

```
import ogr2osm
```

```
class Dedicated_IPA_2020_tagsTranslation(ogr2osm.TranslationBase):
```

```
    def filter_tags(self, attrs):
```

```
        if not attrs:
```

```
            return
```

```
        tags = {}
```

```
        if "NAME" in attrs:
```

```
            tags.update({"name": attrs["NAME"].strip(" ") + " " + attrs["TYPE"].strip(" ")})
```

```
        if "ENVIRON" in attrs:
```

```
            if attrs["ENVIRON"].strip() == "M":
```

```
                tags.update({"name": attrs["NAME"].strip(" ") + " " + attrs["TYPE"].strip(" ") + " sea country"})
```

```
        if "IPA_ID" in attrs:
```

```
            tags.update({"ref:DedicatedIPA2020:pa_id": "CWITH_IPA" + attrs["IPA_ID"].strip(" ")})
```

```
        if "IUCN_CODE" in attrs:
```

```
            tags.update({"iucn_level": attrs["IUCN_CODE"].replace(", ", ";").strip(" ")})
```

```
        if "IUCN_CODE" in attrs:
```

```
            if attrs["IUCN_CODE"].strip() == "II":
```

```
                tags["protect_class"] = "2"
```

```
            elif attrs["IUCN_CODE"].strip() == "II,VI":
```

```
                tags["protect_class"] = "2;6"
```

```
            elif attrs["IUCN_CODE"].strip() == "III,IV,V":
```

```
                tags["protect_class"] = "3;4;5"
```

```
            elif attrs["IUCN_CODE"].strip() == "III,IV,V,VI":
```

```
                tags["protect_class"] = "3;4;5;6"
```

```
            elif attrs["IUCN_CODE"].strip() == "III,IV,VI":
```

```
                tags["protect_class"] = "3;4;6"
```

```
            elif attrs["IUCN_CODE"].strip() == "III,V":
```

```
                tags["protect_class"] = "3;5"
```

```
            elif attrs["IUCN_CODE"].strip() == "III,VI":
```

```
                tags["protect_class"] = "3;6"
```

```
            elif attrs["IUCN_CODE"].strip() == "IV":
```

```
                tags["protect_class"] = "4"
```

```
            elif attrs["IUCN_CODE"].strip() == "IV,V":
```

```
                tags["protect_class"] = "4;5"
```

```
            elif attrs["IUCN_CODE"].strip() == "IV,VI":
```

```
                tags["protect_class"] = "4;6"
```

```
            elif attrs["IUCN_CODE"].strip() == "V":
```

```
                tags["protect_class"] = "5"
```

```
            elif attrs["IUCN_CODE"].strip() == "V,VI":
```

```
                tags["protect_class"] = "5;6"
```

```
            elif attrs["IUCN_CODE"].strip() == "VI":
```

```
                tags["protect_class"] = "6"
```

```
        if "AUTHORITY" in attrs:
```

```
            if attrs["AUTHORITY"].strip() == "IMG":
```

```
                tags["operator"] = "Indigenous Management Group"
```

```
            elif attrs["AUTHORITY"].strip() == "LILC":
```

```
                tags["operator"] = "Local Indigenous Land Council"
```

```
            elif attrs["AUTHORITY"].strip() == "TSRA":
```

```
                tags["operator"] = "Torres Strait Regional Authority"
```

```
        tags["leisure"] = "nature_reserve"
```

```
        tags["access"] = "private"
```

```
        tags["source"] = "Dedicated Indigenous Protected Areas 2020"
```

```
tags["boundary"] = "protected_area"
```

```
tags["type"] = "boundary"
```

```
return tags
```

```
'''
```

type=boundary tag needs deleting for non multipolygons.

I will delete this tag from non-multipolygons using a JOSM search for 'type=way type=boundary' and delete the type=boundary tag.

```
'''
```