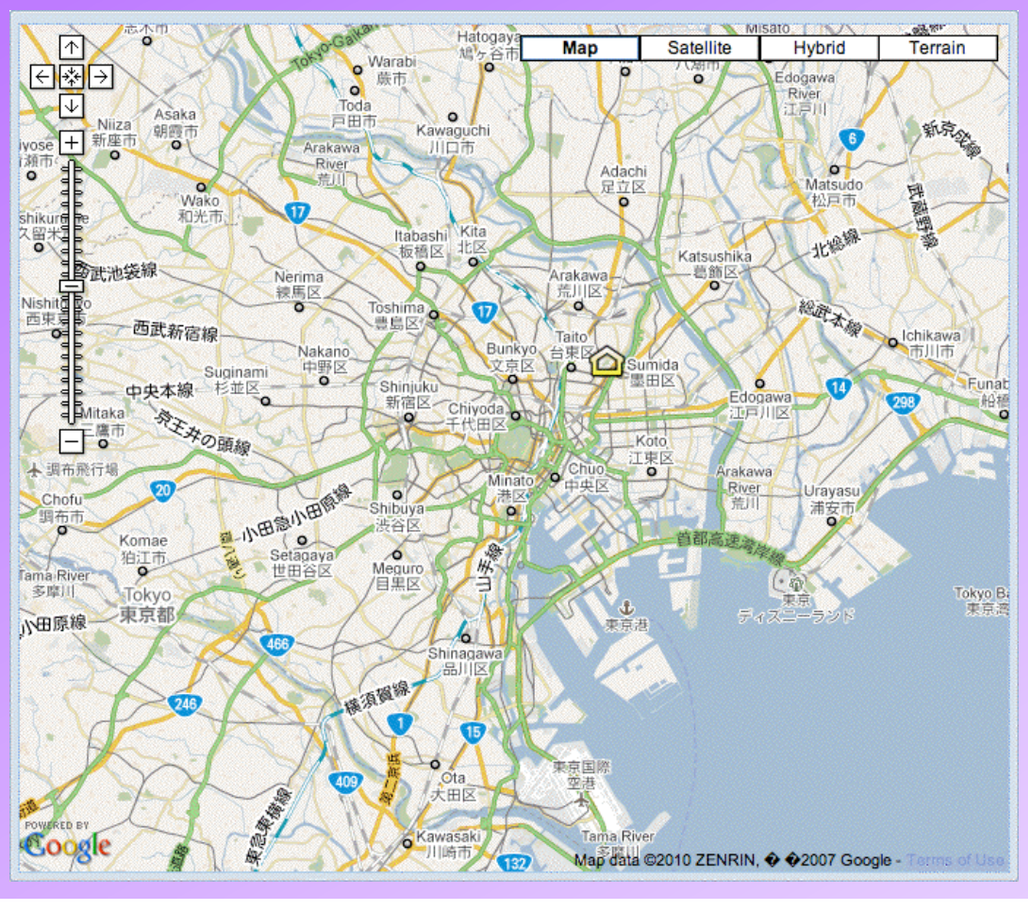
GTiler: *A Python Script to Generate Google Tiles From a Georeferenced Image*



Andy Anderson

Academic Technology Services
Amherst College
Amherst, Massachusetts

Simplifying Geographic Discovery

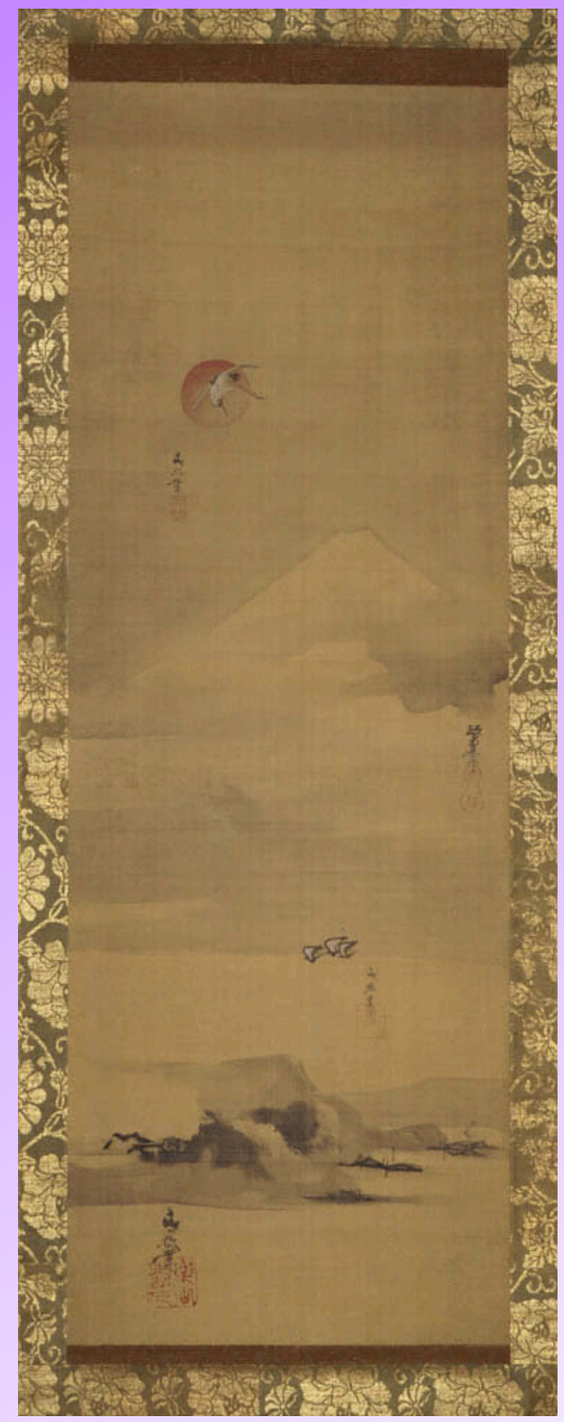
- Many professors understand the value of thinking geographically.
- Many tools for geographic discovery aren't as time-consuming as full GIS.
- **But:** what can we provide students that not only lets them explore but also lets them easily build integrated geographic multimedia presentations?

Encyclopedia of Edo-Meiji-Tokyo Digital Map or *Superimposed Map* (Epipi Company, 2004)

Reinventing Tokyo

- We considered many possibilities for a class taught in Fall 2009:

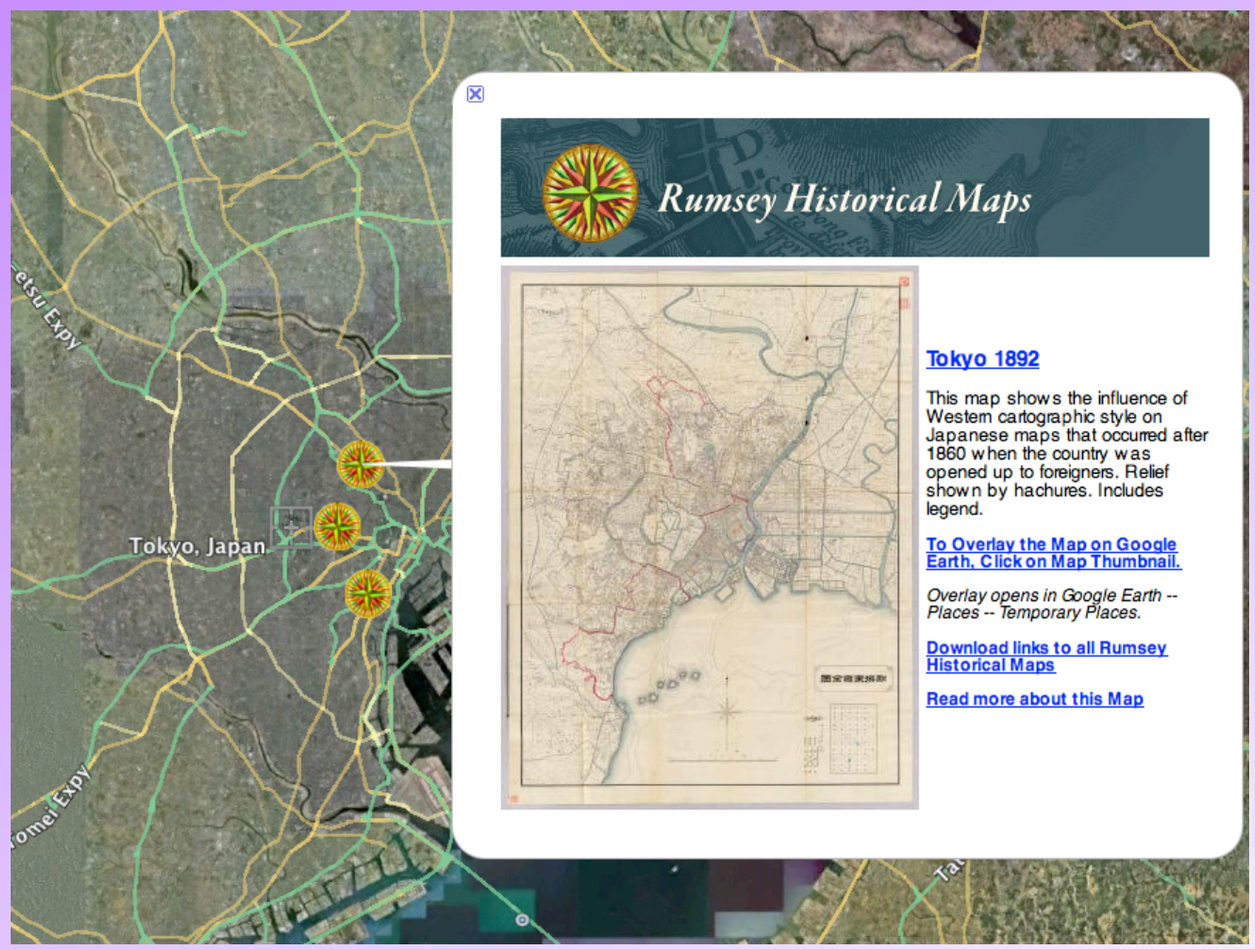
  Reinventing Tokyo: The Art, Literature, and Politics of Japan's Modern Capital
  "Tokyo is the political, cultural, and economic center of Japan, the largest urban conglomeration on the planet, holding 35 million people…. Since its founding 400 years ago, when a small fishing village became Edo, the castle headquarters of the Tokugawa shoguns, the city has been reinvented multiple times…"



Fishing Village, Flying Crane, and Mount Fuji

Google Tokyo

- Google Earth's Rumsey Historical Maps would make it a valuable tool.
- But creating presentations requires knowledge of HTML!
- Also not very "Web 2.0"
- Solution: Build our own platform with Google Maps

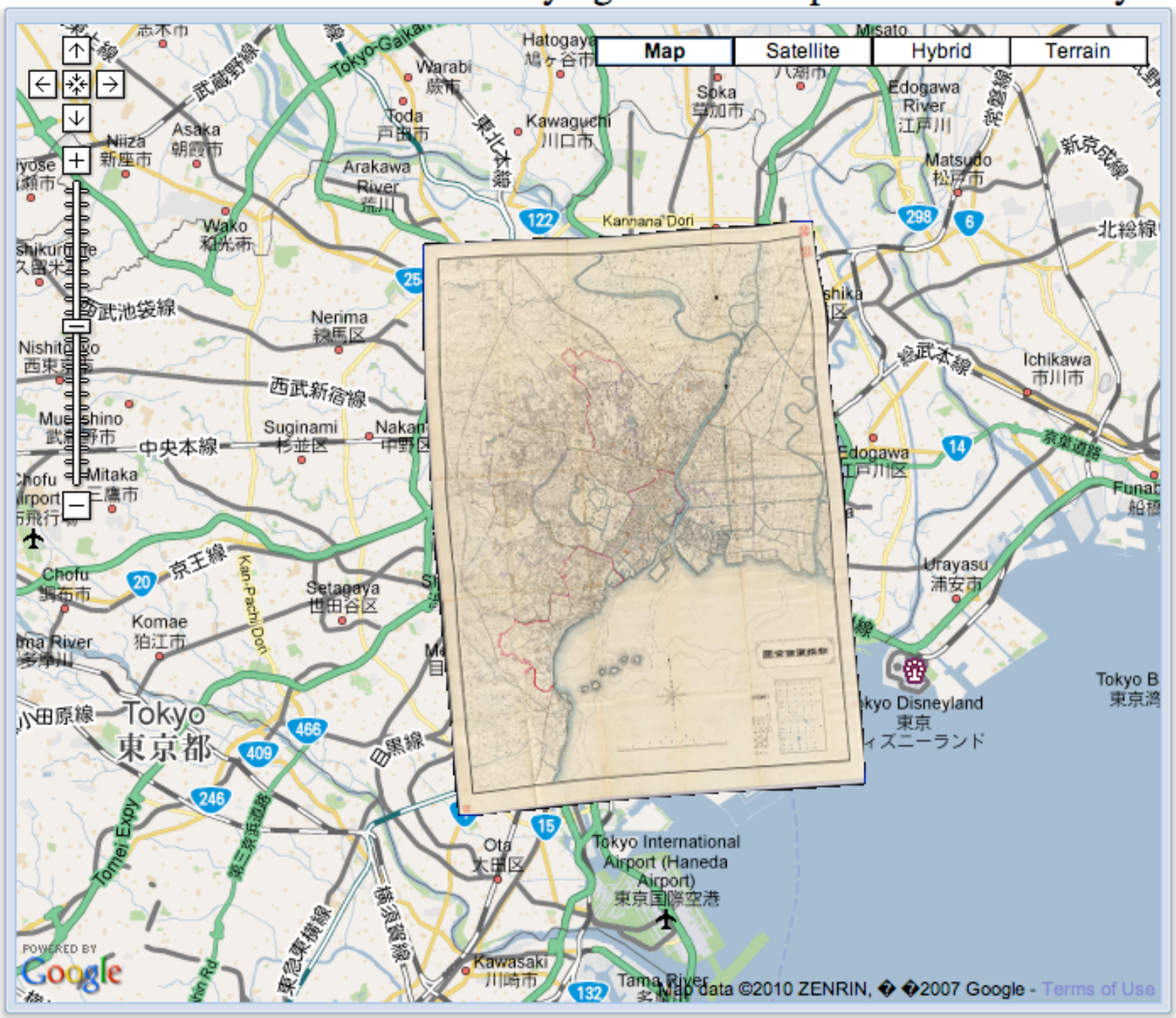


Cityscapes Tokyo

http://ats.amherst.edu/tokyodemo

Tokyo Demo Site:

Demonstration of Overlaying Historic Maps on a Modern City Using MapApp



Drag icon to map location

Map Size

Enlarge Map Reduce Map

Opacity Controls

1680:

1799:

1858:

1892:

1945:

Groups

Go To

Enter an Address or Location Name

Go

Cityscapes Paris

http://ats.amherst.edu/parisdemo

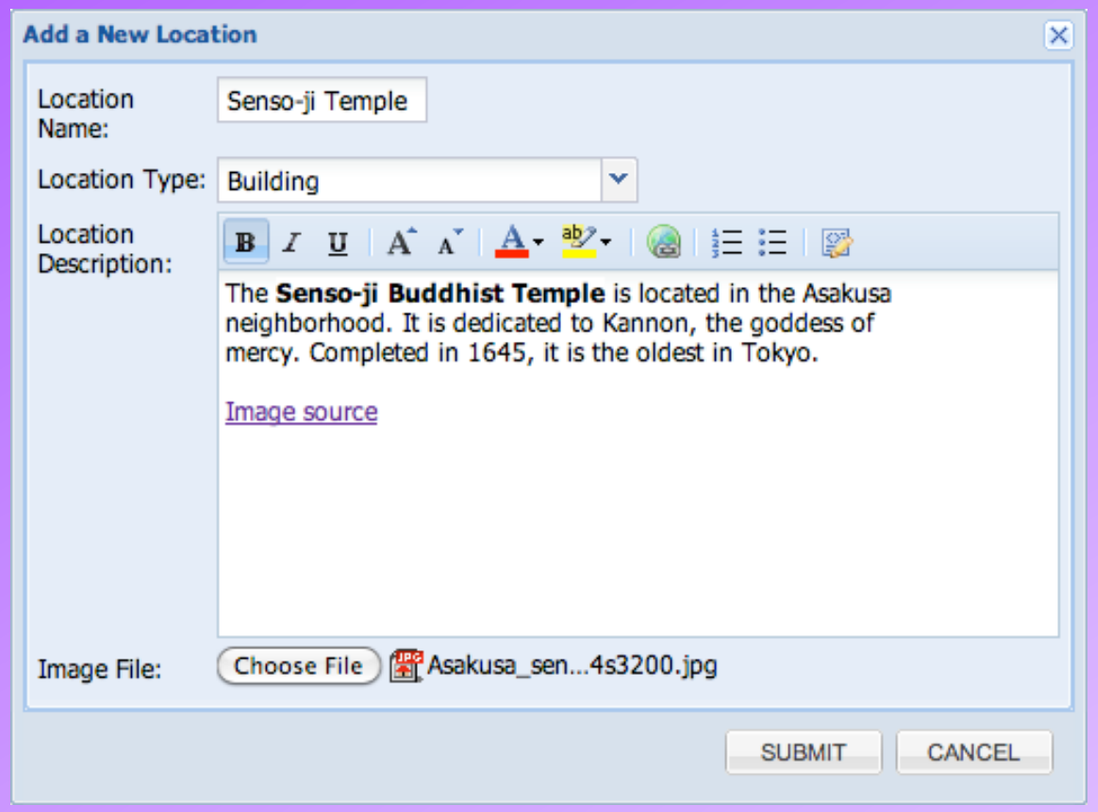
GTiler: A Python Script to Generate Google Tiles From a Georeferenced Raster

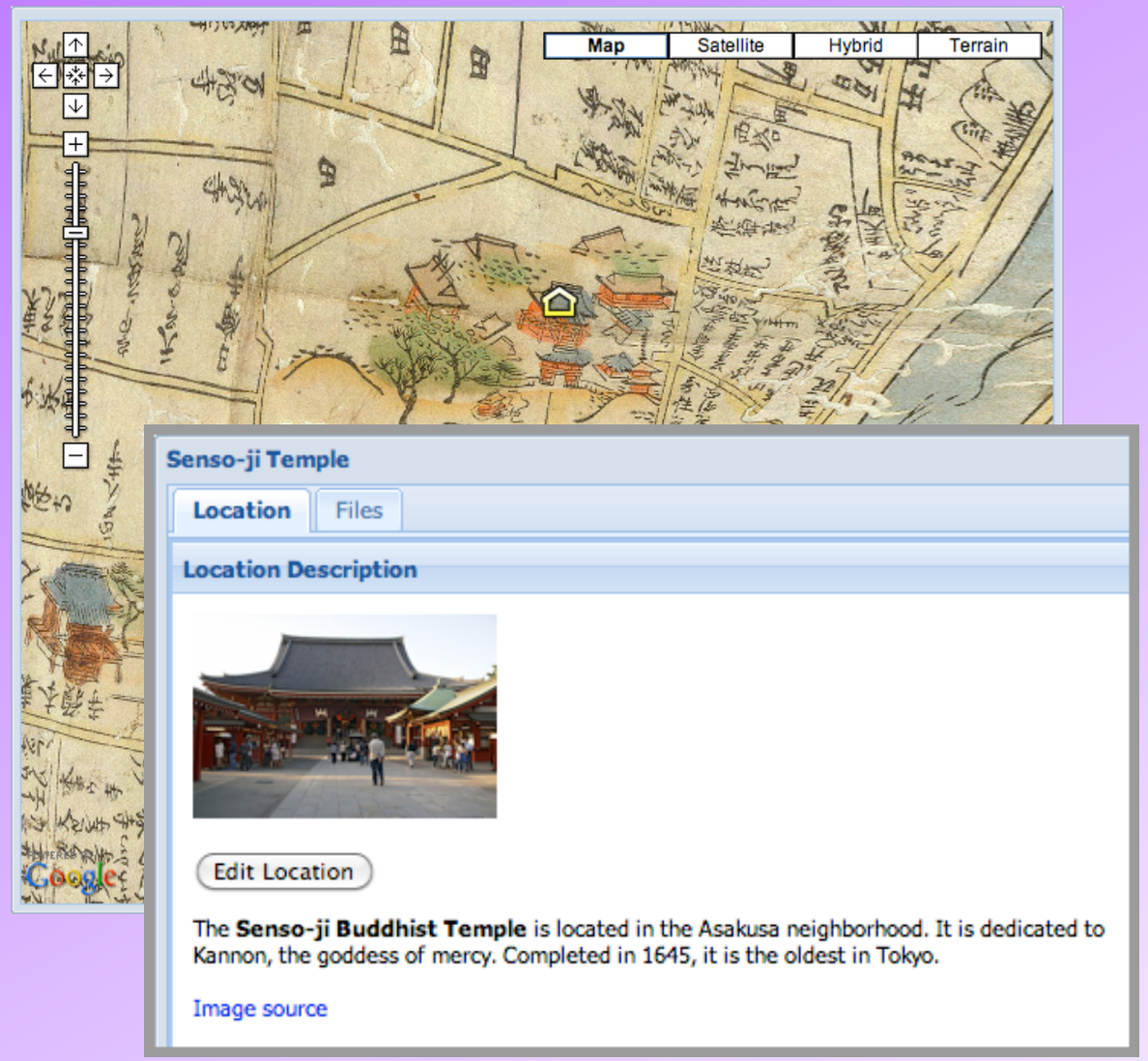
Cityscapes Features

- ⊕ zooming
- ⊕ layering
- ⊕ transparency
- ⊕ fully web-based
- ⊕ Google-provided map service with vector features, satellite imagery, and terrain
- ⊕ and more!

Cityscapes Feature: Content Balloons

⊕ content balloons provide structured media inclusion



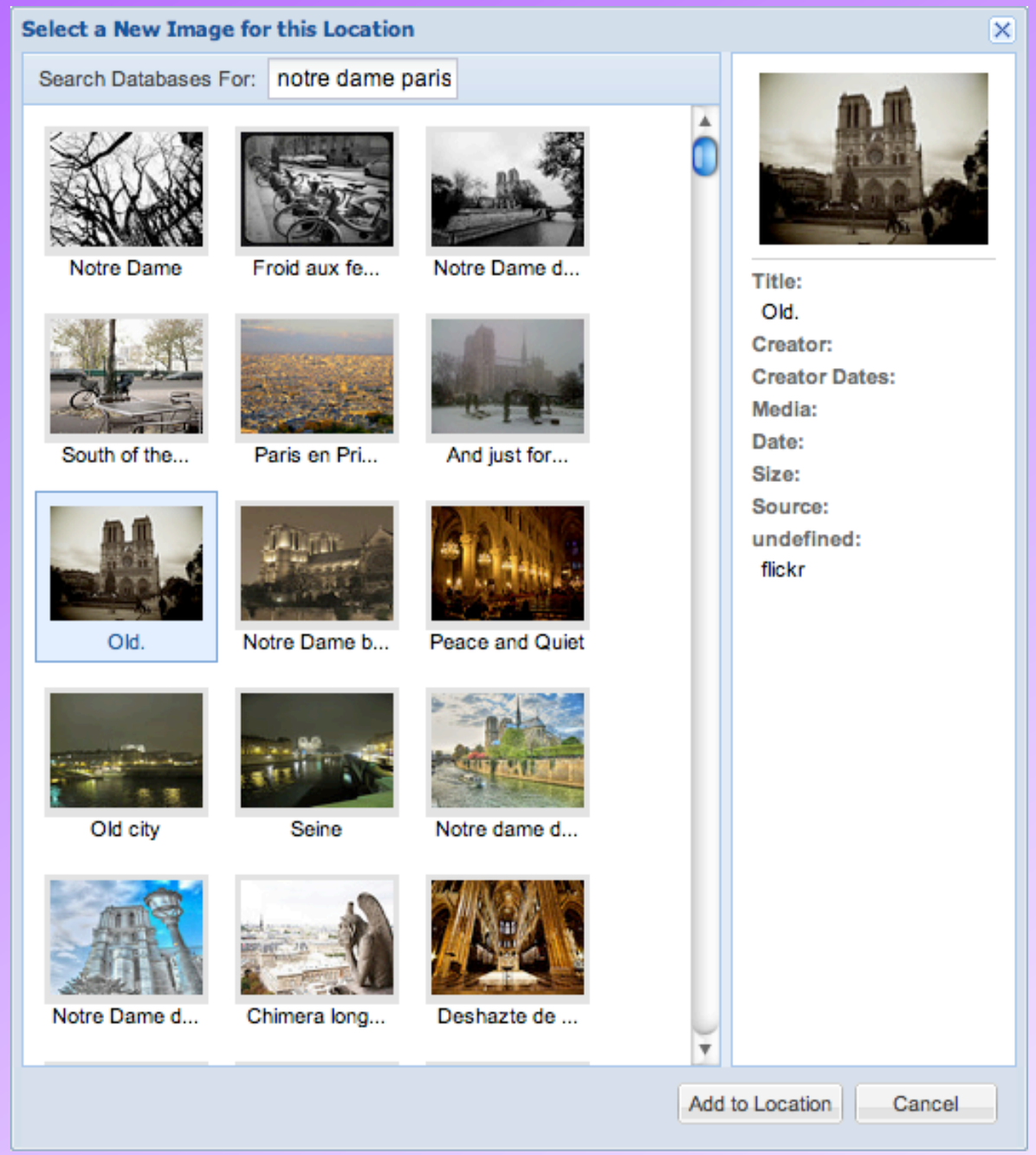


⊕ dialogues don't require knowledge of HTML

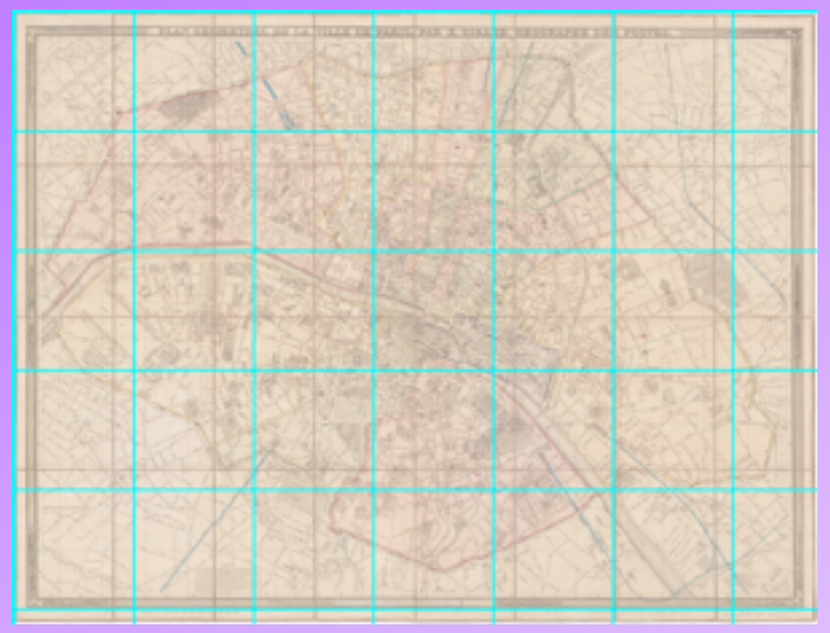
⊕ created using the Ext JS "Rich Internet Application Framework"

Other Features

- ⊕ Balloons can have blogs.
- ⊕ Balloons can reference images and display them in a dedicated viewer.
- ⊕ Images can be pulled directly from several different collections:
  - Luna (Smith College)
  - DigiTool (Amherst College)
  - Flickr (provided by students and others)
- ⊕ Built-in access control and groups for classes.



Map Image Processing

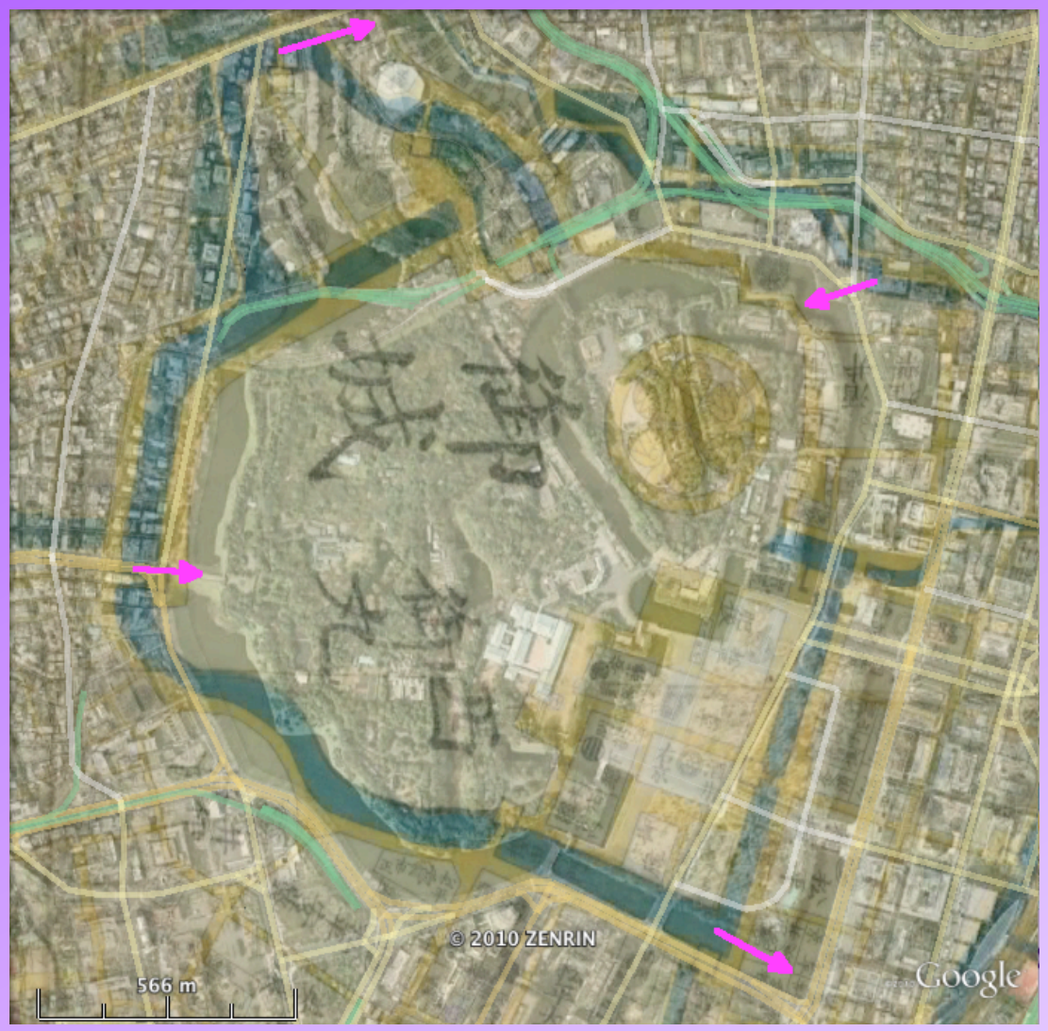


- Map images were downloaded at the highest available resolution, usually on the order of 10K pixels^2, with a scale of 1-2 meters per pixel.
- They were georeferenced using ArcMap, with the projection **WGS84 Web Mercator (Auxiliary Sphere)**, corresponding to Google Maps. *Don't* use **Google Mercator** (it's an inconsistent datum)!
- As a background, the ArcGIS Map Service **World Street Map** had great performance.

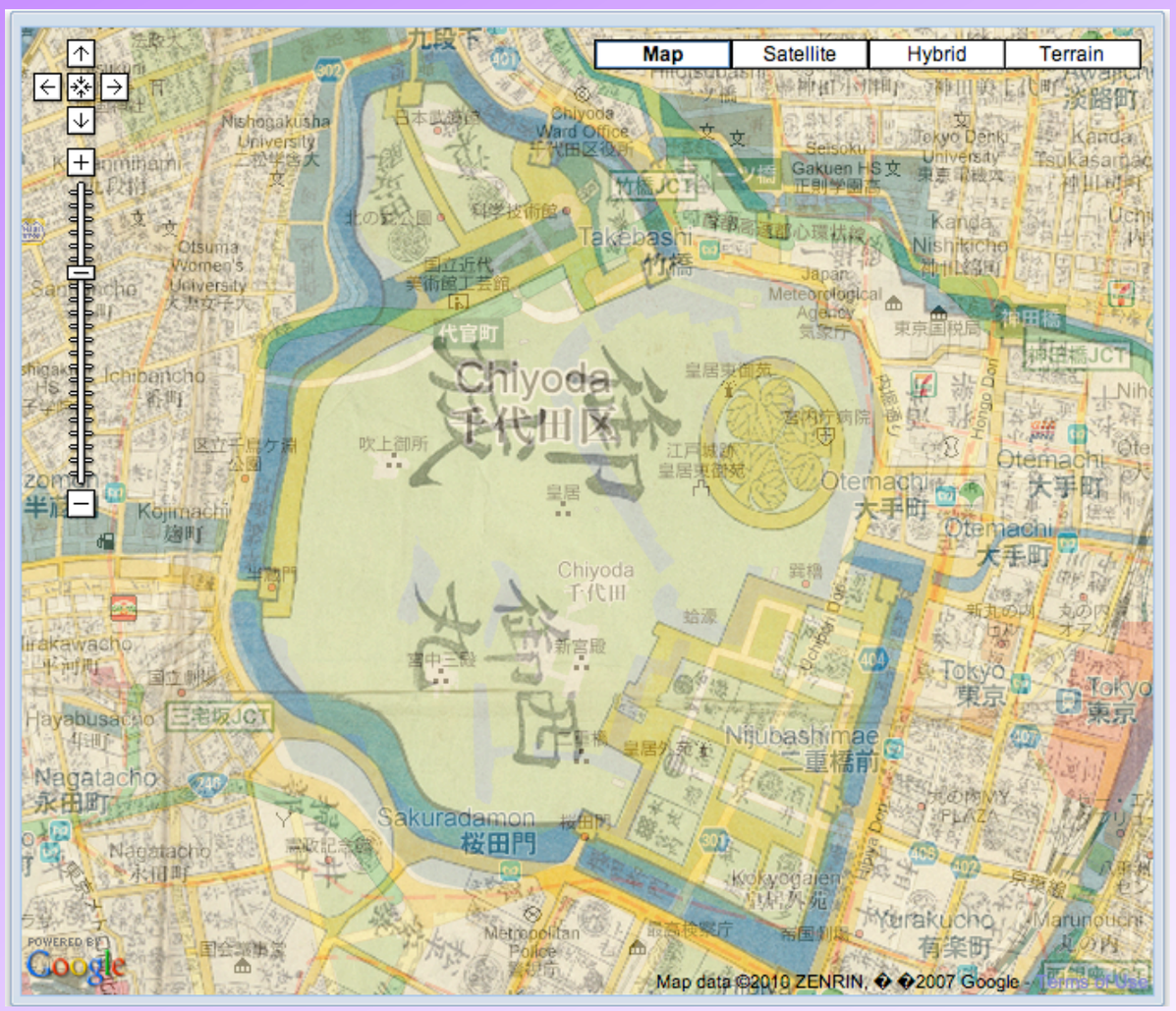
Cityscapes Feature: Georeferencing

⊕ Cityscapes historical maps are closely georeferenced:

Google Earth



Cityscapes



Google Tiles

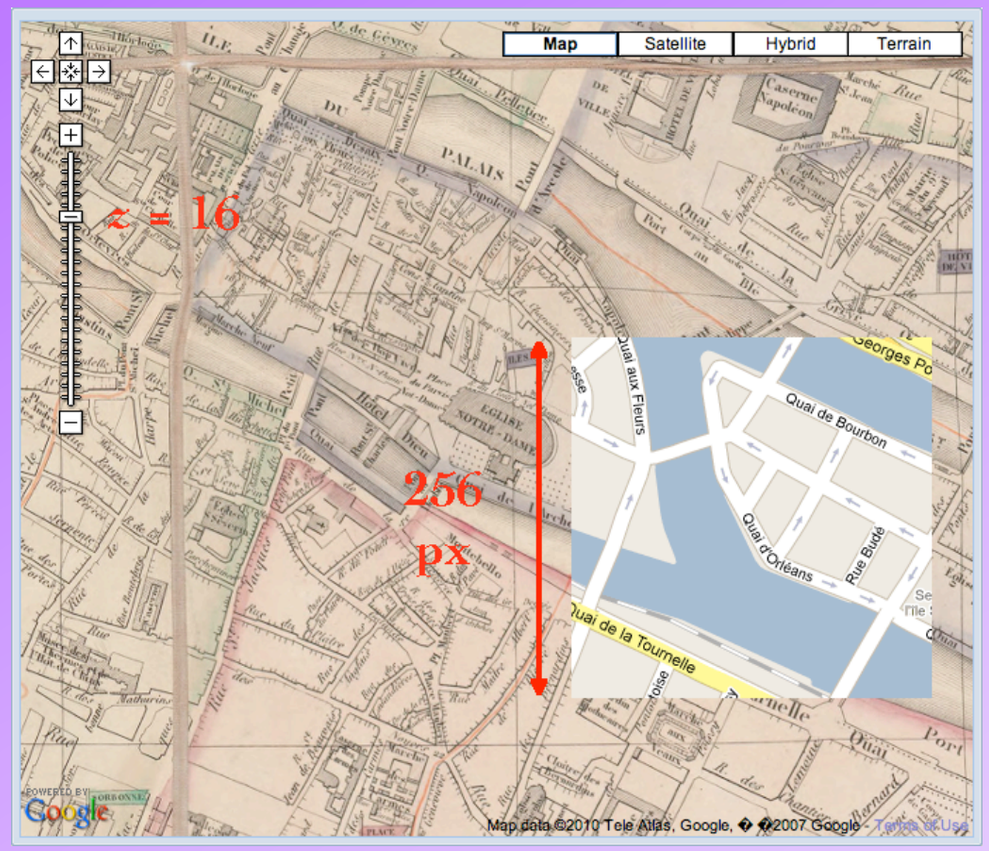
- Google tiles are square and $256 = 2^8$ pixels across.
- The Mercator pixel size is:

$$\text{cellSize}(z) = 2\pi r / 2^{z+8}$$

where $r = 6378137$ m is the equatorial radius, and z is the Google zoom level.

  - By definition, $z = 0$ produces a single tile covering the Earth exclusive of the polar regions (a distance of $2\pi r$).
  - Incrementing the zoom level doubles the number of tiles.

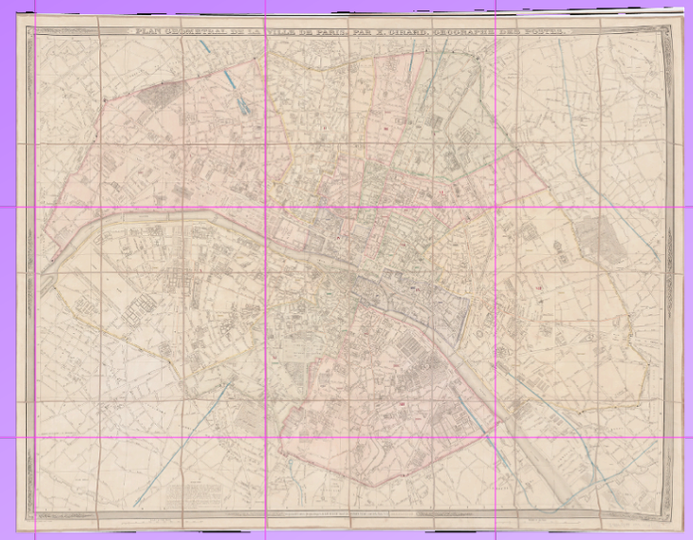
- The zoom levels for large cities like Tokyo or Paris range from about 17 (1.2 m/px) to 7 (1.2 Km/px).

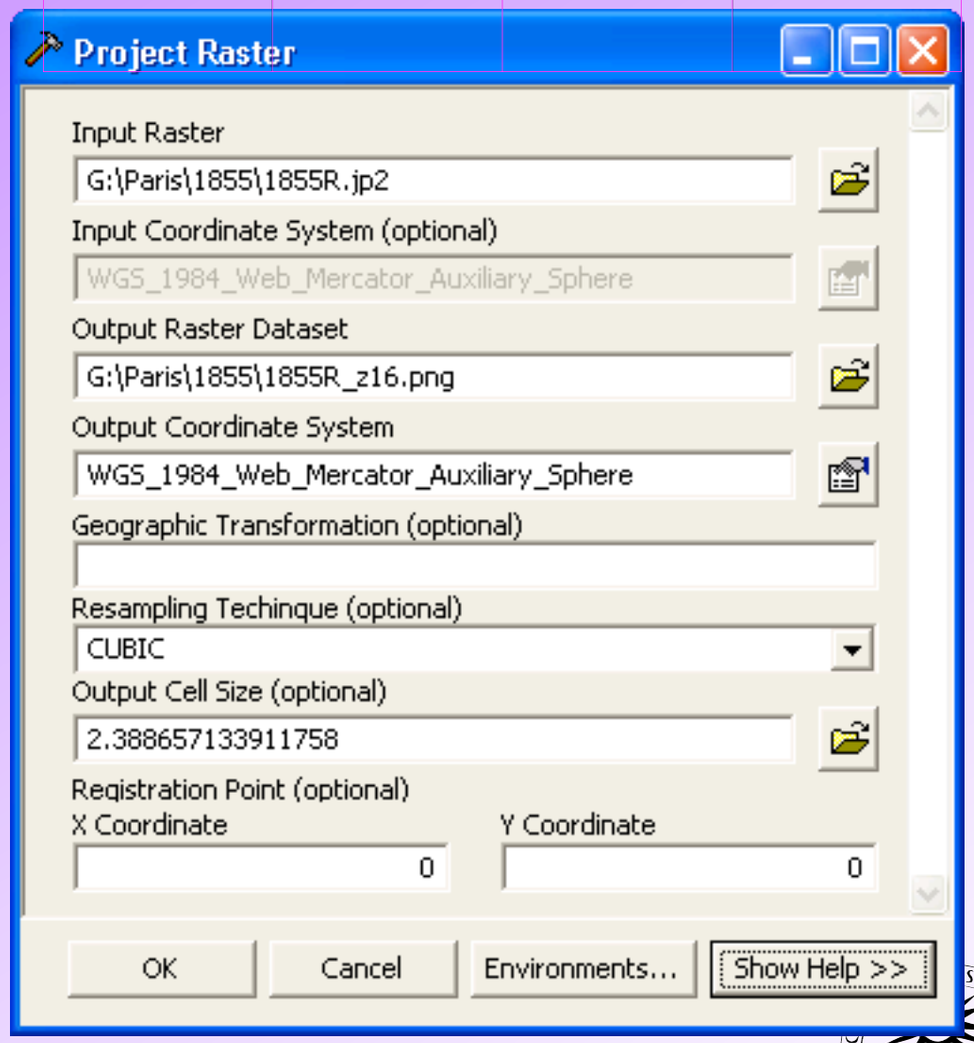


Rescaling to Tile Space

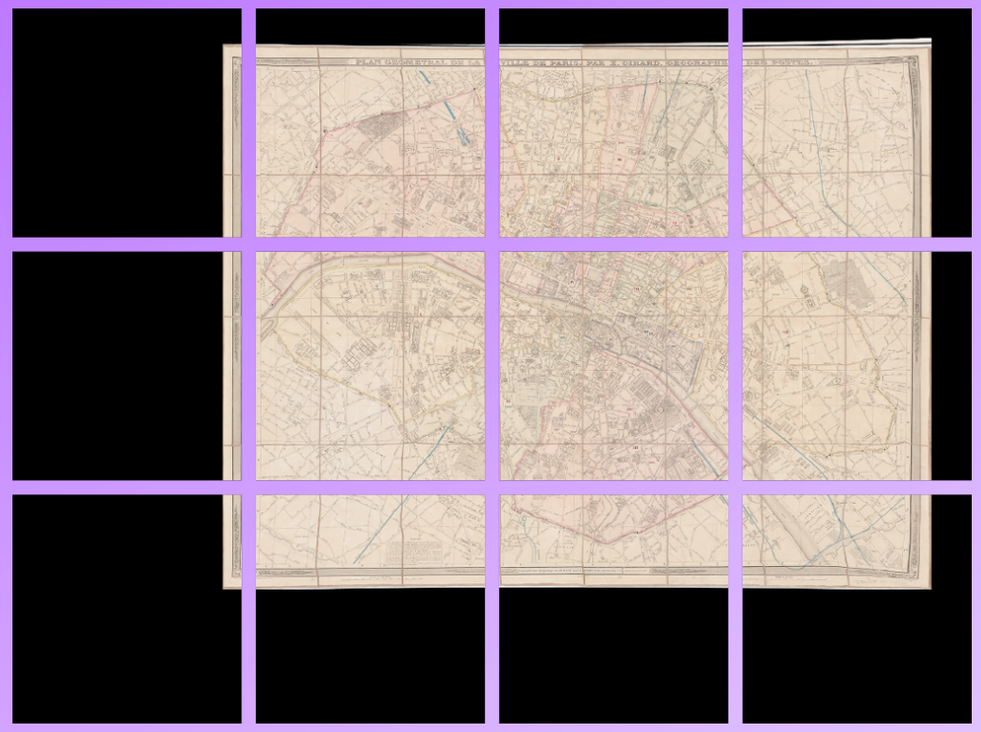
- A scanned, rectified map be rescaled to the appropriate cell size using:
  ArcToolbox >
  Data Management >
  Projections and Transformations >
  Raster > Project Raster

- Project Raster can force a raster to a Google pixel boundary by specifying a registration point of (0, 0) — unlike Resample.





Tile Coverage



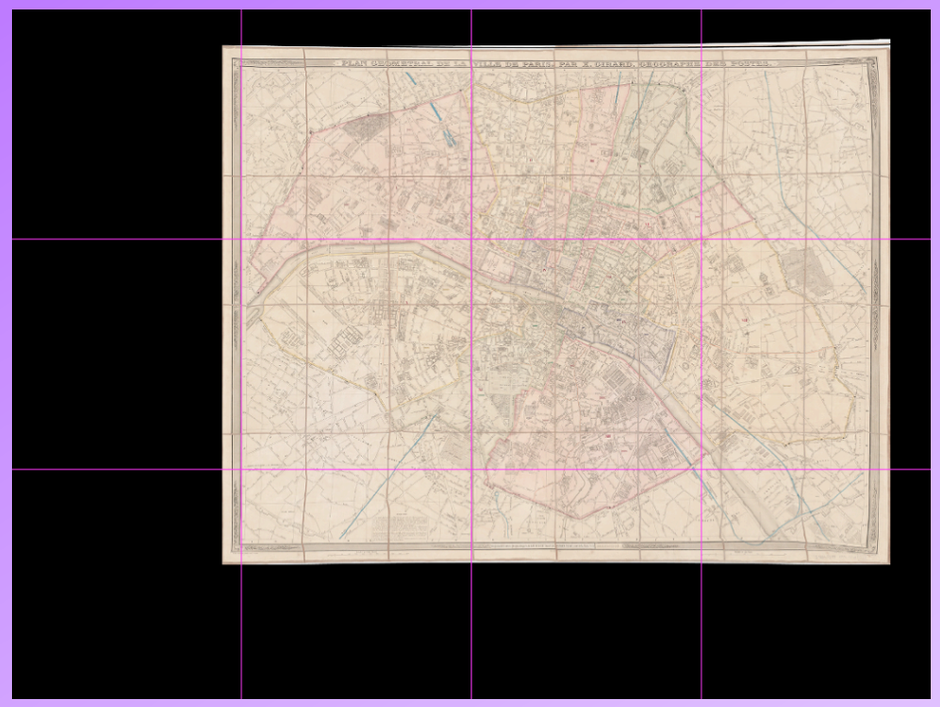
- Rescaled images typically overlap many tiles, but lie completely within one tile at the smallest scales.
- For a given zoom level, Google tiles are indexed from 0 to $2^z - 1$, west to east and *north to south*.
- For a given spherical Mercator position (lx, ly), the Google tile indices are given by:

$$tx(lx, z) = \text{floor}(2^{z-1} (1 + lx / (\pi r)))$$

$$ty(ly, z) = \text{floor}(2^{z-1} (1 - ly / (\pi r)))$$

- So, the image extent yields the tile coverage.

Extending the Image



- An image must be padded with NoData around its edge to fill it out to the extent of the covering tiles.
- The Mercator coordinates of the northwest corner of a tile (tx, ty) are given by the inverse formulae:

$$lx(tx, z) = \pi r \left(tx / 2^{z-1} - 1 \right)$$

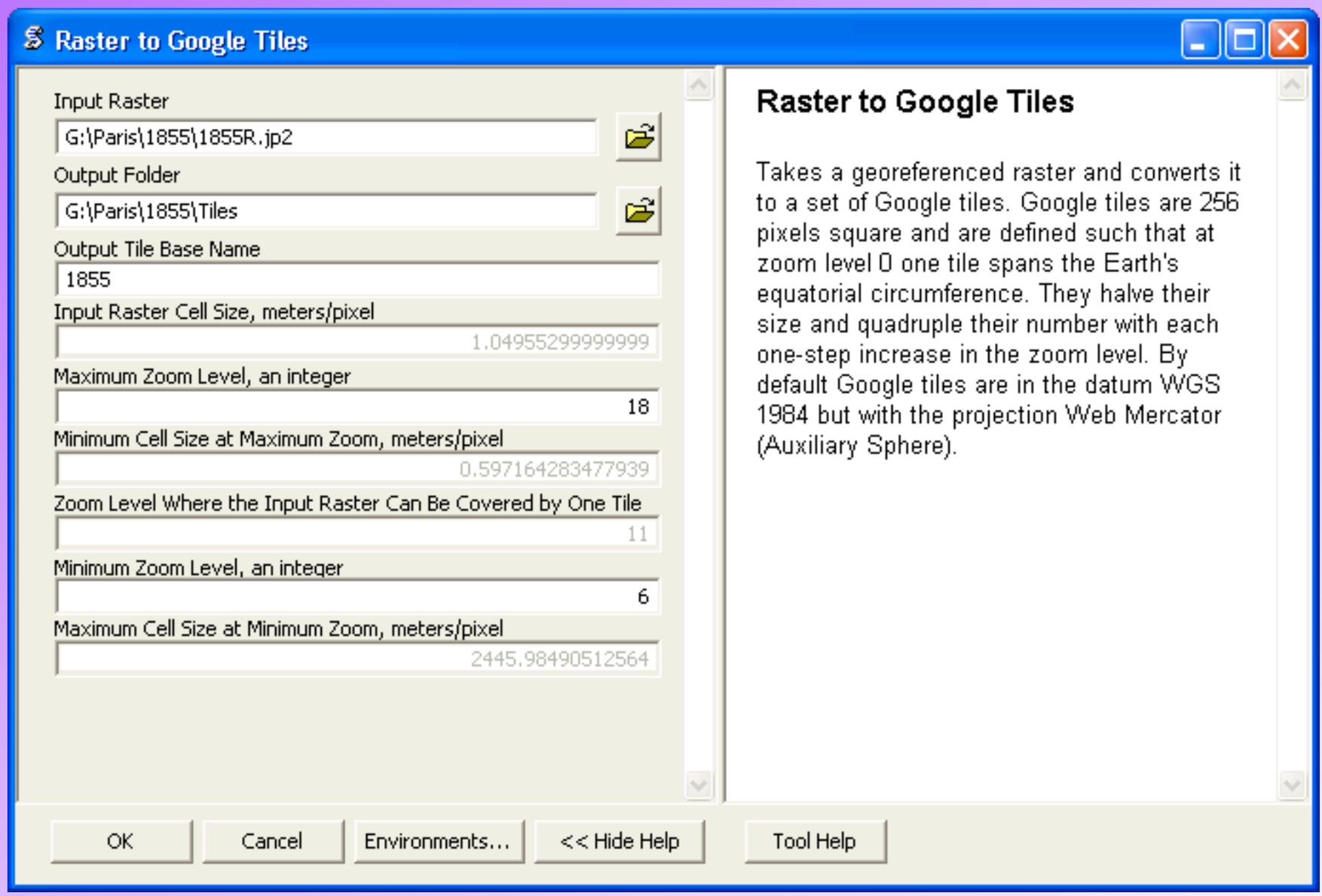
$$ly(ty, z) = \pi r \left(1 - ty / 2^{z-1} \right)$$

  which can be used to calculate the padding.
- So, the covering tile indices yield the size of the extended image, always a multiple of 256 pixels.

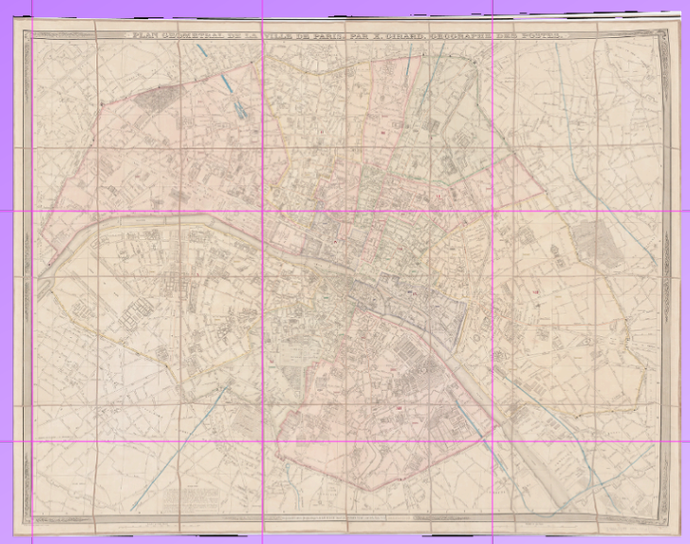
GTiler Script

- Tiling a georeferenced and rescaled image can be effected with programs such as Photoshop, etc.

- But, to automate the process, I wrote a Python script.



GTiler Procedure 1: Project Raster and Tiles



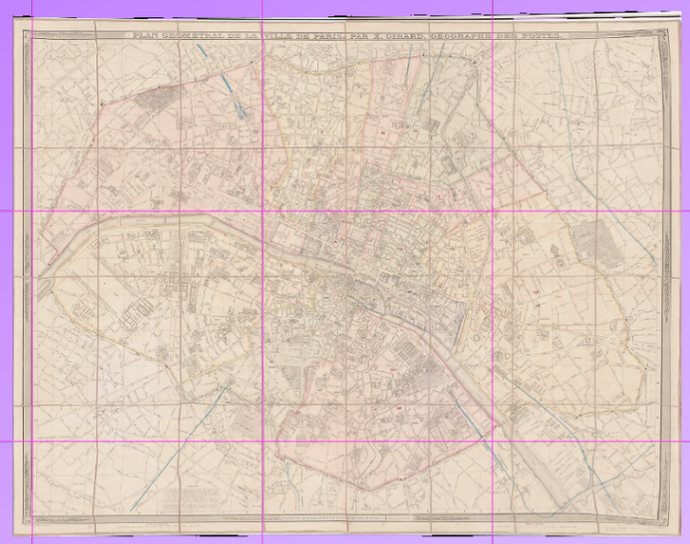
- The georeferenced image is first rescaled using:

```
gp.ProjectRaster_management( inputRaster, zRaster,
spatialReference, "CUBIC", cellSize, "", "0 0" )
```

- Using the formulae described previously, the four corners of the covering tiles are calculated:

```
corner = gp.CreateObject("Point")
boundary = gp.CreateObject("Array")
corner.ID = 0;  corner.X = xMin;  corner.Y = yMin  # four times
boundary.Add(corner)  # four times
```

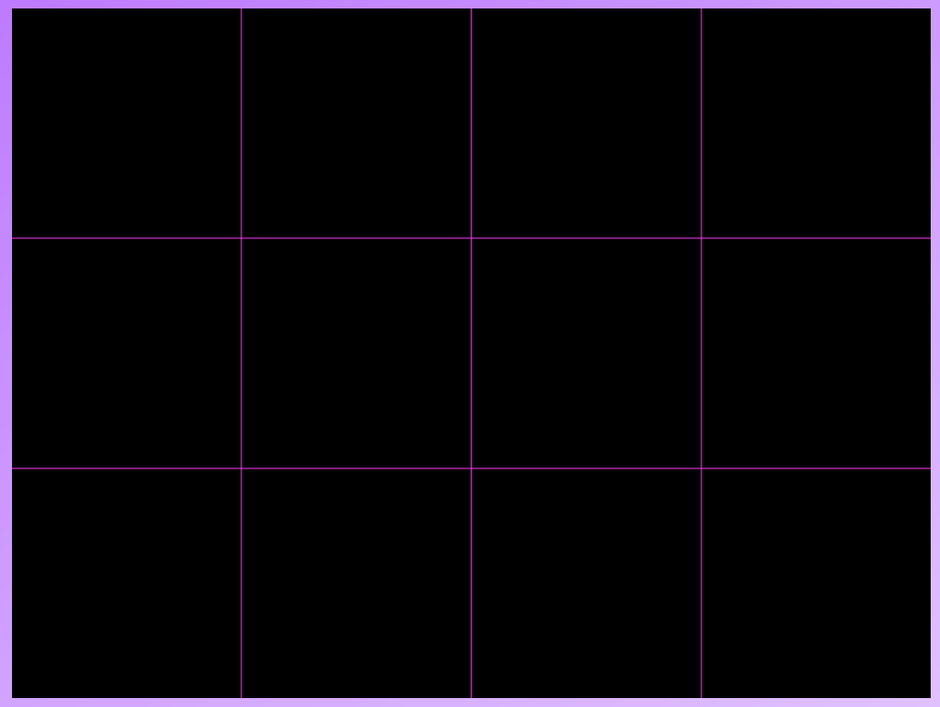
GTiler Procedure II: Create Extent Polygon



- Using the boundary list, an extent rectangle is created:

```
gp.CreateFeatureClass_management( gp.Workspace,
        bPolygon, "Polygon", "", "", "", spatialReference )
cursor = gp.InsertCursor( bPolygon )
polygon = cursor.NewRow()
polygon.shape = boundary  # the field polygon.Id defaults to 0
cursor.InsertRow( polygon )
```

GTiler Procedure III: Create a Background



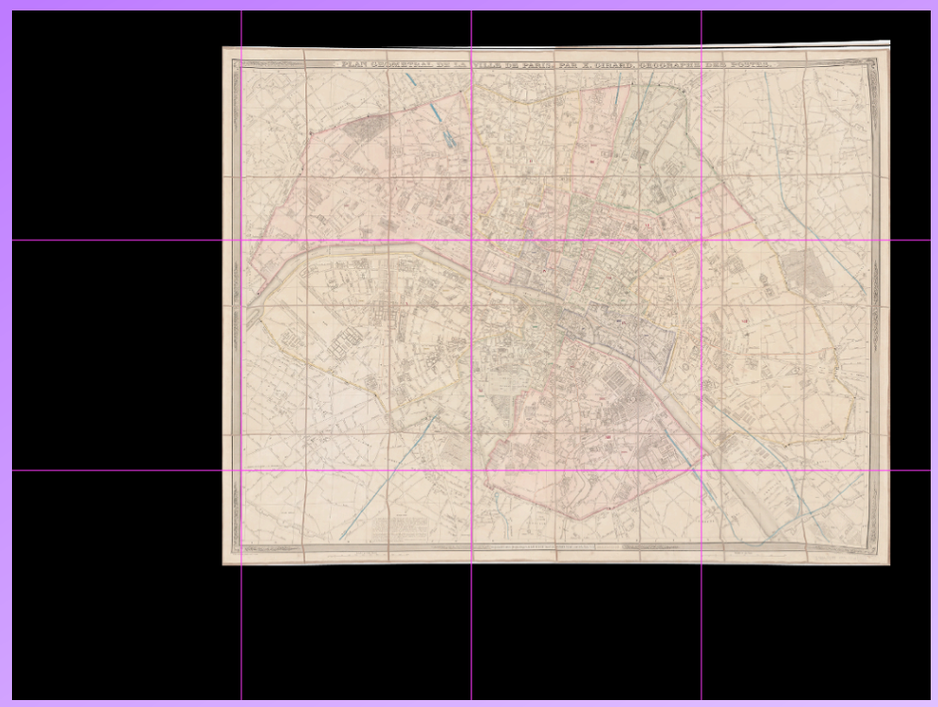
- A background raster the size of the final image is created with all NoData:

  gp.FeatureToRaster_conversion(
  bPolygon, "Id", bRaster, tileSize = 256 * cellSize)

- If the input raster has more than one band, the previous step is repeated for each and then:

  gp.CompositeBands_management(
  "band_1; band_2; band_3", bRaster)

GTiler Procedure IV: Mosaic the Image



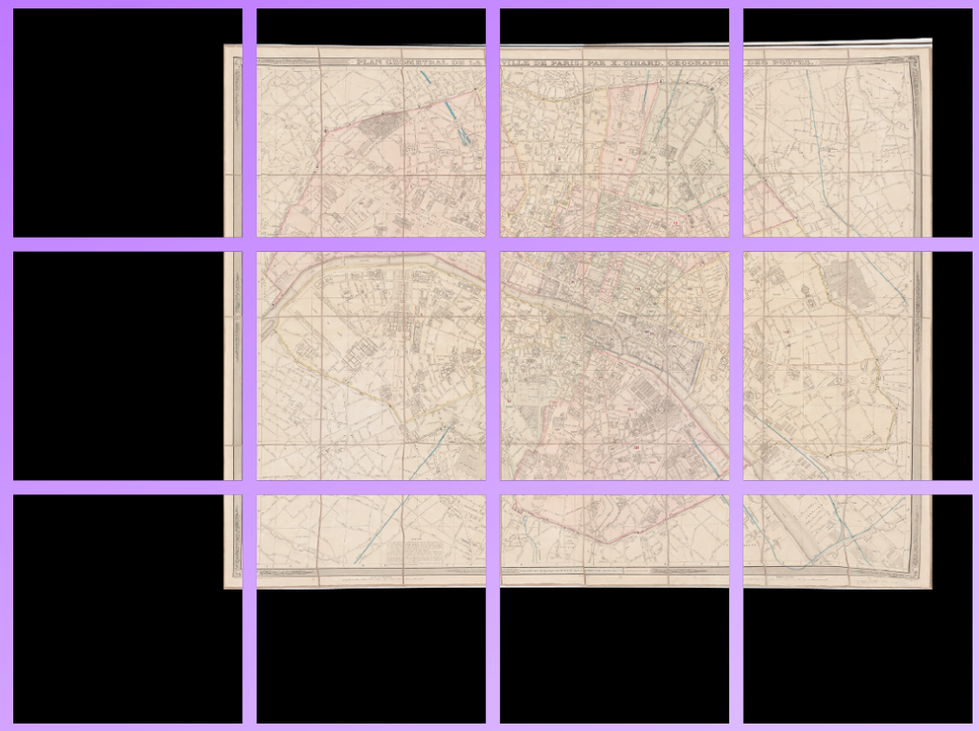
- An empty raster is created to hold the mosaic:

```
gp.CreateRasterDataset_management( gp.Workspace,
xRaster, cellSize, inputRasterPixelType,
spatialReference, inputRasterBandCount )
```

- The background and rescaled image are then mosaicked together:

```
gp.Mosaic_management( bRaster + ";" + zRaster, xRaster,
"LAST", "LAST", 0, "", "NONE", 0.1, "NONE" )
```

GTiler Procedure V: Clip the Mosaic

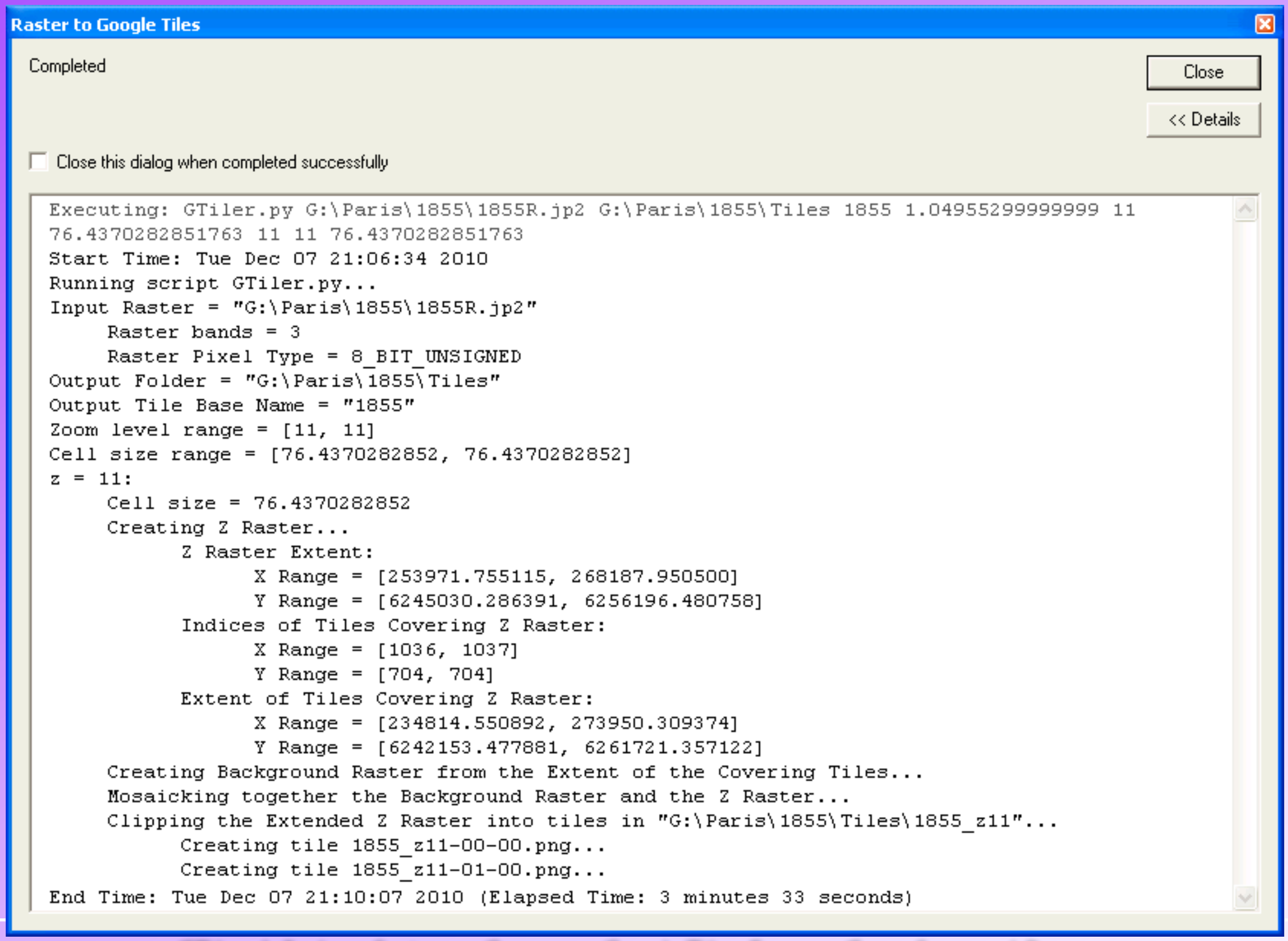


- The mosaicked image is turned into tiles by clipping it multiple times:

```
gp.Clip_management( xRaster,
        "%f %f %f %f" % ( xMin + cellOffset, yMin + cellOffset,
        xMax - cellOffset, yMax - cellOffset ),
        tRaster )
```

- In the above, `cellOffset = cellSize/2` to intersect only the boundary pixels of the raster and avoid rounding errors.

GTiler Execution



Acknowledgements

- Samuel Morse, professor of History of Art and of Asian Languages and Civilizations, the initiator.
- Scott Payne, director of Academic Technology Services, who suggested we build our own platform using Google Maps.
- Paul Chapin, who built the platform in JavaScript.
- Miodrag Glumac, who built the database interface.
- Smith College folks: Prof. Hélène Visentin (French), GIS Specialist Jon Caris, and especially student georeferencing specialist Zoë Zaferiou.

Image Sources

- Scanned maps were obtained from historical map collections at our institutions,
- And from those available on-line, specifically:

The University of California at Berkeley's East Asian Library

Photo Courtesy of Sébastien Bertrand

Photo Courtesy of Will Hart

The Harvard University Libraries

The David Rumsey Historical Map Collection

Contact Information

- Andy Anderson, Academic Technology Specialist:
  - Academic Technology Services, Amherst College, Amherst, MA 01002-5000
  - aanderson@amherst.edu
  - 413-542-2255
  - https://www.amherst.edu/people/facstaff/aanderson
  - Twitter @GeoObservatory