

# Bringing Validation to Users

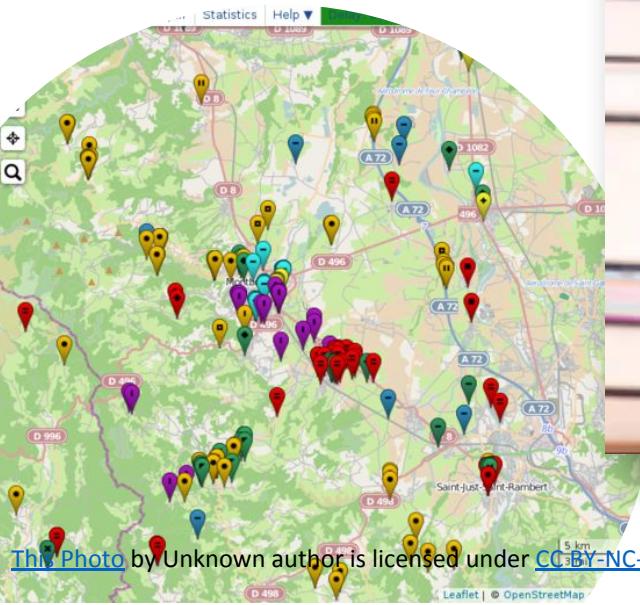
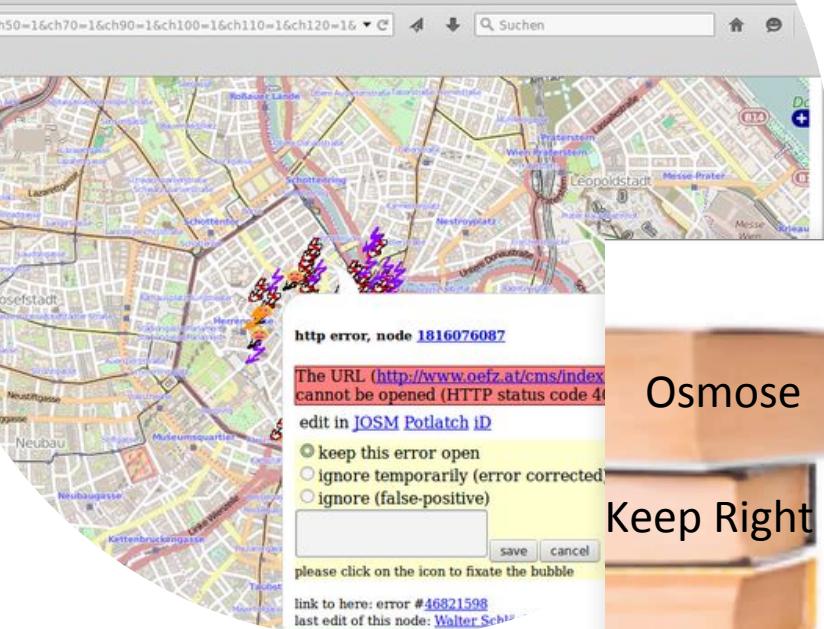
Integrating Quality Assurance Checks  
into Map Editors

Clarissee Abalos  
@map2save

Matthew Gibb  
@giblet22

# The Problem

- Abundance of data issues waiting to be cleaned up
- There are more mappers than validators
- Mappers need more guidance while mapping



# Quality Assurance Approach

Stop bad or incomplete data from being created



# Solutions



Validation rules in  
JOSM and iD

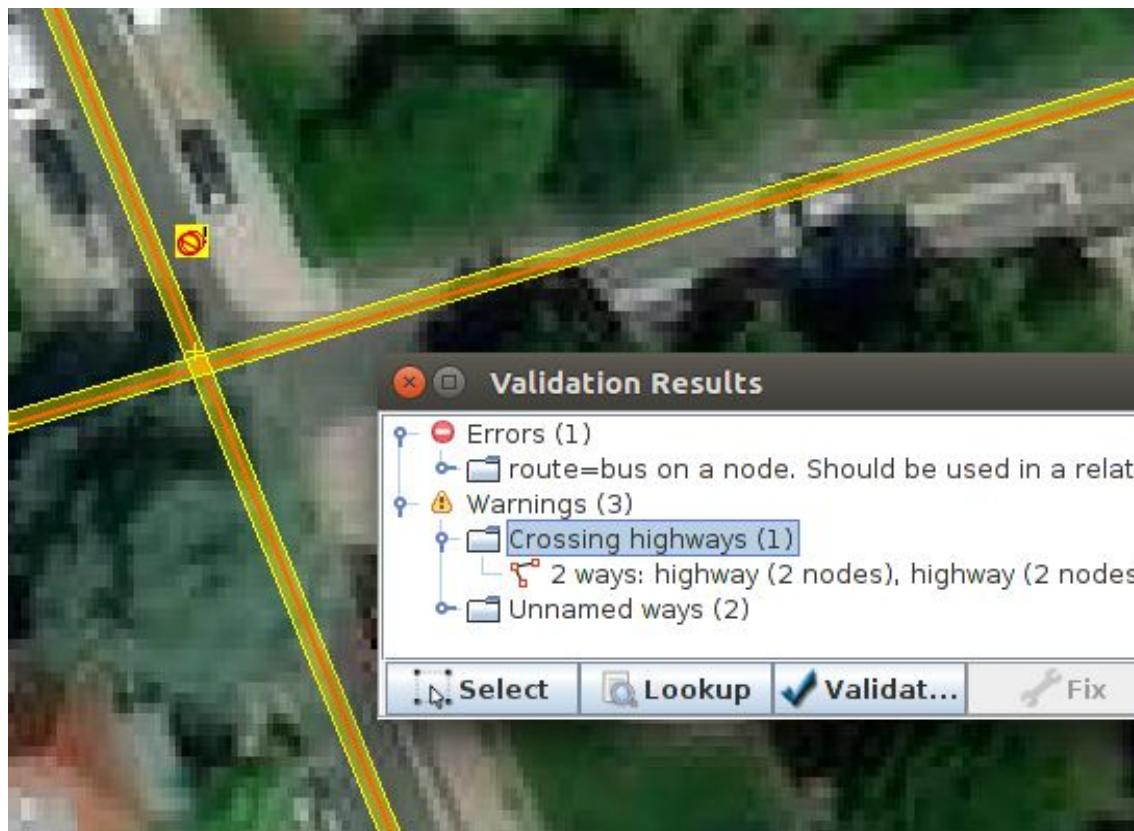


Custom Presets and  
Validation Rules

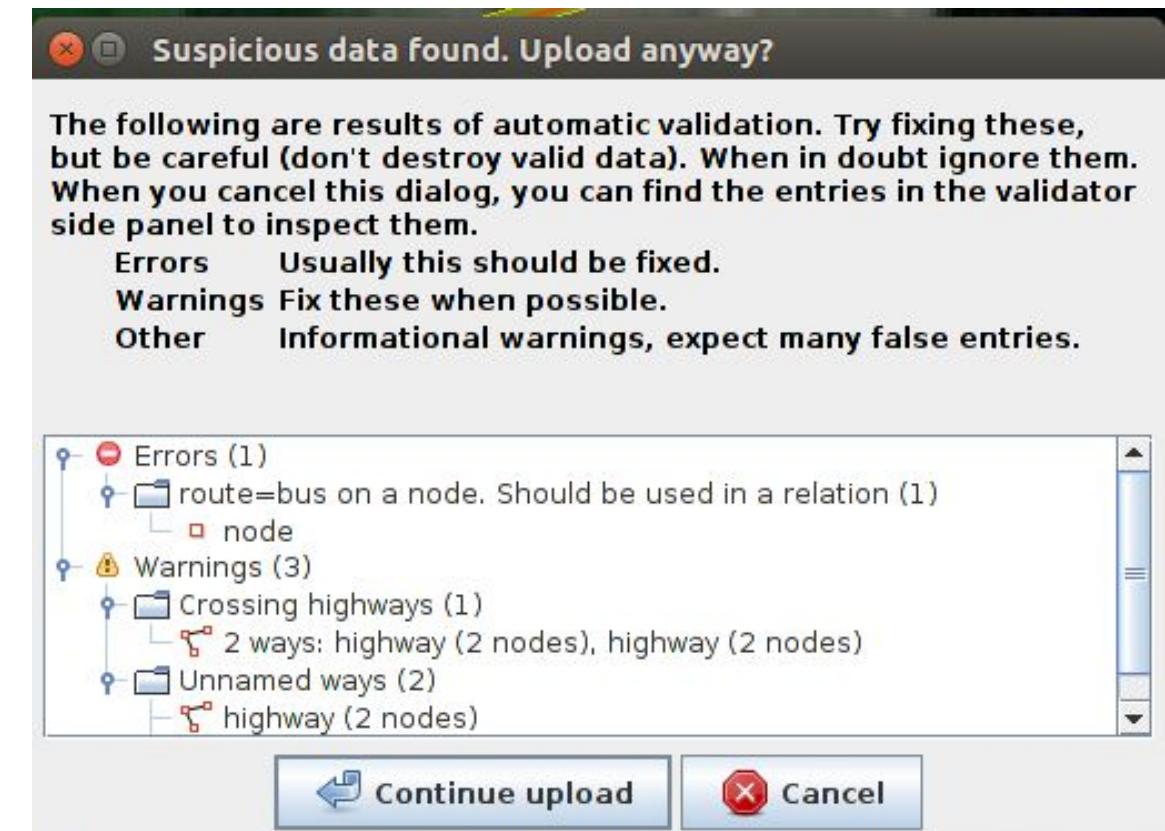


Validation-centric  
Overpass Queries

# Validation in JOSM



Validation on dataset while editing features



Warnings before upload

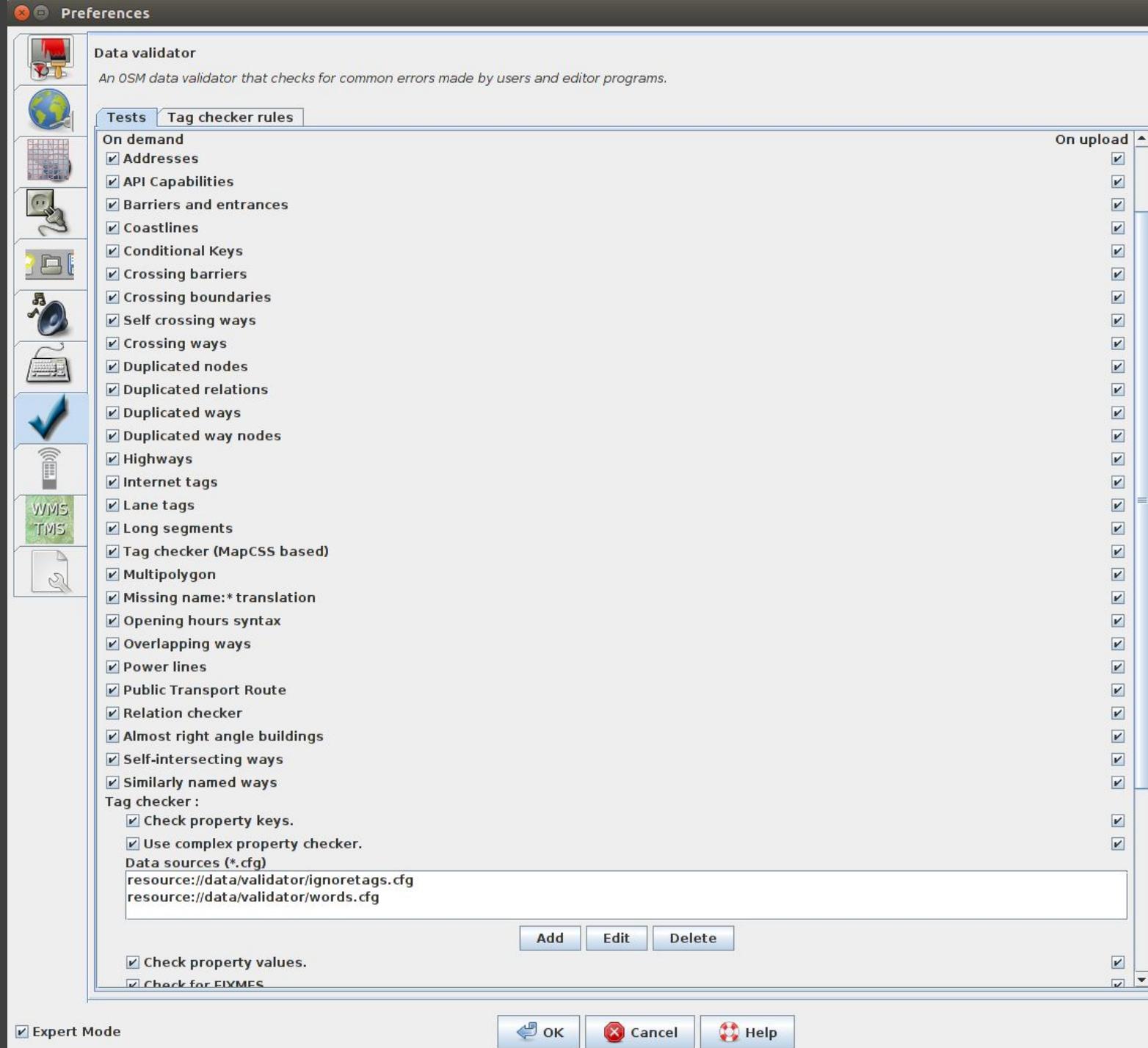
# JOSM Data Validator Types

## Tag Checker Rules (MapCSS)

- Great for basic topology and tag checks

## Tests (Java)

- Able to handle more complex geometry checks





# What is MapCSS?

- CSS-like language for map stylesheets
- Used in JOSM for map styling and data validation



Each color represents a building  
with a different name

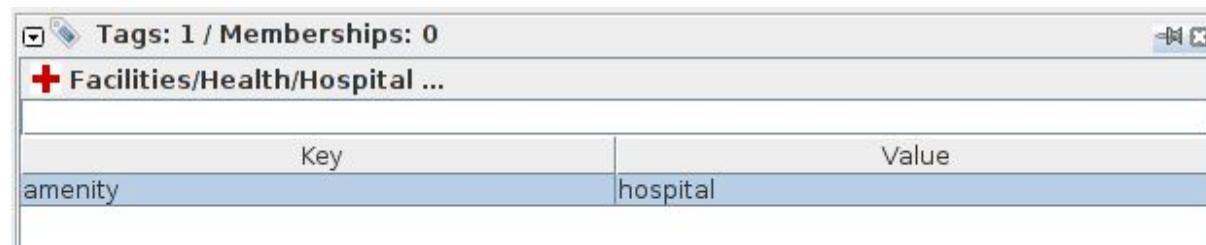
# Basic MapCSS Validation Checks

```
geometry[key=value] {  
    throwWarning: "Fix this";  
}
```

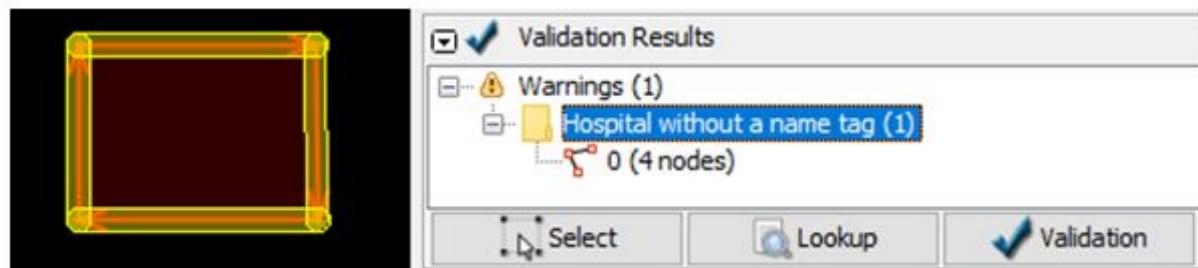
Selector: what feature the rule is looking for  
Instructions on what to do with the feature

```
*[amenity=hospital][!name] {  
    throwWarning: "Hospital without a name tag";  
}
```

Selector: \* = all geometries (node,way,relation),  
Features tagged as **amenity=hospital** that does  
not have a **name** tag.



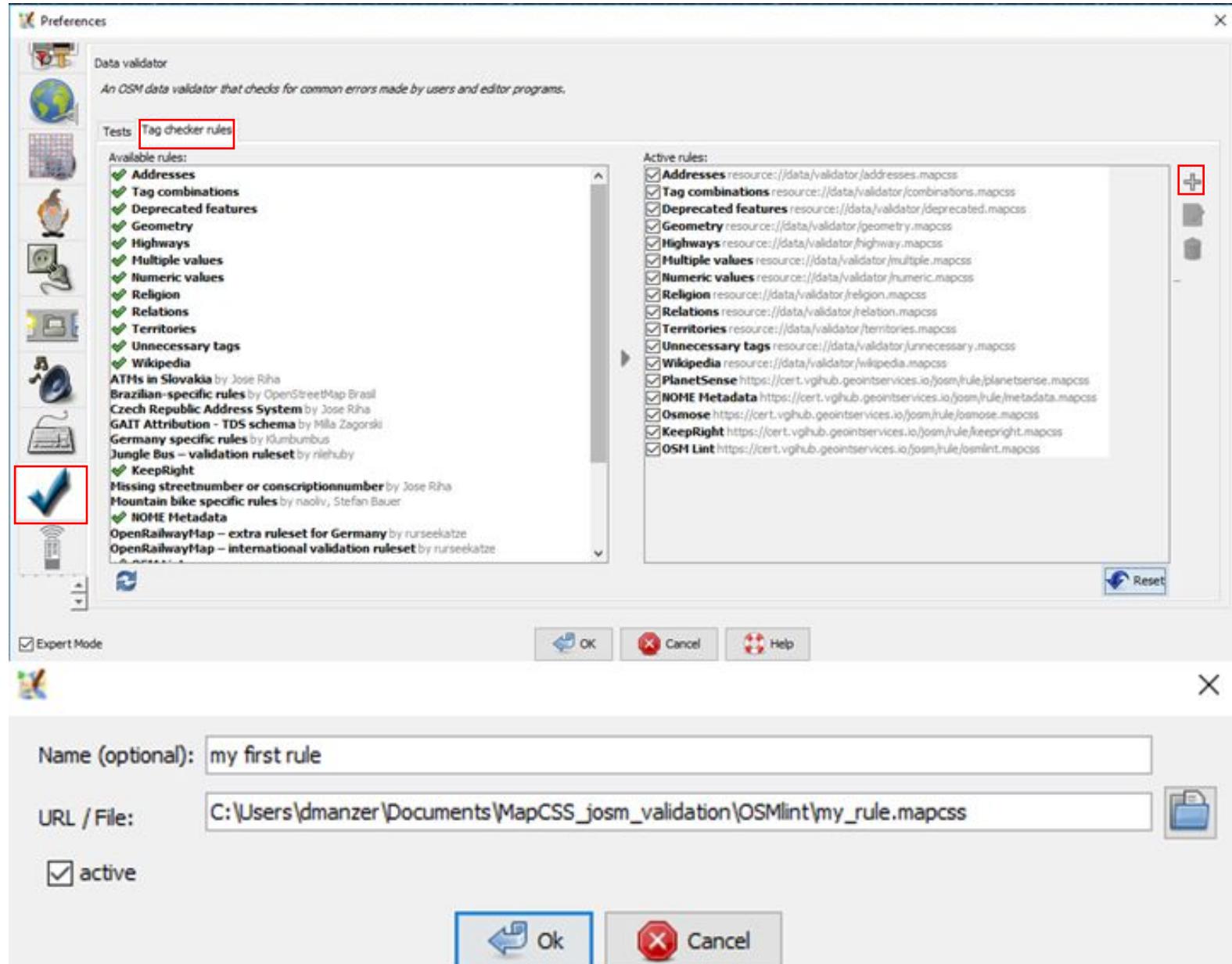
Warning in JOSM validator window



Features that meet the rule criteria appear in the JOSM Validation Results window

# Create your own MapCSS checks locally

1. Write the rule - save as .mapcss or .validator.mapcss file
2. Add as an Active Rule in JOSM
3. Click “Validation”



# Converting other QA tools to MapCSS

```
/* Keep Right motorways connected directly 270 */
/* Message: This node is a junction of a motorway and a highway other than motorway, motorway_link, trunk, rest_area or construction.
Service or unclassified is only valid if it has access=no/private or if it is a service=parking_aisle. */

way[highway=motorway] >[index=1] node:connection,
way[highway=motorway] >[index=-1] node:connection {
    set kr_270_end_motorway;
}

way[highway][highway!=motorway] node:connection {
    set hwycon270;
}

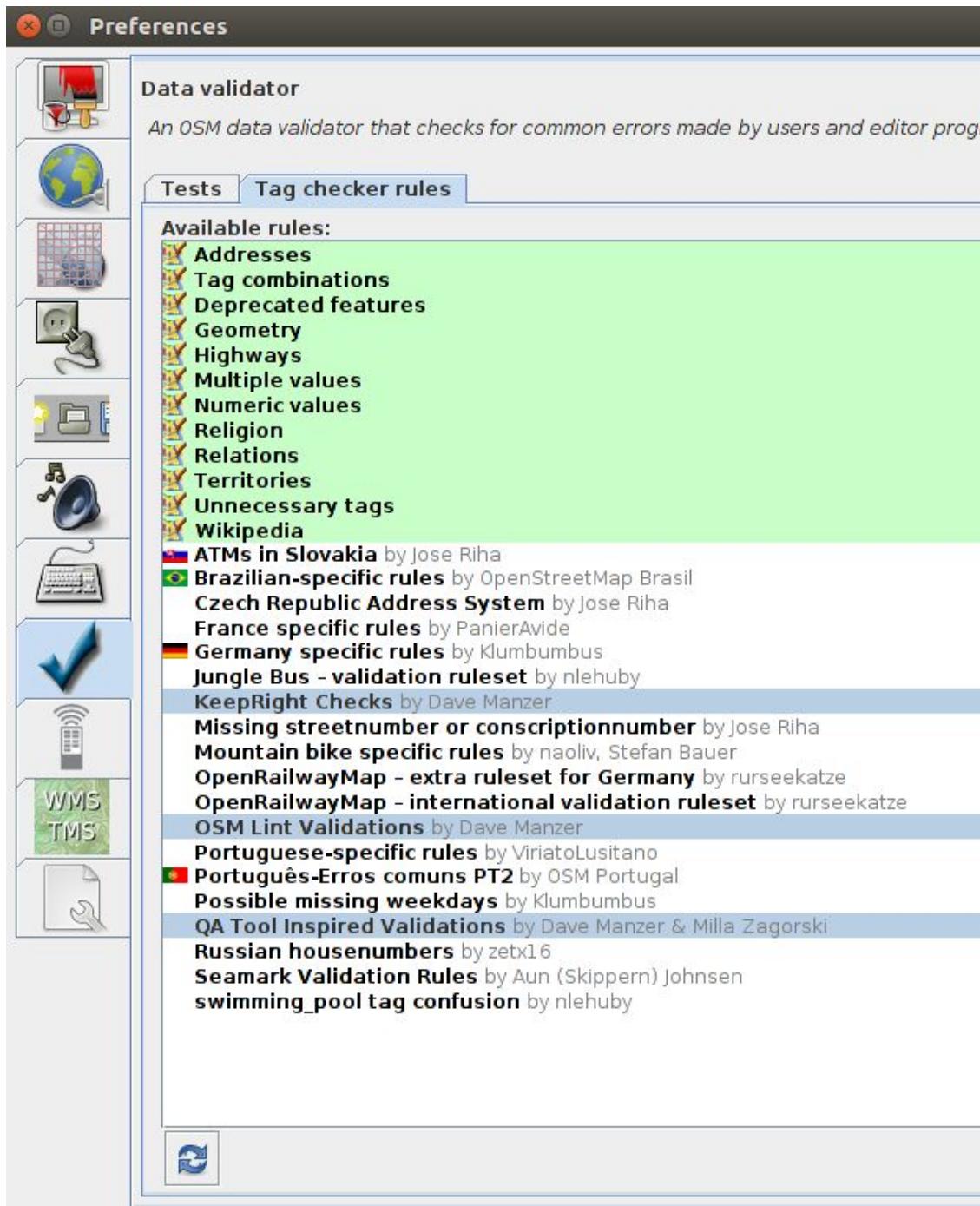
/* allowed junctions */
way[highway=service][access=no] node:connection,
way[highway=service][access=private] node:connection,
way[highway=service][service=parking_aisle] node:connection,
way[highway=unclassified][access=no] node:connection,
way[highway=unclassified][access=private] node:connection,
way[highway=unclassified][service=parking_aisle] node:connection,
way[highway=motorway_link] node:connection,
way[highway=motorway_junction] node:connection,
way[highway=construction] node:connection {
    set kr_270_allowed;
}

way[highway=motorway] node:connection.hwycon270!.kr_270_allowed!.kr_270_end_motorway {
    throwWarning: "270: motorways connected directly";
    group: "KeepRight";
}

/* KeepRight missing maxspeed 300 */
/* Message: missing maxspeed tag */
way[highway=~/(motorway|trunk|primary|secondary)$/][junction!=roundabout][!bridge][!tunnel][!maxspeed] {
    throwError: "300: missing maxspeed on major road";
    group: "KeepRight";
}
```

# Available JOSM Tag Checker Rules

- Osmose (QA Tool Inspired Validations)
- KeepRight
- OSM Lint



# JOSM Wiki Rules

<https://josm.openstreetmap.de/wiki/Rules>

## Create new rules

If you want to add your rules to this list use one of the two options:

- Simply create a page in this wiki with your rule, e.g. [Rules/MyRuleName](#). This is similar to [Styles](#) and [Presets](#). See there for more information about this topic.
- Externally hosted rule URLs can be entered when editing the [RulesSource](#) page.

If your rule applies to a limited area, use [Territory selectors](#), to limit the execution to the area.

## Rules managed in JOSM wiki

JOSM wiki based Rules is the preferred method.

- [Rules/AtmsSlovakia](#)
- [Rules/CzechRepublicAddressSystem](#)
- [Rules/FranceSpecificRules](#)
- [Rules/GermanySpecific](#)
- [Rules/KeepRight](#)
- [Rules/MissingStreetnumberOrConscriptionnumber](#)
- [Rules/MissingWeekdays](#)
- [Rules/OSMLint](#)
- [Rules/Portuguese-Errors-PT2](#)
- [Rules/Portuguese-Specific](#)
- [Rules/QAToolInspiredValidations](#)
- [Rules/SuspiciousSwimming\\_Pool](#)

## Osmose-QA : MapCSS, OpenData and Mapillary

Focus on MapCSS rules set as common rules validation. Explore integration of OpenData and Mapillary AI vision extracted data.

Room: Großer Hörsaal

*Frédéric Rodrigo (Makina Corpus)*

Monday, 09:30

At the beginning, Osmose-QA is a Quality Assurance tool pointing issues on map, but also able to compare external data set to OpenStreetMap.

[Back to schedule](#)

First talk part is about the new support of MapCSS rules in Osmose-QA. MapCSS language was primarily designed as way to style map objects, it look like the web styling language CSS. The base of MapCSS is to use selectors to filter objects. JOSM already use it for map styling and validator. This new implementation of MapCSS in Osmose-QA allows to write rules more easily. It also allow to reuse JOSM validator rules set. Next step would be to run Osmose rules in JOSM. It's a great opportunity to share rules across tools and projects. But come the question on how to share and manage this rules while considering tools particularities.

## What's behind JOSM?

Presentation of the JOSM development model: how is it made, by who? How can I help?

Room: Großer Hörsaal

*Vincent Privat*

Sunday, 14:00

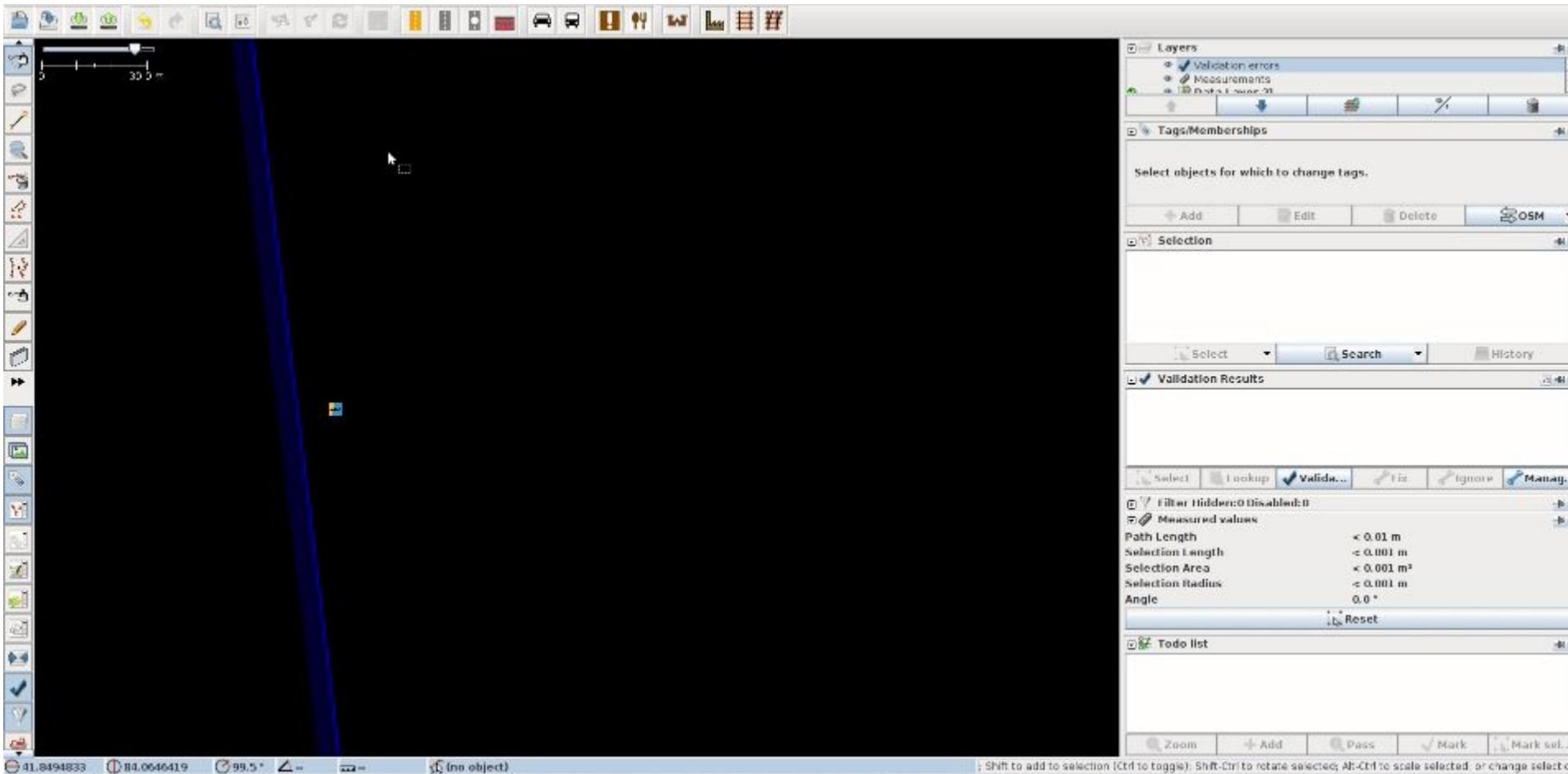
JOSM is almost as old as OSM but few people really know what it takes to maintain your preferred editor. We'll present the development model of JOSM and who's part of its active community: developers, translators, testers, plugin authors, end users, sponsors, etc. We'll talk about the project difficulties, the major achievements made in the past years, what work is currently in progress and what will happen in the near future!

[Back to schedule](#)

# JOSM Validator – Water Structures

Water Structures

Inspired by Osmose analyser: #1221 “Far from water”

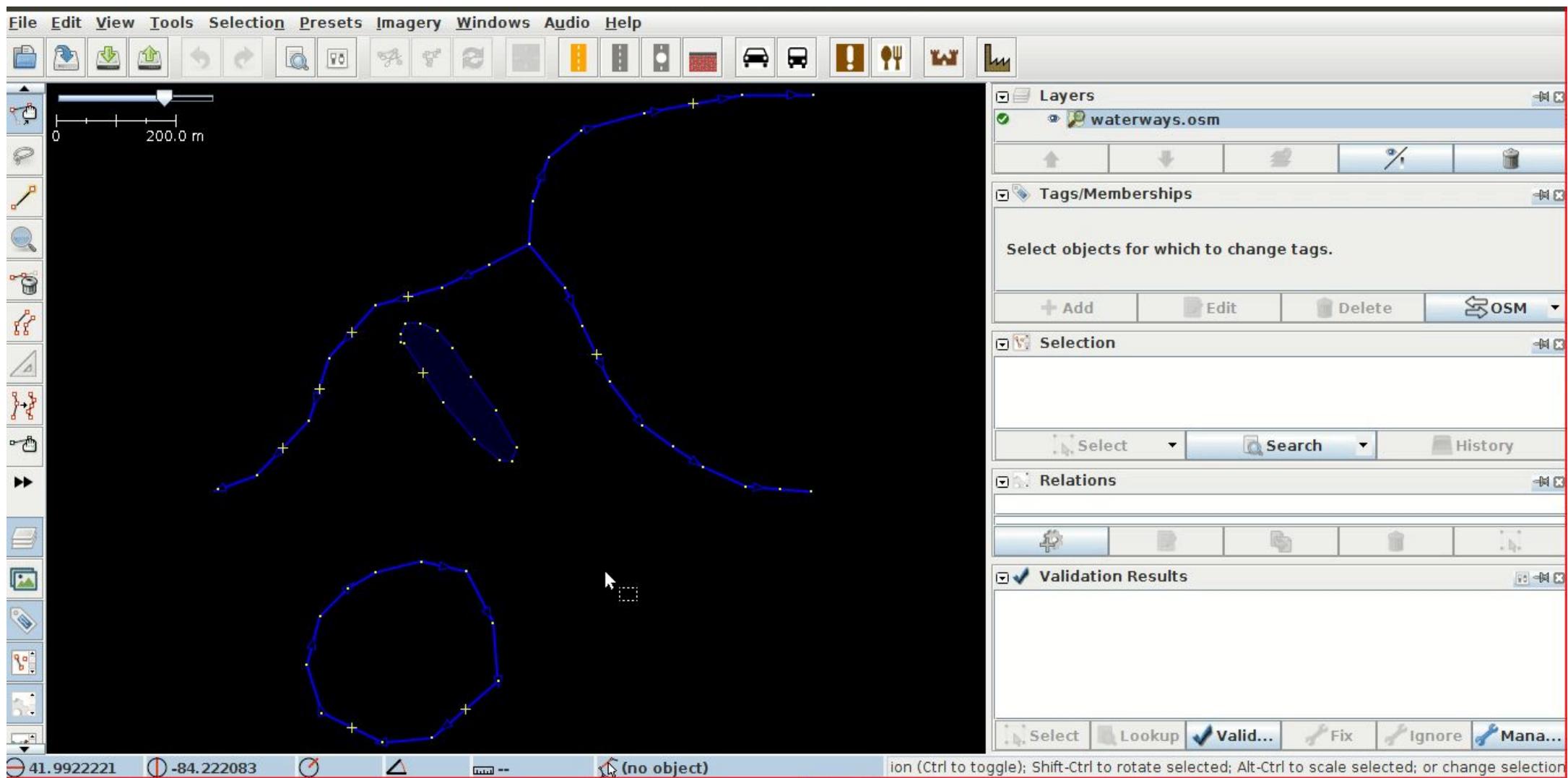


(Developed by: Milla Zagorski)

# JOSM Validator - Waterways

Waterways

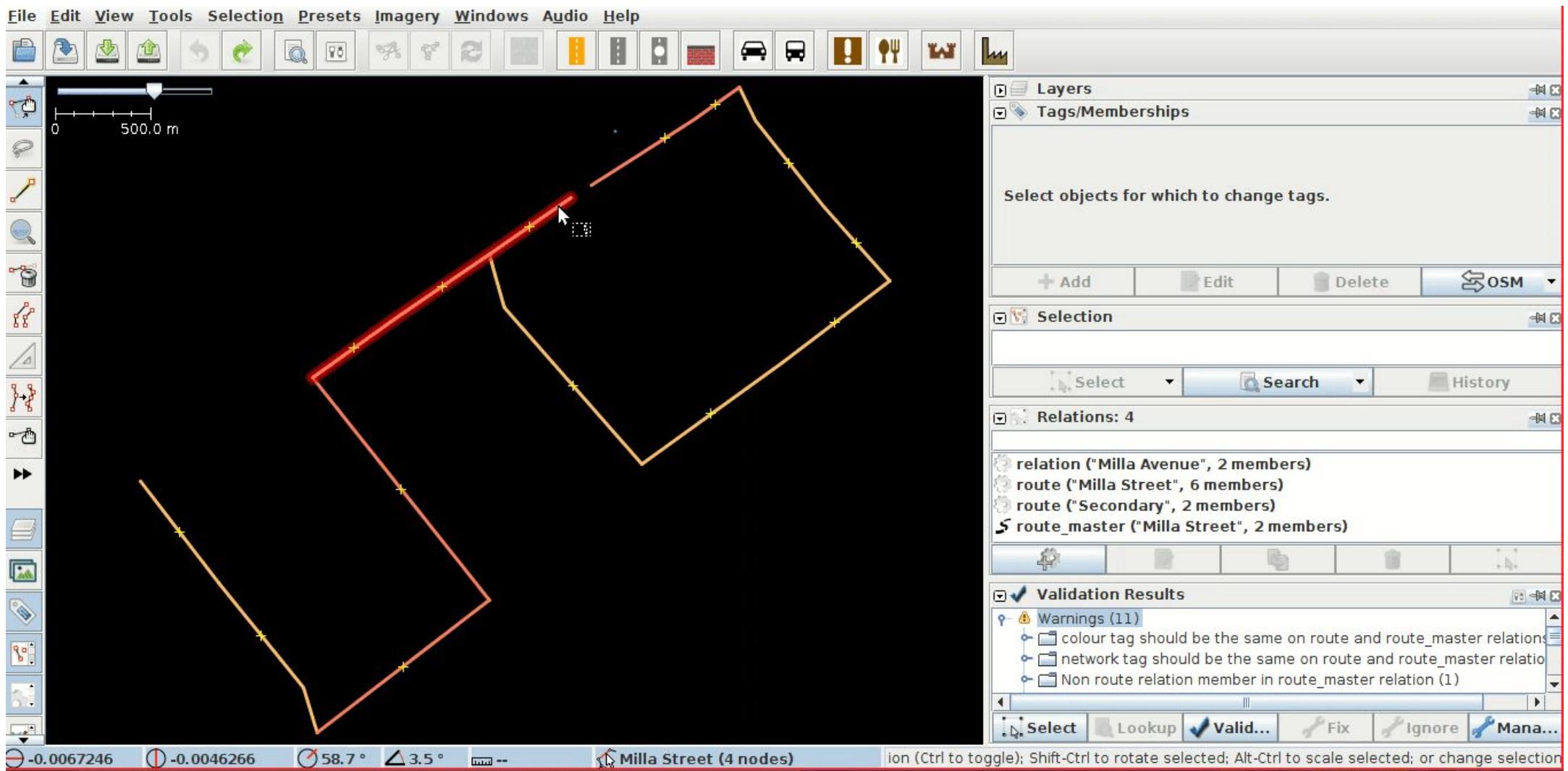
Inspired by Osmose analyser: #1220 “Waterway”



# JOSM Validator – Public Transport

Public Transport

Inspired by Osmose analyser: #1260 “Public Transport”



## Validation Results

- ⌚⚠️ Warnings (1)
  - ⌚ OSMOSE - 1210: Unconnected Highway (1)
- ⌚ℹ️ Other (5)
  - ⌚ KeepRight - 300: missing maxspeed on major road (1)
  - ⌚ OSM Lint - missing destination:ref tag (1)
  - ⌚ OSMOSE (2)
    - ⌚ 3120: natural=tree without species tag (1)
    - ⌚ 3120: natural=tree without type tag (1)
  - ⌚ Way looks crudely drawn according to Osmose algorithm (1)

# What's new in JOSM?

58 new validators!

- Osmose – 41
- KeepRight – 7
- OSM Lint – 9
- Atlas – 1

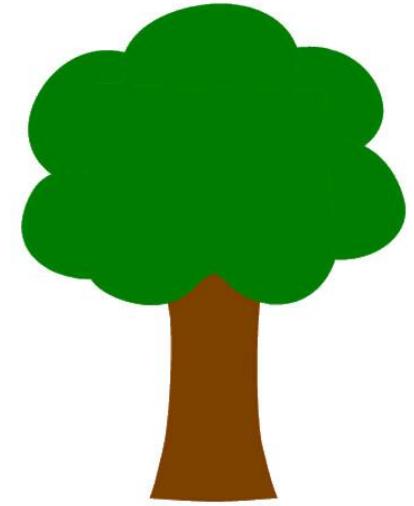
# Mainly addressing...



## highway issues

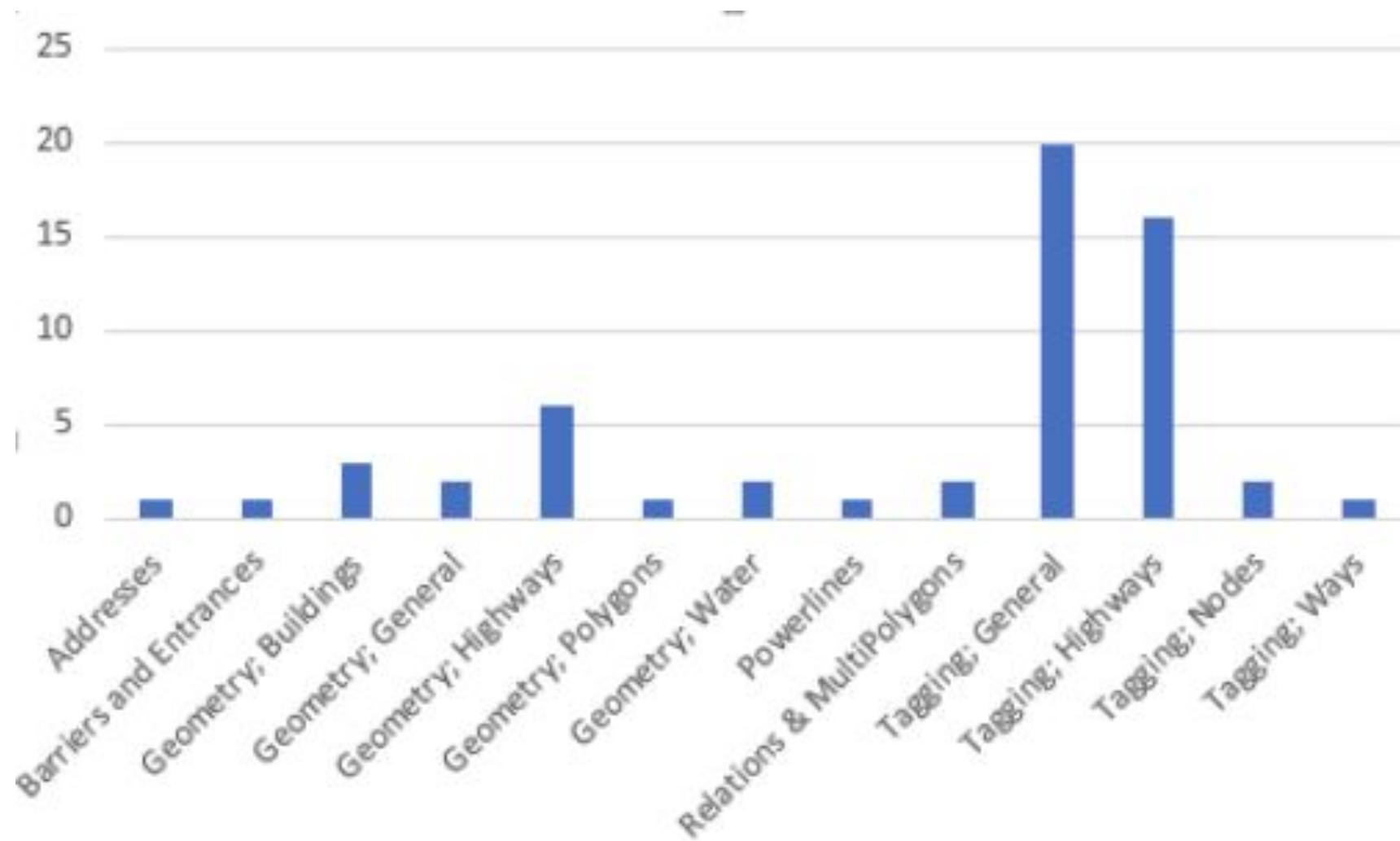


# tagging issues



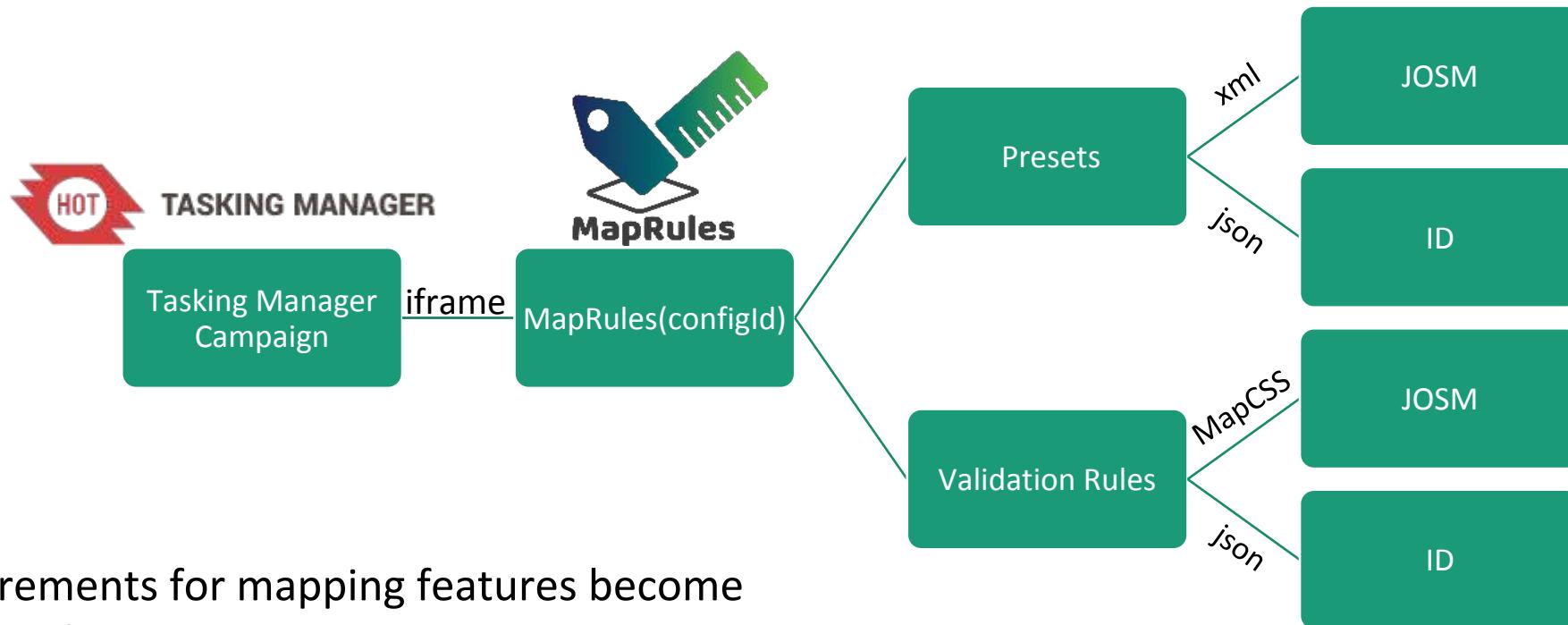
## tree-trimming issues

# A really sweet bar chart



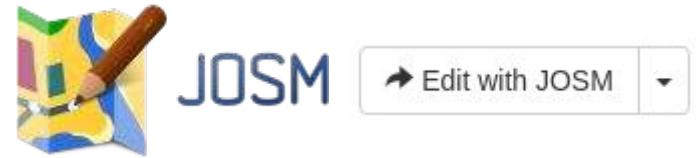
# Custom Presets and Validation Rules

Simple UI to create guidelines that can be tied to campaigns and generate presets and validation rules that can tailor the editors to your campaign/organization focus specific needs



Requirements for mapping features become

- Simplified
- Standardized
- Streamlined



# Creating Attribution Guidelines

## Presets

The screenshot shows the taginfo interface with two main sections: 'Identifying Tags' and 'Fields'.

**Identifying Tags:** A table with 'Key' and 'Value' columns. It contains one row: 'building' (Key) and 'hospital' (Value). Buttons for 'Add Identifying Tag' and 'Delete' are at the bottom right.

Key	Value
building	hospital

**Fields:** A table with 'Key Condition' and 'Value Condition' columns. It contains two rows:

Key Condition	Key	Value Condition	Value(s)
must have	name	Add...	Add...
may have	emergency	Add...	Add...

**Geometry:** Radio buttons for Area (selected), Line, and Point.

**Fields:** A section for defining key-value pairs. It includes dropdowns for 'Key Condition' (must have, may have), 'Key' (name, emergency), 'Value Condition' (Add...), and 'Value(s)' (Add...).

**Assisted with taginfo popular combinations**



# taginfo

hospital

Geometry

Area  Point

Identifying Tags

building = hospital

Required Tags

must have name

Suggested Tags

may have emergency

may have healthcare must be clinic doctor hospital

sports field

Geometry

Point  Area

Identifying Tags

leisure = pitch

Required Tags

must have name

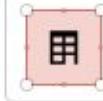
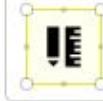
Suggested Tags

may have sport may be soccer american\_football

# Campaign Focused Presets

Select feature type

Search

-  Area
-  hospital
-  sports field
-  school

Tools

Point Line Area



# Realtime Validation

- Only show fields and values relevant to campaign
- Throw validation errors or suggestions for further information while editing

Edit feature

sports field

Zoom to this

Issues (2)

- ! sports field 'sports field' preset must include name
- ⚠ sports field 'sports field' preset may include sport

All fields

Name	i
Common name (if any)	+

Sport	i
...	▼
soccer	
american_football	

All tags (7)

leisure	pitch	trash	info
▼	▼		

Tools

Point

DigitalGlobe

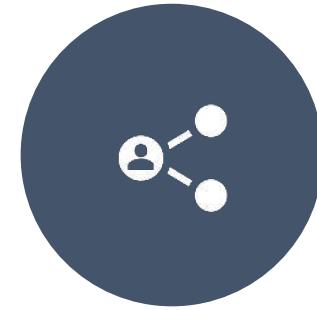
45 m

# In Progress

<https://github.com/radiant-maxar/maprules>



USER PRESETS



SHARING AND  
REUSING PRESETS



TASKING MANAGER

INTEGRATING INTO  
TASKING MANAGER 4

# Validation Based Overpass Queries

# Why?

- Stale errors in QA databases
  - Pulling in validation targeted data over larger areas
  - Bulk operations

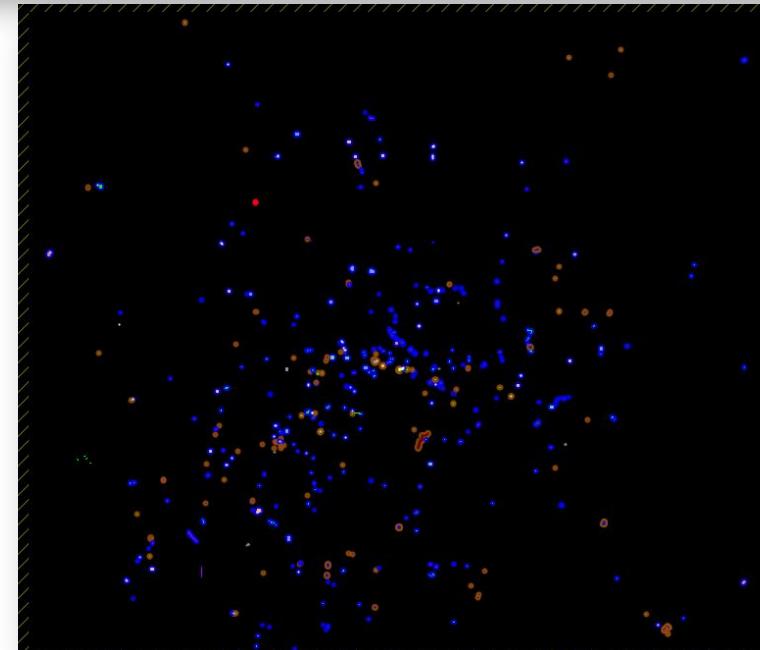
The screenshot shows the Overpass Turbo interface with the following components:

- Overpass query:** The query text is as follows:

```
//OSMLint: tunnel and bridge without layer
[out:json][bbox:{bbox}];

(
  way[bridge][!layer];
  way[tunnel][!layer];
);

out body;
>;
out skel qt;
```
- Your saved queries:** A list of saved queries on the right side, with "tunnel\_bridge\_without\_layer" highlighted in blue:
  - orphan\_nodes
  - tunnel\_maxheight
  - combinations
  - tunnel\_bridge\_without\_layer
  - buildings
  - place without name
- Map:** A map of the Twin Cities area, centered on Saint Paul. A large red rectangular area highlights the city of Saint Paul. Other labels visible include "Eau Claire" and a small green area labeled "DNR".
- Toolbar:** A bottom toolbar with icons for "lippy map", "Bookmarks", "Bounding Box", "Areas around places", and "Tile Numbers".



[Edit Layout](#)

Current layout: Task Completion \*

# MapRoulette Challenges

## GeoJSON

- Local File
- Remote Url

## Overpass Query

### Rebuild Challenge Tasks

Rebuilding will re-run the Overpass query and rebuild the challenge tasks with the latest data:

- Existing tasks included in the latest data will be updated
- New tasks will be added
- If you choose to first remove incomplete tasks (below), existing **incomplete** tasks will first be removed
- If you do not first remove incomplete tasks, they will be left as-is, possibly leaving tasks that have already been addressed outside of MapRoulette

**Warning: Rebuilding can lead to task duplication if your feature ids are not setup properly or if matching up old data with new data is unsuccessful. This operation cannot be undone!**

[Learn More](#) First remove incomplete tasks[Cancel](#)[Proceed](#)

## add layer to bridges/tunnels in Portugal ↗

wormitok's Project | [Contact Owner](#)

### INSTRUCTIONS

add a layer to the bridge: **minimum layer=1** ascending

add a layer to the tunnel: **minimum layer=-1** descending

### COMPLETION

[Add MR Tags](#)Current Editor: [Edit in iD \(web editor\)](#) \*[Edit](#)[Not an Issue](#)[Skip](#)[Other...](#)

### LOCATION



### Properties

osmid	243199808
bridge	yes
source	bing
highway	residential



# Overpass API/Overpass API by Example/QA Tool Overpass Queries

< Overpass API | Overpass API by Example  
(Redirected from Overpass API/Validation)



Validation Query	Description	Reference
Boundary Missing Source	An administrative boundary does not contain tag source=* sourcing the origin	OSMOSE 2040: Source
Building Road Intersection	Flags buildings that intersect/touch center lines of roads. This does not address cases where buildings get really close to roads, but do not overlap them.	Atlas: Building Road Intersection
Combinations	Missing tags on a given combination, suspicious tag combinations	JOSM: combinations
Deprecated Tags	Deprecated tagging that needs to be updated	JOSM: deprecated
Elevation	Some elements, including the peak ( <code>natural=peak</code> ) and mountain_pass ( <code>mountain_pass=yes</code> ), has an elevation. This is shown in OSM with tag <code>ele=*</code> in meters.	OSMOSE 2020: Elevation
fixme	Items tagged with fixme	KeepRight 170: fixme tagged items
Geometry	Incorrect geometry used or topological issues	JOSM: geometry
Highway	Highway validations	JOSM: highway
Missing Highway	<p>The way seems to be a <code>highway=*</code>.</p> <ul style="list-style-type: none"><li>• Class 20800 "Tag highway missing on junction" : the way have a tag <code>junction=*</code> but without <code>highway=*</code>.</li><li>• Class 20801 "Tag highway missing on oneway" : the way have a tag <code>oneway=*</code> but without <code>highway=*</code>.</li><li>• Class 20803 "Tag highway missing for tracktype or lanes" :</li></ul>	OSMOSE 2080: Missing Highway CLASS 20800, 20801, 20803, 20804
Missing Oneway on Motorlink	Detecting motorway links without oneway tag , which is connected to a highway=motorway with a oneway	OSMLint: missing oneway on motorway link
Orphan Nodes	Untagged nodes that do not belong to a way	
Place without Name	The tag <code>place=*</code> must always be used in combination with the tag <code>name=*</code> .	OSMOSE 6030: place without name
Powerlines	This analysis mainly verifies the continuity of powerlines <ul style="list-style-type: none"><li>• Class 1 "Lone power tower or pole" :</li></ul>	OSMOSE 7040: power lines Class 1
Relation	Relation validations	JOSM: relation
Tunnel and Bridge without Layer	This validator detects all the bridges that does not have the tag <code>layer=*</code> .	OSMLint: tunnel and bridge without layer
Tunnel/Bridge Max Height	Missing <code>maxheight=*</code> or maxheight: * for a tunnel or a way under a bridge.	OSMOSE 7130:Tunnel/Bridge
Two Names	tag <code>name=*</code> contains two street names, separated by a semicolon, a "/" or a "\". This issue was probably produced by the fusion of two way and the concatenation of the names of streets.	OSMOSE 5030 - Two names
Unnecessary	Features with unnecessary tags that can be removed	JOSM: unnecessary
Wrong Highway on Roundabout	Class 1 "Wrong highway on roundabout" : it must match the highest level of connected routes, except highway and trunk.	OSMOSE 3010: Class 1 wrong highway on roundabout
Wrong Railway Crossing	Checks level crossings if all ways have same <code>layer=*</code> and any way has tag <code>bridge=*</code> or <code>tunnel=*</code> .	KeepRight 160: Wrongly Used Railway Crossing Tag



## Overpass API/Overpass API by Example

### 4 Quality Assurance

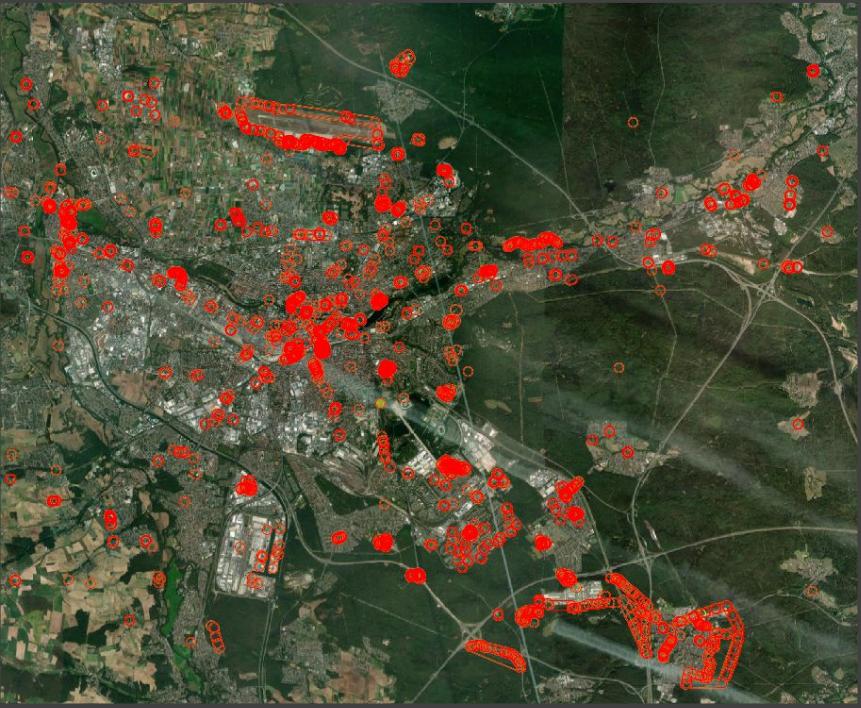
- 4.1 Checking boundary relations for correct tagging
- 4.2 Find all bus stops which are not included in a relation
- 4.3 Bus relations, which are not part of a master relation
- 4.4 Counties without fire station
- 4.5 Buildings crossed by roads that are not tagged as tunnel or covered
- 4.6 Find duplicate nodes (since 0.7.54)
  - 4.6.1 Find duplicate nearby nodes with same tag
- 4.7 Search for untagged ways
- 4.8 Orphaned nodes (nodes without tags, not part of any way or relation)
- 4.9 Find broken associatedStreet-relations
- 4.10 Find house numbers without street
- 4.11 Non-matching addr:postcode=XXXXX tags inside a boundary relation with postal code ≠ XXXXX
- 4.12 Non-matching addr:postcode=XXXXX tags inside multiple boundary relation with postal code ≠ XXXXX (since 0.7.54)
- 4.13 Ways that are part of a "Bundesstraßenrelation", but do not have a ref-tag
- 4.14 POIs with websites but with incorrect opening hours

## Overpass turbo/Examples

### Contents [hide]

- 1 Complete route relation
- 2 Part of a (route) relation
- 3 Power substations without proper voltage tag
- 4 Living streets with a potentially wrong speed limit
- 5 Highways with lanes=1 but no other explaining tag
- 6 Nodes that only have a name tag
- 7 Bridges without maxweight=...
- 8 Tunnels without maxheight=...
- 9 More
  - 9.1 Overpass API by Example / DE:Overpass API/Beispielsammlung
  - 9.2 Quality assurance of postal codes
  - 9.3 Soldier Boy's Examples
  - 9.4 Parking areas example
  - 9.5 More various Overpass examples
  - 9.6 Public transport Quality Assurance
  - 9.7 Overpass Queries based on Quality Assurance Tools

# Automatically Fix Validation Errors



Tags: 26 / Memberships: 0

Man Made/Man Made/Residential Building ...  
Annotation/Address ...

Key	Value
addr:city	Nürnberg
addr:country	DE
addr:housenumber	12
addr:postcode	90475
addr:street	Habsburgerstraße
amenity	hotel
angle	242.8
beds	9
building	hotel
feature_area	208.2
internet_access	wlan

+ Add Edit Delete OSM

Validation Results

- deprecated tagging (595)
  - amenity=advertising is deprecated, use advertising=\* instead (6)
  - amenity=artwork is deprecated, use tourism=artwork instead (1)
  - amenity=education is deprecated, use amenity=college or amenity=school or a
  - amenity=hotel is deprecated, use tourism=hotel instead (1)
    - Avenue Altenfurt (6 nodes)
  - amenity=kiosk is deprecated, use shop=kiosk instead (1)
  - amenity=nursery is deprecated, use amenity=childcare or amenity=kindergarte
  - amenity=preschool is deprecated, use amenity=kindergarten instead (1)
  - amenity=public\_building is deprecated, use amenity=townhall or amenity=comm
  - amenity=sauna is deprecated, use leisure=sauna instead (24)

Select Lookup Validation Fix Ignore Manage...

```
*[amenity=hotel] {
    throwWarning: tr("{0} is deprecated", "{0.tag}");
    suggestAlternative: "tourism=hotel";
    group: tr("deprecated tagging");
    fixAdd: "tourism=hotel";
    fixRemove: "amenity";
}
```

# Why does this matter?

- Custom validations give users and groups the directed focus they need.
  - If you're only interested in mapping railways, why not share your `railway=switch` validation MapCSS checker?
- Thematic validations can assist with directed editing and cleanup

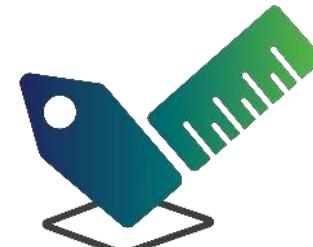
# Create and Share Validation Rules



ADD TO MAPCSS  
VALIDATION RULES



ADD TO JOSM  
VALIDATORS



MapRules

CREATE MAPRULES

clarisse.abalos@maxar.com  
@map2save

matthew.gibb@maxar.co  
m  
@giblet22

# Questions