Classical Mapping Approach ➜ transfers the image source into a **geo-referenced canvas** for the GIS -Application
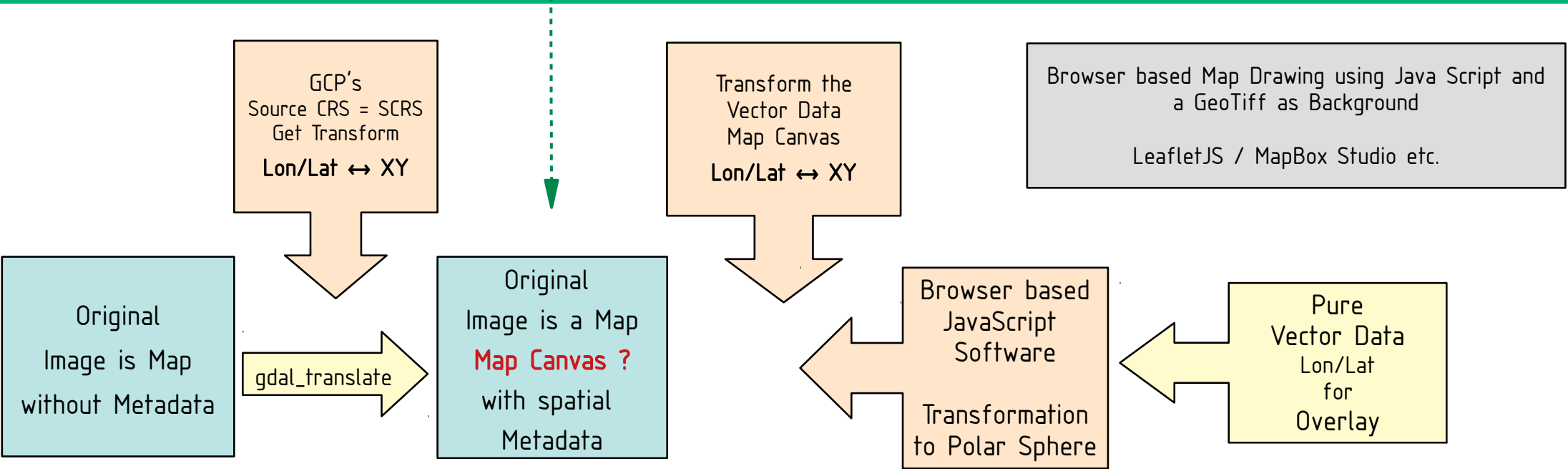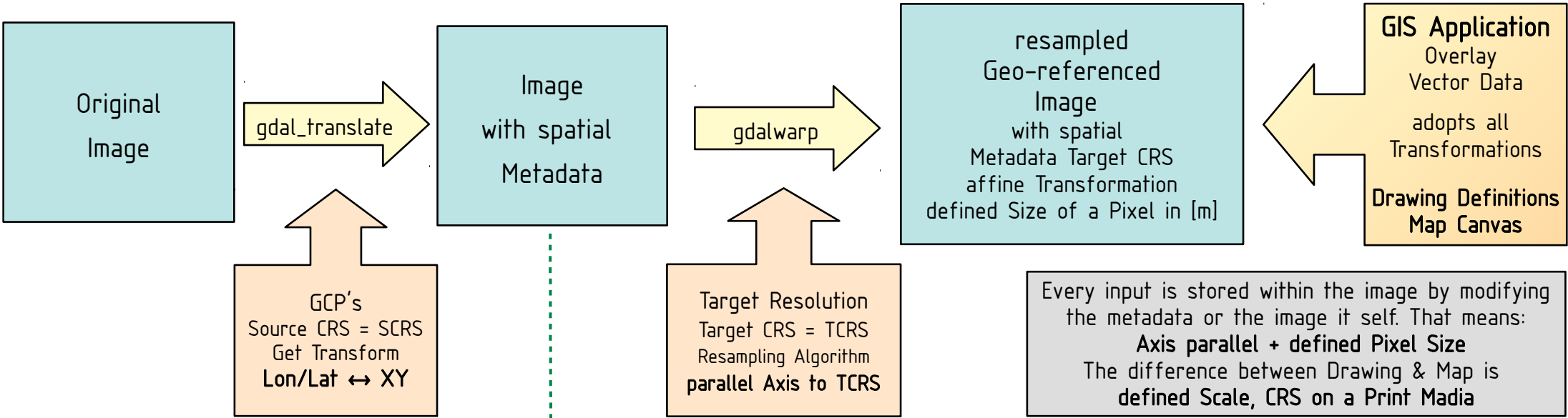
Original Image

gdal_translate

Image with spatial Metadata

gdalwarp

resampled Geo-referenced Image with spatial Metadata Target CRS affine Transformation defined Size of a Pixel in [m]

GIS Application
Overlay Vector Data
adopts all Transformations
Drawing Definitions Map Canvas

GCP's
Source CRS = SCRS
Get Transform
Lon/Lat ↔ XY

Target Resolution
Target CRS = TCRS
Resampling Algorithm
parallel Axis to TCRS

Every input is stored within the image by modifying the metadata or the image it self. That means:
Axis parallel + defined Pixel Size
The difference between Drawing & Map is
defined Scale, CRS on a Print Madia

GCP's
Source CRS = SCRS
Get Transform
Lon/Lat ↔ XY

Transform the Vector Data Map Canvas
Lon/Lat ↔ XY

Browser based Map Drawing using Java Script and a GeoTiff as Background
LeafletJS / MapBox Studio etc.

Original Image is Map without Metadata

gdal_translate

Original Image is a Map
Map Canvas ?
with spatial Metadata

Browser based JavaScript Software
Transformation to Polar Sphere

Pure Vector Data
Lon/Lat for Overlay

Map Recycling Approach ➜ augments an existing image to represent a Map Canvas but needs a JS - Application for Drawing