

# Open Street Map for Multi-Modal Freight Transport Planning



Markus Mayr, Gerhard Navratil  
Department of Geodesy and Geoinformation  
TU Wien

# Overview

- 1 Transportation
- 2 OpenStreetMap
- 3 Existing Systems
- 4 Routing with OSM
- 5 Missing in OSM

# Transport

Persons

vs.

Goods



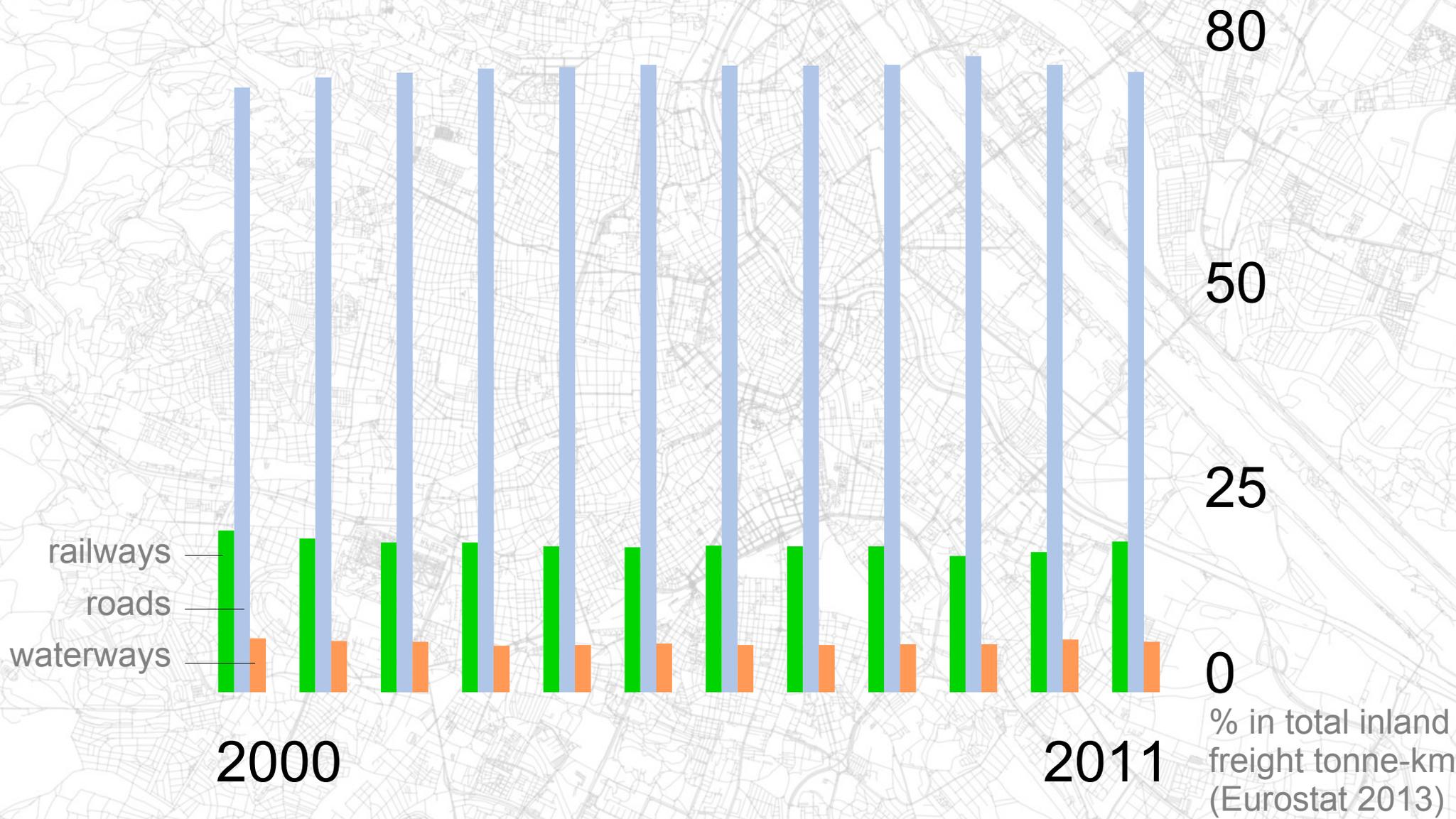
Is a basic requirement for economy

# Goods Transport

Factors affecting movement of goods: (according to Ortúzar, 2011)

- location factors (*dependency on source materials*)
- chain of dependencies
- physical factors of goods (*e.g. steel vs. milk*)
- operational factors (*e.g. company size and internal regulations*)
- **geographical factors** (*e.g. population density*)
- **dynamic factors** (*e.g. seasonal changes*)
- pricing factors (*usually not published like with passenger transport*)
- weather
- traffic

# Modal Split



# MultiModal Transport

In contrast to MonoModal / InterModal

Change needs time / is timely constrained

Change only at specific locations

Optimizes for multiple criterions

Not always the fastest route is best (e.g. pirates)

Legal constraints of certain goods

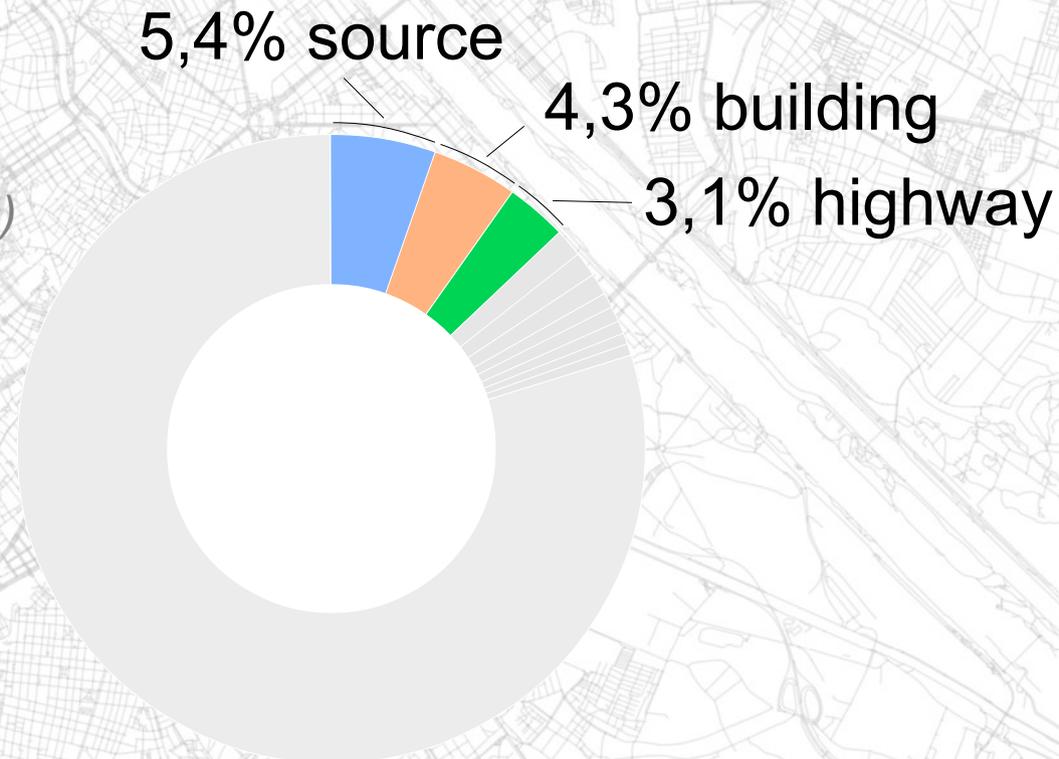
# OSM

OSM data types:

- nodes (*geographical points + attributes*)
- ways (*list of nodes + attributes*)
- relations (*list of nodes or ways + attributes*)

Especially streets of higher order are of sufficient quality

Routing with OSM data is already done



# Existing Systems



AnachB  
*(personal, designed for specific region)*



FLAVIA  
*(intermodal freight, no real-time traffic)*



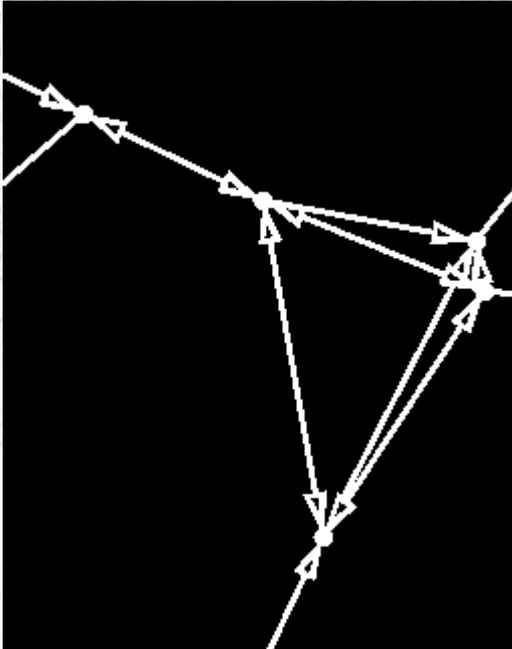
OpenTripPlanner  
*(personal, integrates multiple sources,  
uses GTFS as storage format, OSM based)*

gives examples for multimodal freight transport planning

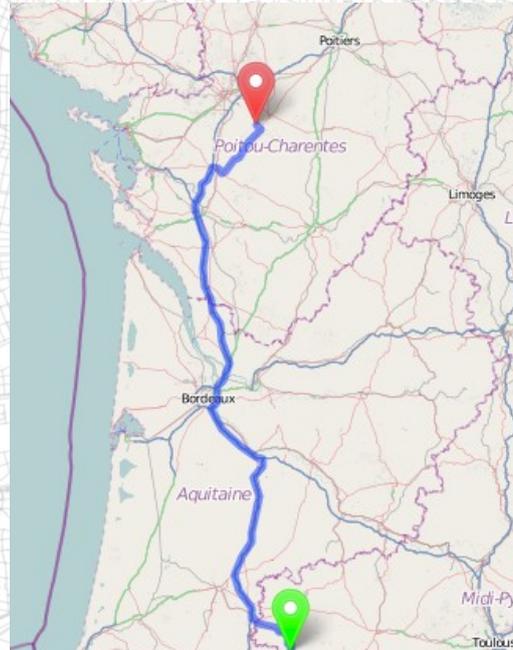
# Routing

Basis for any route planning application

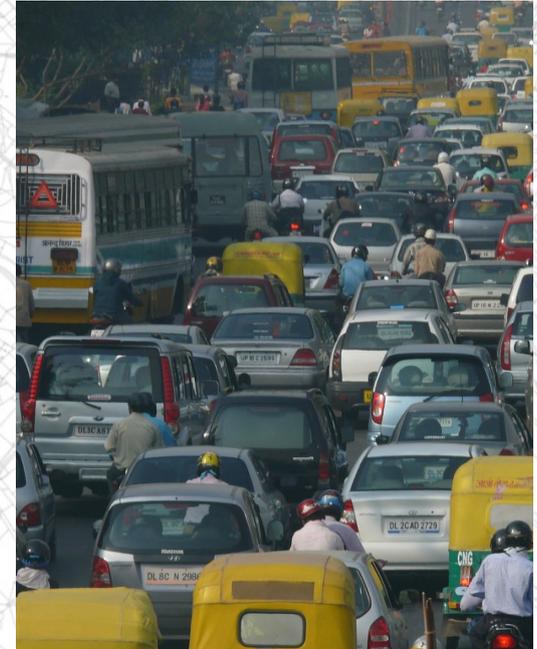
routable graph



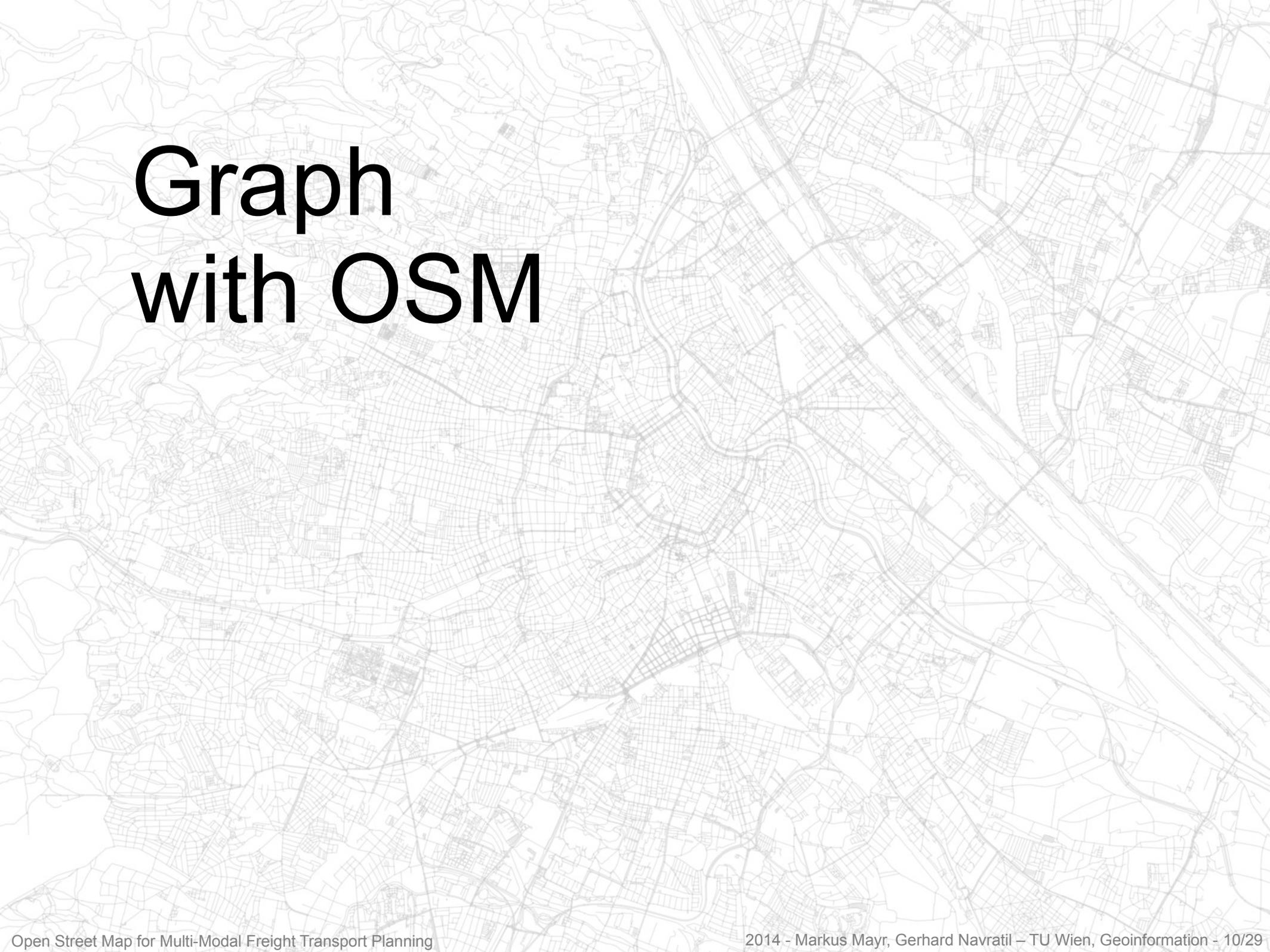
routing engine



additional (real-time) data



# Graph with OSM



# Graph OSM - Tags

aircraft (*no routes but infrastructure*)

railways (*lines as ways, stops as nodes, trains as relations, routes as relations*)

ferries (*water as ways or area, some attributive information*)

connections (*switchpoints*) :

- monomodal (*two ways share a common node, restrictions by relations*)
- multimodal (*public\_transport=platform/station/stop\_area(\_group), relation*)
- multimodal (*connecting end-points*)

# Aircraft

Item ↕	Description ↕
Aerodrome	Can be tagged as a <i>node</i> , an <i>area</i> or a <i>multipolygon</i> . As well as having a <code>aeroway=aerodrome</code> it should also have a <code>name=*</code> and may have an <code>iata=*</code> or <code>icao=*</code> code and many others. This node should also contain <code>type=*</code> to indicate what type of aerodrome it is: (public, military, private or some other type). For complex boundaries a <code>relation:multipolygon</code> can be used (for example, see <a href="#">Norwich airport</a> ). Using a relation in this way allows a boundary to be constructed from a number of elements, such as fences, walls and parts of buildings. The boundary may be for the entire aerodrome complex, or for 'airside only'.
Runway	Each <i>runway</i> should be drawn as a <i>way</i> and tagged as <code>aeroway=runway</code> with <code>ref=*</code> . Use <code>surface=*</code> for the runway surface. Runways can be defined as ways or as areas. A proposed new runway, or a runway under construction should be tagged with <code>aeroway=construction</code> (or <code>aeroway=proposed</code> ) and <code>construction=runway</code> . A disused, but generally function runway should be tagged with either <code>disused=yes</code> or as <code>disused:aeroway=runway</code> . A abandoned runway can be tagged with <code>abandoned:aeroway=runway</code> .
Taxiway	Each <i>taxiway</i> should be drawn as a <i>way</i> and tagged as <code>aeroway=taxiway</code> with <code>ref=*</code> . Use <code>surface=*</code> for the taxiway surface. A proposed new taxiway, or a taxiway under construction should be tagged with <code>aeroway=construction</code> (or <code>aeroway=proposed</code> ) and <code>construction=taxiway</code> . A disused, but generally function taxiway should be tagged with either <code>disused=yes</code> or as <code>disused:aeroway=taxiway</code> . A abandoned taxiway can be tagged with <code>abandoned:aeroway=taxiway</code> .
Holding position	Use <code>aeroway=holding_position</code> on nodes along the taxiway, runway to identify holding positions.
Apron	Use <code>aeroway=apron</code> to identify areas where planes are parked or are serviced. If an apron has a particular purpose or designation then provide this using <code>name=*</code> (eg 'Parking area 15' or 'de-icing').
Aircraft Parking	Use <code>aeroway=parking_position</code> to show aircraft parking positions. Name and location doesn't necessarily correspond to <code>aeroway=gate</code> .
Terminal	Each <i>terminal</i> should be drawn as an <i>area</i> tagged as <code>aeroway=terminal</code> with a <code>name=*</code> . Tag the entrance to the terminal using a <code>building=entrance</code> node. use <code>name=*</code> for the terminal name (eg 'terminal 1', 'main hall').
Helipad	Mark helipads with <code>aeroway=helipad</code> using a node or area and ideally also a <code>ref=*</code> .
Gate	Use <code>aeroway=gate</code> to identify each departure gate and include a <code>ref=*</code> if this is known.
Beacon	Use <code>man_made=beacon</code> nodes to mark navigation lights.
Grass	Mark areas of grass with <code>landuse=grass</code> . If appropriate use <code>aeroway=*</code> , <code>surface=grass</code>
Hangar	Buildings where aircraft are stored or repaired should be tagged with <code>aeroway=hangar</code> .
Windsock	Mark <code>aeroway=windsocks</code> as a <i>node</i> . If it is lighted, include <code>lit=yes</code> .
Navigation aid	Use <code>aeroway=navigationaid</code> on nodes representing navigation aids such as VORs and NDBs.
Papi/Vasi	A <code>W</code> PAPI or <code>V</code> VASI should be marked by a <i>node</i> and tagged as <code>aeroway=papi</code> or <code>aeroway=vasi</code> .
Public Parking	Many aerodromes also include various forms of <i>parking</i> , mapped as <i>areas</i> tagged with <code>amenity=parking</code> . Multistorey Car Parks should be mapped as <i>areas</i> tagged as <code>amenity=parking</code> and <code>parking=multi-storey</code> . Provide an identifier or name using <code>ref=*</code> .
Service road	Mark internal service roads as ways tagged with <code>highway=service</code> . Include <code>access=private</code> if the public are not able to use them.
Tower	Map any control towers using <code>man_made=tower</code> and <code>service=aircraft_control</code> . <b>(Proposal Only)</b>

# Aircraft



# Graph OSM - Tags

aircraft (*no routes but infrastructure*)

railways (*lines as ways, stops as nodes, trains as relations, routes as relations*)

ferries (*water as ways or area, some attributive information*)

connections (*switchpoints*) :

- monomodal (*two ways share a common node, restrictions by relations*)
- multimodal (*public\_transport=platform/station/stop\_area(\_group), relation*)
- multimodal (*connecting end-points*)

# Railways

Schlagwörter: 19 / Mitgliedschaften: 4

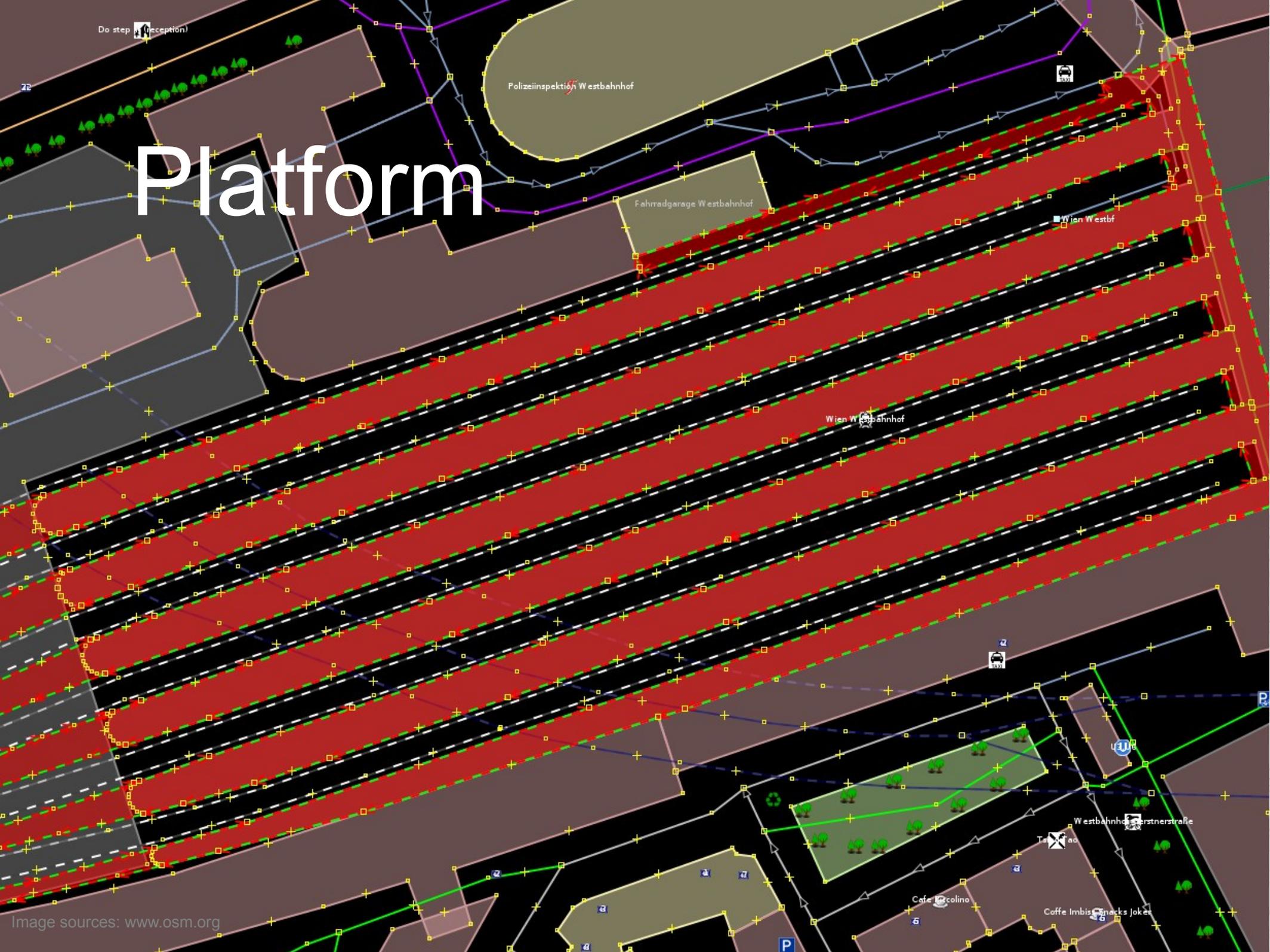
Transport/Gleise/Eisenbahn ...

Schlüssel	Wert
electrified	contact_line
frequency	16.7
gauge	1435
maxspeed	40
name	Westbahn
operator	ÖBB
railway	rail
railway:bidirectional	regular
railway:pzb	yes
railway:radio	gsm-r
railway:track_class	D4
railway:track_ref	9
railway:traffic_mode	passenger
ref	101 01
structure_gauge	GC
tracks	1
usage	main
voltage	15000
wikipedia:de	Westbahn (Österreich)

Element von	Rolle	Posit...
Route ("111: Westbahn St. Pölten-Tullnerfeld-Wien", 86 Ele...		73
Route ("150 Wien Westbf-Linz-Wels-Neumarkt Kallham -Schä...		778
Route ("Westbahn", 201 Elemente, unvollständig)		38
Route ("railjet München", 753 Elemente, unvollständig)		746

Do step (reception)

# Plattform



# Graph OSM - Tags

aircraft (*no routes but infrastructure*)

railways (*lines as ways, stops as nodes, trains as relations, routes as relations*)

ferries (*water as ways or area, some attributive information*)

connections (*switchpoints*) :

- monomodal (*two ways share a common node, restrictions by relations*)
- multimodal (*public\_transport=platform/station/stop\_area(\_group), relation*)
- multimodal (*connecting end-points*)

# Ferries

Schlagwörter: 4 / Mitgliedschaft

 Wasser/Schiffahrt/Fährroute ...

Schlüssel	Wert
bicycle	yes
foot	yes
route	ferry
vehicle	no

Schlagwörter: 3 / Mitgliedschaft

 Wasser/Schiffahrt/Fährstelle ...

Schlüssel	Wert
amenity	ferry_terminal
cargo	passengers
name	Sankt Wolfgang

# Graph OSM - Tags

aircraft (*no routes but infrastructure*)

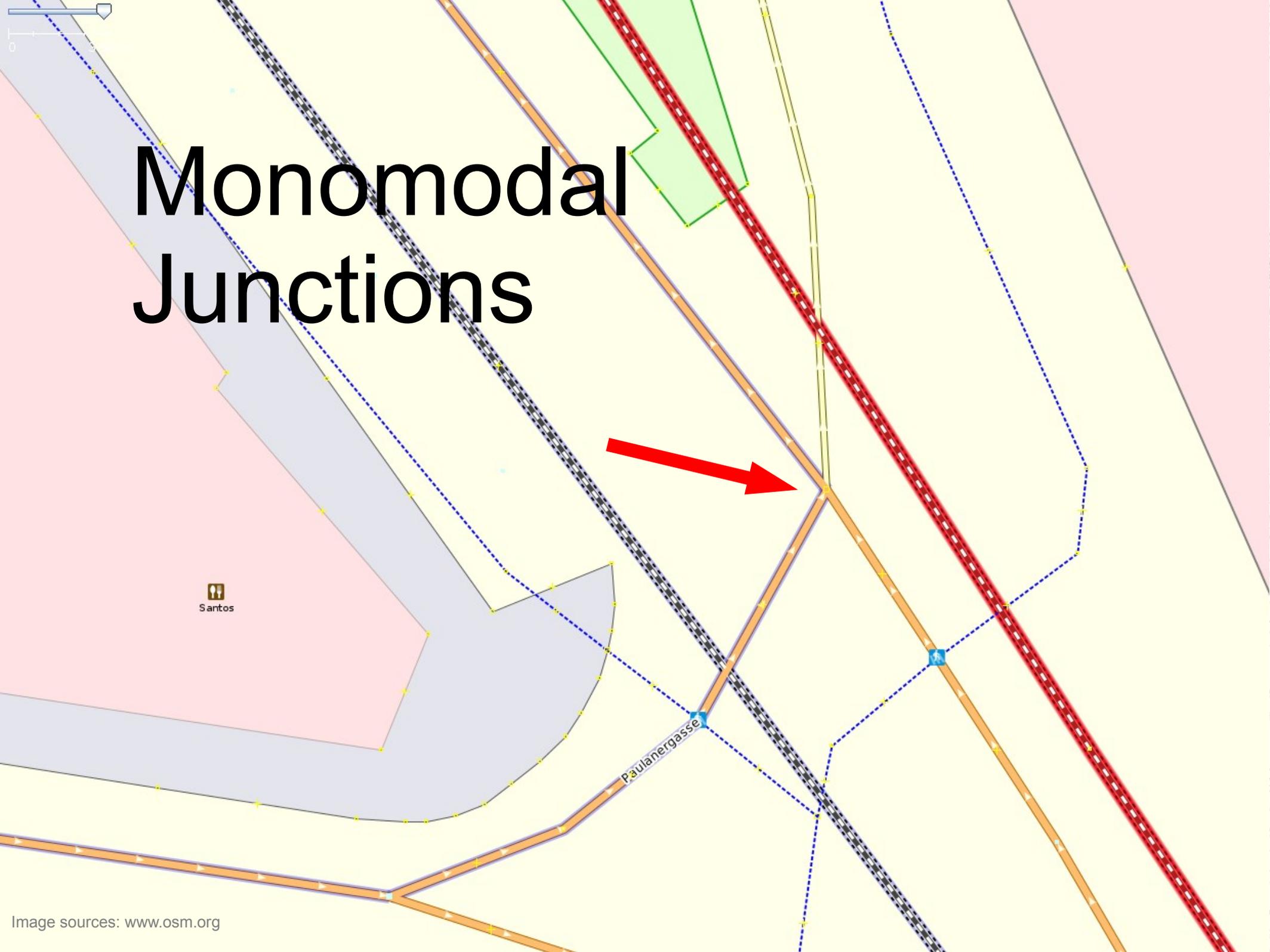
railways (*lines as ways, stops as nodes, trains as relations, routes as relations*)

ferries (*water as ways or area, some attributive information*)

connections (*switchpoints*) :

- monomodal (*two ways share a common node, restrictions by relations*)
- multimodal (*public\_transport=platform/station/stop\_area(\_group), relation*)
- multimodal (*connecting end-points*)

# Monomodal Junctions



Santos

Paulanergasse

# Graph OSM - Tags

aircraft (*no routes but infrastructure*)

railways (*lines as ways, stops as nodes, trains as relations, routes as relations*)

ferries (*water as ways or area, some attributive information*)

connections (*switchpoints*) :

- monomodal (*two ways share a common node, restrictions by relations*)
- multimodal (*public\_transport=platform/station/stop\_area(\_group), relation*)
- multimodal (*connecting end-points*)

# MultiModal Relations

Ändere Relation 3.314.805 in Ebene "Datenebene 1"

Merkmalen und Elemente Eltern-Relationen Kind-Relationen

Merkmalen

Transport/Öffentlicher Verkehr/Haltebereich ...

Schlüssel	Wert
name	Salzburg Hbf
operator	ÖBB
public_transport	stop_area
type	public_transport
uic_ref	8100002

Elemente

Rolle	Referenziert auf	
platform	37517515 (33 Punkte)	
stop	Salzburg Hbf (47.8130402, 13.0452858)	
stop	Salzburg Hbf (47.8130011, 13.0454115)	
platform	2 + 3 (25 Punkte)	
stop	Salzburg Hbf (47.8129432, 13.045591)	
stop	Salzburg Hbf (47.8129238, 13.0456545)	
platform	4 + 5 (26 Punkte)	
stop	Salzburg Hbf (47.8128682, 13.0458318)	
stop	Salzburg Hbf (47.8128513, 13.0458902)	
platform	6 + 7 (21 Punkte)	
stop	Salzburg Hbf (47.8128068, 13.0460423)	
stop	Salzburg Hbf (47.8127872, 13.0461053)	
platform	8 + 9 (23 Punkte)	
stop	Salzburg Hbf (47.8127403, 13.0462624)	
	Hausnummer 1 in Südtiroler Platz (12 Punkte)	

Auswahl

Auswahl

- Salzburg Hbf (47.8128068, 13.0460423)

Rolle anwenden:

# Graph OSM - Tags

aircraft (*no routes but infrastructure*)

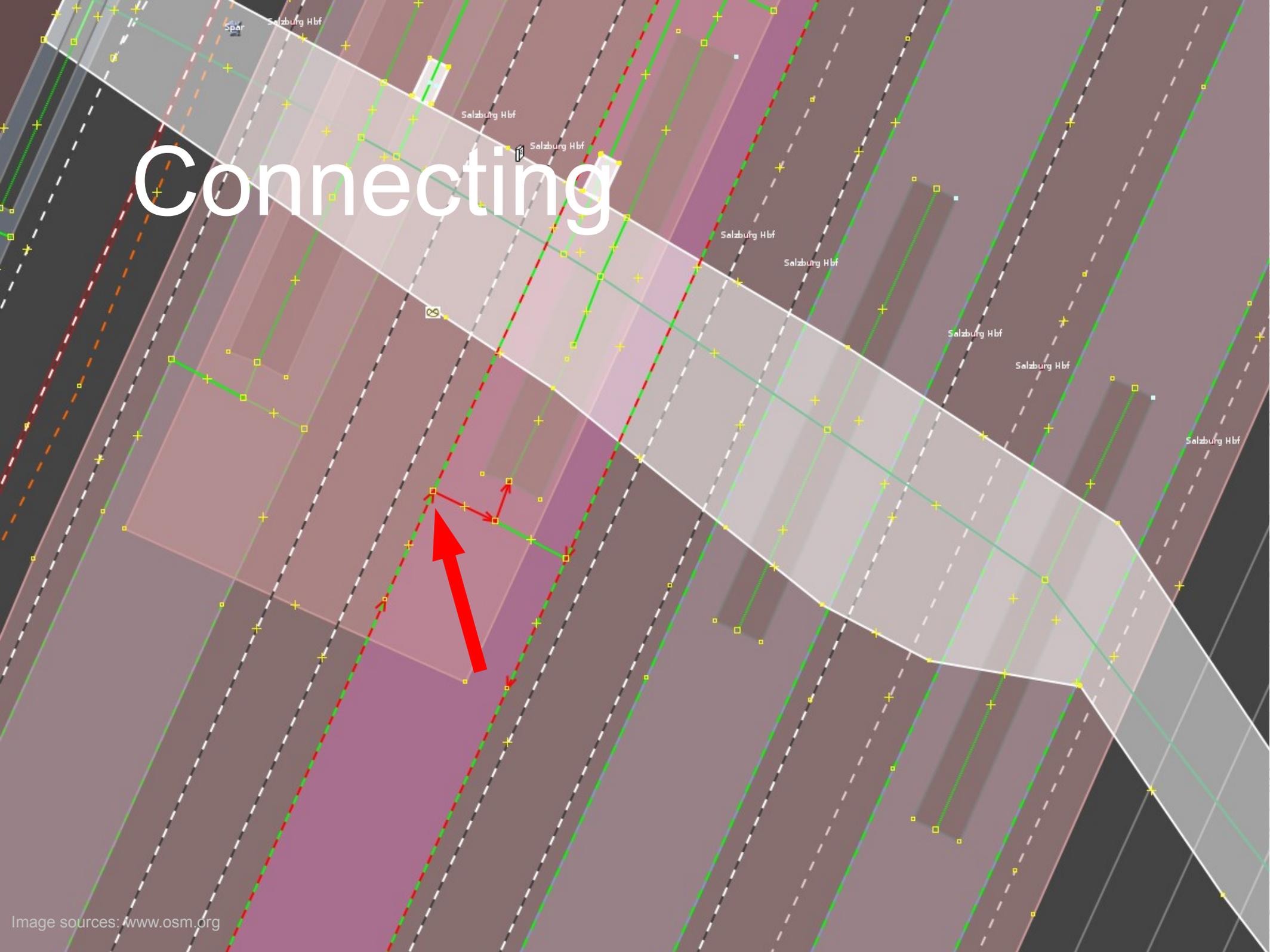
railways (*lines as ways, stops as nodes, trains as relations, routes as relations*)

ferries (*water as ways or area, some attributive information*)

connections (*switchpoints*) :

- monomodal (*two ways share a common node, restrictions by relations*)
- multimodal (*public\_transport=platform/station/stop\_area(\_group), relation*)
- multimodal (*connecting end-points*)

# Connecting



# Route-Planning with OSM

Building a routable graph is no problem

Many libraries doing routing on OSM exist

- *Java*: GraphHopper, Brouter, OpenTripPlanner, osmNavigation
- *C/C++*: Gosmore, MoNav, Navit, OpenSourceRoutingMachine, Routino
- *C#*: IMORTIS, OsmSharp, SimpleMapRouting
- *Scala*: osm\_routing
- *Ruby*: Mormon
- *Python*: PyrouteLib, SimpleOsmRouter
- ClosedSource Solutions

...

# Additional (real-time) Data with OSM

No real time data in OpenStreetMap

Additional data may be

- too detailed
- be too specific
- updated too often

# Missing in OSM

<i>Type</i>	<i>Description</i>	<i>Example</i>
<b>Quality</b>	missing information that cannot be gathered	<i>volume of traffic, storehouse capacity</i>
<b>Frequency</b>	information that is changing too often	<i>real-time data on usage</i>
<b>Quantity</b>	information that is too specific	<i>train schedules</i>
<b>Availability</b>	service that is not provided	<i>routing algorithm</i>

# Using OSM

OSM is a useable additional source for multimodal transport planning

OSM can not be the only source for multimodal transport planning

When mixing different datasets, thoughts about legal issues are necessary

MultiModal Routing with OSM Data is already done

