Wikidata Analysis

T282790

Vertical data:

- All non-English labels
- All labels that are covered by fallbacks (names duplicated across 200 languages etc)
- All labels
- all descriptions
- All aliases
- All labels, description and aliases
- External identifiers

Set up Spark

In [2]:
```
import wmfdata
import time
import matplotlib.pyplot as plt
from matplotlib.pyplot import text
import numpy as np
import pandas as pd
from statistics import mean
```

In [4]:
```
spark = wmfdata.spark.get_session(type='yarn-large', app_name="wdqs-analysis") ##ship_python_env=False
```
```
PySpark executors will use /usr/lib/anaconda-wmf/bin/python3.
```

What are all the label/description/aliases properties?

In [3]:
```
props = ["<http://www.w3.org/2000/01/rdf-schema#label>", #rdfs
         "<http://schema.org/description>", #schema
         "<http://www.w3.org/2004/02/skos/core#altLabel>"] #skos
propsStr = "('"+ "','".join(props) +"')"
```

Total triples

In [3]:
```
# date = "20210712"
date = "20210719"
```

In [5]:
```
total_triples = spark.sql(f"Select count(distinct subject, predicate, object) as count from discovery.wikibase_rdf where date={date} and wiki='wikidata'")\
                    .collect()[0]['count']
```

In [107]:
```
total_triples
```
Out[107]:
```
12671768950
```

WD Growth rate

See Grafana Board

In [1]:
```
stats_start = [12276861067, 12276834059, 12276878537, 12276911948, 12276878947, 12276825993] # 6-6-21
stats_end = [12692176657, 12692306226, 12692317874, 12692317904, 12692349229, 12692308900] # 11-3-21
```

In [8]:
```
## growth rate percentage
[(end-start)*100/start for start, end in zip(stats_start, stats_end)]
```
Out[8]:
```
[3.382913496645828,
 3.384196324584369,
 3.3839166507019747,
 3.3836355409201473,
 3.3841685968690363,
 3.3842860299307005]
```

In [14]:
```
## growth per day
days = 20 + 30 + 31 + 6
rates = [(end-start)/days for start, end in zip(stats_start, stats_end)]
rates, mean(rates)
```
Out[14]:
```
([4773742.413793104,
  4775542.149425288,
  4775164.793103448,
  4774781.1034482755,
  4775520.482758621,
  4775665.597701149],
 4775069.4233716475)
```

In [23]:
```
total_triples = 12819818340 #20210719

def how_many_days(num, rate=4770000, total=total_triples):
    print(round(num/rate, 3), "days", round(num*100.0/total, 3), "% of triples")
```

Labels

How many label triples?

In [7]:
```
label_count = spark.sql(f"""Select count(distinct subject, predicate, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{props[0]}'""").collect()[0]['count']
label_count, (label_count/total_triples)*100
```
Out[7]:
```
(499663174, 3.9431209326145424)
```

In [121]:
```
how_many_days(label_count)
```
```
104.751 days 3.943 % of triples
```

How many non-English label triples?

In [8]:
```
ne_label_count = spark.sql(f"""Select count(distinct subject, predicate, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{props[0]}' and object not like '%en' """)\
                        .collect()[0]['count']
ne_label_count, (ne_label_count/total_triples)*100
```
Out[8]:
```
(419885045, 3.313547198159733)
```

In [123]:
```
how_many_days(ne_label_count)
```
```
88.026 days 3.314 % of triples
```

English labels

In [9]:
```
e_label_count = spark.sql(f"""Select count(distinct subject, predicate, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{props[0]}' and object like '%en' """)\
                        .collect()[0]['count']
e_label_count, (e_label_count/total_triples)*100
```
Out[9]:
```
(79778129, 0.6295737344548095)
```

In [124]:
```
how_many_days(e_label_count)
```
```
16.725 days 0.63 % of triples
```

Number of labels per item

In [10]:
```
## takes time
spark.sql(f"""
        select count(distinct subject) as num_item_that_have_a_label, sum(label_count) as total_labels,
        avg(label_count), max(label_count), min(label_count),
        sum(CASE WHEN label_count=1 THEN 1 ELSE 0 END) as one_label_count
        from (
        select subject, count(distinct subject, predicate, object) as label_count
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[0]}'
        group by subject
        ) as intermediate_table
        """)\
    .show()
```
```
+--------------------------+------------+-----------------+----------------+----------------+---------------+
|num_item_that_have_a_label|total_labels| avg(label_count)|max(label_count)|min(label_count)|one_label_count|
+--------------------------+------------+-----------------+----------------+----------------+---------------+
|                  93474062|   499663174|5.345474063168454|             446|               1|       20084825|
+--------------------------+------------+-----------------+----------------+----------------+---------------+
```

In [125]:
```
num_item_that_have_a_label = 93474062
one_label_item = 20084825
items_with_more_than_one_label = num_item_that_have_a_label - one_label_item
items_with_more_than_one_label
```
Out[125]:
```
73389237
```

In [126]:
```
one_label_item/num_item_that_have_a_label, items_with_more_than_one_label/num_item_that_have_a_label
```
Out[126]:
```
(0.21487057019090494, 0.7851294298090951)
```

Which items have that many labels?

In [ ]:
```
max_value = 446

spark.sql(f"""
        select subject
        from (
        select count(distinct subject, predicate, object) as label_count, subject
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[0]}'
        group by subject
        ) as intermediate_table
        where label_count = {max_value}
        limit 10
        """)\
    .show(10, truncate=False)
```

In [37]:
```
value = 1
spark.sql(f"""
        select subject
        from (
        select count(distinct subject, predicate, object) as label_count, subject
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[0]}'
        group by subject
        ) as intermediate_table
        where label_count = {value}
        limit 10
        """)\
    .show(10, truncate=False)
```
```
+------------------------------------------+
|subject                                   |
+------------------------------------------+
|<http://www.wikidata.org/entity/Q30010331>|
|<http://www.wikidata.org/entity/Q94069522>|
|<http://www.wikidata.org/entity/Q61276028>|
|<http://www.wikidata.org/entity/Q49996258>|
|<http://www.wikidata.org/entity/Q41070541>|
|<http://www.wikidata.org/entity/Q82880440>|
|<http://www.wikidata.org/entity/Q88387788>|
|<http://www.wikidata.org/entity/Q88104623>|
|<http://www.wikidata.org/entity/Q27452862>|
|<http://www.wikidata.org/entity/Q32035066>|
+------------------------------------------+
```

How many don't have any english label? i.e have only other language labels

Meaning if we remove the other language labels, it won't have any label at all.

In [11]:
```
spark.sql(f"""
        select count(distinct subject) as num_item_that_have_a_english_label
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[0]}' and object like '%en'
        """)\
    .show()
```
```
+----------------------------------+
|num_item_that_have_a_english_label|
+----------------------------------+
|                          79778129|
+----------------------------------+
```

In [127]:
```
num_item_that_have_a_english_label = 79778129
num_item_that_have_only_other_language_labels = num_item_that_have_a_label - num_item_that_have_a_english_label

num_item_that_have_only_other_language_labels, num_item_that_have_only_other_language_labels/num_item_that_have_a_label
```
Out[127]:
```
(13695933, 0.14652121355333847)
```

Distribution of language tags

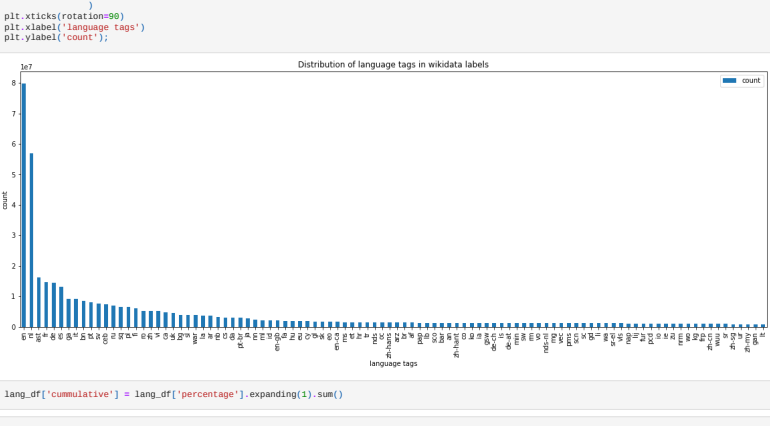
In [12]:
```
lang_df = spark.sql(f"""
        select lang_tag, count(lang_tag) as count from
        (select distinct subject, SUBSTRING(object, length(object) - locate('@',reverse(object)) + 2, length(object)) as lang_tag
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[0]}') as lang_tag_table
        group by lang_tag
        """)\
    .toPandas()
```

In [13]:
```
lang_df['percentage'] = (lang_df['count']*100/lang_df['count'].sum()).round(decimals=2)
lang_df = lang_df.sort_values("count", ascending=False)
lang_df.head(20)
```
Out[13]:

	lang_tag	count	percentage
252	en	79778129	15.97
224	nl	56940665	11.40
154	ast	16106324	3.22
214	fr	14594937	2.92
87	de	14352435	2.87
143	es	13005130	2.60
389	ga	9162180	1.83
144	it	9090037	1.82
256	bn	8531392	1.71
335	pt	7966495	1.59
354	sv	7641820	1.53
272	ceb	7295129	1.46
25	ru	6942345	1.39
229	sq	6593056	1.32
62	pl	6572662	1.32
68	fi	6126639	1.23
19	ro	5149707	1.03
364	zh	5128715	1.03
245	vi	5080149	1.02
32	ca	4648876	0.93

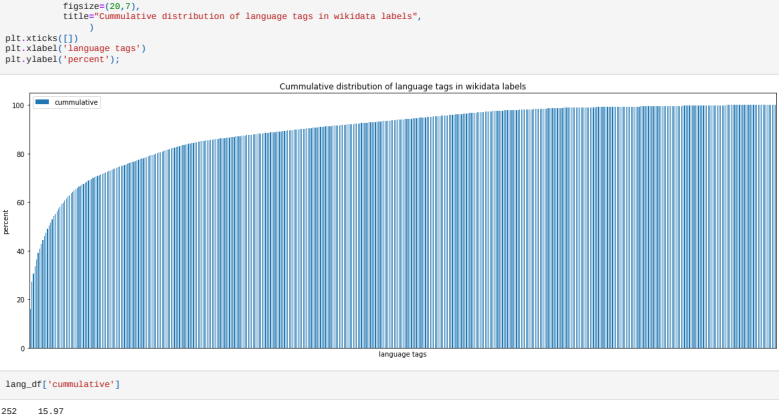
In [14]:
```
len(lang_df)
```
Out[14]:
```
476
```

In [15]:
```
lang_df.head(100).plot(x="lang_tag", y="count",
                kind='bar',
                figsize=(20,7),
                title="Distribution of language tags in wikidata labels",
                )
plt.xticks(rotation=90)
plt.xlabel('language tags')
plt.ylabel('count');
```



In [16]:
```
lang_df['cumulative'] = lang_df['percentage'].expanding(1).sum()
```

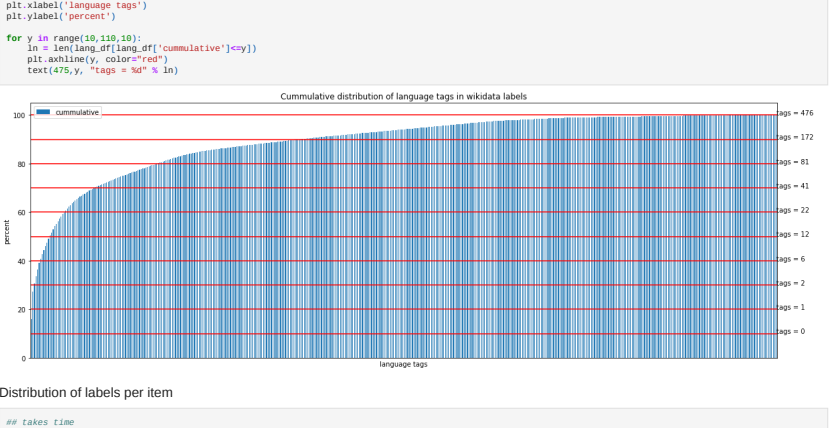
In [17]:
```
lang_df.plot(y="cumulative",
             kind='bar',
             figsize=(20,7),
             title="Cummulative distribution of language tags in wikidata labels",
             )
plt.xticks([])
plt.xlabel('language tags')
plt.ylabel('percent')
```



In [18]:
```
lang_df['cumulative']
```
Out[18]:
```
252    15.97
224    27.37
154    30.59
214    33.51
87     36.38
       ...
342    99.97
137    99.97
298    99.97
206    99.97
8      99.97
Name: cumulative, Length: 476, dtype: float64
```

In [19]:
```
for y in range(10,110,10):
    ln = len(lang_df[lang_df['cumulative']<=y])
    print("Number of language tags under {}% is {}".format(y,ln))
```
```
Number of language tags under 10% is 0
Number of language tags under 20% is 1
Number of language tags under 30% is 2
Number of language tags under 40% is 6
Number of language tags under 50% is 12
Number of language tags under 60% is 22
Number of language tags under 70% is 41
Number of language tags under 80% is 91
Number of language tags under 90% is 172
Number of language tags under 100% is 476
```

In [20]:
```
lang_df.plot(y="cumulative",
             kind='bar',
             figsize=(20,7),
             title="Cummulative distribution of language tags in wikidata labels",
             )
plt.xticks([])
plt.xlabel('language tags')
plt.ylabel('percent')

for y in range(10,110,10):
    ln = len(lang_df[lang_df['cumulative']<=y])
    plt.axhline(y, color="red")
    text(475,y, "tags = %d" % ln)
```



Distribution of labels per item

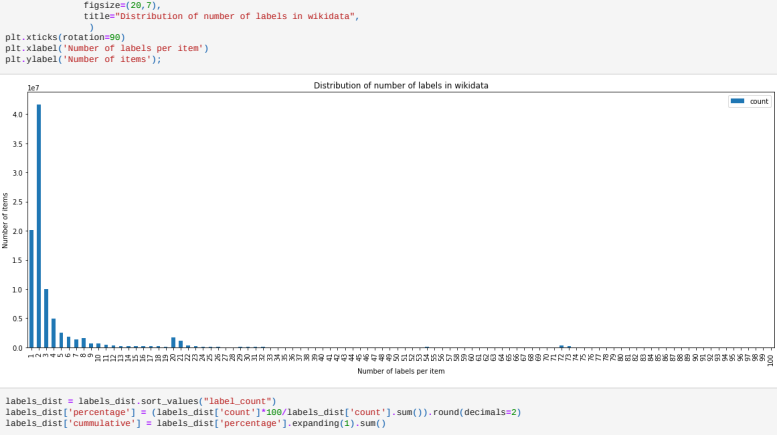
In [21]:
```
## takes time
labels_dist = spark.sql(f"""
        select label_count, count(1) as count
        from (select count(distinct subject, object) as label_count
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[0]}'
        group by subject) as intermediate_table
        group by label_count
        """).toPandas()
```

In [22]:
```
labels_dist = labels_dist.sort_values("count", ascending=False)
```

In [23]:
```
labels_dist.head()
```
Out[23]:

	label_count	count
93	2	41697507
104	1	20084825
316	3	10030890
50	4	4988361
325	5	2568004

In [24]:
```
labels_dist.sort_values("label_count")\
        .head(100)\
        .plot(x="label_count", y="count",
            kind='bar',
            figsize=(20,7),
            title="Distribution of number of labels in wikidata",
            )
plt.xticks(rotation=90)
plt.xlabel('Number of labels per item')
plt.ylabel('Number of items');
```

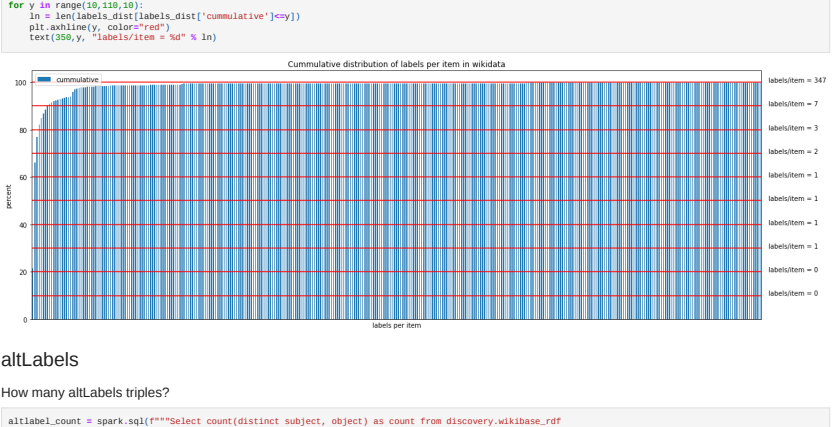


In [25]:
```
labels_dist = labels_dist.sort_values("label_count")
labels_dist['percentage'] = (labels_dist['count']*100/labels_dist['count'].sum()).round(decimals=2)
labels_dist['cummulative'] = labels_dist['percentage'].expanding(1).sum()

labels_dist.head(20)
```
Out[25]:

	label_count	count	percentage	cummulative
104	1	20084825	21.49	21.49
93	2	41697507	44.61	66.10
316	3	10030890	10.73	76.83
50	4	4988361	5.34	82.17
325	5	2568004	2.75	84.92
216	6	1857881	1.99	86.91
265	7	1368083	1.46	88.37
53	8	1592460	1.70	90.07
257	9	683102	0.73	90.80
254	10	731273	0.78	91.58
72	11	416371	0.45	92.03
96	12	348728	0.37	92.40
125	13	277841	0.30	92.70
332	14	274199	0.29	92.99
58	15	202929	0.22	93.21
256	16	194319	0.21	93.42
64	17	185169	0.20	93.62
342	18	168883	0.18	93.80
285	19	156680	0.17	93.97
85	20	1720307	1.84	95.81

In [26]:
```
## Avg labels per item
## matches previously found avg
(labels_dist['label_count']*labels_dist['count']).sum() / labels_dist['count'].sum()
```
Out[26]:
```
5.345474083168454
```

In [27]:
```
labels_dist.plot(y="cumulative",
             kind='bar',
             figsize=(20,7),
             title="Cummulative distribution of labels per item in wikidata",
             )
plt.xticks([])
plt.xlabel('labels per item')
plt.ylabel('percent')

for y in range(10,110,10):
    ln = len(labels_dist[labels_dist['cummulative']<=y])
    plt.axhline(y, color="red")
    text(350,y, 'labels/item = %d' % ln)
```



altLabels

How many altLabels triples?

In [108]:
```
altlabel_count = spark.sql(f"""Select count(distinct subject, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{props[2]}' """)\
                        .collect()[0]['count']
altlabel_count, (altlabel_count/total_triples)*100
```
Out[108]:
```
(102593854, 0.8096253522678063)
```

In [109]:
```
how_many_days(altlabel_count)
```
```
21.508 days 0.81 % of triples
```

In [20]:
```
l_p, alt_p, both_p = label_count/total_triples, altlabel_count/total_triples, (altlabel_count+label_count)/total_triples
print("Percent of label triples",l_p*100)
print("Percent of altlabel triples",alt_p*100)
print("Percent of both labels triples",both_p*100)
```
```
Percent of label triples 3.9431209326145424
Percent of altlabel triples 0.8096253522678063
Percent of both labels triples 4.752746284882348
```

See some alt labels

In [30]:
```
spark.sql(f"""Select *
        from discovery.wikibase_rdf
        where date={date} and wiki='wikidata' and
        predicate='{props[2]}'
        limit 10""")\
    .show(10, truncate=False)
```
```
+------------------------------------------+------------------------------------------+----------------------------------------------+------------------------------
|context                                   |subject                                   |predicate                                     |object
|date    |wiki    |
+------------------------------------------+------------------------------------------+----------------------------------------------+------------------------------
|<http://www.wikidata.org/entity/Q66237906>|<http://www.wikidata.org/entity/Q66237906>|<http://www.w3.org/2004/02/skos/core#altLabel>|"\uC7D0\uB3E7\uB77C\uC0D
4"@ko                                    |20210517|wikidata|
|<http://www.wikidata.org/entity/Q51845726>|<http://www.wikidata.org/entity/Q51845726>|<http://www.w3.org/2004/02/skos/core#altLabel>|"\u306F\u3088\u304E"@ood
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q80807931>|<http://www.wikidata.org/entity/Q80807931>|<http://www.w3.org/2004/02/skos/core#altLabel>|"Rottschalk"@scc
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q51133609>|<http://www.wikidata.org/entity/Q51133609>|<http://www.w3.org/2004/02/skos/core#altLabel>|"A. Depla"@es
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q51177560>|<http://www.wikidata.org/entity/Q51177560>|<http://www.w3.org/2004/02/skos/core#altLabel>|"Andreini"@ky
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q51881250>|<http://www.wikidata.org/entity/Q51881250>|<http://www.w3.org/2004/02/skos/core#altLabel>|"serine/arginine-rich sp
licing factor SR45-like"@en|20210517|wikidata|
|<http://www.wikidata.org/entity/Q51845796>|<http://www.wikidata.org/entity/Q51845796>|<http://www.w3.org/2004/02/skos/core#altLabel>|"\u6642\u91CE\uDE37"@pt
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q51635083>|<http://www.wikidata.org/entity/Q51635083>|<http://www.w3.org/2004/02/skos/core#altLabel>|"\u307F\u3054\u3082\u308
8"@en                                    |20210517|wikidata|
|<http://www.wikidata.org/entity/Q51881927>|<http://www.wikidata.org/entity/Q51881927>|<http://www.w3.org/2004/02/skos/core#altLabel>|"Sebestyn\u00E9n"@luz
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q51845784>|<http://www.wikidata.org/entity/Q51845784>|<http://www.w3.org/2004/02/skos/core#altLabel>|"\u3067\u3070\u3068"@ch
|20210517|wikidata|
+------------------------------------------+------------------------------------------+----------------------------------------------+------------------------------
```

How many non-English altlabel triples?

In [30]:
```
ne_alt_label_count = spark.sql(f"""Select count(distinct subject, object) as count
                        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
                        predicate = '{props[2]}' and object not like '%en' """)\
                        .collect()[0]['count']
ne_alt_label_count, (ne_alt_label_count/total_triples)*100
```
Out[30]:
```
(93436016, 0.7373628555163295)
```

In [110]:
```
how_many_days(ne_alt_label_count)
```
```
19.588 days 0.737 % of triples
```

English altlabels

In [31]:
```
e_alt_label_count = spark.sql(f"""Select count(distinct subject, object) as count
                        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
                        predicate = '{props[2]}' and object like '%en' """)\
                        .collect()[0]['count']
e_alt_label_count, (e_alt_label_count/total_triples)*100
```
Out[31]:
```
(9157838, 0.07226329675147683)
```

In [111]:
```
how_many_days(e_alt_label_count)
```
```
1.92 days 0.072 % of triples
```

Number of altlabels per item

In [32]:
```
## takes time
spark.sql(f"""
        select count(distinct subject) as num_item_that_have_a_label, sum(label_count) as total_labels,
        avg(label_count), max(label_count), min(label_count),
        sum(CASE WHEN label_count=1 THEN 1 ELSE 0 END) as one_label_count
        from (
        select subject, count(distinct subject, predicate, object) as label_count
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[2]}'
        group by subject
        ) as intermediate_table
        """)\
    .show()
```
```
+--------------------------+------------+------------------+----------------+----------------+---------------+
|num_item_that_have_a_label|total_labels|  avg(label_count)|max(label_count)|min(label_count)|one_label_count|
+--------------------------+------------+------------------+----------------+----------------+---------------+
|                   9924856|   102593854|10.336915685163414|            3879|               1|        4855544|
+--------------------------+------------+------------------+----------------+----------------+---------------+
```

In [114]:
```
num_item_that_have_a_altlabel = 9924856
one_altlabel_item = 4855544
items_with_more_than_one_altlabel = num_item_that_have_a_altlabel - one_altlabel_item

print("Number of items with more than one altlabels ", items_with_more_than_one_altlabel)
```
```
Number of items with more than one altlabels  5069312
```

In [115]:
```
one_altlabel_item/num_item_that_have_a_altlabel, items_with_more_than_one_altlabel/num_item_that_have_a_altlabel
```
Out[115]:
```
(0.48922564716097516, 0.5107743528390248)
```

Which items have that many altlabels?

In [ ]:
```
max_value = 3879

spark.sql(f"""
        select subject
        from (
        select count(distinct subject, object) as label_count, subject
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[2]}'
        group by subject
        ) as intermediate_table
        where label_count = {max_value}
        limit 10
        """)\
    .show(10, truncate=False)
```

How many don't have any english label? i.e have only other language labels

Meaning if we remove the other language labels, it won't have any label at all.

In [33]:
```
spark.sql(f"""
        select count(distinct subject) as num_item_that_have_a_english_altlabel
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[2]}' and object like '%en'
        """)\
    .show()
```
```
+-------------------------------------+
|num_item_that_have_a_english_altlabel|
+-------------------------------------+
|                              5475585|
+-------------------------------------+
```

In [116]:
```
num_item_that_have_a_english_altlabel = 5475585
num_item_that_have_only_other_language_altlabels = num_item_that_have_a_altlabel - num_item_that_have_a_english_altlabel

num_item_that_have_only_other_language_altlabels, num_item_that_have_only_other_language_altlabels/num_item_that_have_a_altlabel
```
Out[116]:
```
(4449271, 0.44829950418127713)
```

Distribution of language tags

In [34]:
```
alt_lang_df = spark.sql(f"""
        select lang_tag, count(lang_tag) as count from
        (select distinct subject, SUBSTRING(object, length(object) - locate('@',reverse(object)) + 2, length(object)) as lang_tag
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[2]}') as lang_tag_table
        group by lang_tag
        """)\
    .toPandas()
```

In [35]:
```
alt_lang_df['percentage'] = (alt_lang_df['count']*100/alt_lang_df['count'].sum()).round(decimals=2)
alt_lang_df = alt_lang_df.sort_values("count", ascending=False)
alt_lang_df.head(20)
```
Out[35]:

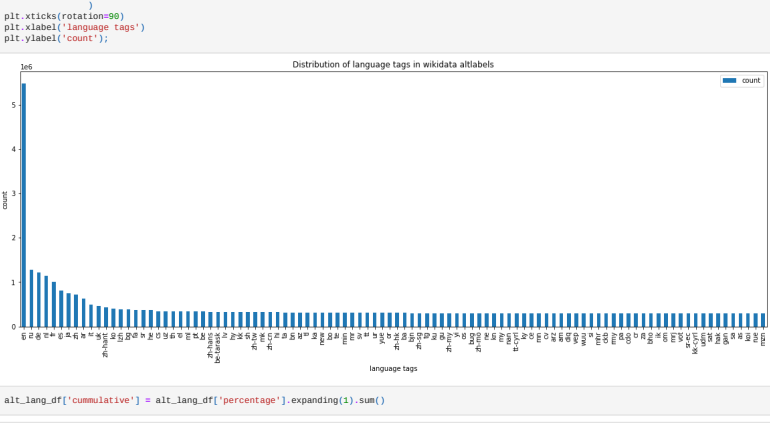
	lang_tag	count	percentage
243	en	5475505	6.43
23	ru	1279056	1.50
62	de	1217710	1.43
215	nl	1184401	1.33
205	fr	1005149	1.18
126	es	802758	0.94
284	ja	742082	0.87
351	zh	723015	0.85
416	sr	632716	0.74
137	it	484438	0.57
75	uk	460405	0.54
241	zh-hant	435917	0.50
226	ko	398506	0.47
389	lzh	378639	0.45
186	bg	378971	0.45
107	fa	373284	0.44
391	sr	369775	0.43
387	he	368946	0.43
161	cs	346115	0.41
363	uz	345385	0.40

In [36]:
```
alt_lang_df.tail(20)
```
Out[36]:

	lang_tag	count	percentage
178	tum	39842	0.05
185	skr-arab	39842	0.05
326	roy	39841	0.05
119	simple	374	0.00
396	ddg	125	0.00
159	mi-arab	95	0.00
85	wes	54	0.00
142	mad	31	0.00
117	bbj	22	0.00
29	fkl	19	0.00
299	bas	16	0.00
66	abs	12	0.00
56	nyu	11	0.00
370	alt	11	0.00
272	diua	7	0.00
144	pwo	6	0.00
37	kiu	5	0.00
414	guc	3	0.00
74	nia	3	0.00
33	nan-hani	3	0.00

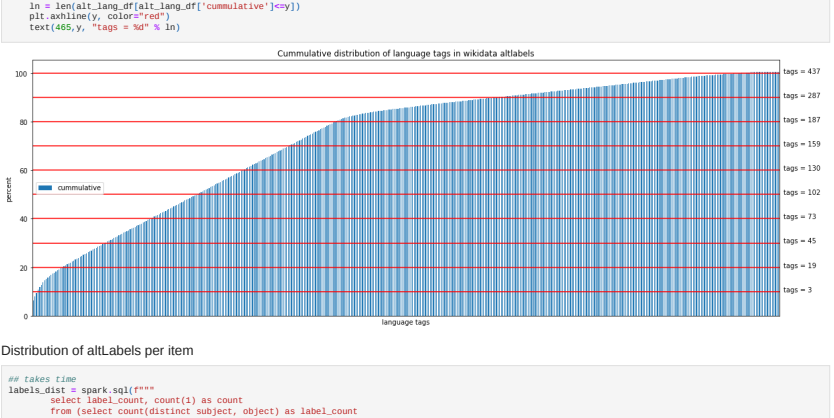
In [37]:
```
len(alt_lang_df)
```
Out[37]:
```
463
```

In [38]:
```
alt_lang_df.head(100).plot(x="lang_tag", y="count",
                kind='bar',
                figsize=(20,7),
                title="Distribution of language tags in wikidata altlabels",
                )
plt.xticks(rotation=90)
plt.xlabel('language tags')
plt.ylabel('count');
```



In [39]:
```
alt_lang_df['cumulative'] = alt_lang_df['percentage'].expanding(1).sum()
```

In [40]:
```
alt_lang_df.plot(y="cumulative",
             kind='bar',
             figsize=(20,7),
             title="Cummulative distribution of language tags in wikidata altlabels",
             )
plt.xticks([])
plt.xlabel('language tags')
plt.ylabel('percent')

for y in range(10,110,10):
    ln = len(alt_lang_df[alt_lang_df['cumulative']<=y])
    plt.axhline(y, color="red")
    text(465,y, "tags = %d" % ln)
```



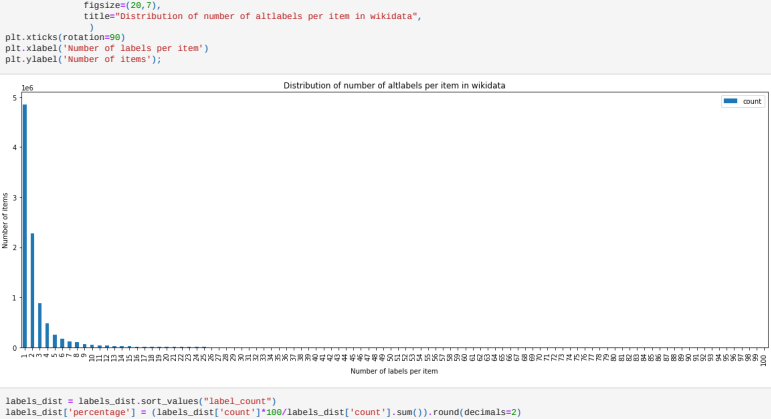
Distribution of altLabels per item

In [41]:
```
## takes time
labels_dist = spark.sql(f"""
        select label_count, count(1) as count
        from (select count(distinct subject, object) as label_count
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[2]}'
        group by subject) as intermediate_table
        group by label_count
        """).toPandas().sort_values("count", ascending=False)
```

In [42]:
```
labels_dist.sort_values("label_count", ascending=False).head(20)
```
Out[42]:

	label_count	count
79	3879	1
174	3012	1
373	2476	1
251	2358	1
50	2354	2
117	2058	1
298	1635	1
511	1498	1
489	1416	1
283	1339	14
531	1290	1
429	1239	1
312	1238	1
67	1237	1
119	1090	1
60	1080	1
172	1073	1
281	1071	1
44	1070	1
432	1065	1

In [43]:
```
labels_dist.sort_values("label_count")\
        .head(100)\
        .plot(x="label_count", y="count",
            kind='bar',
            figsize=(20,7),
            title="Distribution of number of altlabels per item in wikidata",
            )
plt.xticks(rotation=90)
plt.xlabel('Number of labels per item')
plt.ylabel('Number of items');
```



In [44]:
```
labels_dist = labels_dist.sort_values("label_count")
labels_dist['percentage'] = (labels_dist['count']*100/labels_dist['count'].sum()).round(decimals=2)
labels_dist['cumulative'] = labels_dist['percentage'].expanding(1).sum()

labels_dist.head(20)
```
Out[44]:

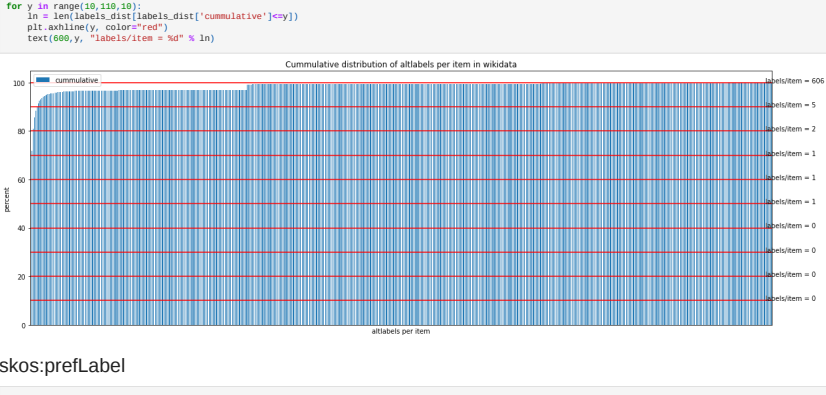
	label_count	count	percentage	cumulative
189	1	4855544	48.92	48.92
170	2	2281944	22.99	71.91
563	3	882232	8.89	80.80
92	4	484178	4.88	85.68
576	5	254312	2.56	88.24
374	6	176893	1.78	90.02
455	7	124748	1.26	91.28
46	8	110114	1.11	92.39
441	9	67277	0.68	93.07
438	10	51055	0.51	93.58
130	11	44799	0.45	94.03
176	12	40144	0.40	94.43
226	13	26751	0.27	94.70
583	14	22164	0.22	94.92
105	15	20185	0.20	95.12
409	16	17490	0.18	95.30
113	17	13308	0.13	95.43
600	18	11943	0.12	95.55
503	19	9931	0.10	95.65
153	20	9894	0.10	95.75

In [45]:
```
len(labels_dist)
```
Out[45]:
```
608
```

In [46]:
```
labels_dist['label_count'].describe()
```
Out[46]:
```
count     608.000000
mean      398.386138
std       389.722732
min         1.000000
25%       152.250000
50%       307.500000
75%       503.500000
max      3879.000000
Name: label_count, dtype: float64
```

In [47]:
```
## Avg labels per item
## matches previously found avg
(labels_dist['label_count']*labels_dist['count']).sum() / labels_dist['count'].sum()
```
Out[47]:
```
10.336915685163414
```

In [48]:
```
labels_dist.plot(y="cumulative",
             kind='bar',
             figsize=(20,7),
             title="Cummulative distribution of altlabels per item in wikidata",
             )
plt.xticks([])
plt.xlabel('altlabels per item')
plt.ylabel('percent')

for y in range(10,110,10):
    ln = len(labels_dist[labels_dist['cumulative']<=y])
    plt.axhline(y, color="red")
    text(600,y, 'labels/item = %d' % ln)
```



skos:prefLabel

In [1]:
```
prefLabel = "<http://www.w3.org/2004/02/skos/core#prefLabel>"
```

In [4]:
```
prefLabel_count = spark.sql(f"""Select count(distinct subject, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{prefLabel}' """)\
                        .collect()[0]['count']
prefLabel_count, (prefLabel_count/total_triples)*100
```
Out[4]:
```
(0, 0.0)
```

schema:name

In [49]:
```
name = "<http://schema.org/name>"

name_count = spark.sql(f"""Select count(distinct subject, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{name}' """)\
                        .collect()[0]['count']

name_count, (name_count/total_triples)*100
```
Out[49]:
```
(78785768, 0.6217424606375152)
```

In [50]:
```
ne_count = spark.sql(f"""Select count(distinct subject, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{name}' and object not like '%en' """)\
                        .collect()[0]['count']
ne_count, (ne_count/total_triples)*100
```
Out[50]:
```
(65064444, 0.5134598354567277)
```

In [51]:
```
e_count = spark.sql(f"""Select count(distinct subject, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{name}' and object like '%en' """)\
                        .collect()[0]['count']
e_count, (e_count/total_triples)*100
```
Out[51]:
```
(13721324, 0.10828262458178738)
```

In [118]:
```
how_many_days(name_count), how_many_days(ne_count), how_many_days(e_count)
```
```
16.517 days 0.622 % of triples
13.64 days 0.513 % of triples
2.877 days 0.108 % of triples
```
Out[118]:
```
(None, None, None)
```

In [14]:
```
# see english names
spark.sql(f"""Select subject, object from discovery.wikibase_rdf
        where date={date} and wiki='wikidata' and predicate = '{name}' and object like '%en' limit 10""")\
    .show(10, truncate=False)
```
```
+-----------------------------------------------------------------------------------------------+----------------------------------------------------------------
-+
|subject                                                                                        |object
 |
+-----------------------------------------------------------------------------------------------+----------------------------------------------------------------
-+
|<https://en.wikipedia.org/wiki/Libertine_Enlightenment>                                        |"Libertine Enlightenment"@en
 |
|<https://en.wikipedia.org/wiki/Squamigera>                                                     |"Squamigera"@en
 |
|<https://en.wikipedia.org/wiki/Template:Taxonomy/Gogo>                                         |"Template:Taxonomy/Gogo"@en
 |
|<https://en.wikipedia.org/wiki/Category:Same-sex_marriage_in_the_United_States_by_Circuit>    |"Category:Same-sex marriage in the United States by Circuit"@
en|
|<https://commons.wikimedia.org/wiki/Category:Kompagnistr%C3%A6de_4>                            |"Category:Kompagnistr\u00E6de 4"@en
 |
|<https://en.wikipedia.org/wiki/Category:Medalists_at_the_2013_Summer_Deaflympics>             |"Category:Medalists at the 2013 Summer Deaflympics"@en
 |
|<https://en.wikipedia.org/wiki/1952_Michigan_gubernatorial_election>                           |"1952 Michigan gubernatorial election"@en
 |
|<https://en.wikipedia.org/wiki/Template:Taxonomy/Calichodes>                                   |"Template:Taxonomy/Calichodes"@en
 |
|<https://en.wikipedia.org/wiki/Category:Earls_of_Aldborough>                                   |"Category:Earls of Aldborough"@en
 |
|<https://commons.wikimedia.org/wiki/Category:Media_from_Blake_2020_-_10.11646/zootaxa.4730.1.1>|"Category:Media from Blake 2020 - 10.11646/zootaxa.4730.1.1"@
en|
+-----------------------------------------------------------------------------------------------+----------------------------------------------------------------
-+
```

In [15]:
```
all_e = 13721324+9157838+79778129
print(all_e)
how_many_days(all_e)
```
```
102657291
21.521 days 0.81 % of triples
```

In [16]:
```
all_ne = 65064444+65064444+419885045
print(all_ne)
how_many_days(all_ne)
```
```
550013933
115.307 days 4.34 % of triples
```

In [119]:
```
total_label = all_e+all_ne
total_label, how_many_days(total_label)
```
```
136.828 days 5.151 % of triples
```
Out[119]:
```
(652671224, None)
```

Description

How many description triples?

In [52]:
```
desc_count = spark.sql(f"""Select count(distinct subject, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{props[1]}' """)\
                        .collect()[0]['count']
desc_count, (desc_count/total_triples)*100
```
Out[52]:
```
(2471378661, 19.503028116187796)
```

In [104]:
```
how_many_days(desc_count)
```
```
518.109 days 19.503 % of triples
```

How many non-English description triples?

In [53]:
```
ne_desc_count = spark.sql(f"""Select count(distinct subject, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{props[1]}' and object not like '%en' """)\
                        .collect()[0]['count']
ne_desc_count, (ne_desc_count/total_triples)*100
```
Out[53]:
```
(2398769645, 18.930029851189951)
```

In [106]:
```
how_many_days(ne_desc_count)
```
```
502.887 days 18.93 % of triples
```

In [13]:
```
spark.sql(f"""Select * from discovery.wikibase_rdf
        where date={date} and wiki='wikidata' and predicate = '{props[1]}' and object not like '%en' limit 10""")\
    .show(10, truncate=False)
```
```
+------------------------------------------+------------------------------------------+--------------------------------+------------------------------------
|context                                   |subject                                   |predicate                       |object
|date    |wiki    |
+------------------------------------------+------------------------------------------+--------------------------------+------------------------------------
|<http://www.wikidata.org/entity/Q53289951>|<http://www.wikidata.org/entity/Q53289951>|<http://schema.org/description>|"\u0638\u0627\u0644\u0639\u0628\u0
644 \u0641\u0642\u0630\u0642\u0646\u0642"@ru                                             |20210517|wikidata|
|<http://www.wikidata.org/entity/Q52544859>|<http://www.wikidata.org/entity/Q52544859>|<http://schema.org/description>|"\u0630\u0637\u0628\u0643\u0
E0C\u0630\u0633\u0638\u0634A\u0630C\u0630E\u0630\u0635"@el                                |20210517|wikidata|
|<http://www.wikidata.org/entity/Q51774489>|<http://www.wikidata.org/entity/Q51774489>|<http://schema.org/description>|"artikulong pang-agham"@tl
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q80511883>|<http://www.wikidata.org/entity/Q80511883>|<http://schema.org/description>|"\u0634\u0630\u0647\u0644\u0630\u0
648\u0648\u0648\u0648 \u0630\u0648\u0634\u0649\u0648"@fa                                  |20210517|wikidata|
|<http://www.wikidata.org/entity/Q51328879>|<http://www.wikidata.org/entity/Q51328879>|<http://schema.org/description>|"1970\u5E74\u5B66\u672F\u8AD6\u6587"@zh
-hans                                                                                    |20210517|wikidata|
|<http://www.wikidata.org/entity/Q66602839>|<http://www.wikidata.org/entity/Q66602839>|<http://schema.org/description>|"categor\u00EDa de Wikimedia"@es
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q51779247>|<http://www.wikidata.org/entity/Q51779247>|<http://schema.org/description>|"vitskaplig artikel"@sv
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q86421165>|<http://www.wikidata.org/entity/Q86421165>|<http://schema.org/description>|"arti\u0219olul cient\u00EDfico"@ast
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q51854486>|<http://www.wikidata.org/entity/Q51854486>|<http://schema.org/description>|"scienca artikolo"@eo
|20210517|wikidata|
|<http://www.wikidata.org/entity/Q52729871>|<http://www.wikidata.org/entity/Q52729871>|<http://schema.org/description>|"wissenschaftlicher Artikel"@de
|20210517|wikidata|
+------------------------------------------+------------------------------------------+--------------------------------+------------------------------------
```

English descriptions

In [54]:
```
e_desc_count = spark.sql(f"""Select count(distinct subject, object) as count from discovery.wikibase_rdf
                        where date={date} and wiki='wikidata' and predicate = '{props[1]}' and object like '%en' """)\
                        .collect()[0]['count']
e_desc_count, (e_desc_count/total_triples)*100
```
Out[54]:
```
(72609016, 0.5729982831962865)
```

In [120]:
```
how_many_days(e_desc_count)
```
```
15.222 days 0.573 % of triples
```

Number of description triple per item

In [55]:
```
## takes time
spark.sql(f"""
        select count(distinct subject) as num_item_that_have_a_desc, sum(desc_count) as total_ldesc,
        avg(desc_count), max(desc_count), min(desc_count),
        sum(CASE WHEN desc_count=1 THEN 1 ELSE 0 END) as one_desc_count
        from (
        select subject, count(distinct subject, object) as desc_count
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[1]}'
        group by subject
        ) as intermediate_table
        """)\
    .show()
```
```
+-------------------------+-----------+------------------+---------------+---------------+--------------+
|num_item_that_have_a_desc|total_ldesc|   avg(desc_count)|max(desc_count)|min(desc_count)|one_desc_count|
+-------------------------+-----------+------------------+---------------+---------------+--------------+
|                 87648591| 2471378661|28.396623883184535|            258|              1|       9916891|
+-------------------------+-----------+------------------+---------------+---------------+--------------+
```

In [144]:
```
num_item_that_have_description = 87648591
one_description_item = 9916891
items_with_more_than_one_description = num_item_that_have_description - one_description_item

print("Number of items with more than one descriptions ", items_with_more_than_one_description)
```
```
Number of items with more than one descriptions  77731700
```

In [146]:
```
one_description_item/num_item_that_have_description, items_with_more_than_one_description/num_item_that_have_description
```
Out[146]:
```
(0.11314562496091172, 0.8868543750390883)
```

Which items have that many descriptions?

In [ ]:
```
max_value = 258
spark.sql(f"""
        select subject
        from (
        select count(distinct subject, object) as desc_count, subject
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[1]}'
        group by subject
        ) as intermediate_table
        where desc_count = {max_value}
        limit 10
        """)\
    .show(10, truncate=False)
```

How many don't have any english description? i.e have only other language descriptions

Meaning if we remove the other language descriptions, it won't have any description at all.

In [56]:
```
spark.sql(f"""
        select count(distinct subject) as num_item_that_have_a_english_description
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[1]}' and object like '%en'
        """)\
    .show()
```
```
+----------------------------------------+
|num_item_that_have_a_english_description|
+----------------------------------------+
|                                72609016|
+----------------------------------------+
```

In [144]:
```
num_item_that_have_a_english_description = 72609016
num_item_that_have_only_other_language_descriptions = num_item_that_have_description - num_item_that_have_a_english_description

num_item_that_have_only_other_language_descriptions, num_item_that_have_only_other_language_descriptions/num_item_that_have_description
```
Out[144]:
```
(14439495, 0.1658780174847514)
```

16.6% of all items that have a description dont have english description.

Distribution of language tags

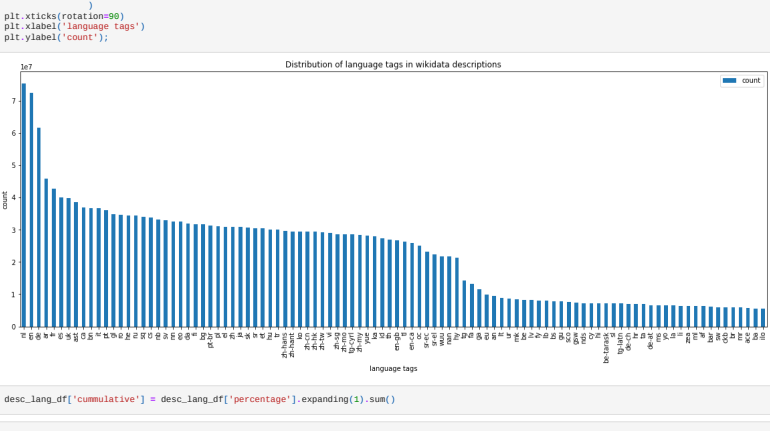
In [57]:
```
desc_lang_df = spark.sql(f"""
        select lang_tag, count(lang_tag) as count from
        (select distinct subject, SUBSTRING(object, length(object) - locate('@',reverse(object)) + 2, length(object)) as lang_tag
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[1]}') as lang_tag_table
        group by lang_tag
        """)\
    .toPandas()
```

In [58]:
```
desc_lang_df['percentage'] = (desc_lang_df['count']*100/desc_lang_df['count'].sum()).round(decimals=2)
desc_lang_df = desc_lang_df.sort_values("count", ascending=False)
desc_lang_df.head(20)
```
Out[58]:

	lang_tag	count	percentage
204	nl	75405965	3.05
231	en	72609016	2.94
76	de	61716292	2.50
384	ar	46309196	1.86
193	fr	42861255	1.73
127	es	39968399	1.62
70	uk	39659946	1.61
137	ast	38643901	1.56
29	ca	36951411	1.49
235	bn	36750936	1.49
128	it	36747322	1.49
307	pl	36086242	1.46
189	gl	34915253	1.41
16	ro	34794021	1.41
364	he	34511500	1.40
22	sq	34471330	1.39
208	sv	34049636	1.38
151	cs	33764437	1.37
277	nb	33272409	1.35
13	hy	33064591	1.34

In [59]:
```
len(desc_lang_df)
```
Out[59]:
```
449
```

In [60]:
```
desc_lang_df.head(100).plot(x="lang_tag", y="count",
                kind='bar',
                figsize=(20,7),
                title="Distribution of language tags in wikidata descriptions",
                )
plt.xticks(rotation=90)
plt.xlabel('language tags')
plt.ylabel('count');
```

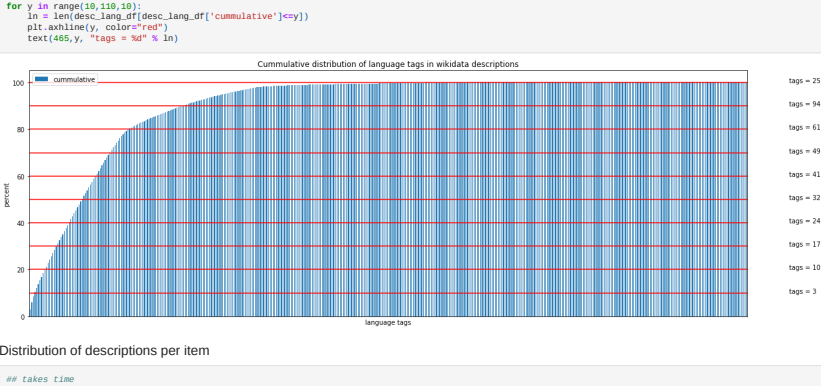


In [61]:
```
desc_lang_df['cumulative'] = desc_lang_df['percentage'].expanding(1).sum()
```

In [62]:
```
desc_lang_df.head(10)
```
Out[62]:

	lang_tag	count	percentage	cumulative
204	nl	75405965	3.05	3.05
231	en	72609016	2.94	5.99
76	de	61716292	2.50	8.49
384	ar	46309196	1.86	10.35
193	fr	42861255	1.73	12.08
127	es	39968399	1.62	13.70
70	uk	39659946	1.61	15.31
137	ast	38643901	1.56	16.87
29	ca	36951411	1.49	18.36
235	bn	36750936	1.49	19.85

In [63]:
```
desc_lang_df.plot(y="cumulative",
             kind='bar',
             figsize=(20,7),
             title="Cummulative distribution of language tags in wikidata descriptions",
             )
plt.xticks([])
plt.xlabel('language tags')
plt.ylabel('percent')

for y in range(10,110,10):
    ln = len(desc_lang_df[desc_lang_df['cumulative']<=y])
    plt.axhline(y, color="red")
    text(445,y, "tags = %d" % ln)
```



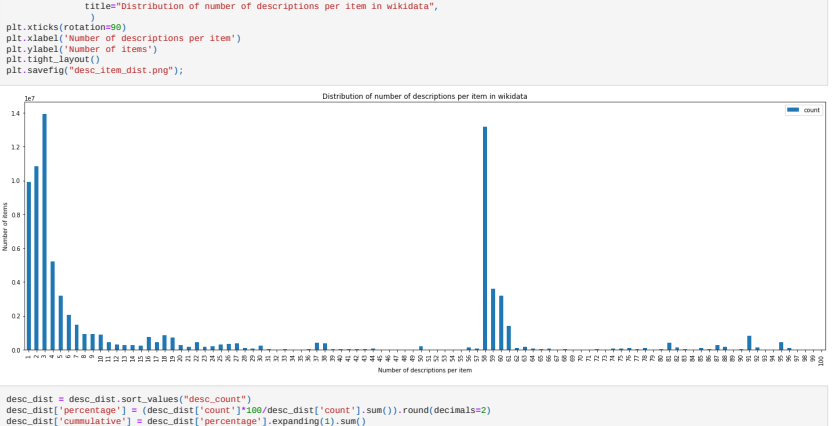
Distribution of descriptions per item

In [64]:
```
## takes time
desc_dist = spark.sql(f"""
        select desc_count, count(1) as count
        from (select count(distinct subject, object) as desc_count
        from discovery.wikibase_rdf where date={date} and wiki='wikidata' and
        predicate = '{props[1]}'
        group by subject) as intermediate_table
        group by desc_count
        """).toPandas().sort_values("count", ascending=False)
```

In [65]:
```
desc_dist.sort_values("desc_count", ascending=False).head(20)
```
Out[65]:

	desc_count	count
38	258	1
69	246	2
17	245	1
28	244	14
143	243	221
15	242	1909
76	241	26863
6	240	1598
3	239	605
102	238	1537
83	237	6346
177	236	82731
8	235	1036
2	234	732
162	233	26114
104	232	5718
99	231	97
93	230	2
84	229	36
153	228	823

In [66]:
```
desc_dist.sort_values("desc_count")\
        .head(100)\
        .plot(x="desc_count", y="count",
            kind='bar',
            figsize=(20,7),
            title="Distribution of number of descriptions per item in wikidata",
            )
plt.xticks(rotation=90)
plt.xlabel('Number of descriptions per item')
plt.ylabel('Number of items')
plt.tight_layout()
plt.savefig("desc_item_dist.png");
```



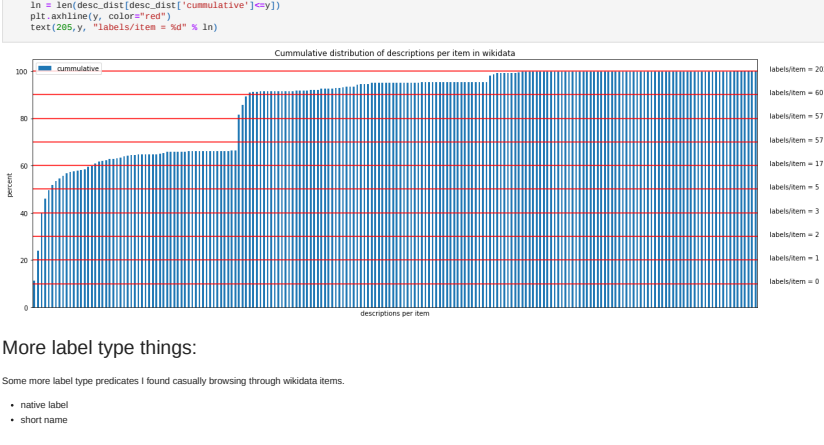
In [67]:
```
desc_dist = desc_dist.sort_values("desc_count")
desc_dist['percentage'] = (desc_dist['count']*100/desc_dist['count'].sum()).round(decimals=2)
desc_dist['cumulative'] = desc_dist['percentage'].expanding(1).sum()

desc_dist.head(20)
```
Out[67]:

	desc_count	count	percentage	cumulative
66	1	9916891	11.38	11.38
61	2	10945750	12.46	23.84
180	3	13930579	16.01	39.85
29	4	5221876	6.00	45.85
168	5	3190051	3.65	49.50
130	6	2081753	2.37	51.87
150	7	1458336	1.67	53.54
19	8	936750	1.08	54.62
149	9	919864	1.06	55.68
146	10	886663	1.02	56.70
64	11	440260	0.51	57.21
62	12	315345	0.36	57.57
77	13	275582	0.32	57.89
191	14	282010	0.32	58.21
34	15	248248	0.29	58.50
147	16	780979	0.87	59.37
39	17	431761	0.50	59.87
199	18	840330	0.97	60.84
164	19	709074	0.81	61.65
54	20	208642	0.30	61.95

In [68]:
```
len(desc_dist)
```
Out[68]:
```
258
```

In [69]:
```
## Avg desc per item
## matches previously found avg
(desc_dist['desc_count']*desc_dist['count']).sum() / desc_dist['count'].sum()
```
Out[69]:
```
28.39662388184535
```

In [70]:
```
desc_dist.plot(y="cumulative",
             kind='bar',
             figsize=(20,7),
             title="Cummulative distribution of descriptions per item in wikidata",
             )
plt.xticks([])
plt.xlabel('descriptions per item')
plt.ylabel('percent')

for y in range(10,110,10):
    ln = len(desc_dist[desc_dist['cumulative']<=y])
    plt.axhline(y, color="red")
    text(255,y, 'labels/item = %d' % ln)
```



More label type things:

Some more label type predicates I found casually browsing through wikidata items.

- native label
- short name

External IDs

In [5]:
```
import sys
from SPARQLWrapper import SPARQLWrapper, JSON

endpoint_url = "https://query.wikidata.org/sparql"

query = """SELECT distinct ?item ?itemLabel
WHERE
{
  ?item wikibase:propertyType <http://wikiba.se/ontology#ExternalId>.
  SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO_LANGUAGE],en". }
}"""


def get_results(endpoint_url, query):
    user_agent = "akhatun-ctr-wmf/"
    sparql = SPARQLWrapper(endpoint_url, agent=user_agent)
    sparql.setQuery(query)
    sparql.setReturnFormat(JSON)
    return sparql.query().convert()


results = get_results(endpoint_url, query)

results_df = pd.json_normalize(results['results']['bindings'])
results_df = results_df[['item.value', 'itemLabel.value']].rename(columns = {"item.value":"item", "itemLabel.value":"label"})

results_df
```
Out[5]:

	item	label
0	http://www.wikidata.org/entity/P503	ISO standard
1	http://www.wikidata.org/entity/P494	ICD-10
2	http://www.wikidata.org/entity/P212	ISBN-13
3	http://www.wikidata.org/entity/P236	ISSN
4	http://www.wikidata.org/entity/P234	InChI
...	...	...
6317	http://www.wikidata.org/entity/P9734	Treccani's Enciclopedia Dantesca ID
6318	http://www.wikidata.org/entity/P9760	Treccani's Enciclopedia del Cinema ID
6319	http://www.wikidata.org/entity/P9761	IRIS SNS author ID
6320	http://www.wikidata.org/entity/P9762	U.S. Ski & Snowboard athlete ID
6321	http://www.wikidata.org/entity/P9665	EGID

6322 rows × 2 columns

In [6]:
```
results_df['code'] = results_df['item'].str.split('/').str[-1]
results_df
```
Out[6]:

	item	label	code
0	http://www.wikidata.org/entity/P503	ISO standard	P503
1	http://www.wikidata.org/entity/P494	ICD-10	P494
2	http://www.wikidata.org/entity/P212	ISBN-13	P212
3	http://www.wikidata.org/entity/P236	ISSN	P236
4	http://www.wikidata.org/entity/P234	InChI	P234
...	...	...	...
6317	http://www.wikidata.org/entity/P9734	Treccani's Enciclopedia Dantesca ID	P9734
6318	http://www.wikidata.org/entity/P9760	Treccani's Enciclopedia del Cinema ID	P9760
6319	http://www.wikidata.org/entity/P9761	IRIS SNS author ID	P9761
6320	http://www.wikidata.org/entity/P9762	U.S. Ski & Snowboard athlete ID	P9762
6321	http://www.wikidata.org/entity/P9665	EGID	P9665

6322 rows × 3 columns

Get triple with external identifier as predicate

In [44]:
```
def propIdentity(prop, type_str="direct"):
    if type_str=="direct":
        return "<http://www.wikidata.org/prop/direct/"+prop+">"
    elif type_str=="norm":
        return "<http://www.wikidata.org/prop/direct-normalized/"+prop+">"
    else:
        return "<http://www.wikidata.org/prop/"+prop+">"
```

In [45]:
```
results_df['prop_uri'] = results_df['code'].apply(propIdentity)
results_df['prop_direct_uri'] = results_df['code'].apply(propIdentity, args=('direct',))
results_df['prop_direct_normalized_uri'] = results_df['code'].apply(propIdentity, args=('norm',))
```

In [46]:
```
propStr = "','".join(results_df['prop_uri'].values.tolist())
propDirectStr = "','".join(results_df['prop_direct_uri'].values.tolist())
propDirectNormStr = "','".join(results_df['prop_direct_normalized_uri'].values.tolist())
```

In [47]:
```
prop_df = spark.sql(f"""select predicate, count(distinct subject, predicate, object) as prop_count
                from discovery.wikibase_rdf
                where date={date} and wiki='wikidata'
                and predicate in ('{propStr}')
                group by predicate
                """).toPandas()
prop_df
```
Out[47]:

	predicate	prop_count
0	<http://www.wikidata.org/prop/P8604>	6067
1	<http://www.wikidata.org/prop/P6242>	8484
2	<http://www.wikidata.org/prop/P3615>	14384
3	<http://www.wikidata.org/prop/P3360>	4094
4	<http://www.wikidata.org/prop/P969>	4255
...	...	...
6195	<http://www.wikidata.org/prop/P2961>	227
6196	<http://www.wikidata.org/prop/P3810>	342
6197	<http://www.wikidata.org/prop/P3022>	55
6198	<http://www.wikidata.org/prop/P6401>	31
6199	<http://www.wikidata.org/prop/P8229>	30

6200 rows × 2 columns

In [48]:
```
prop_direct_df = spark.sql(f"""select predicate, count(distinct subject, predicate, object) as prop_direct_count
                from discovery.wikibase_rdf
                where date={date} and wiki='wikidata'
                and predicate in ('{propDirectStr}')
                group by predicate
                """).toPandas()
prop_direct_df
```
Out[48]:

	predicate	prop_direct_count
0	<http://www.wikidata.org/prop/direct/P7741>	8228
1	<http://www.wikidata.org/prop/direct/P7042>	1876
2	<http://www.wikidata.org/prop/direct/P8254>	1659
3	<http://www.wikidata.org/prop/direct/P6299>	1349
4	<http://www.wikidata.org/prop/direct/P4768>	2520
...	...	...
6195	<http://www.wikidata.org/prop/direct/P8079>	43
6196	<http://www.wikidata.org/prop/direct/P6687>	113
6197	<http://www.wikidata.org/prop/direct/P6423>	43
6198	<http://www.wikidata.org/prop/direct/P6483>	12
6199	<http://www.wikidata.org/prop/direct/P5667>	3

6200 rows × 2 columns

In [49]:
```
prop_direct_norm_df = spark.sql(f"""select predicate, count(distinct subject, predicate, object) as prop_direct_norm_count
                from discovery.wikibase_rdf
                where date={date} and wiki='wikidata'
                and predicate in ('{propDirectNormStr}')
                group by predicate
                """).toPandas()
prop_direct_norm_df
```
Out[49]:

	predicate	prop_direct_norm_count
0	<http://www.wikidata.org/prop/direct-normalize...	1330
1	<http://www.wikidata.org/prop/direct-normalize...	30
2	<http://www.wikidata.org/prop/direct-normalize...	775
3	<http://www.wikidata.org/prop/direct-normalize...	8090
4	<http://www.wikidata.org/prop/direct-normalize...	22208
...	...	...
179	<http://www.wikidata.org/prop/direct-normalize...	45210
180	<http://www.wikidata.org/prop/direct-normalize...	28771
181	<http://www.wikidata.org/prop/direct-normalize...	2714
182	<http://www.wikidata.org/prop/direct-normalize...	682848
183	<http://www.wikidata.org/prop/direct-normalize...	454

184 rows × 2 columns

In [69]:
```
full_prop_df = results_df.merge(prop_df, how="outer", left_on="prop_uri", right_on="predicate")\
            .drop(columns=['predicate'])[['code', 'label', 'item', 'prop_uri', 'prop_direct_uri', 'prop_direct_normalized_uri', 'prop_count']]

full_prop_df = full_prop_df.merge(prop_direct_df, how="outer", left_on="prop_direct_uri", right_on="predicate")\
            .drop(columns=['predicate'])[['code', 'label', 'item', 'prop_uri', 'prop_direct_uri', 'prop_direct_normalized_uri', 'prop_count', 'prop_direct_count']]

full_prop_df = full_prop_df.merge(prop_direct_norm_df, how="outer", left_on="prop_direct_normalized_uri", right_on="predicate")\
            .drop(columns=['predicate'])[['code', 'label', 'item', 'prop_uri', 'prop_direct_uri', 'prop_direct_normalized_uri', 'prop_count', 'prop_direct_count', 'prop_direct_norm
```

In [70]:
```
print(len(full_prop_df[full_prop_df['prop_direct_count'].isna()]), \
      len(full_prop_df[full_prop_df['prop_direct_norm_count'].isna()]),
      len(full_prop_df[full_prop_df['prop_count'].isna()])
     )

## either they are not used (weird?)
## or, were simply deleted (therefore not used)
## spark is taking previous data, sparql is taking new data. maybe thats where the inconsistency happens

full_prop_df[full_prop_df['prop_count'].isna()]
```
```
122 6138 122
```
Out[70]:

	code	label	item	prop_uri	prop_direct_uri	prop_direct_normalized_uri	prop_count	prop_direct_count
1514	P3640	National Drug Code	http://www.wikidata.org/entity/P3640	<http://www.wikidata.org/prop/P3640>	<http://www.wikidata.org/prop/direct/P3640>	<http://www.wikidata.org/prop/direct-normalize...	NaN	NaN
1660	P3717	MEROPS enzyme ID	http://www.wikidata.org/entity/P3717	<http://www.wikidata.org/prop/P3717>	<http://www.wikidata.org/prop/direct/P3717>	<http://www.wikidata.org/prop/direct-normalize...	NaN	NaN
1685	P3619	WarSampo army unit ID	http://www.wikidata.org/entity/P3619	<http://www.wikidata.org/prop/P3619>	<http://www.wikidata.org/prop/direct/P3619>	<http://www.wikidata.org/prop/direct-normalize...	NaN	NaN
1709	P3983	Proteopedia ID (find-ID-code)	http://www.wikidata.org/entity/P3983	<http://www.wikidata.org/prop/P3983>	<http://www.wikidata.org/prop/direct/P3983>	<http://www.wikidata.org/prop/direct-normalize...	NaN	NaN
1878	P3972	PASE Domesday place	http://www.wikidata.org/entity/P3972	<http://www.wikidata.org/prop/P3972>	<http://www.wikidata.org/prop/direct/P3972>	<http://www.wikidata.org/prop/direct-normalize...	NaN	NaN
...	...	...	...	...	...	...	...	...
6315	P9756	Schoenberg Database of Manuscripts name ID	http://www.wikidata.org/entity/P9756	<http://www.wikidata.org/prop/P9756>	<http://www.wikidata.org/prop/direct/P9756>	<http://www.wikidata.org/prop/direct-normalize...	NaN	NaN
6316	P9757	Schoenberg Database of Manuscripts place ID	http://www.wikidata.org/entity/P9757	<http://www.wikidata.org/prop/P9757>	<http://www.wikidata.org/prop/direct/P9757>	<http://www.wikidata.org/prop/direct-normalize...	NaN	NaN
6318	P9760	Treccani's Enciclopedia del Cinema ID	http://www.wikidata.org/entity/P9760	<http://www.wikidata.org/prop/P9760>	<http://www.wikidata.org/prop/direct/P9760>	<http://www.wikidata.org/prop/direct-normalize...	NaN	NaN
6319	P9761	IRIS SNS author ID	http://www.wikidata.org/entity/P9761	<http://www.wikidata.org/prop/P9761>	<http://www.wikidata.org/prop/direct/P9761>	<http://www.wikidata.org/prop/direct-normalize...	NaN	NaN
6320	P9762	U.S. Ski & Snowboard athlete ID	http://www.wikidata.org/entity/P9762	<http://www.wikidata.org/prop/P9762>	<http://www.wikidata.org/prop/direct/P9762>	<http://www.wikidata.org/prop/direct-normalize...	NaN	NaN

122 rows × 8 columns

In [71]:
```
full_prop_df = full_prop_df.fillna(0)
```

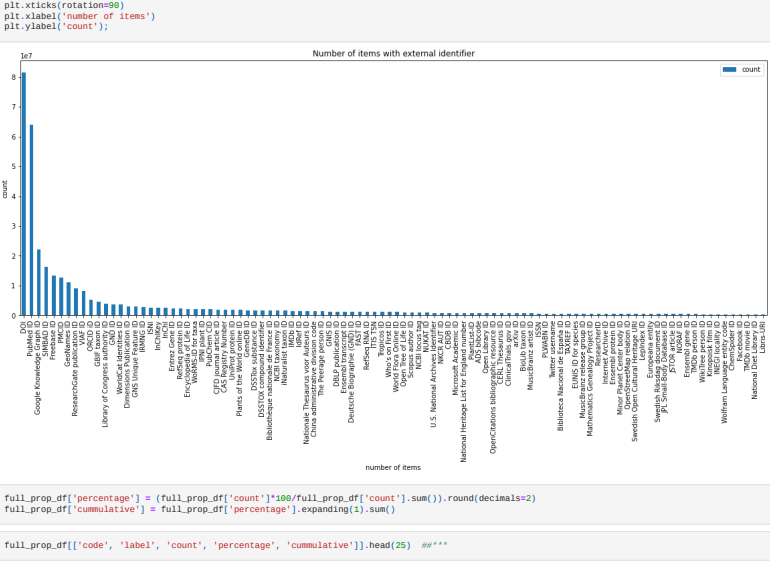
In [72]:
```
full_prop_df['count'] = full_prop_df['prop_count'] + full_prop_df['prop_direct_count'] + full_prop_df['prop_direct_norm_count']
```

In [73]:
```
# number of external property triples
tot_prop = full_prop_df['prop_count'].sum()
tot_direct_prop = full_prop_df['prop_direct_count'].sum()
tot_direct_norm_prop = full_prop_df['prop_direct_norm_count'].sum()
tot = tot_prop+tot_direct_prop+tot_direct_norm_prop

print("external identifiers (prop)", tot_prop, tot_prop*100/total_triples)
print("external identifiers (prop direct)", tot_direct_prop, tot_direct_prop*100/total_triples)
print("external identifiers (prop direct norm)", tot_direct_norm_prop, tot_direct_norm_prop*100/total_triples)
print("external identifiers (prop and prop direct)", tot, tot*100/total_triples)
print(full_prop_df['count'].sum())
```
```
external identifiers (prop) 179679329.0 1.4015766887772972
external identifiers (prop direct) 179486559.0 1.4000768111143919
external identifiers (prop direct norm) 63666217.0 0.4966233944310215
external identifiers (prop and prop direct) 422832096.0 3.298269014317408
422832096.0
```

In [74]:
```
how_many_days(tot)
```
```
88.644 days 3.298 % of triples
```

In [76]:
```
full_prop_df = full_prop_df.sort_values('count', ascending=False)
```

In [77]:
```
full_prop_df.head(100).plot(x="label", y="count",
                kind='bar',
                figsize=(20,7),
                title="Number of items with external identifier",
                )
plt.xlabel('number of items')
plt.ylabel('count');
```

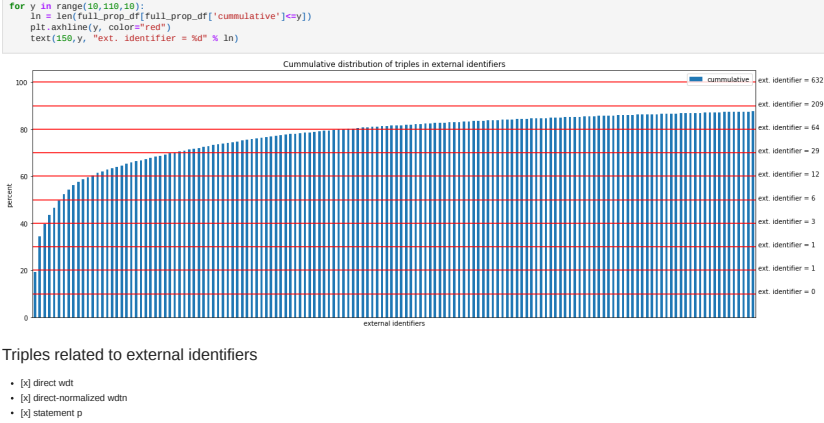


In [78]:
```
full_prop_df['percentage'] = (full_prop_df['count']*100/full_prop_df['count'].sum()).round(decimals=2)
full_prop_df['cumulative'] = full_prop_df['percentage'].expanding(1).sum()
```

In [79]:
```
full_prop_df[['code', 'label', 'count', 'percentage', 'cumulative']].head(25)  #"""
```
Out[79]:

	code	label	count	percentage	cumulative
14	P356	DOI	81478718.0	19.27	19.27
166	P698	PubMed ID	63826510.0	15.12	34.39
927	P2671	Google Knowledge Graph ID	22127888.0	5.23	39.62
1107	P3083	SIMBAD ID	16316711.0	3.86	43.48
106	P646	Freebase ID	13274336.0	3.14	46.62
344	P932	PMCID	12708778.0	3.01	49.63
409	P1566	GeoNames ID	11133098.0	2.63	52.26
3361	P5875	ResearchGate publication ID	9157348.0	2.17	54.43
67	P214	VIAF ID	8199507.0	1.94	56.37
69	P496	ORCID iD	5222820.0	1.24	57.61
123	P846	GBIF taxon ID	4571061.0	1.08	58.69
20	P244	Library of Congress authority ID	3894077.0	0.92	59.61
31	P227	GND ID	3743072.0	0.89	60.50
4775	P7859	WorldCat Identities ID	3607589.0	0.85	61.35
3440	P6179	Dimensions Publication ID	3080555.0	0.73	62.08
755	P2326	GNS Unique Feature ID	2930978.0	0.69	62.77
2625	P5055	IRMNG ID	2717119.0	0.64	63.41
57	P213	ISNI	2659310.0	0.63	64.04
97	P235	InChIKey	2631375.0	0.62	64.66
4	P234	InChI	2616244.0	0.62	65.28
60	P351	Entrez Gene ID	2312303.0	0.55	65.83
56	P637	RefSeq protein ID	2290389.0	0.54	66.37
107	P830	Encyclopedia of Life ID	2199008.0	0.52	66.89
182	P850	WoRMS-ID for taxa	2059088.0	0.49	67.38
394	P961	IPNI plant ID	2045984.0	0.48	67.86

In [80]:
```
full_prop_df.head(150).plot(y="cumulative",
             kind='bar',
             figsize=(20,7),
             title="Cummulative distribution of triples in external identifiers",
             )
plt.xticks([])
plt.xlabel('external identifiers')
plt.ylabel('percent')

for y in range(10,110,10):
    ln = len(full_prop_df[full_prop_df['cumulative']<=y])
    plt.axhline(y, color="red")
    text(150,y, "ext. identifier = %d" % ln)
```



Triples related to external identifiers

- [x] direct with
- [x] direct-normalized with
- [x] statement p
- [ ] everything inside the prop statement p: (?x prop (...other statements...)) => **going one step down the tree**
  - prop/statement ps (has the same P## code)
  - psn (has the same P## code)
- [ ] If we think of removing external identifiers (/prop), there is no point in keeping their items (/entity) and its related triples. So everything in **context = /entity/P####**
  - characteristics of the triples (?/prop ?characters ?obj)
  - of the entities (?/entity ?p ?o)

Get 1 step down from all /prop/P**** nodes (for external identifiers)

Basically this helps finds number of triples in items related to external identifiers

- ?item /prop ?b
- ?item /prop/direct "literal"
- all ?b ?x ?y

In [81]:
```
t = time.time()
spark.sql(f"""select count(subject, predicate, object)
            from discovery.wikibase_rdf
            where date={date} and wiki='wikidata'
                -- subquery
                and subject in (
                select object
                from discovery.wikibase_rdf
                where date={date} and wiki='wikidata'
                and predicate in ('{propStr}')
                )
            """).show()
time.time()-t
```
```
+---------+
|    count|
+---------+
|717745459|
+---------+
```
Out[81]:
```
2347.478277876968
```

In [82]:
```
ext_triples = 717745459
total_ext_triples = ext_triples+tot

ext_triples*100/total_triples, total_ext_triples*100/total_triples
```
Out[82]:
```
(5.598717859820156, 8.896998874152539)
```

In [83]:
```
how_many_days(ext_triples)
```
```
150.471 days 5.599 % of triples
```

In [85]:
```
print(total_ext_triples)
how_many_days(total_ext_triples)
```
```
1140577555.0
239.115 days 8.897 % of triples
```

External identifiers on items make up 8.897% of the dataset

Number of triples for defining the properties themselves

/entity is context

subjects include:

- /prop
- /entity
- /prop/direct

In [86]:
```
entityStr = "','".join(["<"+item+">" for item in results_df['item'].values.tolist()])
```

In [87]:
```
spark.sql(f"""select count(distinct subject, predicate, object)
            from discovery.wikibase_rdf
            where date={date} and wiki='wikidata'
            and context IN ('{entityStr}')
            """).show()
```
```
+------------------------------------------+
|count(DISTINCT subject, predicate, object)|
+------------------------------------------+
|                                   1187001|
+------------------------------------------+
```

In [88]:
```
ext_as_item = 1187001
```

In [89]:
```
how_many_days(ext_as_item)
```
```
0.249 days 0.009 % of triples
```

In [90]:
```
total_ext_triples, total_ext_triples*100/total_triples
```
Out[90]:
```
(1140577555.0, 8.896998874152539)
```

In [91]:
```
(ext_as_item+total_ext_triples)*100/total_triples
```
Out[91]:
```
8.90624598351368
```

In [106]:
```
## Just for fun: how many triples are associated with the famous 'instance of' property?

spark.sql(f"""select count(distinct subject, predicate, object)
            from discovery.wikibase_rdf
            where date={date} and wiki='wikidata'
            and context='<http://www.wikidata.org/entity/P31>'
            """).show(10, truncate=False)
```
```
+------------------------------------------+
|count(DISTINCT subject, predicate, object)|
+------------------------------------------+
|1257                                      |
+------------------------------------------+
```

Combined figures

In [108]:
```
others = total_triples - e_alt_label_count - ne_alt_label_count - e_desc_count - ne_desc_count - e_label_count - ne_label_count - total_ext_triples - ext_as_i

source_data={
    'predicate':['altLabel','altLabel','description','description',
                 'label','label','others','external_id', 'external_id',
                 'schema:name', 'schema:name'],
    'language':['english', 'others', 'english', 'others', 'english',
                'others', '-', 'as_props', 'as_item', 'english',
                'others'],
    'count' : [e_alt_label_count, ne_alt_label_count, e_desc_count, ne_desc_count, e_label_count, ne_label_count, others,
               total_ext_triples, ext_as_item, e_count, ne_count]
}

# Segregating the raw data into usable form
df=pd.DataFrame(source_data)

df['count'] = df['count'].astype(int)

df
```
Out[108]:

	predicate	language	count
0	altLabel	english	9157038
1	altLabel	others	93436016
2	description	english	72609016
3	description	others	2398769645
4	label	english	79778129
5	label	others	419885045
6	others	-	8377582937
7	external_id	as_props	1140577555
8	external_id	as_item	1187001
9	schema:name	english	13721324
10	schema:name	others	65064444

In [109]:
```
df['count'].sum(), total_triples
```
Out[109]:
```
(12671768950, 12671768950)
```

In [110]:
```
df['percentage of total triples'] = (df['count']/df['count'].sum())*100
df
```
Out[110]:

	predicate	language	count	percentage of total triples
0	altLabel	english	9157038	0.072263
1	altLabel	others	93436016	0.737362
2	description	english	72609016	0.572998
3	description	others	2398769645	18.930030
4	label	english	79778129	0.629574
5	label	others	419885045	3.313547
6	others	-	8377582937	66.112182
7	external_id	as_props	1140577555	9.000934
8	external_id	as_item	1187001	0.009367
9	schema:name	english	13721324	0.108283
10	schema:name	others	65064444	0.513460

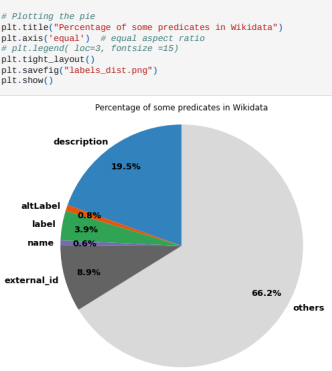
In [111]:
```
agg_df = df.groupby("predicate").sum().reset_index()
agg_df['days to recover'] = agg_df['count']/4770000
agg_df # slight difference because of analyzing external id with new dump
```
Out[111]:

	predicate	count	percentage of total triples	days to recover
0	altLabel	102593854	0.809625	21.508145
1	description	2471378661	19.503028	518.108734
2	external_id	1141764556	9.010301	239.363639
3	label	499663174	3.943121	104.751190
4	others	8377582937	66.112182	1756.306695
5	schema:name	78785768	0.621742	16.516932

Let make nice diagrams!

e_alt_label_count - ne_alt_label_count - e_desc_count - ne_desc_count - e_label_count - ne_label_count - total_ext_triples - ext_as_item - e_count - ne_count

In [117]:
```
df_dc={'predicate':['description', 'altLabel', 'label', 'name', 'external_id', 'others'],
'count':[ne_desc_count+e_desc_count,
        e_alt_label_count+ne_alt_label_count,
        e_label_count+ne_label_count,
        e_count+ne_count,
        total_ext_triples,
        others]}

# Segregating the raw data into usable form
df=pd.DataFrame(df_dc,columns=
['predicate','count'])

# Setting figure colors
cmap = plt.get_cmap("tab20c")
outer_colors = cmap(np.arange(6)*4) ## preds

# Setting the size of the figure
plt.figure(figsize=(8,8))

# Plotting the pie
plt.pie(df.iloc[:,1], labels = df.iloc[:, 0],
        startangle=90, pctdistance =0.8 ,colors=outer_colors,
        autopct = '%1.1f%%', radius= 1.0, labeldistance=1.05,
        textprops ={ 'fontsize':13, 'fontweight': 'bold'},
        wedgeprops = {'linewidth' : 0, 'edgecolor' : "w" } )

# Plotting the pie
plt.title("Percentage of some predicates in Wikidata")
plt.axis('equal')  # equal aspect ratio
# plt.legend( loc=3, fontsize =15)
plt.tight_layout()
plt.savefig("labels_dist.png")
plt.show()
```



New ext ID Count / New Triple Count = 8.896%

New ext ID Count / Old Triple Count = 9%

In [ ]: