Variability-aware data migration tool

David Benavides together with David Romero,
José A. Galindo and José M. Horcas

benavides@us.es @davbencue

DIVERSO LAB
international computing

UNIVERSIDAD DE SEVILLA

Conjecture: Data management as a cause of technical debt

However…..

- No mention to data in Wikipedia technical debt entry
- No explicit mention in most cited paper on technical debt according to google scholar

The Journal of Systems and Software 101 (2015) 193–220

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

A systematic mapping study on technical debt and its management

Zengyang Li[a,*], Paris Avgeriou[a], Peng Liang[b,c]

[a] Department of Mathematics and Computing Science, University of Groningen, Nij
[b] State Key Lab of Software Engineering, School of Computer, Wuhan University, Li
[c] Department of Computer Science, VU University Amsterdam, De Boelelaan 1081a

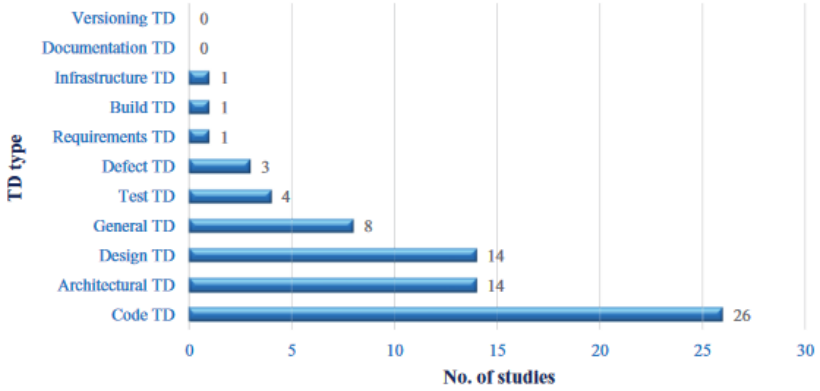
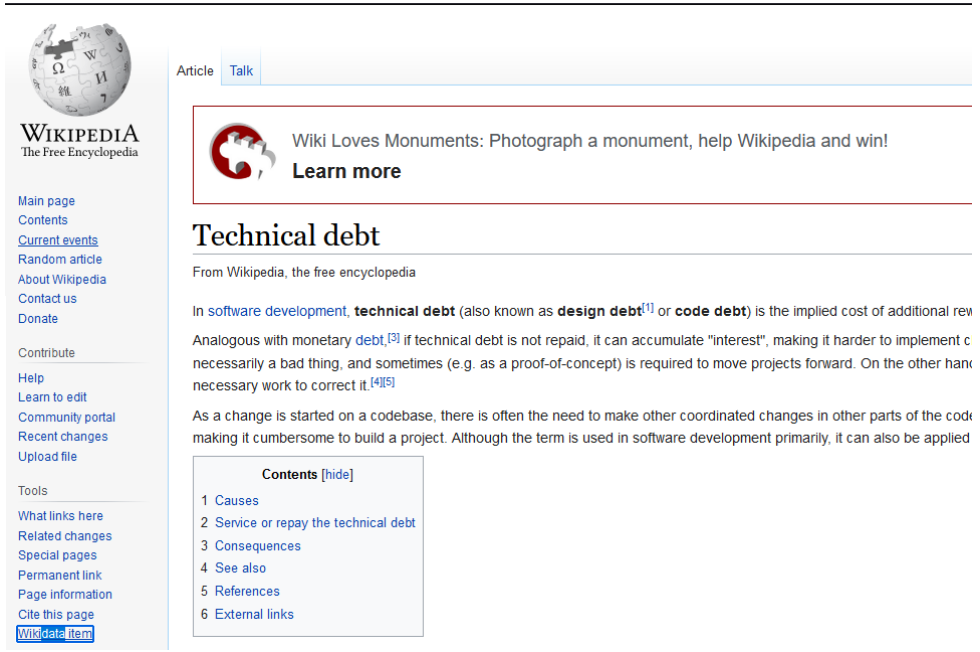


Fig. 12. Distribution of studies on TDM approaches over TD types.



WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Current events
Random article
About Wikipedia
Contact us
Donate

Contribute
Help
Learn to edit
Community portal
Recent changes
Upload file

Tools
What links here
Related changes
Special pages
Permanent link
Page information
Cite this page
Wikidata item

Article Talk

Wiki Loves Monuments: Photograph a monument, help Wikipedia and win!
Learn more

Technical debt

From Wikipedia, the free encyclopedia

In software development, **technical debt** (also known as **design debt**[1] or **code debt**) is the implied cost of additional rew

Analogous with monetary debt,[3] if technical debt is not repaid, it can accumulate "interest", making it harder to implement c
necessarily a bad thing, and sometimes (e.g. as a proof-of-concept) is required to move projects forward. On the other hanc
necessary work to correct it.[4][5]

As a change is started on a codebase, there is often the need to make other coordinated changes in other parts of the code
making it cumbersome to build a project. Although the term is used in software development primarily, it can also be applied

Contents [hide]

1 Causes
2 Service or repay the technical debt
3 Consequences
4 See also
5 References
6 External links

AGENDA

Motivation

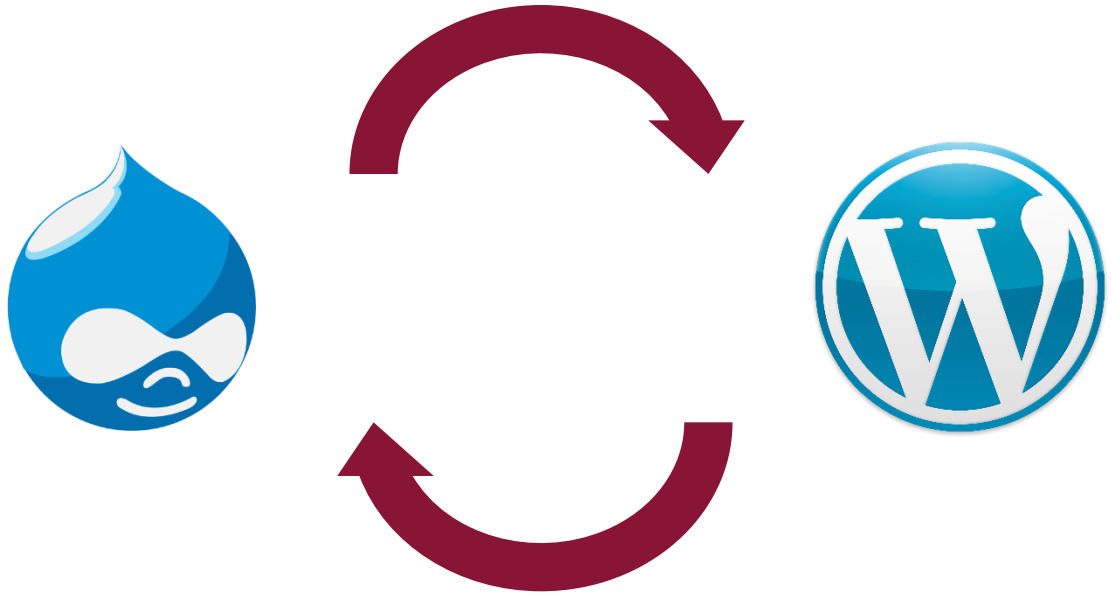
Our solution

Preliminary implementation

Discussion

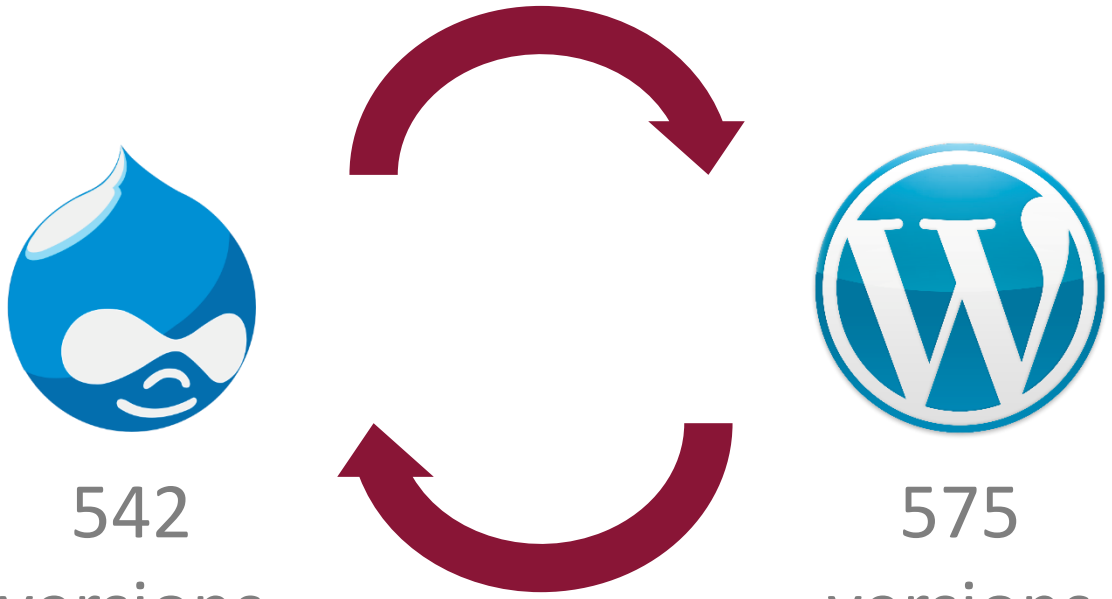
Conclusions

Motivating example



How to migrate from Drupal to WordPress and vice versa?

Motivating example



542
versions

575
versions

$$S = 2 \cdot V_A \cdot V_B = 2 \cdot 542 \cdot 575 = \mathbf{623.300}$$

Conjencture

Can we use variability management techniques to face this complexity?

AGENDA

Our Goal

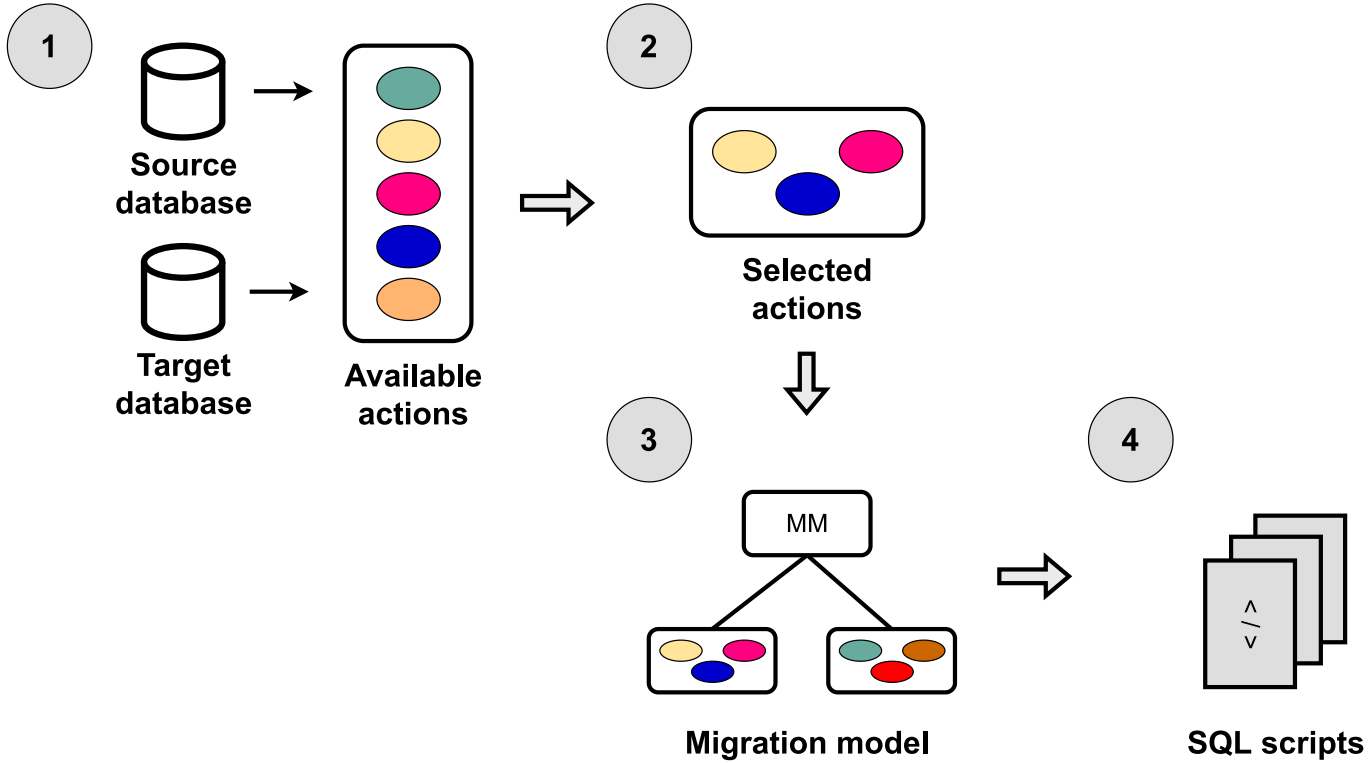
Our solution

Preliminary implementation

Discussion

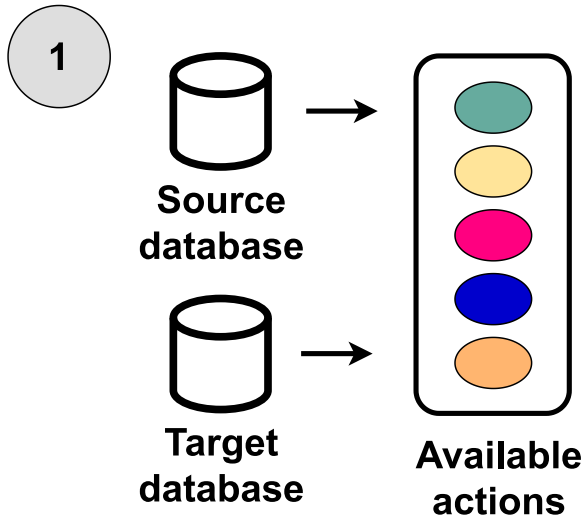
Conclusions

Database transformation using a migration product line



Our solution

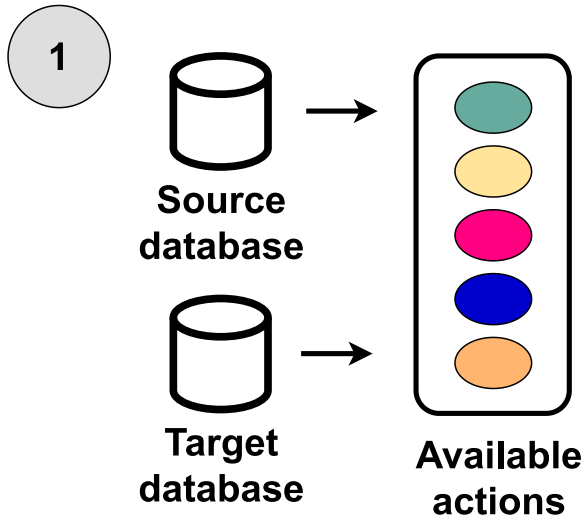
Extract simplified model




```xml
<?xml version="1.0" encoding="utf-8" standalone="yes" ?>
<sdm>

    <entity id="post">
        <name>Post</name>
        <attribute>
            <name>title</name>
            <type>varchar(25)</type>
        </attribute>
        <attribute>
            <name>body</name>
            <type>text</type>
        </attribute>
    </entity>

    <entity id="post_meta">
        <name>PostMeta</name>
        <attribute>
            <name>author</name>
            <type>varchar(25)</type>
        </attribute>
    </entity>

    <relation>
        <one id="post"></one>
        <many id="post_meta"></many>
    </relation>

</sdm>
```

Our solution

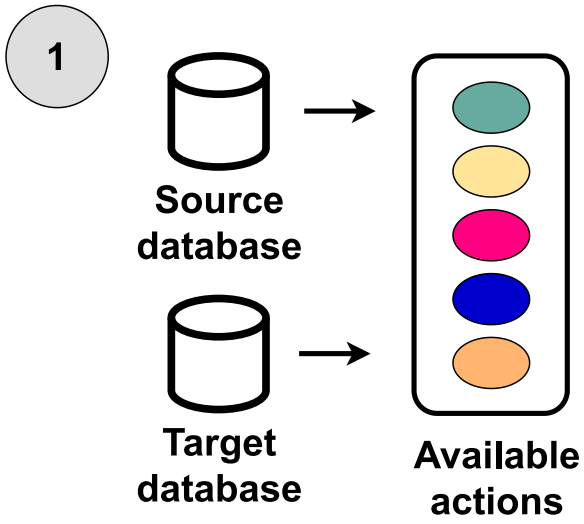
General actions



Concerning entities

- create entity
- rename entity
- delete entity

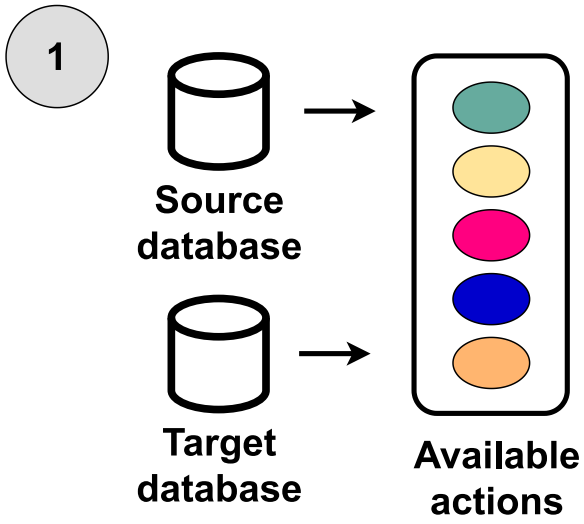
General actions



Concerning attributes

- create attribute
- move attribute
- rename attribute
- change attribute type
- delete attribute

General actions

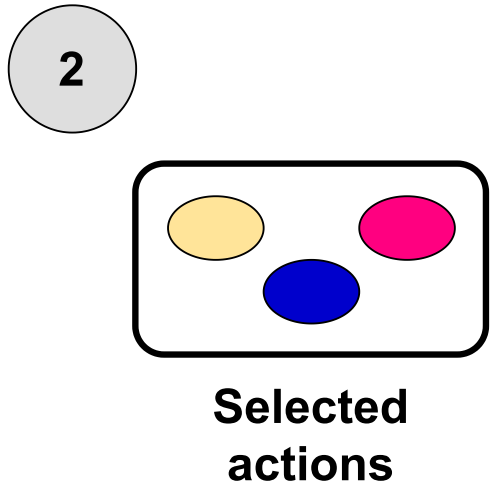


Concerning relationships

- create relationship
- remove relationship
- edit relationship

Our solution

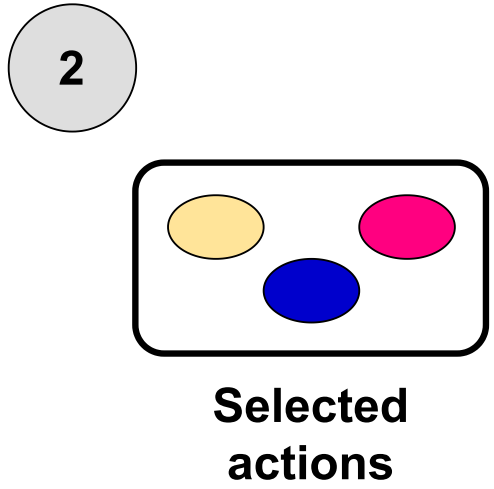
Select actions from the available ones



The selection will affect the structure of the the target database

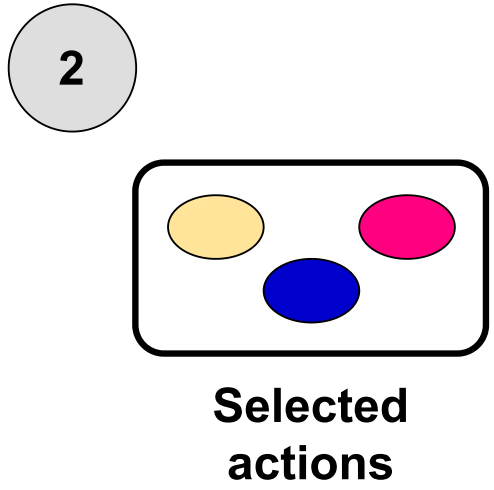
Our solution

Select actions



Option 1: Manually
user selects actions via an interactive menu

Select actions



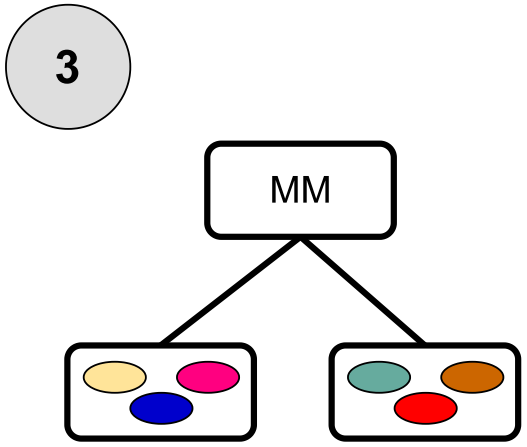
Option 2: By heuristic

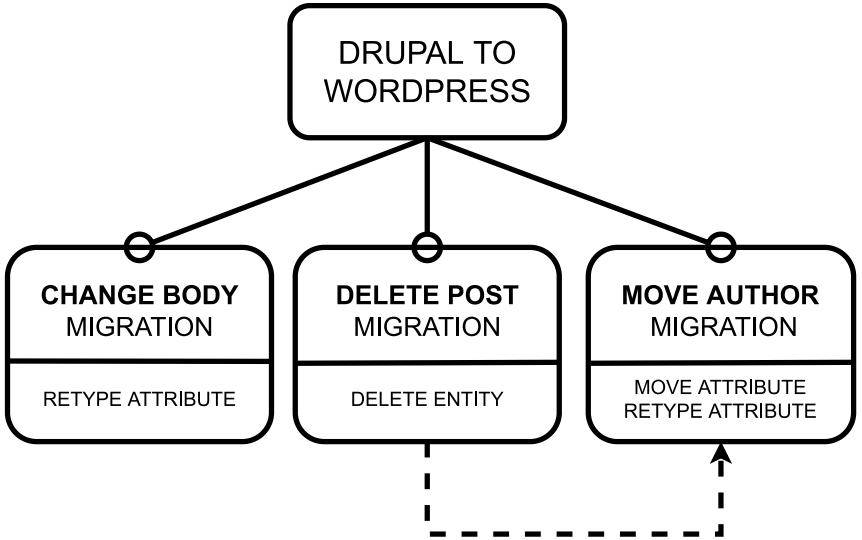
fitness function: difference between entities, attributes and relationships

$$H = dif(E_x, E_y) + \sum_{n=1}^{|E_x|} \sum_{m=1}^{|E_y|} dif(At_n, At_m) + dif(T_n, T_m)$$

Our solution

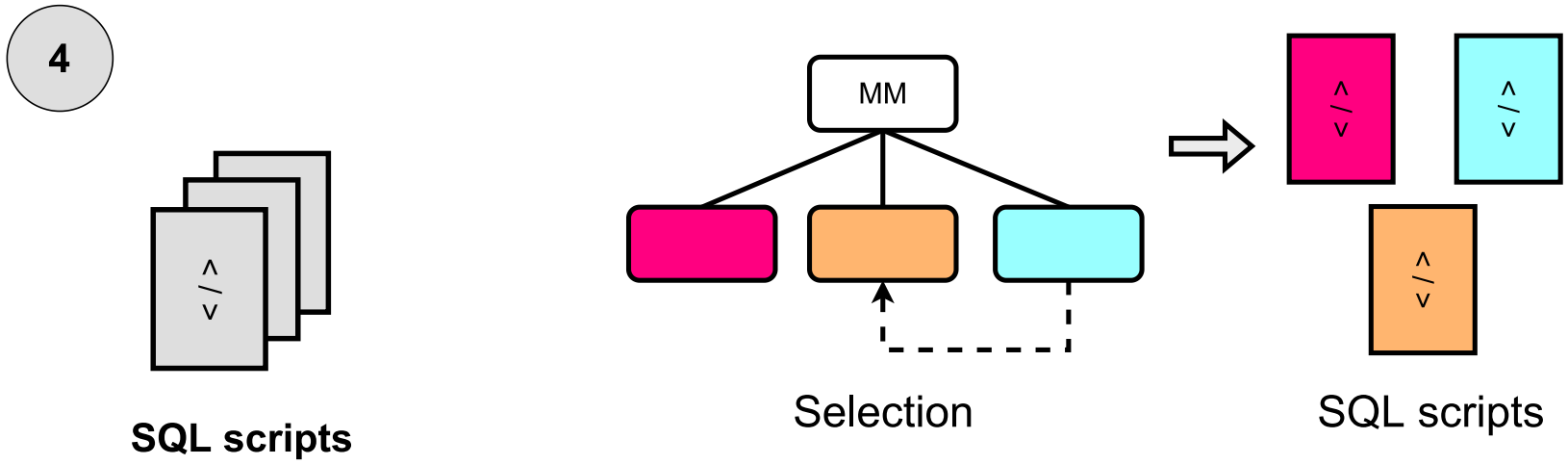
Create the migration model





Our solution

Generate the migration (SQL) scripts



AGENDA

Our Goal

Some methodological aspects

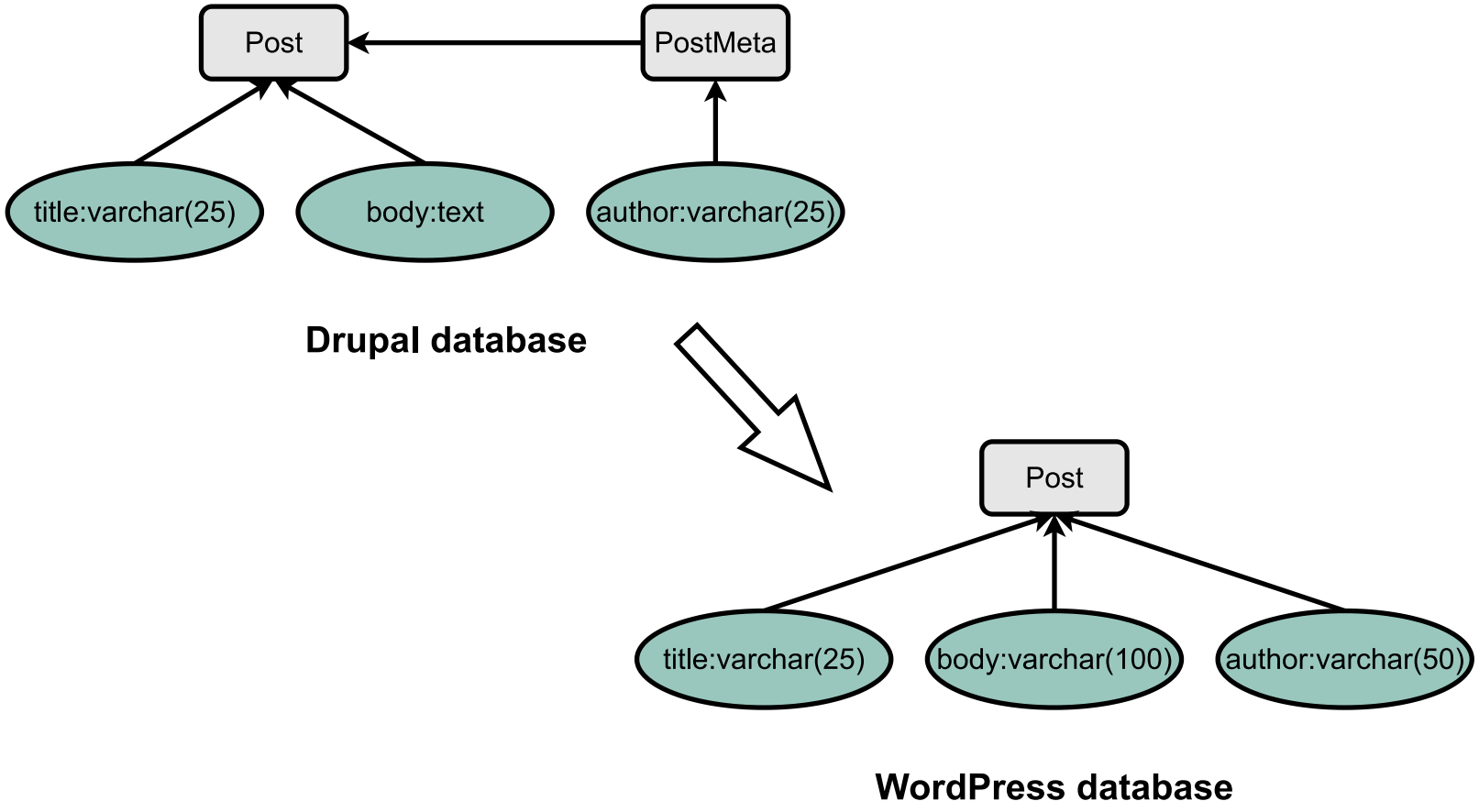
Preliminary implementation

Discussion

Conclusions

Implementation

An example



Selecting actions

##
Available actions
##

6 available actions

0 -> RenameAttributeAction
rename attribute title to author in Post

1 -> RetypeAttributeAction
retype attribute body to varchar(100) in Post

2 -> MoveAttributeAction
move attribute author : varchar(25), from PostMeta to Post

3 -> CreateAttributeAction
new attribute: author : varchar(25) in entity Post

4 -> DeleteEntityAction
delete entity: post_meta

5 -> DeleteAttributeAction
delete attribute body from Post

Select an available action ('q' for quit):

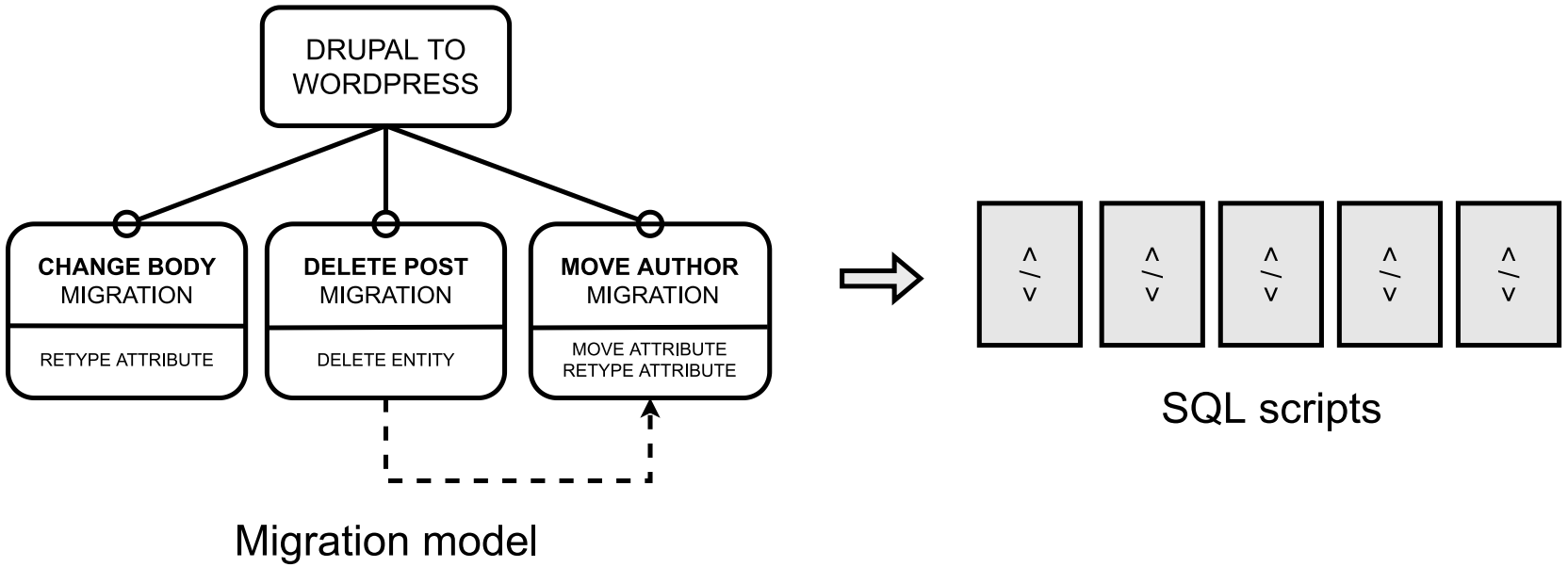
Selecting actions

```
########################################
Available actions
########################################

6 available actions

0 -> RenameAttributeAction
rename attribute title to author in Post

1 -> RetypeAttributeAction
retype attribute body to varchar(100) in Post

2 -> MoveAttributeAction
move attribute author : varchar(25), from PostMeta to Post

3 -> CreateAttributeAction
new attribute:  author : varchar(25) in entity Post

4 -> DeleteEntityAction
delete entity:  post_meta

5 -> DeleteAttributeAction
delete attribute body from Post

Select an available action ('q' for quit):
```

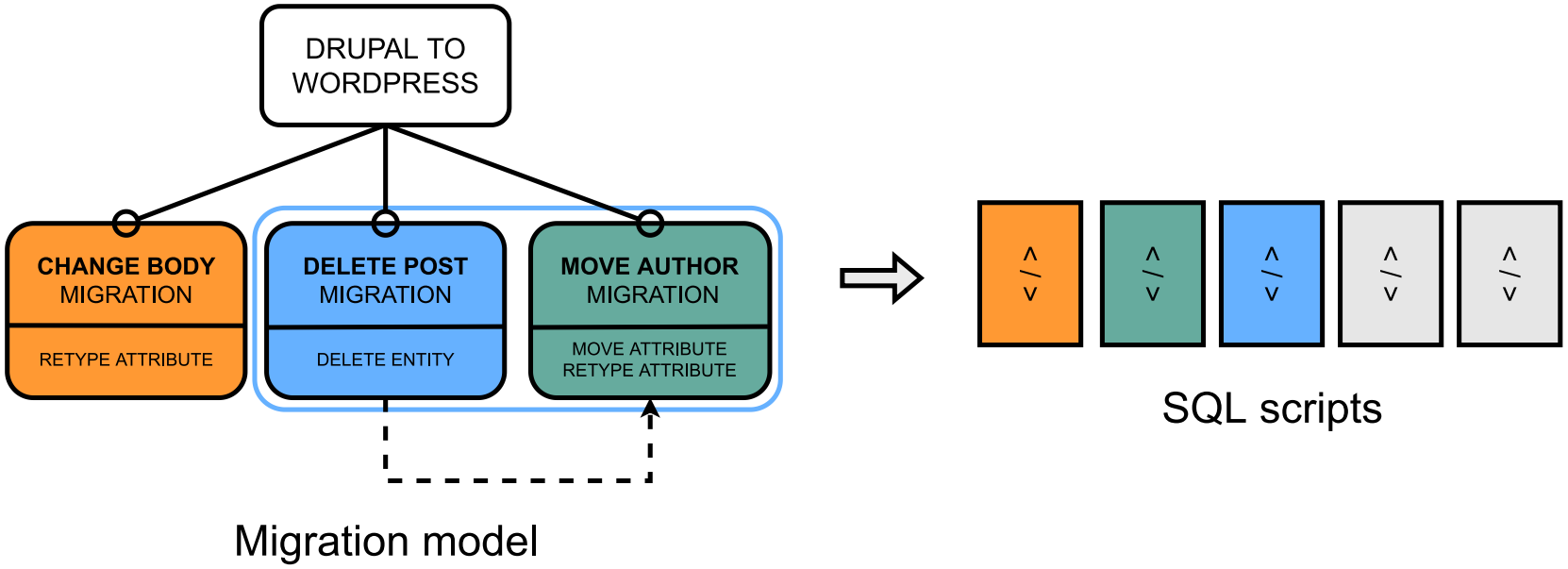
Selecting actions

```
########################################
Available actions
########################################
4 available actions

0 -> RenameAttributeAction
rename attribute title to author in Post

1 -> MoveAttributeAction
move attribute author : varchar(25), from PostMeta to Post

2 -> CreateAttributeAction
new attribute:  author : varchar(25) in entity Post

3 -> DeleteEntityAction
delete entity:  post_meta

Select an available action ('q' for quit):
```

Generate SQL scripts



Generate SQL scripts



AGENDA

Our Goal

Some methodological aspects

Findings

Discussion

Conclusions

Conclusions

- Is data migration related to technical debt?
- We are pioneers in migrating data using SPL techniques (AFAWK)
- Problems when dealing with 2-way migration and identifying commonalities

Variability-aware data migration tool

David Benavides together with David Romero,
José A. Galindo and José M. Horcas

benavides@us.es @davbencue

DIVERSO LAB
international computing

UNIVERSIDAD DE SEVILLA