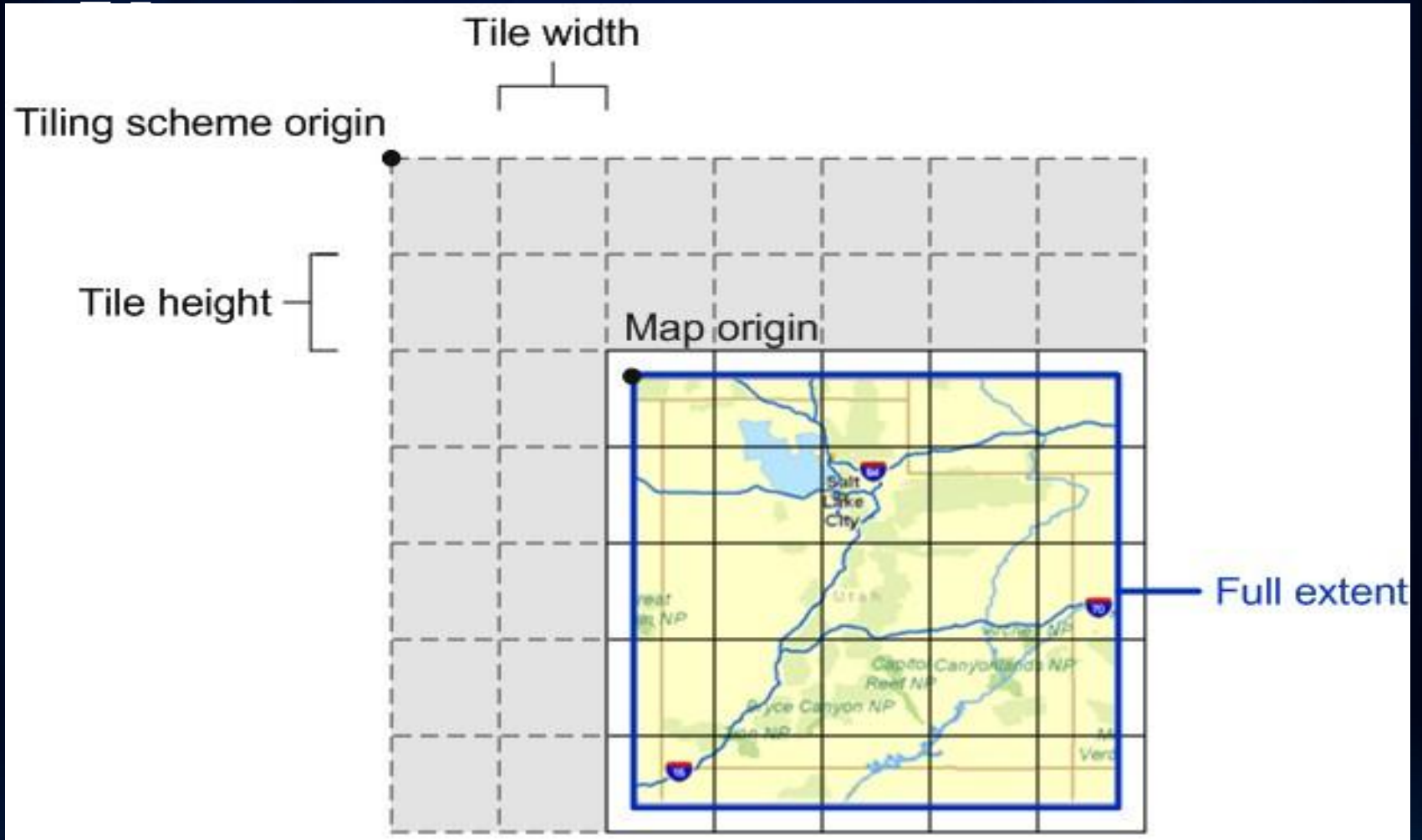
eYalimage Server与ArcGIS Server的结合

*利用ArcGIS Web API构架, 定义CustomLayer
即插即用*

- ArcGIS Server切图原理介绍
- 基于ArcGIS Server Flex的Tile图层实现

eYalimage Server与ArcGIS Server的结合

- ArcGIS Server Tile Schema



eYalimage Server与ArcGIS Server的结合

- ArcGIS Server 切图文件组织

- 切片文件夹名为Map Service名

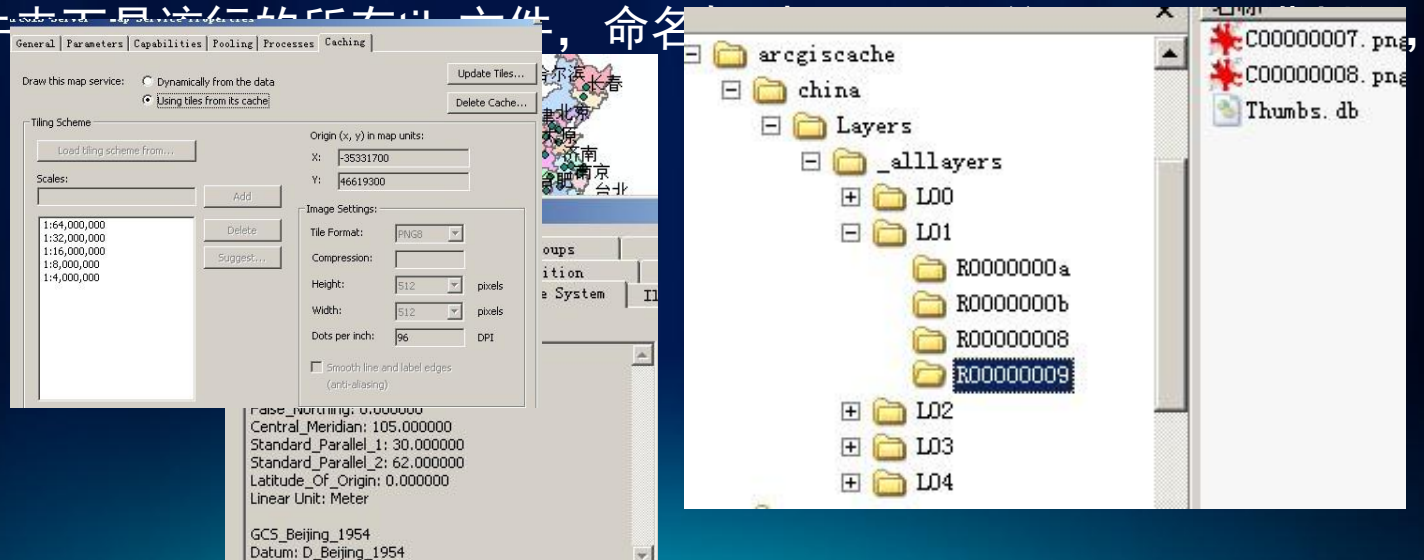
- 地图的DataFrame名: Layers

- Fused方式: _alllayers

- 比例尺级别: L01、L02、*****

- 比例尺文件夹下, 是切图的“行”文件夹, 命名规则是R加上8位行号(16进制), 不足补0。如图中的R0000000a, 表示此比例尺中第10行(16进制中的a)

- 每行文件夹下是这行的所有切图文件, 命名规则是C加上8位列号(16进制), 不足补0



eYalimage Server与ArcGIS Server的结合

- 计算某比例尺下地图某点所在的tile图片行列号

如计算L01中，乌鲁木齐市所在tile的行列号。需要收集三个信息：

- 1、获得乌市的地理坐标：在本地图中是 $x=-1341070$ ， $y=5343697$ ；
- 2、获得tiling scheme： $x=-35331700$ ， $y=46619300$ ；
- 3、获得当前比例尺的resolution，即一个像素所占的地图单位长度：在L01比例尺上是8466.68360003387。

乌市所在的行号： $(35331700-1341070)/(8466.6836*512)=7.84=8$

乌市所在的列号： $(46619300-5343697)/(8466.6836*512)=9.52=10$

所以乌鲁木齐在切图的第二个比例尺中，处于第10行，第8列的tile。

eYalimage Server与ArcGIS Server的结合

- 建立eYalimage Server和ArcGIS Server的对应关系

- 首先新建一个自定义的类

- EyaTiledMapServiceLayer, 这个类必须继承TiledMapServiceLayer类

- 修改重载属性和函数getTileURL

- 这里的关键是通过重载函数getTileURL去获得切片图, 这个函数有3个参数: 缩放级别、行号和列号。根据这3个参数, 我们可以计算出位于row行和col列的图片的范围。

- 算法

- $xmin = originPointX + (col-1)*resolution*width;$

- $ymin = originPointY - (row-1)*resolution*height;$

- $xmax = originPointX + col*resolution*width;$

- $ymax = originPointY - row*resolution*height;$

- 其中, originPointX和originPointY为Tile Schema的原点坐标。



Eis.wrf

演示



谢谢