30 Sept 2019

**Instructions for Xuankai**

**Create 2 GITHUB Folders**

1. **WTO (for wtoscrape)**
2. **ITA (for scrape db)**

**Create two subfolders for each**

1. **Meta Data**
2. **XML**

**Create WTO Dispute Meta Data Master File**

Pseudo SQL code:

TAKE *dispute*

REPLACE *dispute.status\_id* WITH *status.description* AND *status.name*

JOIN *dispute\_country*

REPLACE *dispute\_country.country\_ID*  WITH *country.ISO3*

REPLACE *dispute\_country.role*  WITH *role.role*  (so that each role is a separate column, allow for multiple values separated by comma)

JOIN *dispute\_cited\_agreement.agreement\_id*

REPLACE *dispute\_cited\_agreement.agreement\_id WITH agreement.doclode*

(like here <https://github.com/chanyaaaa/SNF-Project/blob/master/TradeData/disputes_cited_agt.csv>)

JOIN *dispute\_subject.subject\_id*

REPLACE *dispute\_subject.subject\_id* WITH *subject.name*

Final dispute csv rows:

* id
* code
* title
* short\_title
* reports\_adopted
* update\_date
* start\_date
* status\_date
* status.description
* status.name
* dispute\_country. country\_ID[Third\_Party]
* dispute\_country. country\_ID[Respondent]
* dispute\_country. country\_ID[Claimant]
* dispute\_cited\_agreement[doclode - as list]
* dispute\_subject.subject.name

CREATE EXTENSION IF NOT EXISTS tablefunc;

WITH dispute\_role\_country\_list AS (

SELECT \*

FROM crosstab(

'

SELECT dispute\_id,

r.role,

array\_to\_string(array\_agg(DISTINCT c.iso3), '','') c\_iso3

FROM dispute\_country dc,

country c,

"role" r

WHERE dc.country\_id = c.id AND

dc.role = r.id

GROUP BY dispute\_id,

r.role

',

'

SELECT DISTINCT r.role

FROM "role" r

ORDER BY 1

'

) AS ct(

dispute\_id INT,

claimant VARCHAR,

respondent VARCHAR,

third\_party VARCHAR

)

),

dispute\_agreement\_dolcode\_list AS (

SELECT d.id dispute\_id,

array\_to\_string(array\_agg(DISTINCT a.dolcode), ',') dolcode

FROM dispute d,

agreement a,

dispute\_cited\_agreement dca

WHERE dca.dispute\_id = d.id AND

dca.agreement\_id = a.id

GROUP BY d.id

)

SELECT d.id,

d.code,

title,

short\_title,

reports\_adopted,

update\_date,

start\_date,

status\_date,

st.description status\_description,

st.name status\_name,

drcl.third\_party "Country[Third\_Party]",

drcl.respondent "Country[Respondent]",

drcl.claimant "Country[Claimant]",

dadl.dolcode dolcode,

sj.name subject\_name

FROM dispute d,

status st,

subject sj,

dispute\_role\_country\_list drcl,

dispute\_agreement\_dolcode\_list dadl,

dispute\_subject dsj

WHERE d.status\_id = st.id AND

drcl.dispute\_id = d.id AND

dadl.dispute\_id = d.id AND

dsj.dispute\_id = d.id AND

dsj.subject\_id = sj.id

**Create WTO Document Meta Data Master File**

Pseudo SQL code:

TAKE *document*

LIMITED TO *Doctype = 4, 5, 7, 12* AND *reporttype.id = 1,3,4* AND *document\_url.language = 0*

JOIN dispute\_document

REPLACEdispute\_document.dispute\_id WITH *dispute.code*

JOIN *document\_doctype*

REPLACE *document\_doctype.doctype\_id* WITH *doctype.name*

JOIN *document\_reporttype*

REPLACE *document\_reporttype . reporttype\_id* WITH *reporttype.name*

JOIN *document\_product*

REPLACE *document\_product.product\_id* WITH *product*

JOIN *document\_url.url*

JOIN *document\_XML* (link or name to XML file on github)

Final document csv rows:

* id
* document\_id
* symbol
* title
* title\_en
* title\_fr
* title\_sp
* date
* derestricted\_date
* pages\_en
* pages\_fr
* pages\_sp
* status collection
* access\_level
* contents
* dispute.code
* doctype.name
* document\_reporttype.reporttype.name
* document\_product.product
* document\_url.url
* document\_XML

**(FOR wtoscrape)**

SELECT doc.\*,  
 dis.code "dispute.code",  
 dt.name "doctype.name",  
 rt.name "document\_reporttype.reporttype.name",  
 prod.name "document\_product.product.name",  
 durl.url "document\_url.url"  
FROM "document" doc,  
 doctype dt,  
 dispute dis,  
 reporttype rt,  
 product prod,  
 document\_doctype ddt,  
 document\_reporttype drt,  
 document\_url durl,  
 document\_product dprod,  
 dispute\_document dd  
WHERE dt.id IN (4, 5, 7, 12)  
 AND rt.id IN (1, 3, 4)  
 AND durl.language = 0  
 AND ddt.document\_id = doc.id  
 AND ddt.doctype\_id = dt.id  
 AND drt.document\_id = doc.id  
 AND drt.reporttype\_id = rt.id  
 AND durl.document\_id = doc.id  
 AND dprod.document\_id = doc.id  
 AND dprod.product\_id = prod.id  
 AND dd.dispute\_id = dis.id  
 AND dd.document\_id = doc.id;

**Create Investment Dispute Meta Data Master File**

Note: We have two sources for dispute meta data: (1) *dispute* from scrapedb and (2) *case* from icsid scrape. The ICSID case data should be added to the dispute data from scrapedb where entries in both databases exist.

Pseudo SQL code:

TAKE *dispute*

**(IN scrapedb)**

SELECT id, title

FROM dispute

WHERE title ILIKE '%ICSID%';

**(IN icsidscrape)**

SELECT id, caseno

FROM "case";

Then run merge\_scrapedb\_icsid.py. Inspect the output and update rules accordingly (search OUTPUTS for documentation). Outputs are dispute\_title\_with\_icsid\_caseno.csv and merged\_dispute\_icsid.csv.

**Create Investment Document Meta Data Master File**

Pseudo SQL code:

TAKE *document*

LIMITED TO *document\_type.id= 1,3,4,8*

REPLACE *document.dispute\_id* WITH *dispute.title*

JOIN *document\_type.name*

JOIN *document\_subject.name*

JOIN *document\_party.name*

JOIN *document\_detail*

Final document csv rows:

* id
* url
* title
* date
* dispute.title
* document\_type.name
* document\_detail.role [as list, document\_detail.name as values]

**(FOR oldscrapedb)**

CREATE EXTENSION IF NOT EXISTS tablefunc;

WITH doc\_role\_name AS (

SELECT \*

FROM crosstab(

'

SELECT document\_id,

"role",

"name"

FROM document\_detail

',

'

SELECT DISTINCT "role"

FROM document\_detail

ORDER BY 1

'

) AS ct( -- TODO: List dynamically?

document\_id INT,

annulment\_committee\_members VARCHAR,

annulment\_committee\_president VARCHAR,

arbitrator VARCHAR,

claimant\_appointee VARCHAR,

claimant\_counsel VARCHAR,

claimant\_expert VARCHAR,

country VARCHAR,

judge VARCHAR,

other\_counsel VARCHAR,

other\_expert VARCHAR,

president VARCHAR,

respondent\_appointee VARCHAR,

respondent\_counsel VARCHAR,

respondent\_expert VARCHAR

)

),

doc\_filtered AS (

SELECT doc.id document\_id,

url,

doc.title,

"date",

dis.title dispute\_title,

dt.name "document\_type.name"

FROM "document" doc,

dispute dis,

document\_type dt,

document\_document\_type ddt

WHERE ddt.document\_type\_id IN (1, 3, 4, 8)

AND doc.dispute\_id = dis.id

AND ddt.document\_type\_id = dt.id

)

SELECT \*

FROM doc\_filtered

LEFT JOIN doc\_role\_name

USING (document\_id);