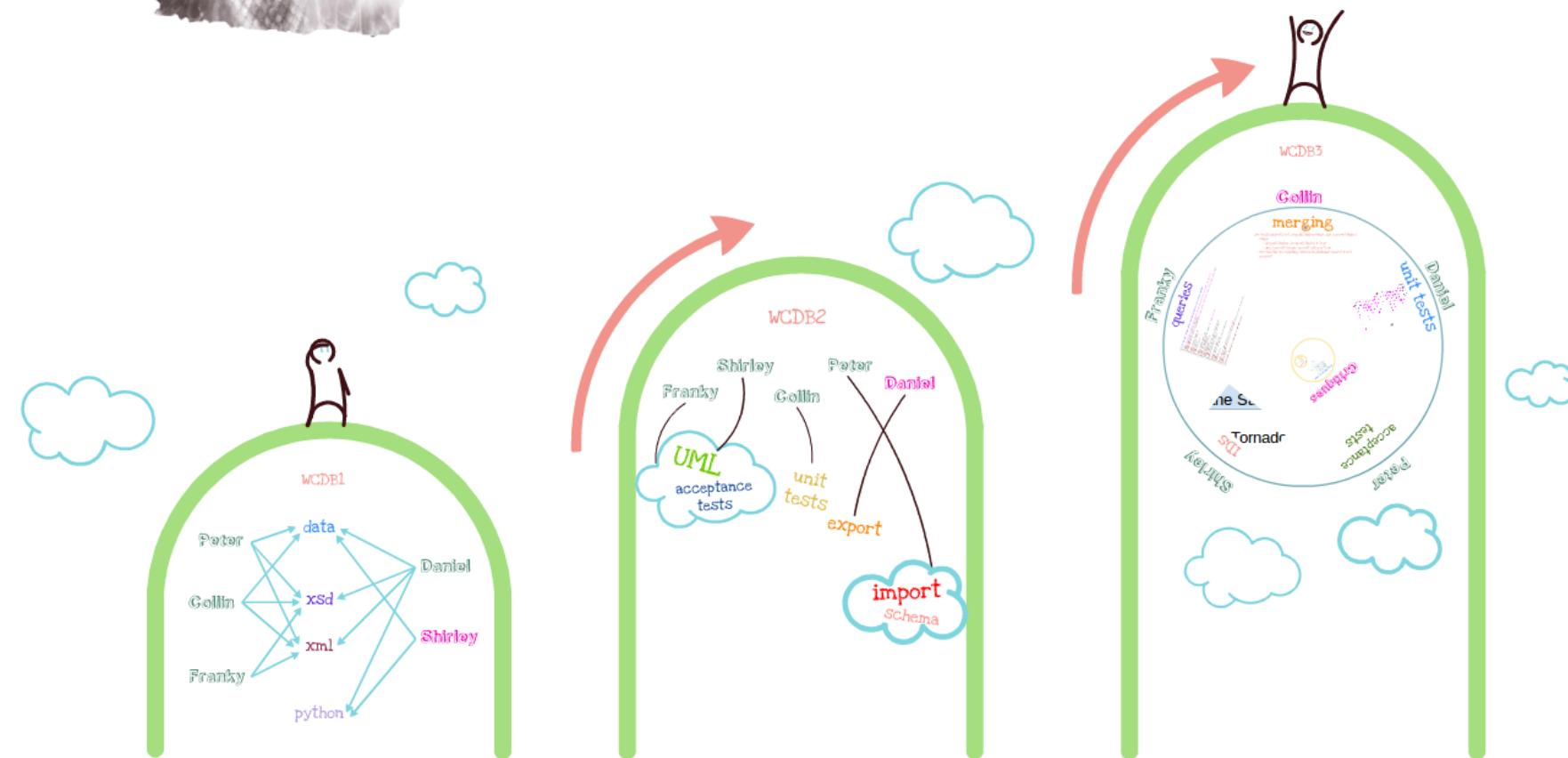


Team Poseidon



Collin Murphy
Shirley Li
Daniel Meng
Peter Li
Franky Prado



Team Poseidon

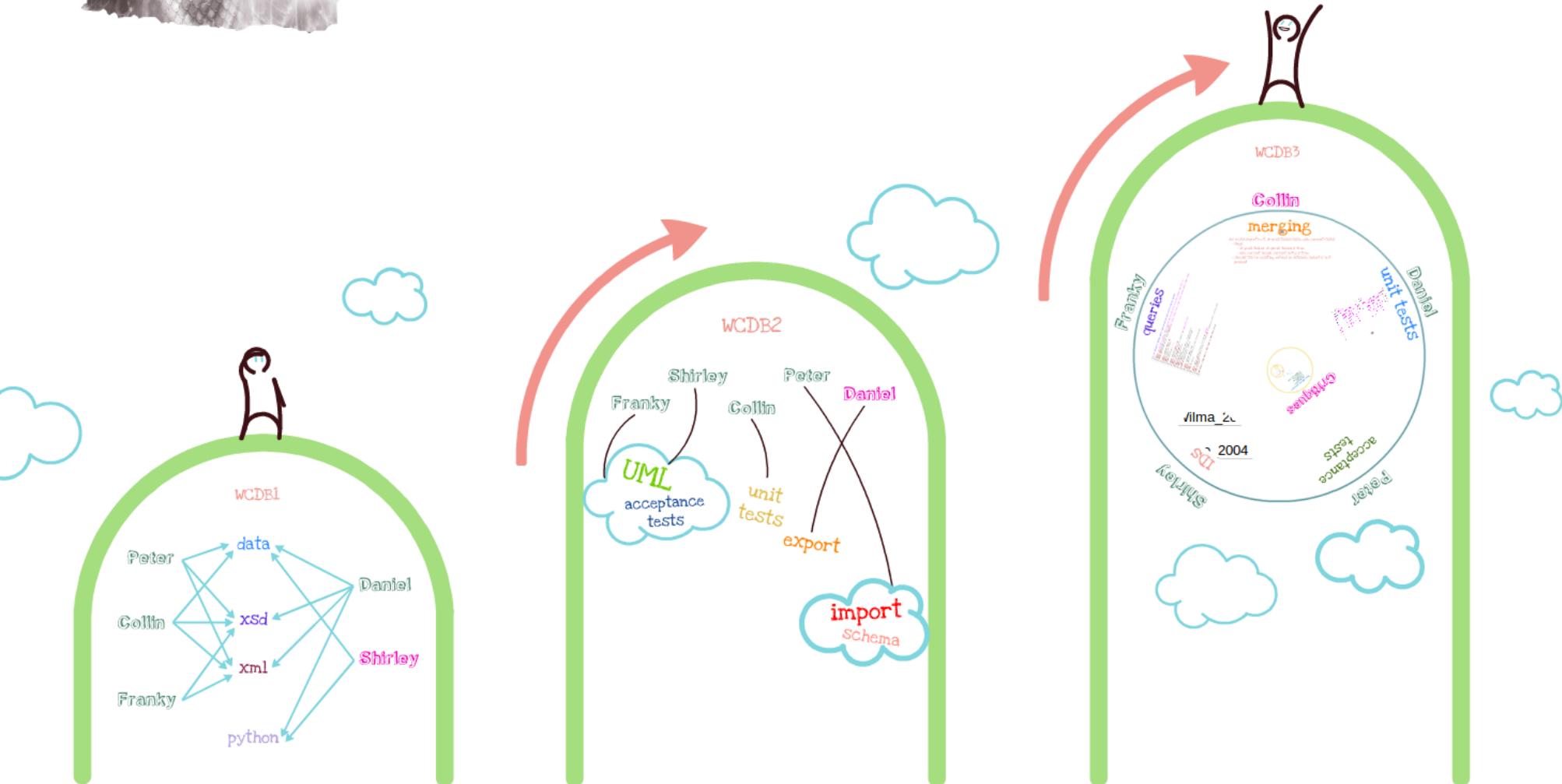


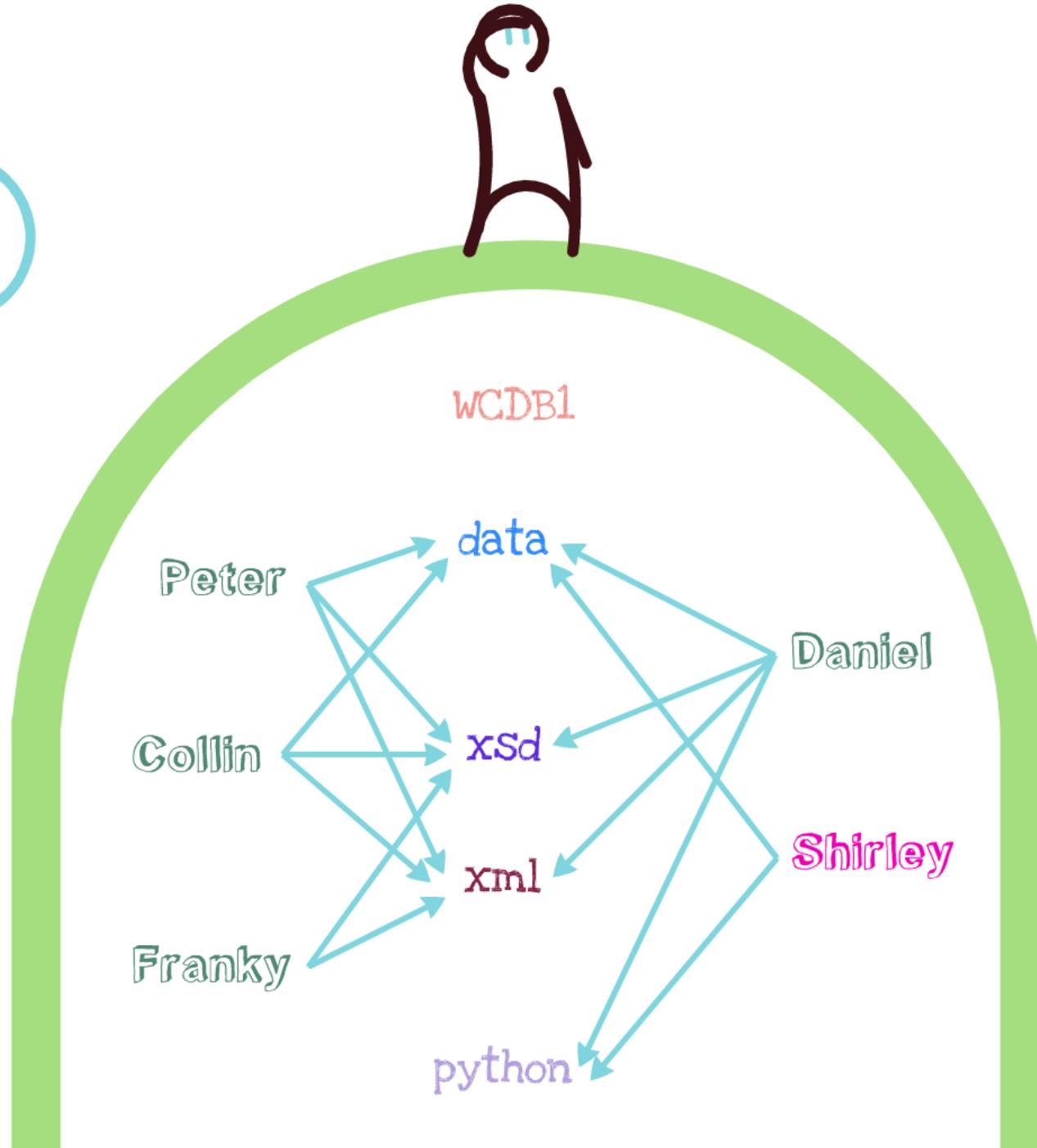
Collin Murphy
Shirley Li
Daniel Meng
Peter Li
Franky Prado

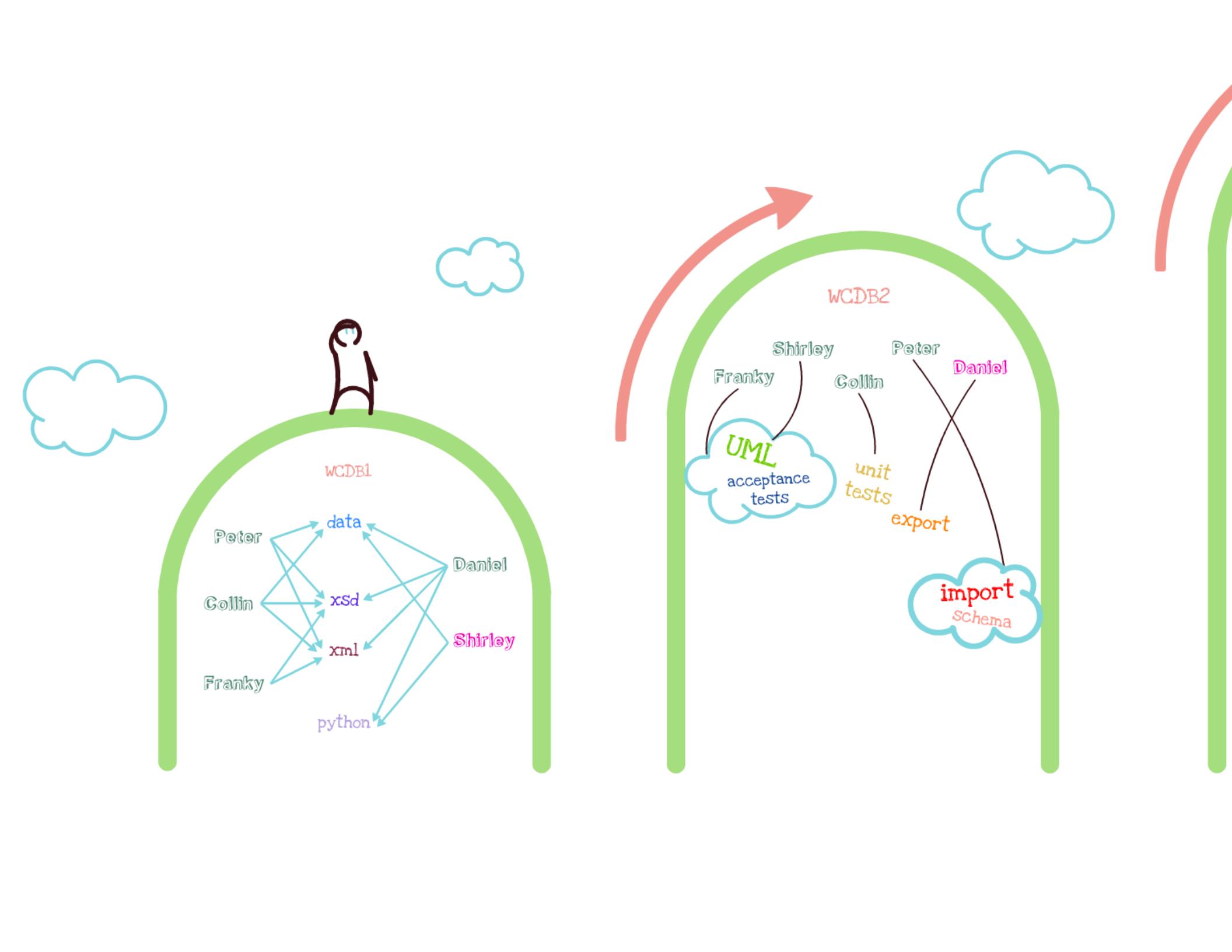
WCDB: A Database for the Web

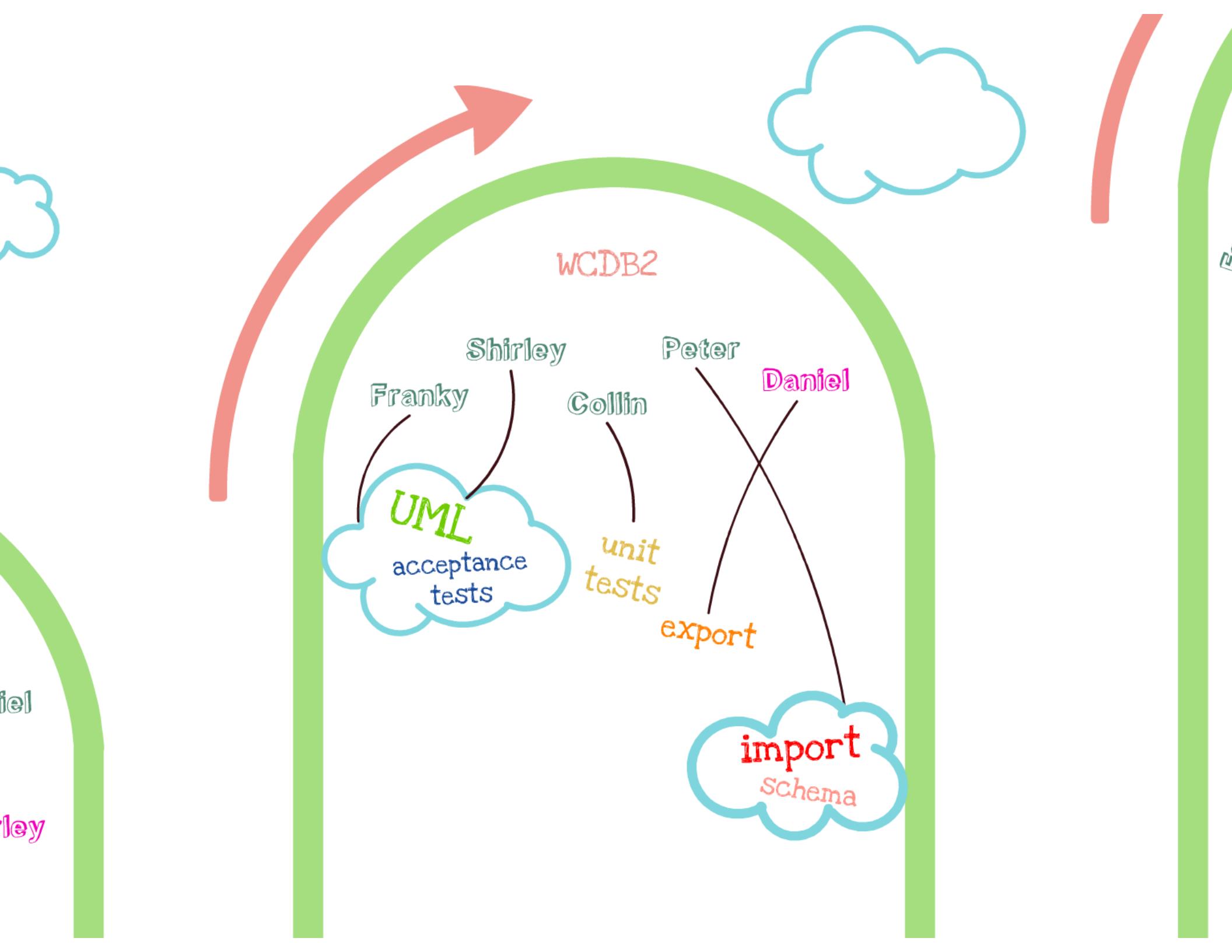


Collin Murphy
Shirley Li
Daniel Meng
Peter Li
Franky Prado

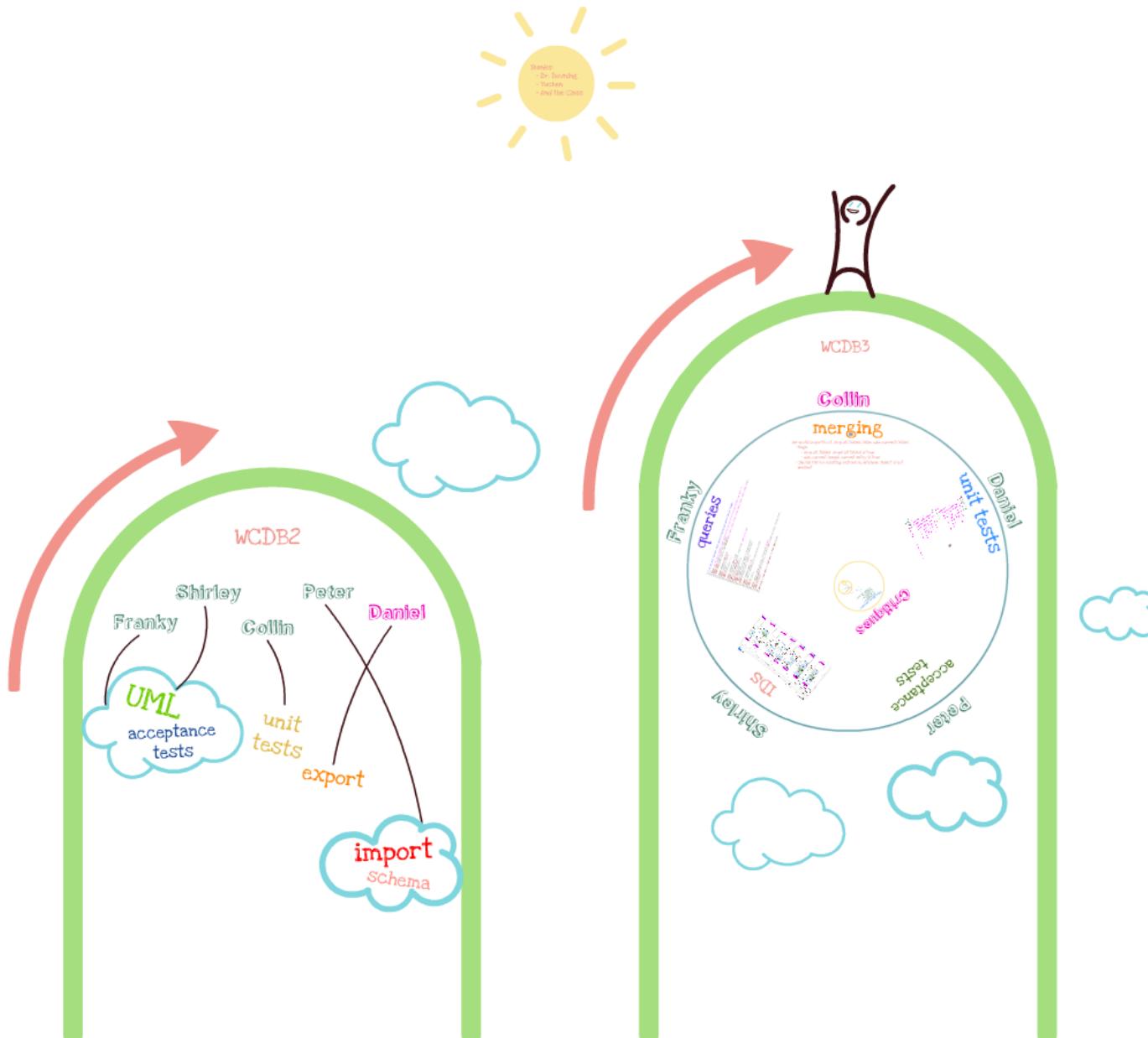
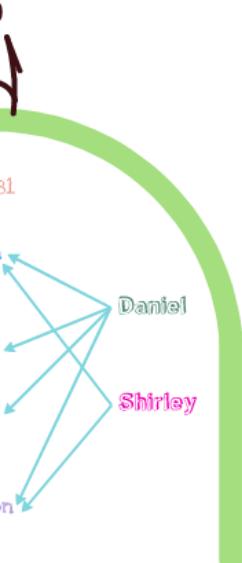


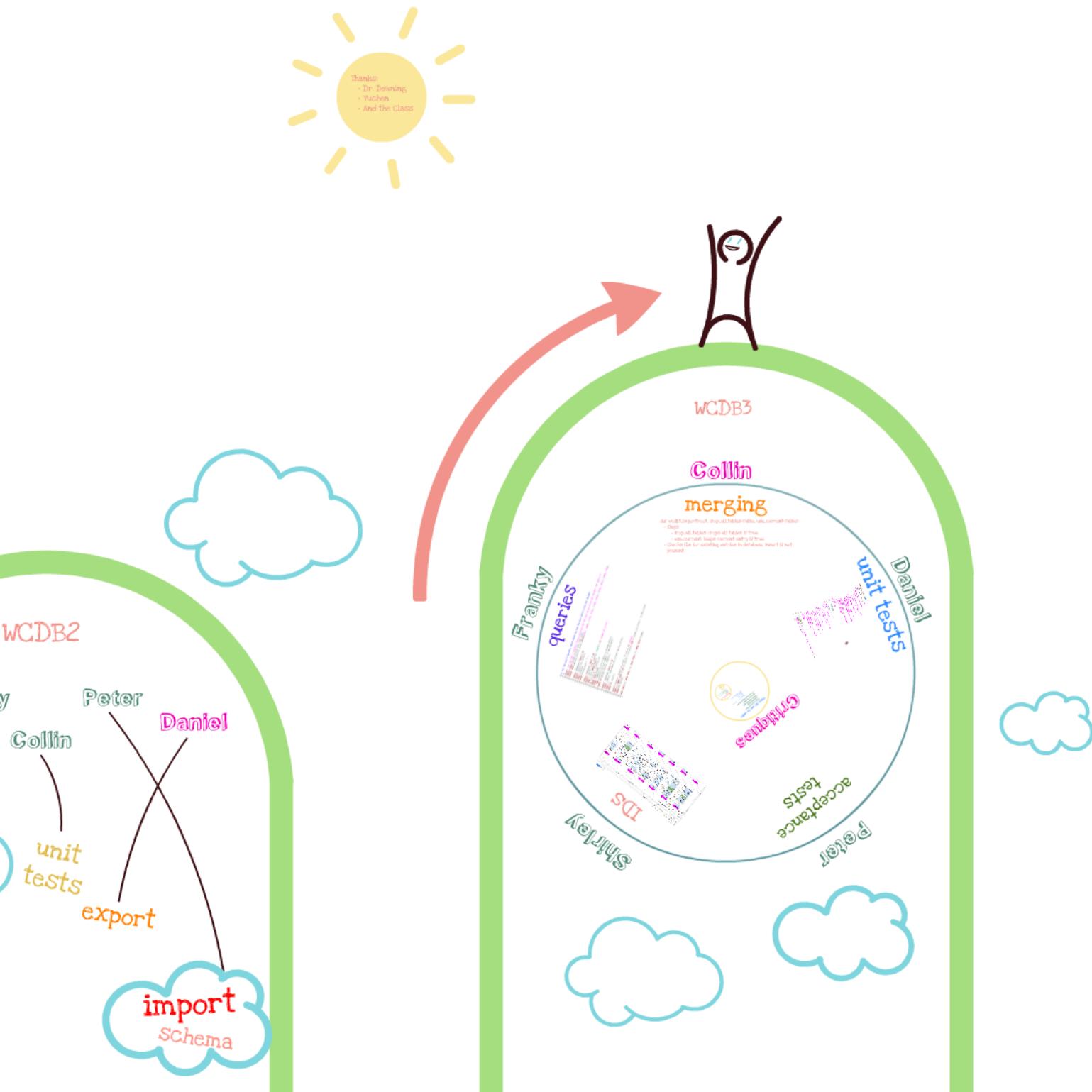


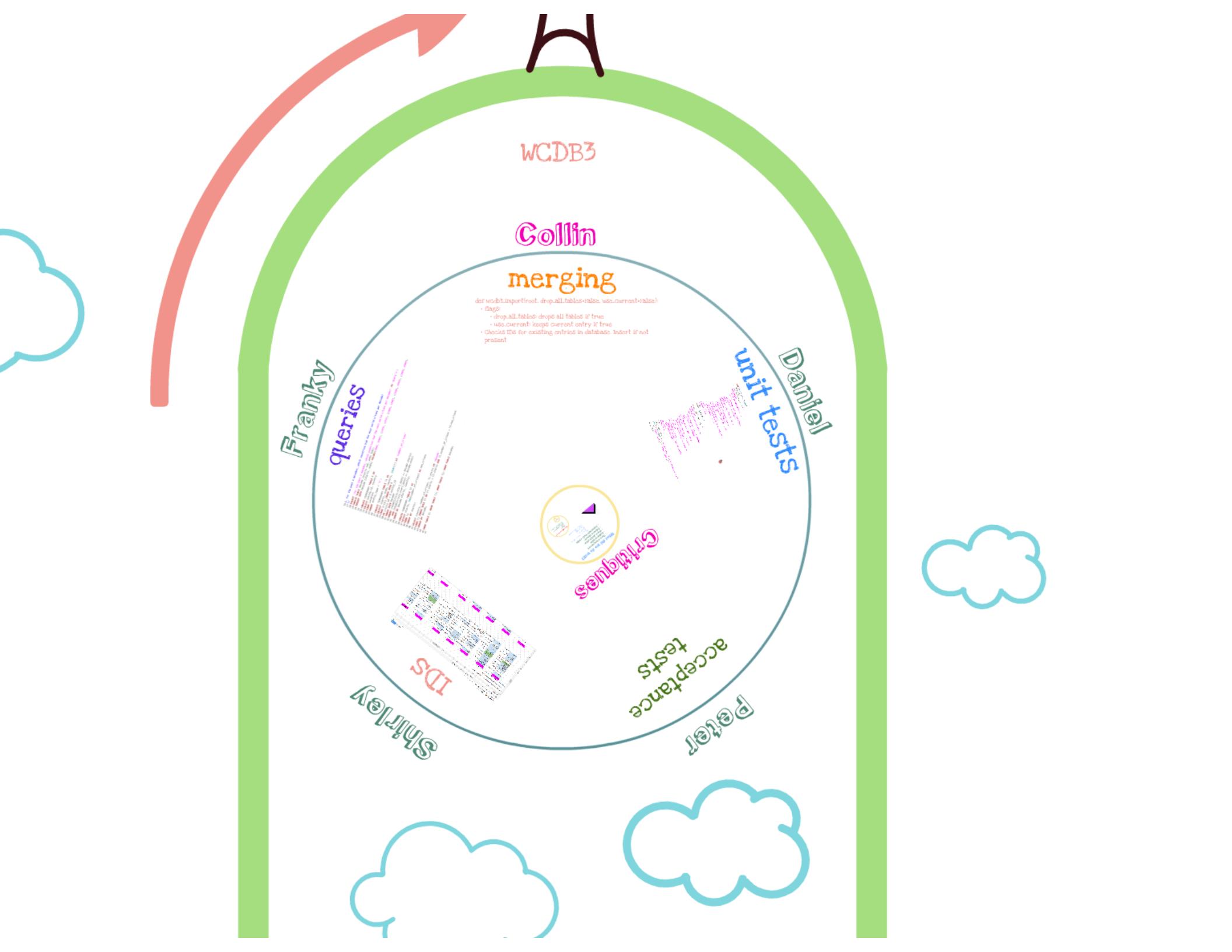




Collin Murphy
Shirley Li
Daniel Meng
Peter Li
Franky Prado







merging

```
def wcdb3_import(root, drop_all_tables=False, use_current=False):
```

- flags:
 - drop_all_tables: drops all tables if true
 - use_current: keeps current entry if true
- Checks IDs for existing entries in database. Insert if not present

```
# Insert Locations of Organizations, Persons, and Crises
for x in ('Organization', 'Crisis', 'Person'):
    for p in root.iter(x):
        entity_id = '"' + escape_quote(p.attrib.get(x.lower()+'Ident')) + '"'
        for n in p.iter('Location'):
            locality = n.find('Locality')
            locality_string = 'NULL' if (locality is None) else get_text(locality)
            region = n.find('Region')
            region_string = 'NULL' if (region is None) else get_text(region)
            country = n.find('Country')
            country_string = 'NULL' if (country is None) else get_text(country)
            if region_string == 'NULL' and locality_string != 'NULL':
                q = 'SELECT * from Location where entity_id = %s and entity_type = "%s" \
                    and country = %s and region is NULL and locality = %s' \
                    %(entity_id, x[0], country_string, locality_string)
            elif locality_string == 'NULL' and region_string != 'NULL':
                q = 'SELECT * from Location where entity_id = %s and entity_type = "%s" \
                    and country = %s and locality is NULL and region = %s' \
                    %(entity_id, x[0], country_string, region_string)
            elif locality_string == 'NULL' and region_string == 'NULL':
                q = 'SELECT * from Location where entity_id = %s and entity_type = "%s" \
                    and country = %s and locality is NULL and region is NULL' \
                    %(entity_id, x[0], country_string)
            else:
                q = 'SELECT * from Location where entity_id = %s and entity_type = "%s" \
                    and country = %s and locality = %s and region = %s' \
                    %(entity_id, x[0], country_string, locality_string, region_string)
            if not query(c, q.encode('ascii','replace')):
                q = 'INSERT INTO Location VALUES (NULL, "%s", %s, %s, %s, %s)' % \
                    (x[0], entity_id, locality_string, region_string, country_string)
                query(c, q.encode('ascii','replace'))
```

WCDB3

Collin

merging

```
def wcdb3.import(root, drop_all_tables=False, use_current=False):
    • flags:
        • drop_all_tables: drops all tables if true
        • use_current: keeps current entry if true
    • Checks IDs for existing entries in database. Insert if not
      present
```

Daniel
tests

Critiques

unit tests

```
def test_import_9(self):
    drop_all()
    root = ET.fromstring("""
<WoldCrises>
    <Person personIdent="SLi">
        <Name>
            <FirstName>Shirley</FirstName>
            <LastName>Li</LastName>
        </Name>
        <Kind personKindIdent="PR"/>
        <Location>
            <Locality>Houston</Locality>
            <Region>Texas</Region>
            <Country>United States</Country>
        </Location>
        <ExternalResources>
            <ImageURL>https://.png</ImageURL>
        </ExternalResources>
    </Person>
</WoldCrises>
""")
    wcdb3_import(root)

    root = ET.fromstring("""
<WoldCrises>
    <Person personIdent="SLi">
        <Name>
            <FirstName>Shirley</FirstName>
            <LastName>Li</LastName>
        </Name>
        <Kind personKindIdent="PR"/>
        <Location>
            <Locality>Houston</Locality>
            <Region>Texas</Region>
            <Country>United States</Country>
        </Location>
        <ExternalResources>
            <ImageURL>https://.jpg</ImageURL>
        </ExternalResources>
    </Person>
</WoldCrises>
""")
    wcdb3_import(root)
    self.assert_(query(c, 'SELECT link from ExternalResource') == \
        (('https://.png',),('https://.jpg',)))
    drop_all()
```



queries

```
16 2. For the past 5 decades, which countries had the most world crises per decade?
17 */
18 SELECT 'For the past 5 decades, which countries had the most world crises per decade?' AS 'Query 2';
19 CREATE TABLE Decade (starts int, ends int);
20 INSERT INTO Decade (starts, ends) VALUES(1960, 1969), (1970, 1979), (1980, 1989), (1990, 1999), (2000, 2009);
21
22 CREATE TEMPORARY TABLE R AS
23 SELECT entity_id, country
24 FROM Location
25 WHERE entity_type = 'C';
26
27 CREATE TEMPORARY TABLE T1 AS
28 SELECT starts, ends, country, COUNT(*) AS 'number_of_crises'
29 FROM Crisis INNER JOIN R
30 ON Crisis.id = R.entity_id
31 INNER JOIN Decade
32 WHERE (YEAR(Crisis.start_date) >= Decade.starts)
33 AND (YEAR(Crisis.start_date) <= Decade.ends)
34 GROUP BY Decade.starts, country;
35
36 CREATE TEMPORARY TABLE T2 AS
37 SELECT starts, MAX(number_of_crises) AS max_crises
38 FROM T1
39 GROUP BY starts;
40
41 SELECT country, number_of_crises, T1.starts AS 'decade'
42 FROM T1 INNER JOIN T2 ON T1.starts = T2.starts AND T1.number_of_crises = T2.max_crises
43 GROUP BY decade;
44
45 DROP TABLE R; DROP TABLE T1; DROP TABLE T2; DROP TABLE Decade;
46
```

Query 2

For the past 5 decades, which countries had the most world crises per decade?

country	number_of_crises	decade
US	1	1960
United States	3	1970
Ukraine	2	1980
United States	3	1990
United States	12	2000

IDS

General Format for WCDB

File Edit View Insert Format Data Tools Help All changes saved in Drive

Comments Share 1 other viewer

Japan_EQ_2011

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	personIdent: first and middle initials and full last name (ex: Vincent Stell => VStell)													
2	crisisIdent: specific name of crisis followed by type and year													
3	organizationIdent: initials or entire name if only one word. (ex: CIA, NATO, AFRICARE)													
4	The Miners	yes	Poseidon	yes	Virus	yes	Byte Me	yes	Bonsai	yes	Nameless	no xml	Is Yuchen Here Today?	
5	Chernobyl	Chernobyl_ACC_1986	2011 Great East Japan Earthquake and Tsunami	Japan_EQ_2011	Deepwater Horizontal Oil Spill	Deepwater_ACC_20	The Great Chicago Fire	Chicago_FR_1871	Cuban Missile Crisis	Cuba_WP_1962	Swine Flu	H1N1_DI_2009	U-2 Incident	
6	Three Mile Island Accident	TMI_ACC_1979	2010 Haitian Earthquake	Haiti_EQ_2010	Eurozone Crisis	Eurozone_EC_2009	2011 Great East Japan Earthquake and Tsunami	Japan_EQ_2011	Hurricane Katrina	HU_Katrina_2005	Bastrop Fires	Bastrop_FR_	2010 Chile Earthquake	
7	SARS Outbreak	SARS_DI_2003	Hurricane Sandy	HU_Sandy_2012	Hurricane Katrina	HU_Katrina_2005	Hurricane Katrina	HU_Katrina_2005	Vietnam War	Vietnam_WAR_1955	Mt. St. Helens Eruption	Helens_VO_	2010 Haiti Earthquake	
8	2010 Haiti Earthquake	Haiti_EQ_2010	Hurricane Katrina	HU_Katrina_2005	Hurricane Sandy	HU_Sandy_2012	Hurricane Andrew	HU_Andrew_1992	Mexican Drug War	Mexico_Drug_WAR_2006	Victorian bushfires	Victoria_FR_	Berlin Blockade	
9	2011 Great East Japan Earthquake and Tsunami	Japan_EQ_2011	Tornado Outbreak (2011)	TO_2011	2004 Indian Ocean Tsunami	TS_2004	Hurricane Ike	HU_Ike_2008	Syrian Civil War	Syrian_WAR_2011	Daulatpur-Saturia Tornados	Daulatpur_TO_	Hurricane Andrew	
10	Joplin Tornado	Joplin_TO_2011	World Trade Center Attacks	WTC_TA_2001	World Trade Center Attacks	WTC_TA_2001	Hurricane Wilma	HU_Wilma_2005	AIDS in Africa	Africa_DI_1970	Rwanda Genocide	Rwanda_GC_	Hurricane Carla	
11	Iran Earthquake	Iran_EQ_2003	Chernobyl	Chernobyl_ACC_1986	Syrian Civil War	Syrian_WAR_2011	Hurricane Ivan	HU_Ivan_2004	Egyptian Revolution (2011)	Egypt_WAR_2011	California Wildfires	California_FR_	Hurricane Emily	
12	Hurricane Katrina	HU_Katrina_2005	Syrian Civil War	Syrian_WAR_2011	West Africa Food Crisis	West_Africa_FO_20	Hurricane Charley	HU_Charley_2004	Eyjafjallajökull Volcano Eruption (2010)	Eyjafjallajökull_VO_2010	Iraqi War	WAR_Terror_2001	Hurricane Hugo	
13	World Trade Center Attacks	WTC_TA_2001	Russian Meteor (2013)	Russia_ME_2013	War in Darfur	Darfur_WAR_2003	Hurricane Sandy	HU_Sandy_2012	Human Trafficking	HT_0001	Deepwater Horizontal Oil Spill	Deepwater_ACC_20	2008 Mumbai Attack	
14	Swine Flu	H1N1_DI_2009	Deepwater Horizontal Oil Spill	Deepwater_ACC_2010	Weapons of mass destruction in North Korea	North_Korea_WP_20(1925)	Tri-State Tornado	Tri-State_TO_1925	Holocaust	Holocaust_GC_1933	SARS Outbreak	SARS_DI_2003	2010 Pakistan Floods	
15														
16														
17														
18														
19														
20														
21														
22														
23														
24	The Miners		Poseidon		Virus		Byte Me		Bonsai		Nameless		Is Yuchen Here Today?	
25	Red Crescent	RC	North Atlantic Treaty Organization	NATO	American Red Cross	ARC	American Red Cross	ARC	Doctors without Borders	DWB	Google	Google	United States Air Force	

Peter

Documentation

WCDB3 [index](#) [/v/filer4b/v20q001/xiaolong/gitrepo/cs327e-wcdb/WCDB3.py](#)

```
# -----
# imports
# -----
```

Modules

[xml.etree](#) [mysql](#) [getpass](#) [sys](#)

Functions

escape_quote(s)
Utility to escape all double quotes of a string that is used as
a string literal in a SQL statement.

get_text(node)
Utility to get the text from a ElementTree node, return "\\""
if there is no text

login()
Modified from cs327e/examples/Login.py
login to the database of the user who runs the program
return a mysql connection object

query(c, s)
This is copied from the cs327e/examples/Query.py
c is a mysql connection object
s is query string
return a tuple of tuples

wcdb3_export(c)
export a DB to an ElementTree
c is mysql connection object
returns outroot, an Element Tree

wcdb3_import(root, drop_all_tables=False, use_current=False)
wcdb3_import(xml, drop_all_tables=False, use_current=False)
Import a valid xml into a database.
By default, tables will not be dropped upon calling the function.
Also by default, new values supercede pre-existing ones with the same key.

wcdb3_read(r)
r is reader
returns root, an Element Tree
copied from WCDB1.py

wcdb3_solve(rlist, w, overwrite)
rlist is a list of readers
w is a writer
if overwrite is true then old information will be overwritten

wcdb3_write(root, w)
w is writer
takes in root
writes XML from Element Tree
copied from WCDB1.py

Issue Tracker

Search or type a command Explore Gist Blog Help shirsureok

PRIVATE DanielMeng / cs327e-wcdb 0 0

Code Network Pull Requests 0 Issues 2 Wiki Graphs

Browse Issues Milestones Search: Issues & Milestones...

Everyone's Issues 57

	Assigned to you	Created by you	Mentioning you
	2	22	0

No milestone selected

Labels

bug	5
enhancement	10
question	1
duplicate	0
invalid	0
wontfix	0

[Manage Labels](#)

New label

New label name

2 Open 57 Closed Sort: Newest

Reopen Label Assignee Milestone

WE HAVE NO ISSUES. WE'RE PROSPEROUS, WE'RE TALKING TO YOU, WE'RE BRIGHT!

Issue	Opened by	Created at
Export wrong tag name ExternalResource, which should be ExternalResources #59	xiaolongli	2 days ago
wait for all groups to update their xmls #58	shirsureok	4 days ago
update Null element of the table? #57	DanielMeng	4 days ago
add more unit test for import #56	DanielMeng	5 days ago
update the logs from Git Repository #55	DanielMeng	8 days ago
update UML diagram. #52	DanielMeng	8 days ago
update Documentation #51	DanielMeng	8 days ago
Implementation: merging the xmls #50	DanielMeng	8 days ago
Update Acceptance Tests when all groups finished pushed their updated xmls #49	DanielMeng	8 days ago
update Unit Tests for imports #48	DanielMeng	8 days ago
Issue Tracker #47	DanielMeng	8 days ago
figure out how Yuchen gonna handle the DB login stuff so we can change #46		

Critiques

What did we do well?

- Time management
- Collaboration
- Work distribution
- Schema chosen
- Automatic login script

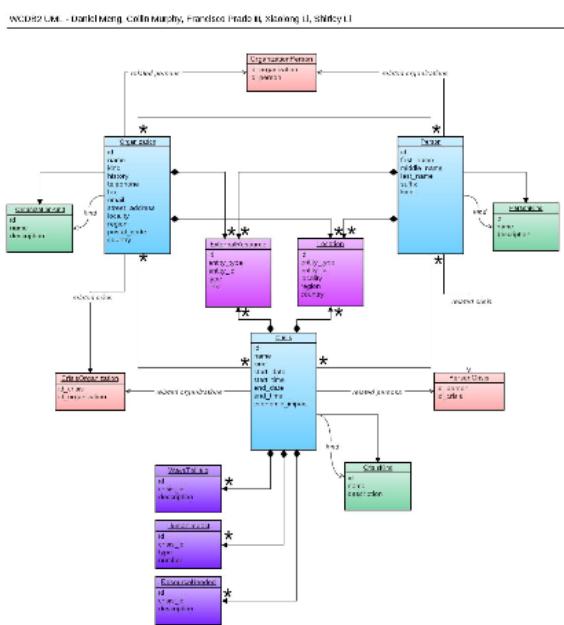


What did we learn?

- XML/XSD
- MySQL
- Advanced python libraries
- Pair programming
- You can't please everyone...

What did we do well?

- Time management
 - Collaboration
 - Work distribution
 - Schema chosen
 - Automatic login script

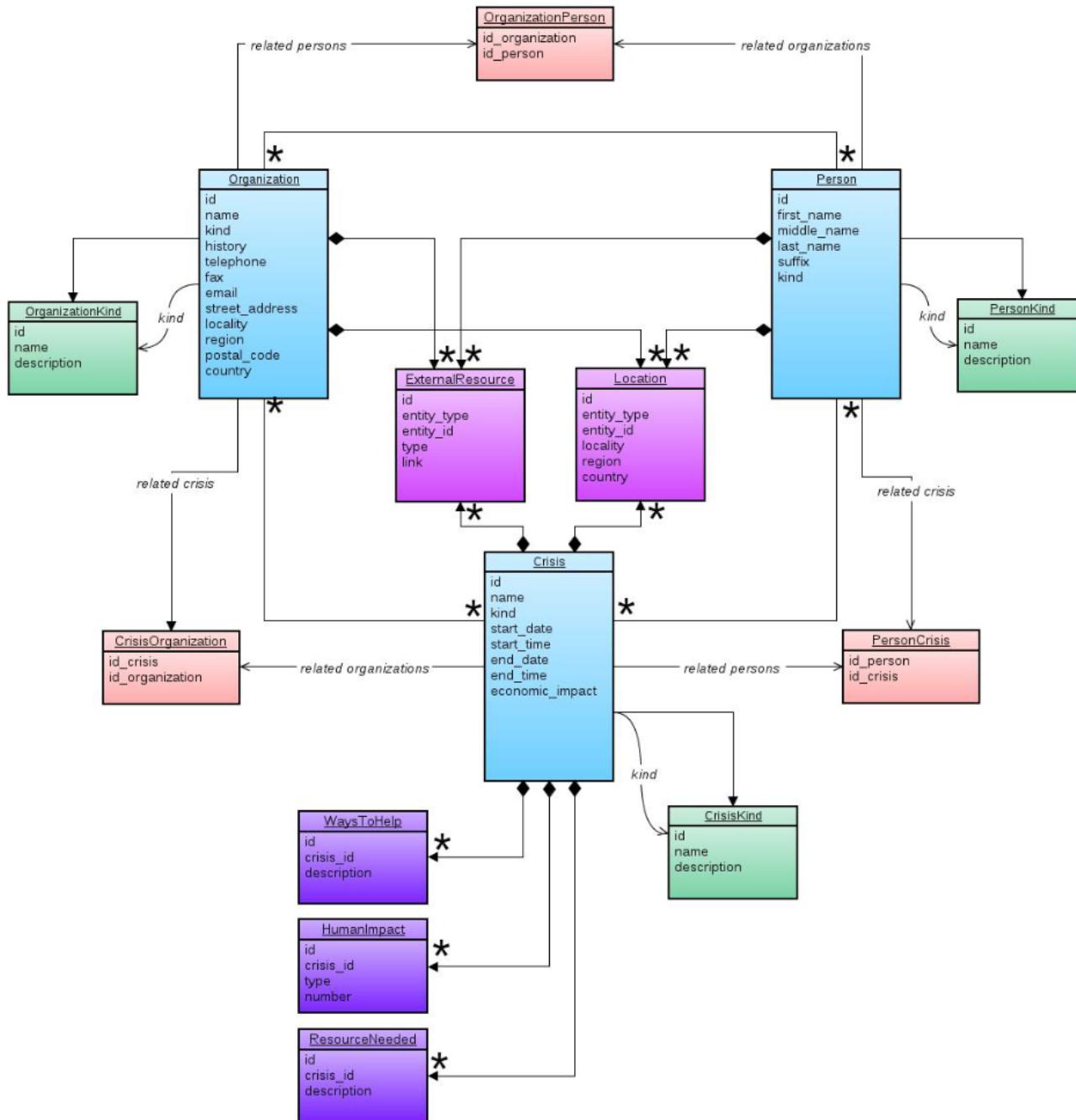


login:

```
for line in $$(`cat /u/z/users/cs327e/$$USER/.zinfo`); \
do \
PASSWORD=$$line; \
done \
&& echo $$PASSWORD \
&& db=cs327e_$$USER \
&& mysql -h z -u $$USER -p$$PASSWORD
```

What did we learn?

- XML/XSD
 - mySQL



What did we learn?

- XML/XSD
- mySQL
- Advanced python libraries
- Pair programming
- You can't please everyone...

What could we do better?

- Change fields in XSD
- More complex merging
- Agreement with other groups on naming

What puzzles me?

What could we do better?

- Change fields in XSD
- More complex merging
- Agreement with other groups on naming

What puzzles us?

- Balance between generality and simplicity
 - Country: String vs ISO3 Code
 - EconomicImpact: String vs Int
- Choice of merging algorithm

What puzzles us?

- Balance between generality and simplicity
 - Country: String vs ISO3 Code
 - EconomicImpact: String vs Int
- Choice of merging algorithm

A yellow circular graphic resembling a sunburst or a stylized flower, centered in the middle of the slide. It has several yellow, rounded, horizontal bars of varying lengths radiating outwards from the center. The bars are slightly darker at the ends and have a thin black outline.

Thanks:

- Dr. Downing
- Yuchen
- And the Class