BIOGRAPHY SYNOPSES FOR EVOLVING RELATIONAL DATABASE SCHEMATA

Theofanis Giachos
Supervisor: Panos Vassiliadis

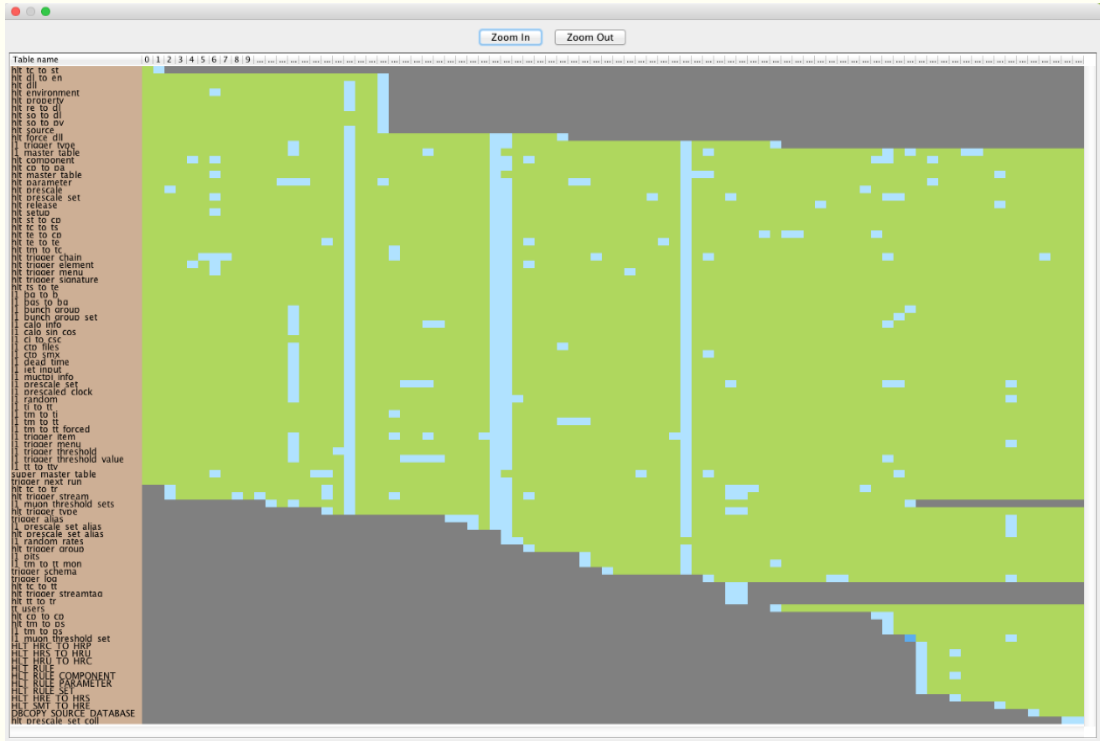


Table name
hlt tc to st
hlt dl to en
hlt dll
hlt environment
hlt property
hlt re to dl
hlt so to dl
hlt so to pv
hlt source
hlt force dll
l1 trigger type
l1 master table
hlt component
hlt cp to pa
hlt master table
hlt parameter
hlt prescale
hlt prescale set
hlt release
hlt setup
hlt st to cp
hlt tc to ts
hlt te to cp
hlt te to te
hlt tm to tc
hlt trigger chain
hlt trigger element
hlt trigger menu
hlt trigger signature
hlt ts to te
l1 bg to b
l1 bgs to bg
l1 bunch group
l1 bunch group set
l1 calo info
l1 calo sin cos
l1 ci to csc
l1 ctp files
l1 ctp smx
l1 dead time
l1 jet input
l1 muctpi info
l1 prescale set
l1 prescaled clock
l1 random
l1 ti to tt
l1 tm to ti
l1 tm to tt
l1 tm to tt forced
l1 trigger item
l1 trigger menu
l1 trigger threshold
l1 trigger threshold value
l1 tt to ttv
super master table
trigger next run
hlt tc to tr
hlt trigger stream
l1 muon threshold sets
hlt trigger type
trigger alias
l1 prescale set alias
hlt prescale set alias
l1 random rates
hlt trigger group
l1 pits
l1 tm to tt mon
trigger schema
trigger log
hlt tc to tt
hlt trigger streamtag
hlt tt to tr
tt users
hlt cp to cp
hlt tm to ps
l1 tm to ps
l1 muon threshold set
HLT HRC TO HRP
HLT HRS TO HRU
HLT HRU TO HRC
HLT RULE
HLT RULE COMPONENT
HLT RULE PARAMETER
HLT RULE SET
HLT HRE TO HRS
HLT SMT TO HRE
DBCOPY SOURCE DATABASE
hlt prescale set coll

Structure

- Introduction
- Problem Specification
- Creating an overview of the history of a schema
- Plutarch’s Parallel Lives Results
- Conclusions and Open Issues

Structure

- **Introduction**
- Problem Specification
- Creating an overview of the history of a schema
- Plutarch’s Parallel Lives Results
- Conclusions and Open Issues

General Facts

- Studying the evolution of database schemata is of great importance.
- A change in the schema of the database can impact the entire ecosystem of applications that are built on top of the database.
- The study of schema evolution entails extracting schema versions and their delta changes from software repositories, subsequently leading to the extraction of patterns and regularities.
- The history of a typical database can consist of hundreds of transitions and includes a potentially large number of tables.

Visual Inspection

- One of the main tools to study schema evolution is the **visual inspection of the history of the schema.**
- This can allow the scientists to construct research hypotheses as well as to drill into the details of inspected phenomena and try to understand what has happened at particular points in time.
- Such a representation is targeted for **a two-dimensional representation target** in a computer screen or a printed paper.
- **The space available in these representation media is simply too small** for encompassing the hundreds of transitions from one version to another and the hundreds of tables involved in such a history.

Our Solution

- The main idea is the **creation of a synopsis** of the history of the schema evolution.
- The number of transitions is abstracted by a limited set of phases and the number of tables is represented by a limited number of table clusters.
- Then, we can represent this synopsis as a 2D diagram, with the details of change in the contents of this 2D space.

How we achieve our solution (1/2)

Phase Extraction

- We introduce a hierarchical agglomerative clustering algorithm that merges the most similar transitions.
- As a result we can have a desired number of phases, each of which encompasses subsequent and similar transitions.

Cluster Extraction

- We introduce another hierarchical agglomerative clustering algorithm that creates a desired number of clusters.
- Within each cluster, the desideratum is to maximize the similarity of the contained tables.

How we achieve our solution (2/2)

Plutarch’s Parallel Lives tool

Plutarch’s Parallel Lives (in short, **PPL**) tool combines our abovementioned contribution and allows an interactive exploration of the history of schema.

Functionalities:

- Production of a detailed visualization of the life of the database, called Parallel Lives Diagram
- Production of an overview for this visualization, which has the extracted phases in its x-axis and the extracted clusters in its y-axis
- Zooming into specific points
- Filters according to specific criteria
- Details on demand

Terminology (1/2)

- **Schema Version**, or simply, **Version**: A snapshot of the database schema, committed to the public repository that hosts the different versions of the system
- **Dataset**: A sequence of versions, respecting the order by which they appear in the repository that hosts the project to which the database belongs
- **Transition**: The fact that the database schema has been migrated from version vi to version vj, $i < j$
- **Revision**: A transition between two sequential versions, i.e., from version vi to version vi+1
- **History** of a database schema: A sequence of revisions

Terminology (2/2)

For each transition, for each relation, we can identify the following data:

- **Old Attributes**: The set of attributes of the relation at the source, old version of the transition
- **New Attributes**: The set of attributes of the relation at the target, new version of the transition
- **Attributes Inserted**: The set of attribute names inserted in the relation in the new version of the transition
- **Attributes Deleted**: The set of attribute names deleted from the relation during the transition from the old to the new version
- **Attributes with Type Alternations**: The set of attributes whose data type changed during a transition.
- **Attributes involved in Key Alternations**: The set of attributes that reversed their status concerning their participation to the primary key of the relation between the old and the new version of the transition

Visual representation of a history of a database (1/3)

Parallel (Table) Lives Diagram of a database schema: a two dimensional rectilinear grid having all the revisions of the schema's history as columns and all the relations of the diachronic schema as its rows

Each cell *PLD*[*i*,j] represents the changes undergone and the status of the relation at row *i* during the revision *j*.

Visual representation of a history of a database (2/3)

- The blue cells correspond to transitions where some form of change occurred to the respective table.
- Dark cells denote that the table was not part of the database at that time.
- Green solid cells denote zero change.

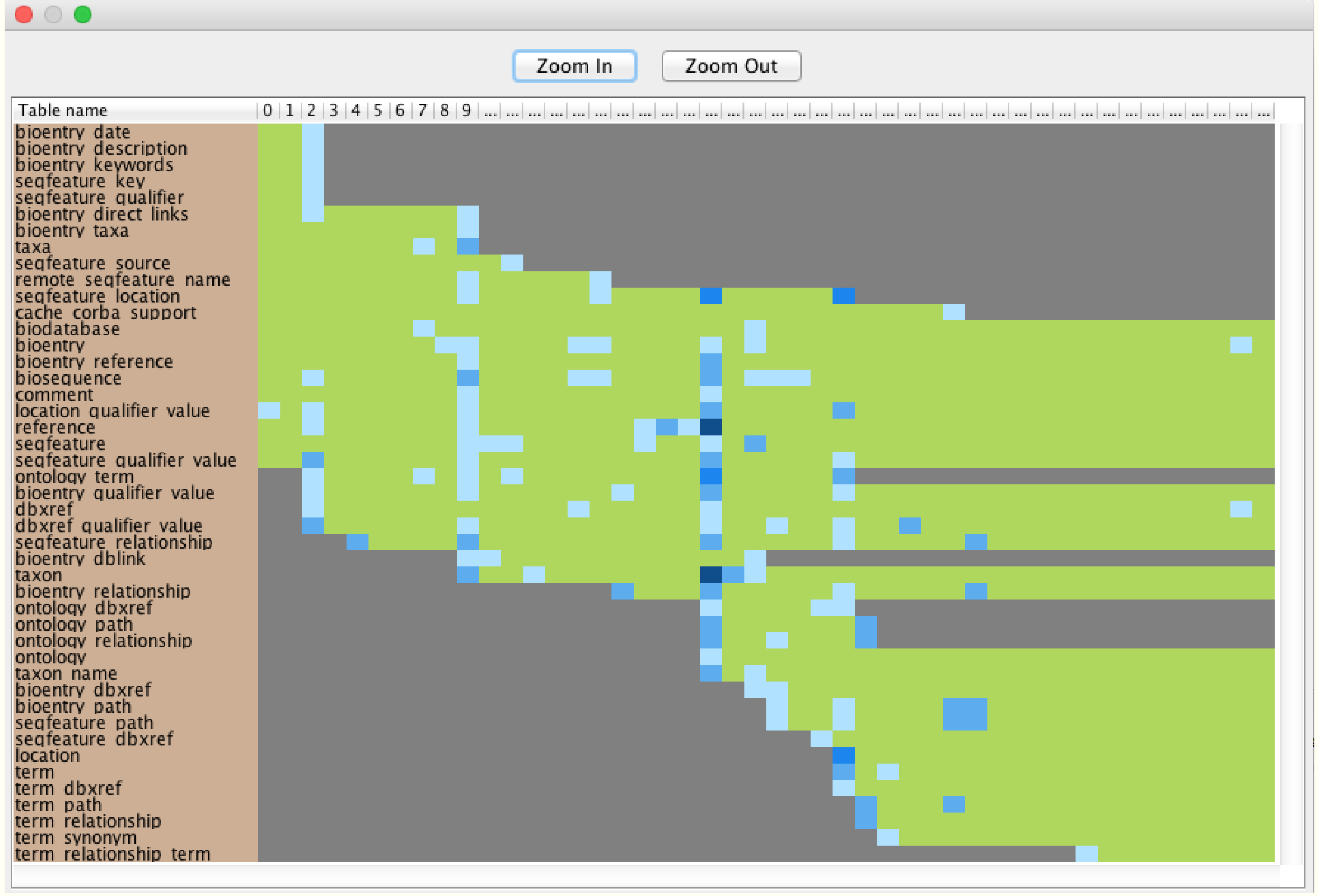
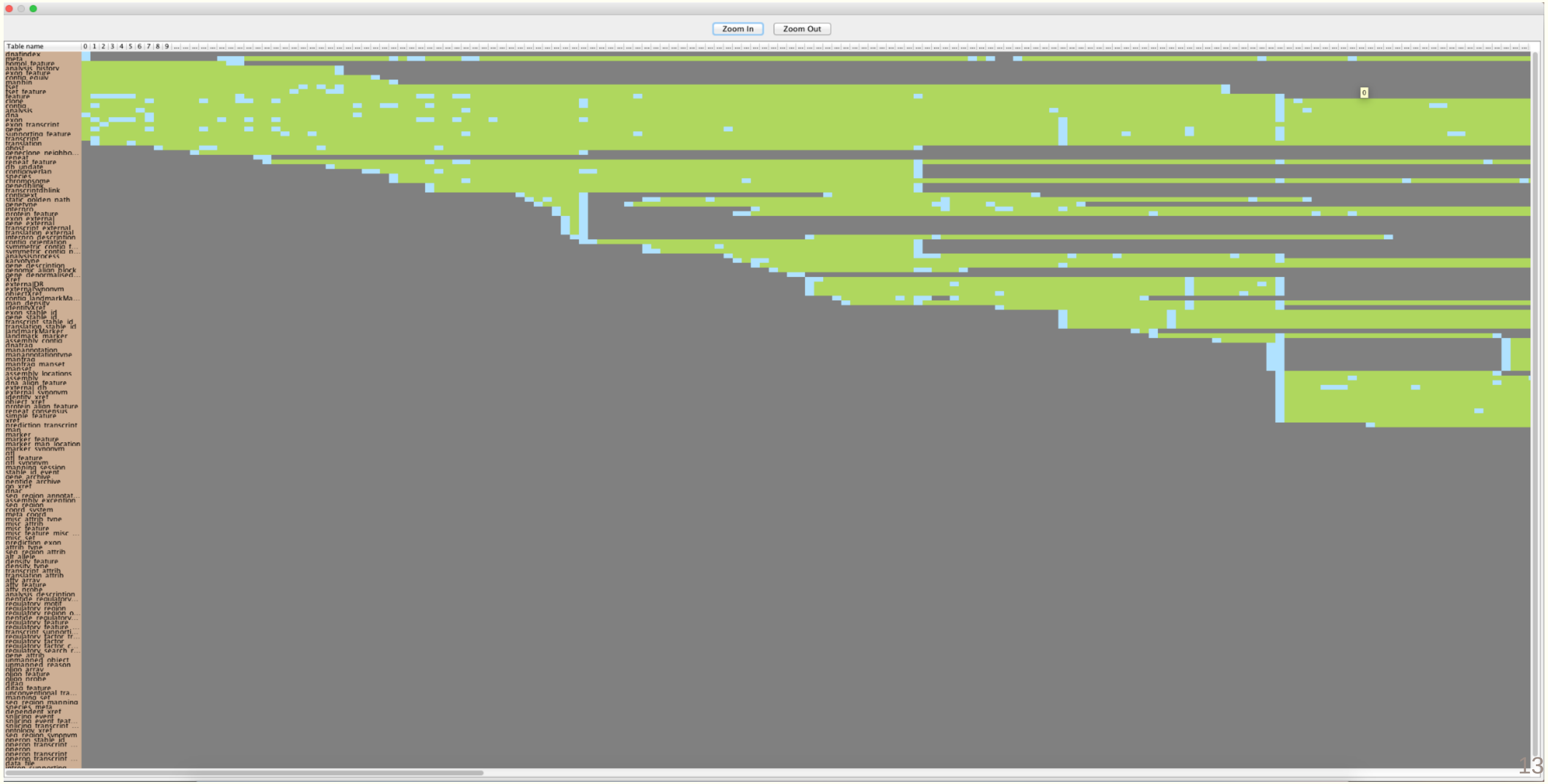


Table name | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ...

bioentry date
bioentry description
bioentry keywords
seqfeature key
seqfeature qualifier
bioentry direct links
bioentry taxa
taxa
seqfeature source
remote seqfeature name
seqfeature location
cache corba support
biodatabase
bioentry
bioentry reference
biosequence
comment
location qualifier value
reference
seqfeature
seqfeature qualifier value
ontology term
bioentry qualifier value
dbxref
dbxref qualifier value
seqfeature relationship
bioentry dblink
taxon
bioentry relationship
ontology dbxref
ontology path
ontology relationship
ontology
taxon name
bioentry dbxref
bioentry path
seqfeature path
seqfeature dbxref
location
term
term dbxref
term path
term relationship
term synonym
term relationship term

Visual representation of a history of a database (3/3)



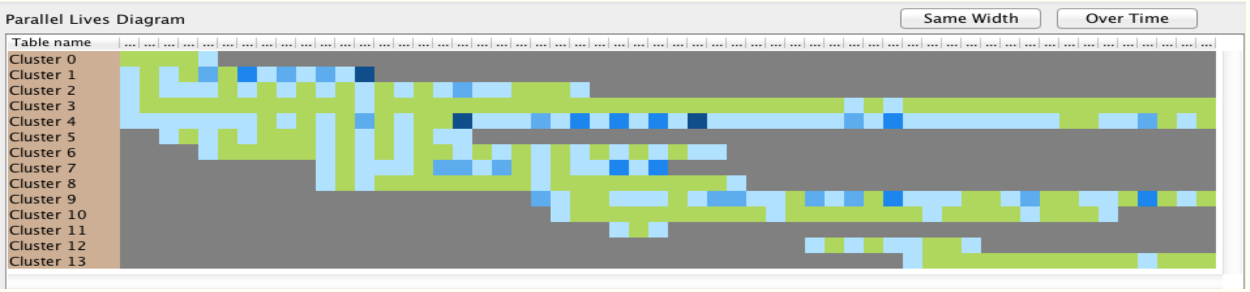
Structure

- Introduction
- **Problem Specification**
- Creating an overview of the history of a schema
- Plutarch’s Parallel Lives Results
- Conclusions and Open Issues

Intuition on the problem

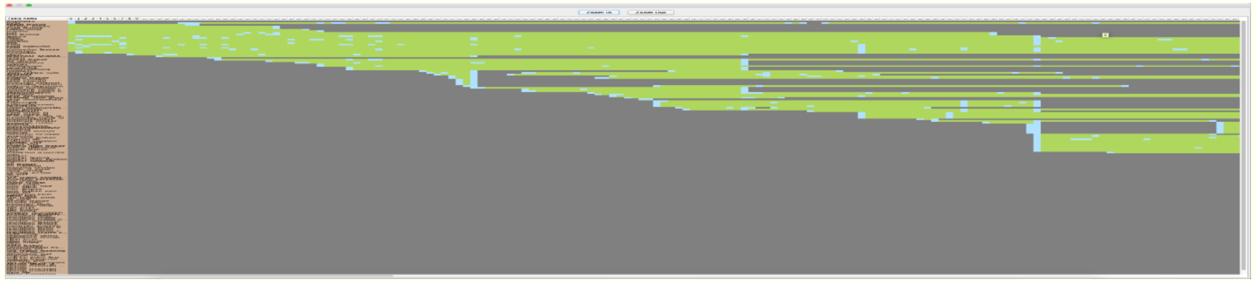
- The idea came from the mantra that Shneiderman underlines in his article at 1996 [Shne96], which is

***Overview first*, zoom and filter, details on demand.**



Construct an overview

instead of a non-fitting diagram



Segmentation of the history into phases

- The idea is that we want to zoom-out on the time/version axis.
- We need to group transitions to *phases*, i.e., partition the set of transitions to disjoint groups of consecutive transitions, such that each phases is “homogeneous” internally
- The formulation of the problem is as follows:

Given the evolution history of a database schema,

group transitions into phases

such that the transitions of each phase share similar

Clustering of tables into groups

- The idea is that we want to zoom-out on the vertical axis with the tables (in case the relations are too many).
- We partition the set of relations into disjoint subsets or else *clusters*. Each cluster has relations with similar lives i.e., lives with similar start, death and heartbeat of changes.
- The formulation of the problem is as follows:

Given the evolution history of a database schema,

group relations into groups of relations with similar lives

such that the relations of each group share similar

Zoom, filter and details on demand

- **Zoom into a specific point of the overview:** if we have a matrix in which the x-axis contains the phases and the y-axis contains the tables of the database or the clusters how we could zoom into a specific cell of this table?
- **Filter the overview:** often there is the desire to isolate a component of an overview including its elements to compare for example how similar are the elements from which it consists of.
- **Details on demand:** if the PLD contains in its x-axis the phases and the clusters in its y-axis what details we could get about a cell of PLD?

Structure

- Introduction
- Problem Specification
- **Creating an overview of the history of a schema**
- Plutarch’s Parallel Lives Results
- Conclusions and Open Issues

Creating an overview of the history of the database

Creation of an overview of the history of the database consists of two parts:

- Computing a segmentation of the history into **phases**
  - Phase extraction algorithm (on the basis of a parameterized distance function)
  - Assessment of the quality of result & parameter tuning
- Grouping tables into **table clusters**
  - Table clustering algorithm (on the basis of a parameterized distance function)
  - Assessment of the quality of result & parameter tuning

Computing a segmentation of the history into phases: The Algorithm

Algorithm: The Phasic Extractor algorithm

Input: A list of schema transitions H = { t_s, t_{s+1}, …, t_{e-1}, t_e}, the desired number of phases *k*, the weight to assign to time w_t, the desired weight to assign to changes w_c , the choice if we want the data to be preprocessed according to the time *preProcessingTime,* the choice if we want the data to be preprocessed according to the changes *preProcessingChanges*.

Output: A partition of H, P = {p_1...p_k}

variable *numPhases=e*, counter of the number of phases.

```
Begin
1. P={$p_1$,…$p_e$} s.t. $p_i$={$t_i$}  i  s…e
2. while(numPhases>k){
      a. for each pair of phases $ph_i$, $ph_{i+1}$,
            i. compute δ($ph_i$, $ph_{i+1}$)
      b. Merge the most similar phases, $p_a$ and $p_{a+1}$ into a new phase p’
      c. P = {p1,…, $p_{a-1}$, p, $p_{a+1}$, …, $p_m$}
      d. numPhases --
   }
3. Return P;
End
```

Computing a segmentation of the history into phases: Parameters

- **Desired number of segments (k)**: refers to the number of phases that we would like to be extracted.
- **Pre-Processing Changes (PPC)**: refers to the preprocessing of the data from the aspect of changes (ON if the data has been preprocessed, OFF otherwise).
- **Pre-Processing Time (PPT)**: refers to the preprocessing of the data from the aspect of time (ON if the data has been preprocessed, OFF otherwise).
- **Weight Change (WC)**: refers to the weight of changes (0.5 normal weight, 0 if changes is not taken into account).
- **Weight Time (WT)**: refers to the weight of time (0.5 normal weight, 0 if time is not taken into account).

Computing a segmentation of the history into phases: Distance Function

$$\delta(p_i, p_{i+1}) = w^T \times \delta^T(p_i, p_{i+1}) + w^C \times \delta^C(\mathrm{p}_{\mathrm{i}}, \mathrm{p}_{i+1})$$

Computing a segmentation of the history into phases: Assessment via divergence from the mean (1/3)

$$E_{pn} = \left(\sum_{\forall phase\ ph_i} \sum_{\forall event\ e_j \in ph_i} \left| \mu_i - e_j \right|^p \right)^{1/p}$$

- μ_i is the average number of changes of each phase
- e_j is the number of changes of each phase's event
- Typically p is equal to one or two

Computing a segmentation of the history into phases: Assessment via divergence from the mean (2/3)

Datasets that were used by Phasic Extractor

- Atlas
- bioSQL
- Coppermine
- Ensembl
- mediaWiki
- Opencart
- phpBB
- typo3

Computing a segmentation of the history into phases: Assessment via divergence from the mean (3/3)

	PPC: OFF PPT: OFF	PPC: ON PPT: OFF	PPC: OFF PPT: ON	PPC: ON PPT: ON
WC = 0.0 **WT = 1.0**	-	-	-	-
WC = 0.5 **WT = 0.5**	-	-	1	1
WC = 1.0 **WT = 0.0**	5	3	-	1

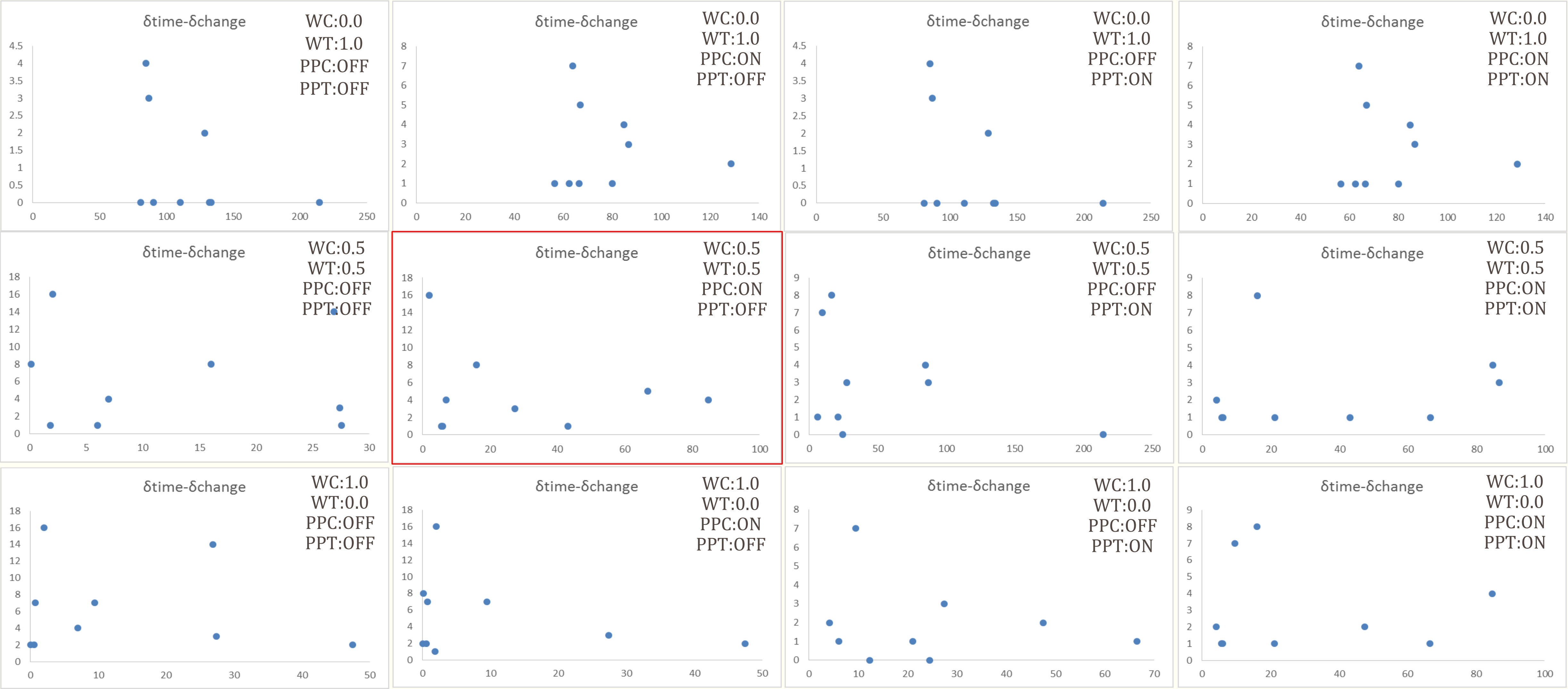
Computing a segmentation of the history into phases: Assessment via spread in the time x change space (1/2)

The second assessment method can be described as follows:

For each pair of phases ph_i and ph_{i+1} :

- compute the term δ^{time}
- compute is the term δ_{change}
- When these two terms have been computed for the whole set of pairs we can represent our results with the scatter plot format.

Computing a segmentation of the history into phases: Assessment via spread in the time x change space (2/2)



Grouping tables into clusters: The Algorithm

Algorithm: The Clustering Extractor algorithm

Input: The entire set of the database’s tables T {tab_1, ... , tab_n}, the desired number of clusters *k*, the weight to assign to birth date $w_{b,}$, the weight to assign to death date $w_{d,}$, the weight to assign to heartbeat of the changes date w_c

Output: A partition of T, C={c_1, … , c_k}

variable *numClusters=n*, counter of the number of clusters

```
Begin

1. C={$c_1$, … , $c_n$} s.t. $c_i$ = {$tab_i$}  i  1…n
2. while(numClusters>k){
       a. for each pair of clusters $c_i$, $c_{i+1}$,
              i. compute the $\delta(c_i,c_{i+1})$
       b. Merge the most similar clusters, $c_a$ and $c_{a+1}$ into a new cluster c’
       c. C = {$c_1$,…, $c_{a-1}$, c, $c_{a+1}$, …, $c_m$}
       d. numClusters –-
}

3. Return C;

End
```

Grouping tables into clusters: Parameters

- **Desired number of clusters (k)**: refers to the number of clusters that we would like to be created.
- **Birth Weight (BW)**: refers to the weight of the distance between birth dates of compared clusters.
- **Death Weight (DW)**: refers to the weight of the distance between death dates of compared clusters.
- **Change Weight (CW)**: refers to the weight of the distance between the changes of compared clusters.

Grouping tables into clusters: Distance Function

$$\delta(cluster_A, cluster_B) = w_b * |\delta_{birth}(c_A, c_B)| + w_d * |\delta_{death}(c_A, c_B)| + w_c * \left|\delta_{change}(c_A, c_B)\right|$$

Grouping tables into clusters: Clustering Validity Techniques

There are two main categories for clustering validity:

- **Internal evaluation**: refers to methods that do not need external knowledge and can measure the quality of the clusters only with the information that they keep and which was used from the clustering algorithm
- **External evaluation**: needs external knowledge, i.e., data have to be classified before the evaluation process, by explicit tracing of human knowledge on the issue.

Grouping tables into clusters: Assessment via External evaluation - Metrics

There is a large amount of methods that have been used previously.

We decided to choose the most common of them

- **Entropy**: is defined as the degree to which each cluster consists of objects of a single class.

For each cluster *j* we compute p_{ij}, which is the probability that a member of cluster *i* belongs to class *j*.

$$p_{ij} = \frac{m_{ij}}{m_i}$$

- m_i is the number of objects in cluster *i.*
- m_{ij} is the number of objects of class *j* in cluster *i*.

The **total entropy of each cluster *i*** is calculated by the following formula:

$$e_i = -\sum_{j=1}^{L} p_{ij} \log_2 p_{ij}$$

- *L* is the number of classes.

Grouping tables into clusters: Assessment via External evaluation – Metrics

The **total entropy of a set of clusters**, is defined as the sum of the entropies of each cluster weighted by the size of each cluster:

$$e = \sum_{i=1}^{K} \frac{m_i}{m} e_i$$

- *K* is the number of clusters and *m* is the total number of data points

- **Precision**: is defined as the fraction of a cluster that consists of objects of a specified class. **Precision of a cluster *i* with respect to class *j*** is:

$$precision(i, j) = p_{ij}$$

Grouping tables into clusters: Assessment via External evaluation - Metrics

- **Recall**: depicts the extent to which a cluster contains all the objects of a specified class. The **recall of cluster *i* with respect to class *j*** is:

$$recall(i,j) = \frac{m_{ij}}{m_j}$$

  - m_j is the number of objects in class *j*.

- **F-measure**: consists of both precision and recall and measures the extent to which a cluster contains only objects of a particular class and all objects of that class.

The **F-measure of cluster *i* with respect to class *j*** is calculated by this formula:

$$F(i,j) = \frac{2 \times precision(i,j) \times recall(i,j)}{precision(i,j) + recall(i,j)}$$

Grouping tables into clusters: Assessment via External evaluation – Classifying Method

- Datasets
  - Atlas
  - bioSQL
  - Coppermine
  - phpBB
- The source of our classification procedure was the PLD (Parallel Live Diagram).
- The most obvious criteria of the PLD are when a table is born (birth date) and when a table died and not as much the count of changes of each table.

Grouping tables into clusters: Assessment via External evaluation – Classifying Method

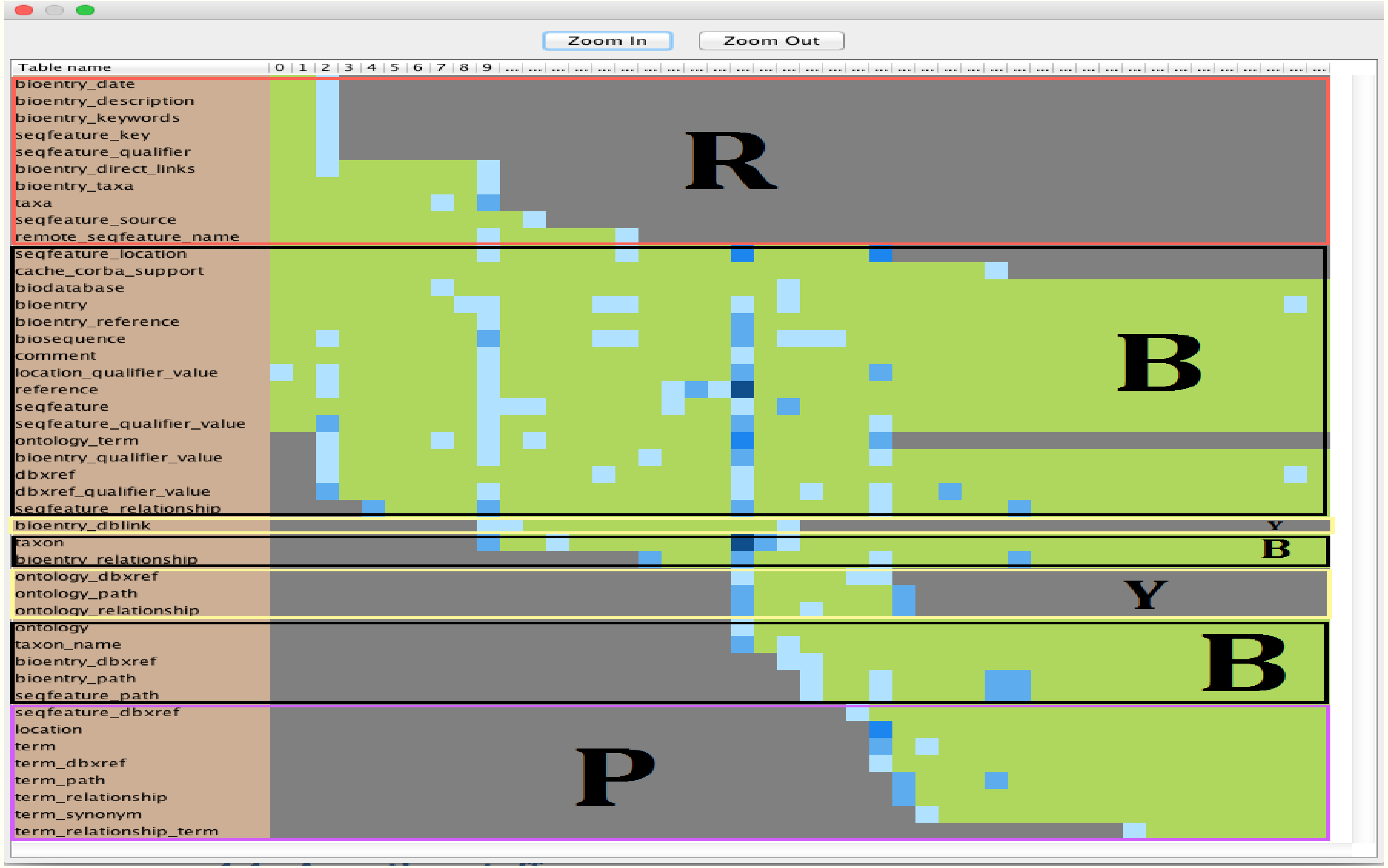


Table name

bioentry_date
bioentry_description
bioentry_keywords
seqfeature_key
seqfeature_qualifier
bioentry_direct_links
bioentry_taxa
taxa
seqfeature_source
remote_seqfeature_name
seqfeature_location
cache_corba_support
biodatabase
bioentry
bioentry_reference
biosequence
comment
location_qualifier_value
reference
seqfeature
seqfeature_qualifier_value
ontology_term
bioentry_qualifier_value
dbxref
dbxref_qualifier_value
seqfeature_relationship
bioentry_dblink
taxon
bioentry_relationship
ontology_dbxref
ontology_path
ontology_relationship
ontology
taxon_name
bioentry_dbxref
bioentry_path
seqfeature_path
seqfeature_dbxref
location
term
term_dbxref
term_path
term_relationship
term_synonym
term_relationship_term

Grouping tables into clusters: Assessment via External evaluation – Results

Average F-Measure

Parameters Set wb-wd-wc	Average F-Measure
0.33 - 0.33 - 0.33	0.19
0.00 - 1.00 - 0.00	0.22
0.00 - 0.50 - 0.50	0.20
0.00 - 0.00 - 1.00	0.17
0.50 - 0.50 - 0.00	**0.25**
0.50 - 0.00 - 0.50	0.16
1.00 - 0.00 - 0.00	0.21

Entropies

w_b	w_d	w_c	Entropy (e)
0.333	0.333	0.333	1.13
0	1	0	0.79
0	0.5	0.5	1.06
0	0	1	1.14
0.5	0.5	0	**0.00**
0.5	0	0.5	0.57
1	0	0	0.52

Grouping tables into clusters: Assessment via Internal evaluation - Metrics

- Internal evaluation contains these types of methods that do not need any external knowledge
- Internal evaluation helps us to decide the right set of parameters for the best quality of the clustering
- We can express the overall cluster validity for a set of *K* clusters as a weighted sum of the validity of individual clusters

$$overall\ validity = \sum_{i=1}^{K} w_i validity(C_i)$$

  - validity function can be expressed by various metrics such as cohesion, separation or even a combination of them

Grouping tables into clusters: Assessment via Internal evaluation - Metrics

- **Cohesion**: can be defined as the sum of the proximities with respect to the prototype (centroid or medoid) of the cluster

$$cohesion(C_i) = \sum\nolimits_{x \in C_i} proximity(x, c_i) = distance(x, c_i)$$

  - c_i is the prototype of cluster C_i

- **Separation** of a cluster is defined as the proximity between the centroid c_i of the cluster and an overall centroid that has been calculated by the whole set of data points

$$separation(C_i) = proximity(c_i, c)$$

  - c is defined the overall centroid of the dataset

- We used the **Euclidean distance** as a measure of proximity.

Grouping tables into clusters: Assessment via Internal evaluation - Results

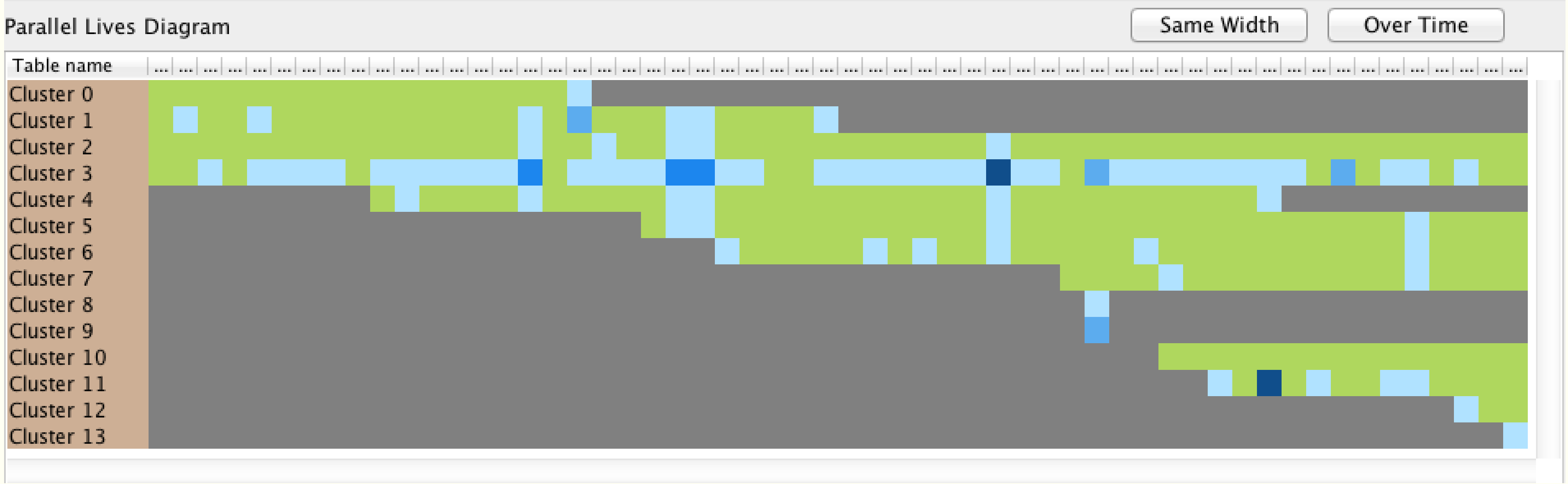
Wb	Wd	Wc	Cohesion
0.00	1.00	0.00	2
0.33	0.33	0.33	2
0.50	0.50	0.00	-
0.50	0.00	0.50	4
1.00	0.00	0.00	-

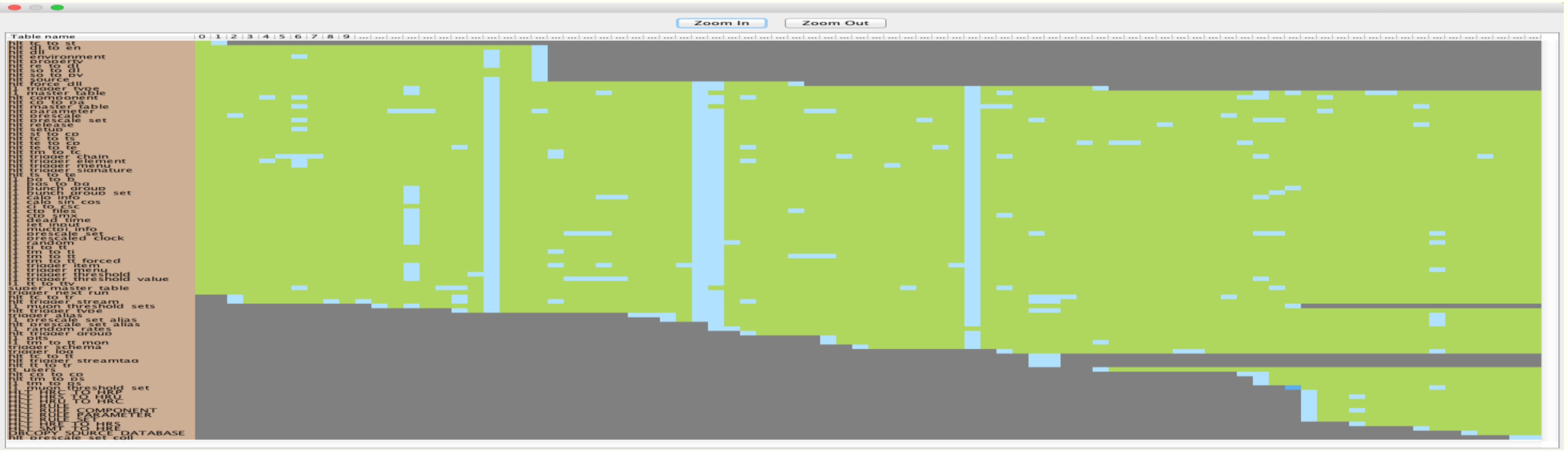
Wb	Wd	Wc	Separation
0.00	1.00	0.00	3
0.33	0.33	0.33	2
0.50	0.50	0.00	-
0.50	0.00	0.50	3
1.00	0.00	0.00	-

Structure

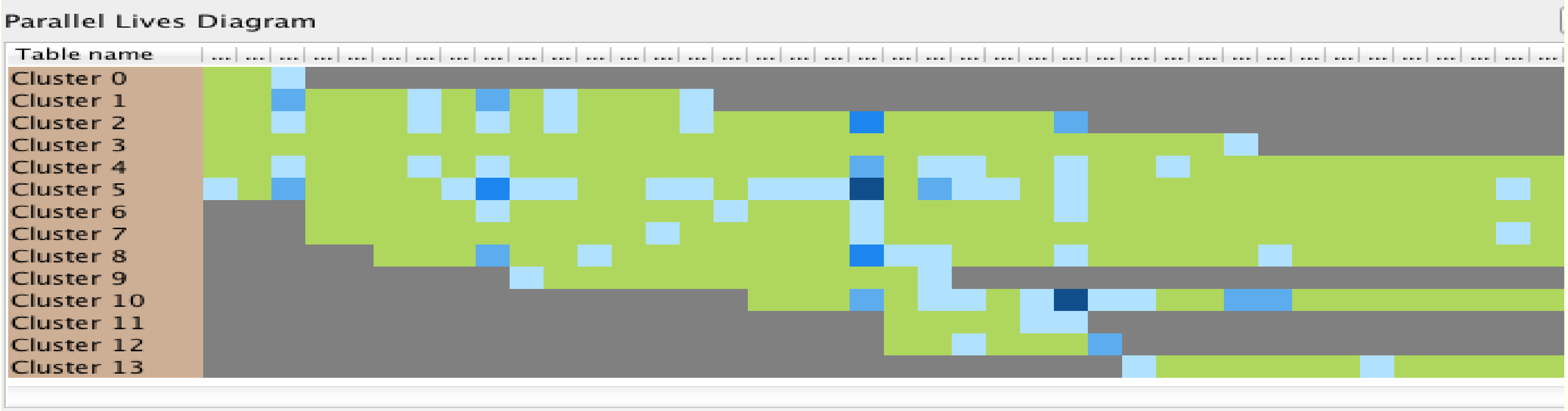
- Introduction
- Problem Specification
- Creating an overview of the history of a schema
- **<u>Plutarch’s Parallel Lives Results</u>**
- Conclusions and Open Issues

Overview vs PLD (Atlas)





Overview vs PLD (bioSQL)



Parallel Lives Diagram

Table name

Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10
Cluster 11
Cluster 12
Cluster 13

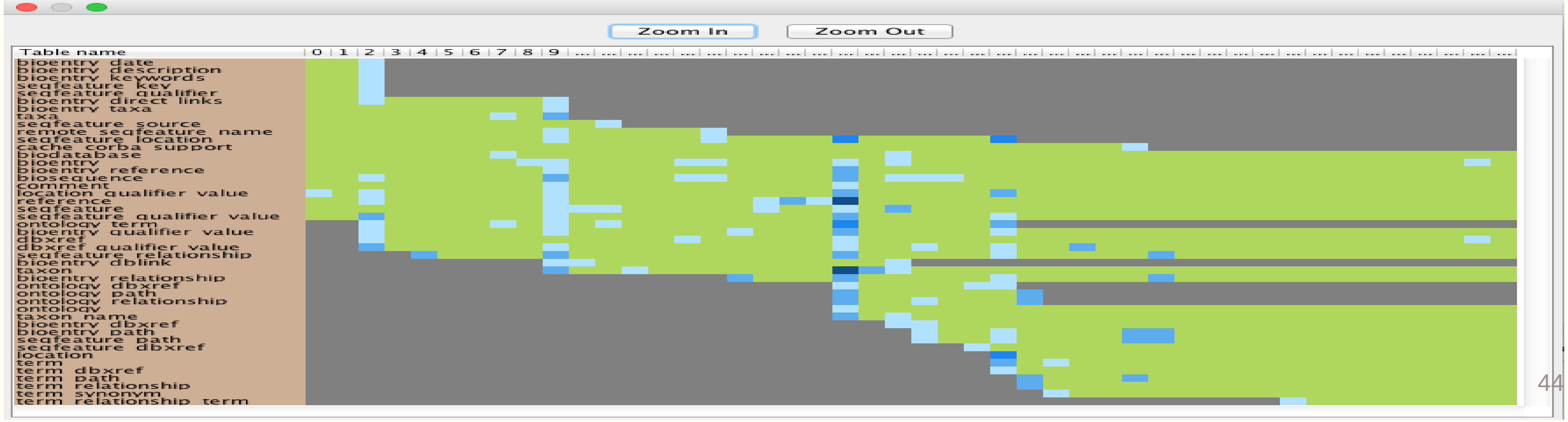
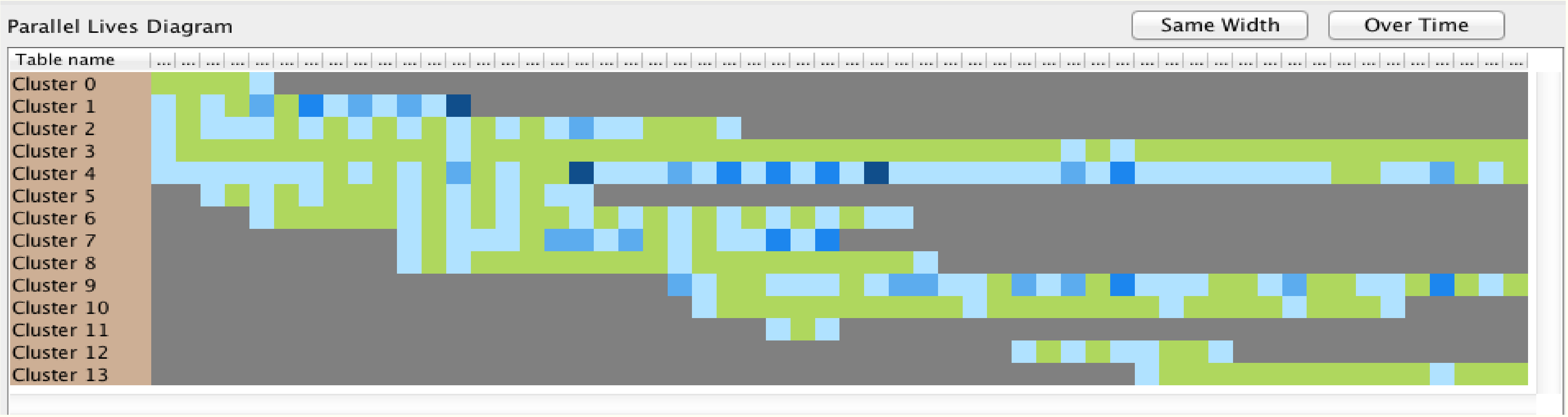
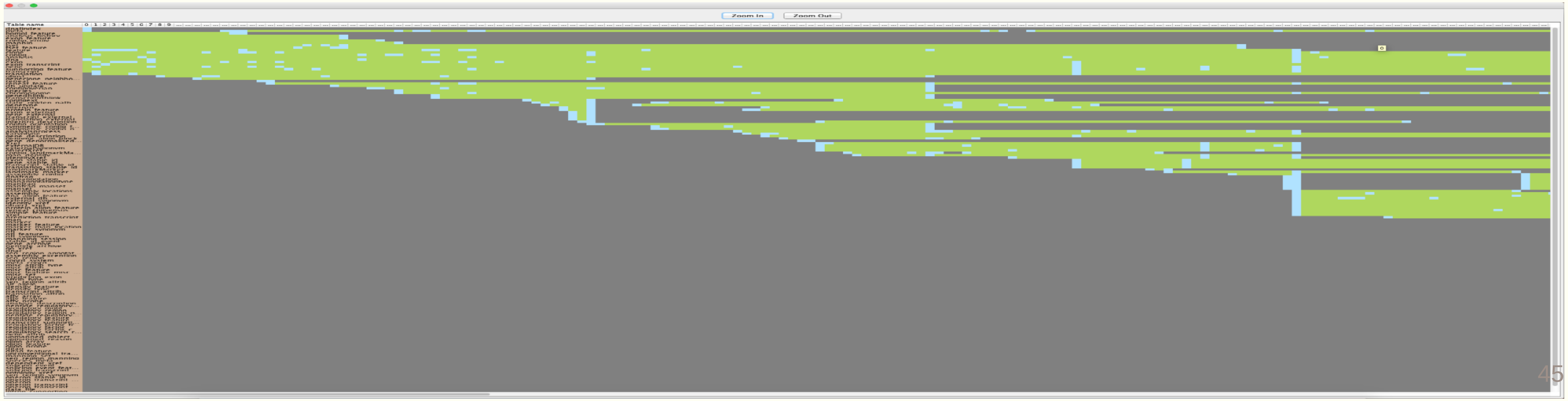


Table name | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

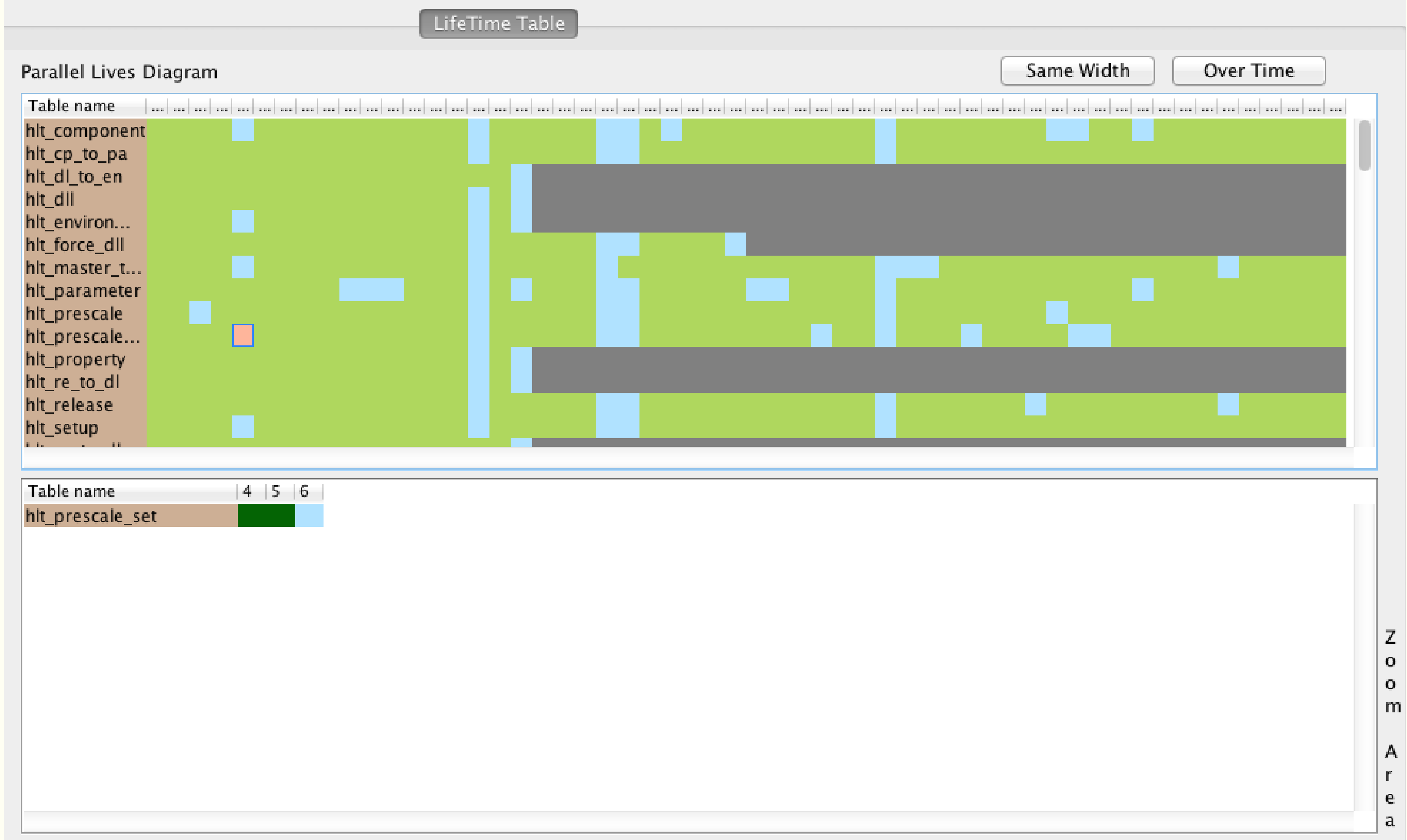
bioentry date
bioentry description
bioentry keywords
seqfeature key
seqfeature qualifier
bioentry direct links
bioentry taxa
taxa
seqfeature source
remote seqfeature name
seqfeature location
cache corba support
biodatabase
bioentry
bioentry reference
biosequence
comment
location qualifier value
reference
seqfeature
seqfeature qualifier value
ontology term
bioentry qualifier value
dbxref
dbxref qualifier value
seqfeature relationship
bioentry dblink
taxon
bioentry relationship
ontology dbxref
ontology path
ontology relationship
ontology
taxon name
bioentry dbxref
bioentry path
seqfeature path
seqfeature dbxref
location
term
term dbxref
term path
term relationship
term synonym
term relationship term

Overview vs PLD (Ensembl)





Zoom into a specific point of overview: PLD phases x tables

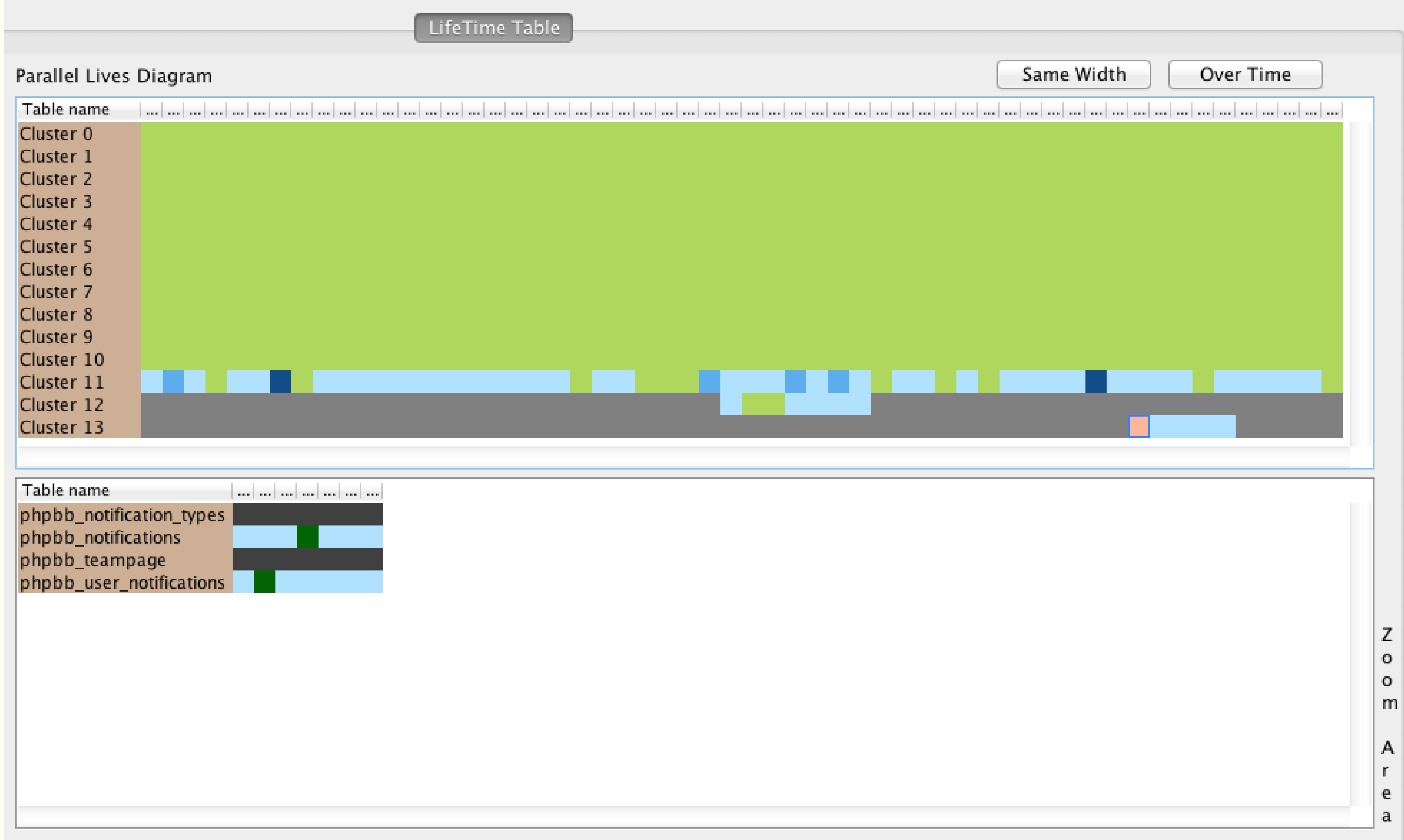


Parallel Lives Diagram

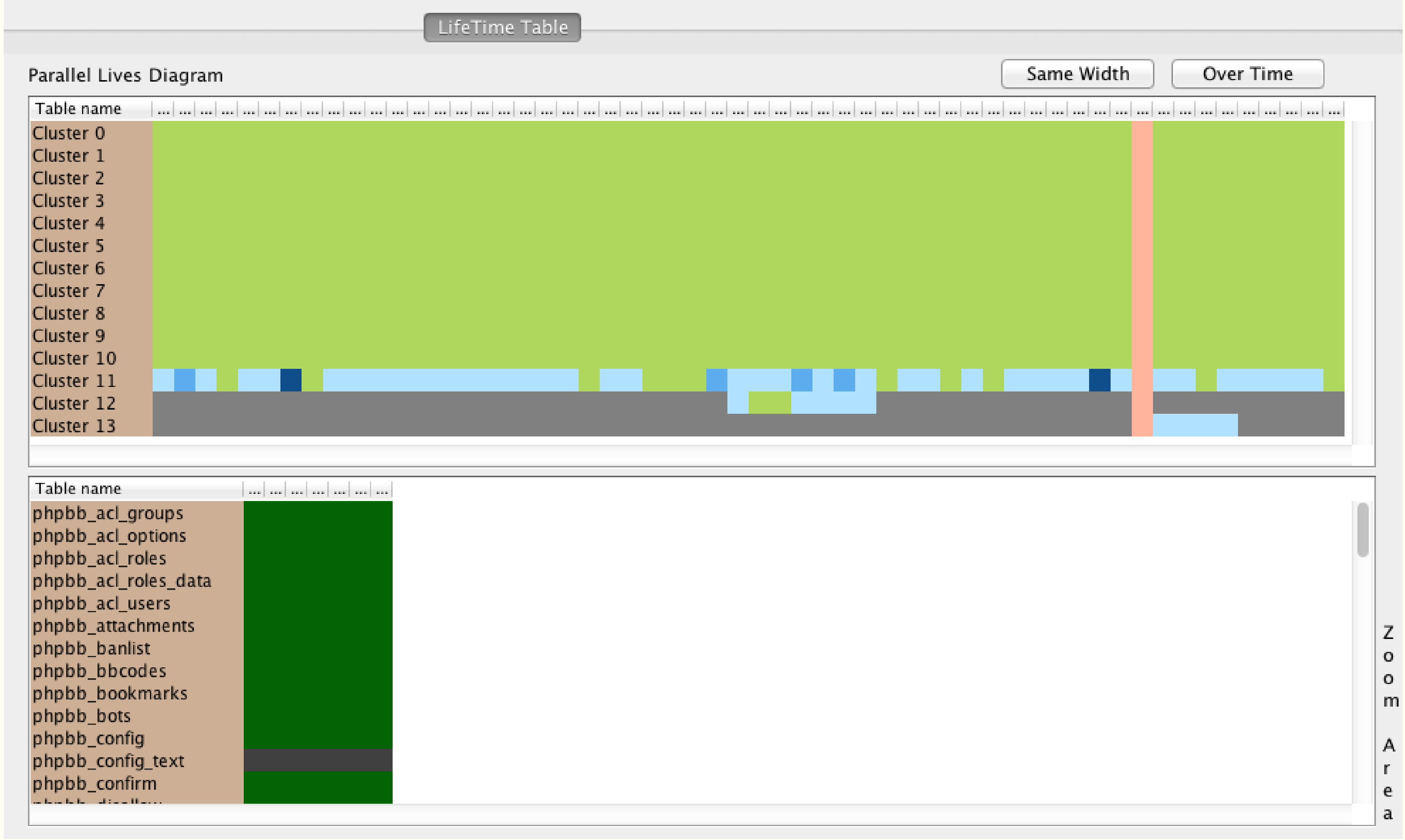
Table name
hlt_component
hlt_cp_to_pa
hlt_dl_to_en
hlt_dll
hlt_environ...
hlt_force_dll
hlt_master_t...
hlt_parameter
hlt_prescale
hlt_prescale...
hlt_property
hlt_re_to_dl
hlt_release
hlt_setup

Table name	4	5	6
hlt_prescale_set	🟩	🟩	🟦

Zoom into a specific point of overview: PLD phases x clusters



Filter the overview of the history: Filter by a specific phase



Filter the overview of the history: Filter by a specific table

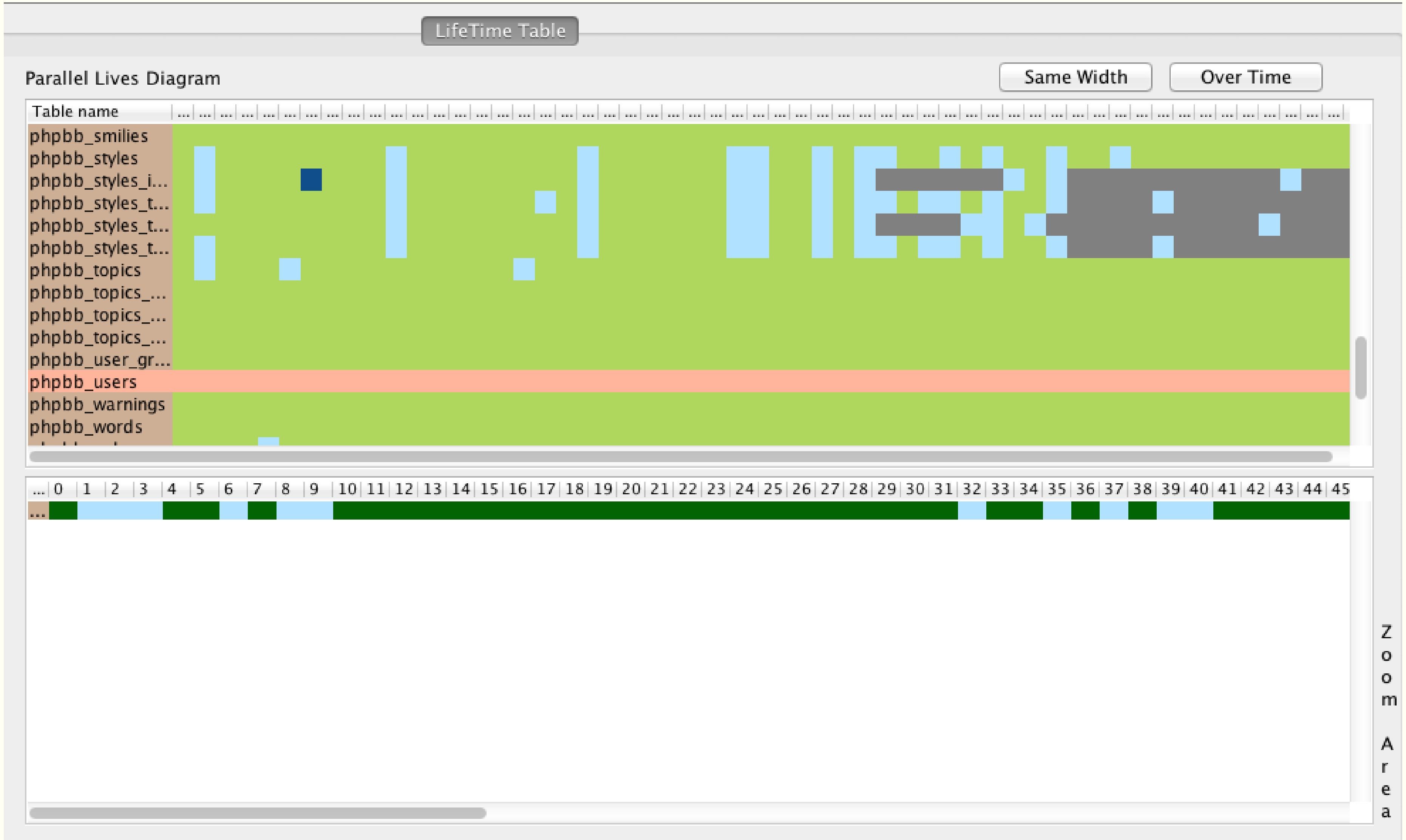


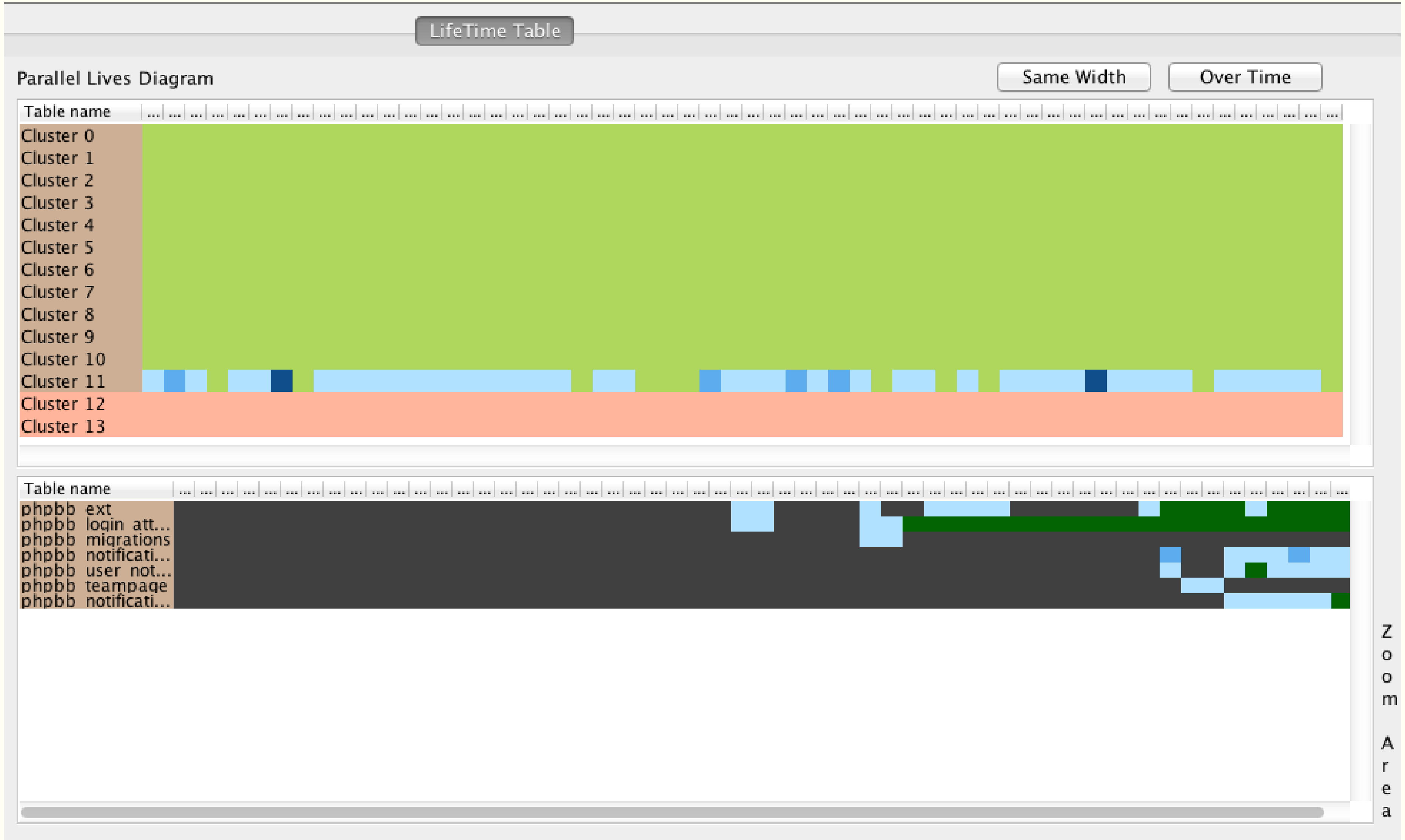
Table name

phpbb_smilies
phpbb_styles
phpbb_styles_i...
phpbb_styles_t...
phpbb_styles_t...
phpbb_styles_t...
phpbb_topics
phpbb_topics_...
phpbb_topics_...
phpbb_topics_...
phpbb_user_gr...
phpbb_users
phpbb_warnings
phpbb_words

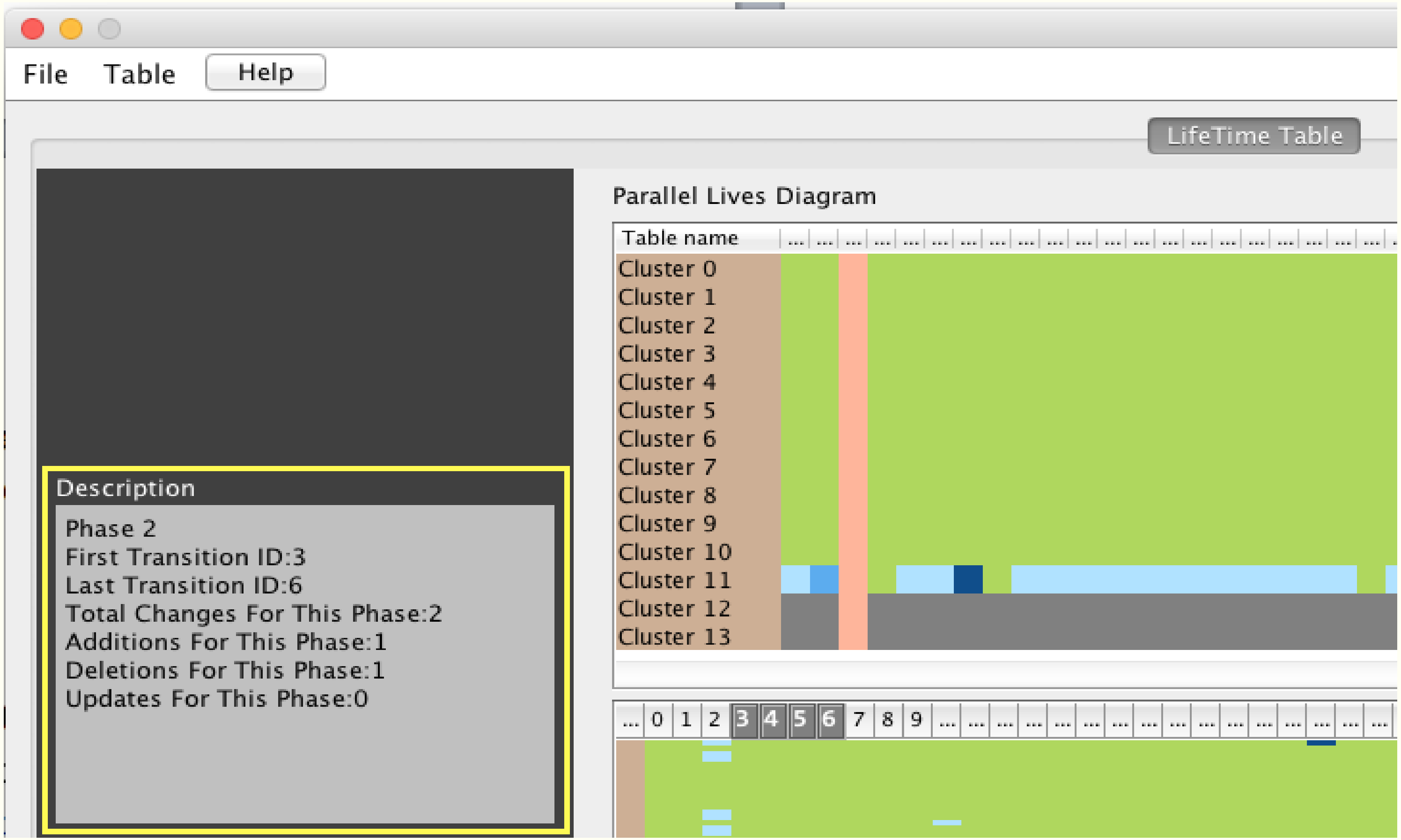
... | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45

...

Filter the overview of the history: Filter by specific clusters



Details on demand for selected phase



Description

Phase 2
First Transition ID:3
Last Transition ID:6
Total Changes For This Phase:2
Additions For This Phase:1
Deletions For This Phase:1
Updates For This Phase:0

Details on demand with a full detailed PLD

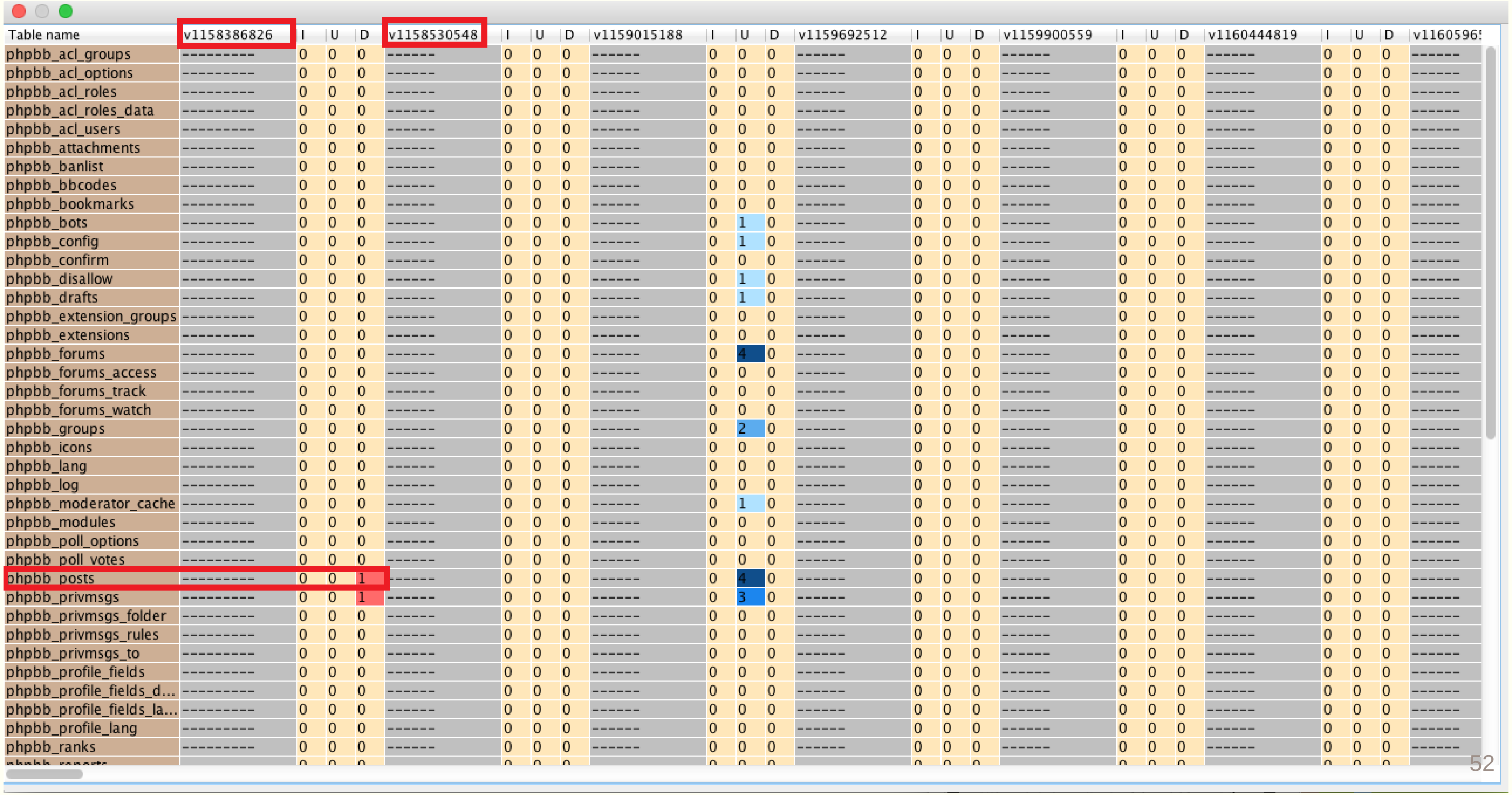


Table name	v1158386826	I	U	D	v1158530548	I	U	D	v1159015188	I	U	D	v1159692512	I	U	D	v1159900559	I	U	D	v1160444819	I	U	D	v1160596
phpbb_acl_groups	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_acl_options	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_acl_roles	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_acl_roles_data	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_acl_users	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_attachments	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_banlist	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_bbcodes	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_bookmarks	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_bots	---------	0	0	0	------	0	0	0	------	0	1	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_config	---------	0	0	0	------	0	0	0	------	0	1	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_confirm	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_disallow	---------	0	0	0	------	0	0	0	------	0	1	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_drafts	---------	0	0	0	------	0	0	0	------	0	1	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_extension_groups	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_extensions	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_forums	---------	0	0	0	------	0	0	0	------	0	4	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_forums_access	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_forums_track	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_forums_watch	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_groups	---------	0	0	0	------	0	0	0	------	0	2	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_icons	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_lang	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_log	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_moderator_cache	---------	0	0	0	------	0	0	0	------	0	1	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_modules	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_poll_options	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_poll_votes	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_posts	---------	0	0	1	------	0	0	0	------	0	4	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_privmsgs	---------	0	0	1	------	0	0	0	------	0	3	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_privmsgs_folder	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_privmsgs_rules	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_privmsgs_to	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_profile_fields	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_profile_fields_d...	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_profile_fields_la...	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_profile_lang	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------
phpbb_ranks	---------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------	0	0	0	------

Structure

- Introduction
- Problem Specification
- Creating an overview of the history of a schema
- Results
- **Conclusions and Open Issues**

Conclusions

Conclusions:

- Creation of an interactive overview from the entire life of a database
- Phase Extraction Algorithm
- Cluster Extraction Algorithm
- Assessment of both of them
- Plutarch’s Parallel Lives tool

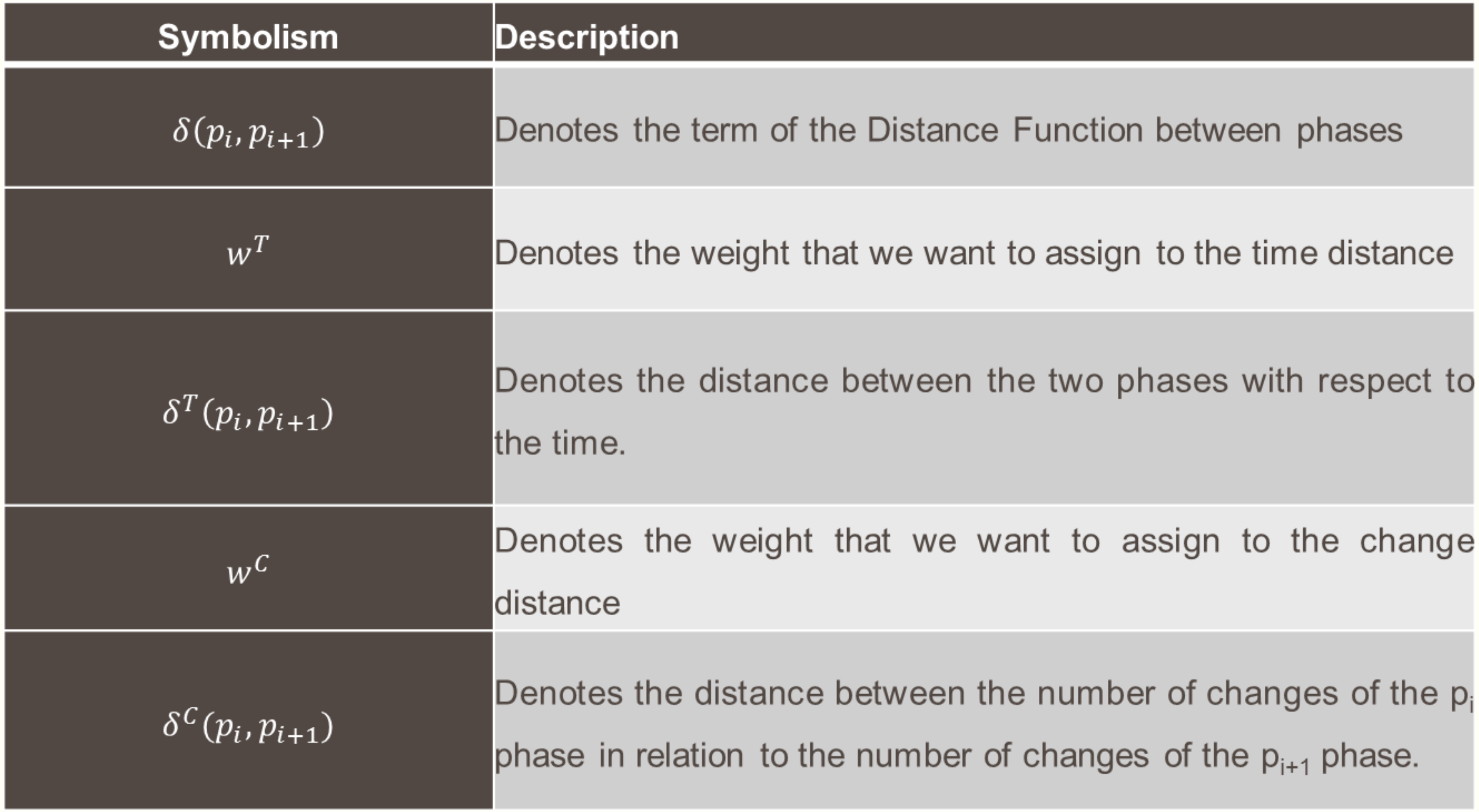
Open Issues

Open Issues

- Implementation of different distance metrics for phase extraction
- Implementation of different distance metrics for cluster extraction
- Enrichment of PPL with more features

Thank you!

Phase Extraction Distance Function Notations



Symbolism	Description
$\delta(p_i, p_{i+1})$	Denotes the term of the Distance Function between phases
w^T	Denotes the weight that we want to assign to the time distance
$\delta^T(p_i, p_{i+1})$	Denotes the distance between the two phases with respect to the time.
w^C	Denotes the weight that we want to assign to the change distance
$\delta^C(p_i, p_{i+1})$	Denotes the distance between the number of changes of the p_i phase in relation to the number of changes of the p_{i+1} phase.

Computing a segmentation of the history into phases: Assessment via divergence from the mean (3/4)

Atlas Dataset

	PPC:OFF	PPC:ON	PPC:OFF	PPC:ON
	PPT:OFF	PPT:OFF	PPT:ON	PPT:ON

WC=0.0
WT=1.0
	898.38	907.51	898.38	907.51

WC=0.5
WT=0.5
	877.94	891.98	840.24	855.17

WC=1.0
WT=0.0
	912.11	912.11	859.56	859.56

bioSQL Dataset

	PPC:OFF	PPC:ON	PPC:OFF	PPC:ON
	PPT:OFF	PPT:OFF	PPT:ON	PPT:ON

WC=0.0
WT=1.0
	380.15	381.22	380.15	381.22

WC=0.5
WT=0.5
	253.84	254.62	375.37	347.37

WC=1.0
WT=0.0
	206.54	206.54	325.82	325.82

Coppermine Dataset

	PPC:OFF	PPC:ON	PPC:OFF	PPC:ON
	PPT:OFF	PPT:OFF	PPT:ON	PPT:ON

WC=0.0
WT=1.0
	136.45	130.74	136.45	130.74

WC=0.5
WT=0.5
	112.54	121.16	130.86	135.71

WC=1.0
WT=0.0
	108.29	135.39	138.20	134.35

Ensembl Dataset

	PPC:OFF	PPC:ON	PPC:OFF	PPC:ON
	PPT:OFF	PPT:OFF	PPT:ON	PPT:ON

WC=0.0
WT=1.0
	4111.28	4115.63	4111.28	4115.63

WC=0.5
WT=0.5
	4081.30	4097.89	4155.04	4083.44

WC=1.0
WT=0.0
	3737.57	4044.81	4124.37	3935.95

Computing a segmentation of the history into phases: Assessment via divergence from the mean (4/4)

mediaWiki Dataset

	PPC:OFF	PPC:ON	PPC:OFF	PPC:ON
	PPT:OFF	PPT:OFF	PPT:ON	PPT:ON

WC=0.0
WT=1.0
	1052.28	1052.28	1052.28	1052.28

WC=0.5
WT=0.5
	1025.91	1042.27	1030.86	1053.47

WC=1.0
WT=0.0
	920.34	920.34	1061.43	1047.30

Opencart Dataset

	PPC:OFF	PPC:ON	PPC:OFF	PPC:ON
	PPT:OFF	PPT:OFF	PPT:ON	PPT:ON

WC=0.0
WT=1.0
	3390.19	3381.58	3390.19	3381.58

WC=0.5
WT=0.5
	1297.10	1294.76	2733.91	2731.19

WC=1.0
WT=0.0
	837.30	837.30	2745.29	2743.91

phpBB Dataset

	PPC:OFF	PPC:ON	PPC:OFF	PPC:ON
	PPT:OFF	PPT:OFF	PPT:ON	PPT:ON

WC=0.0
WT=1.0
	870.53	880.23	870.53	880.23

WC=0.5
WT=0.5
	861.10	941.45	853.23	791.49

WC=1.0
WT=0.0
	843.11	843.11	953.27	872.68

typo3 Dataset

	PPC:OFF	PPC:ON	PPC:OFF	PPC:ON
	PPT:OFF	PPT:OFF	PPT:ON	PPT:ON

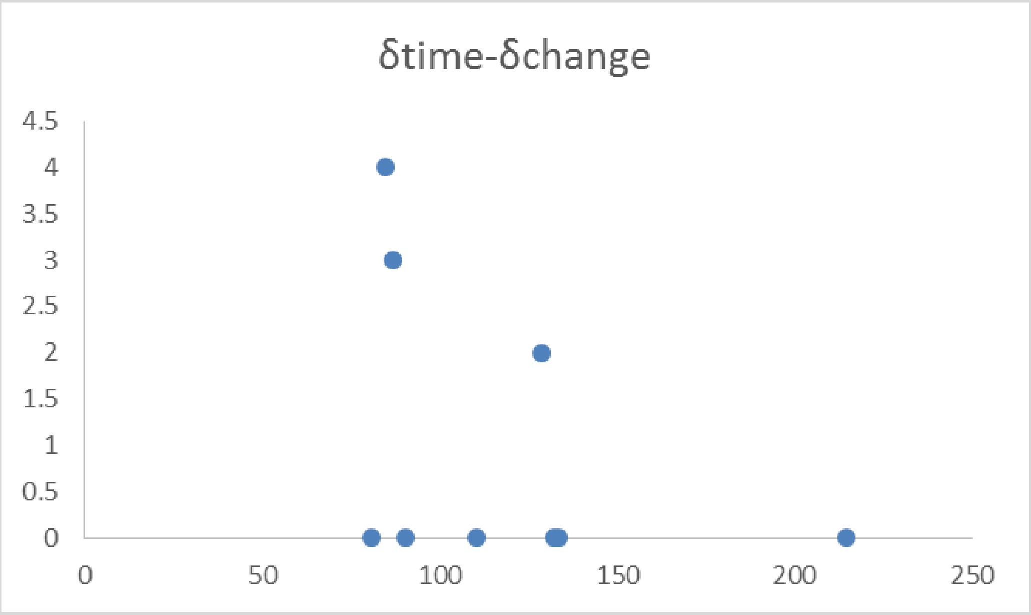
WC=0.0
WT=1.0
	648.59	644.33	648.59	644.33

WC=0.5
WT=0.5
	658.19	664.04	664.39	485.49

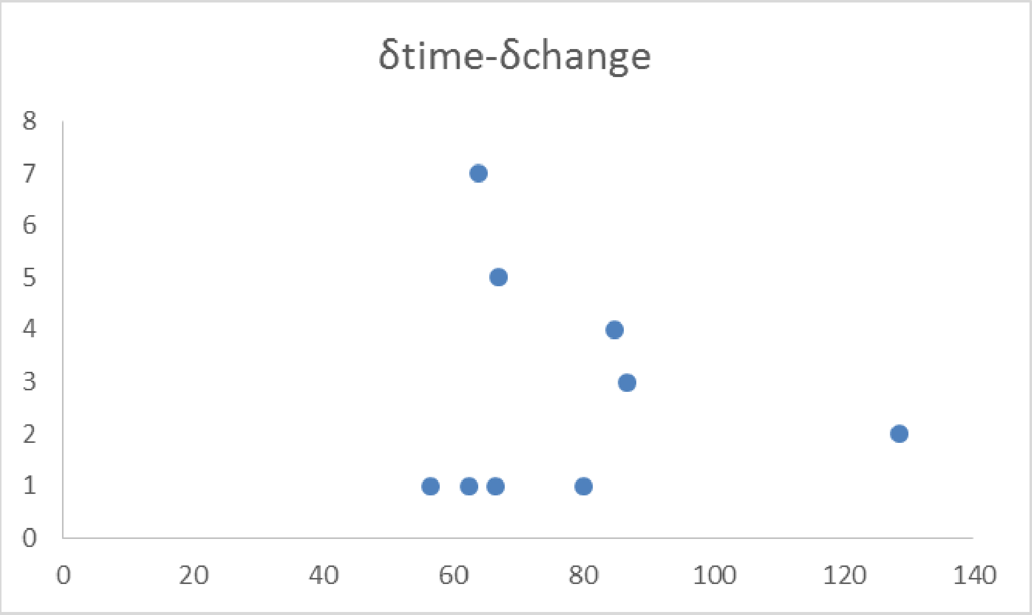
WC=1.0
WT=0.0
	486.84	486.84	477.48	438.35

Computing a segmentation of the history into phases: Assessment via spread in the time x change space (2/7)

WC:0.0, WT:1.0, PPC:OFF, PPT:OFF

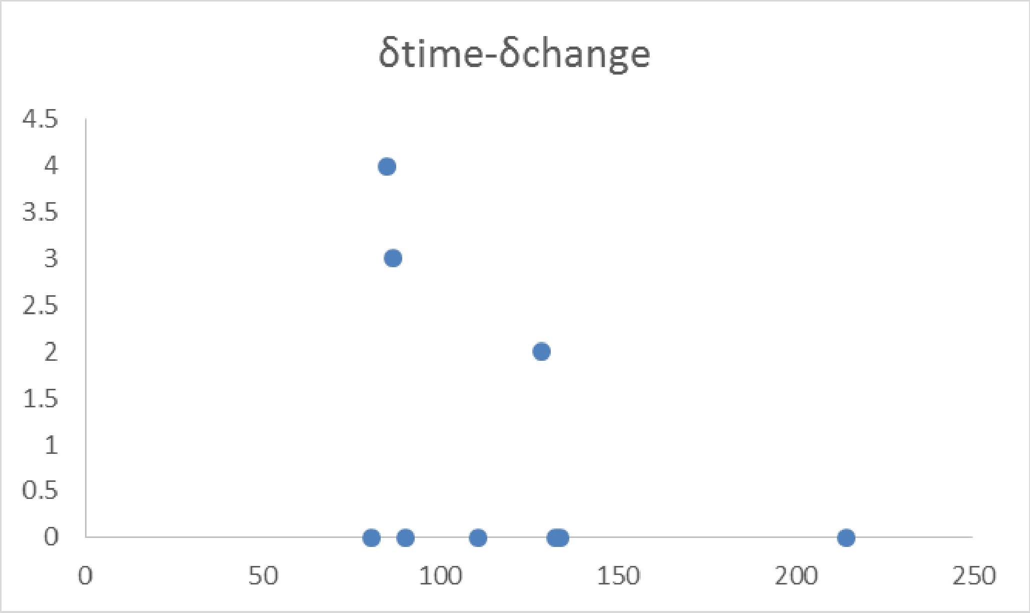


WC:0.0, WT:1.0, PPC:ON, PPT:OFF

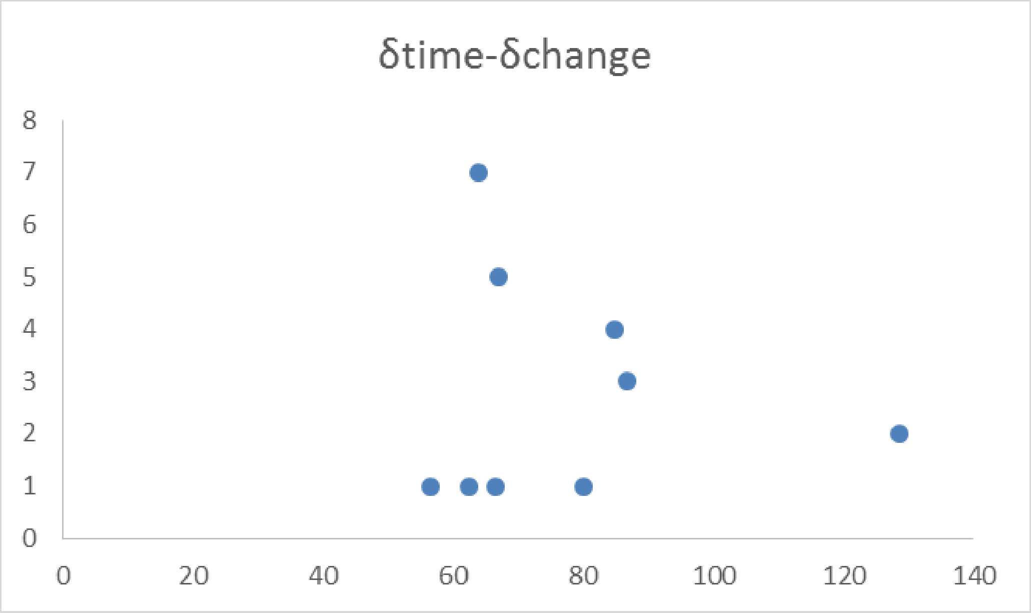


Computing a segmentation of the history into phases: Assessment via spread in the time x change space (3/7)

WC:0.0, WT:1.0, PPC:OFF, PPT:ON

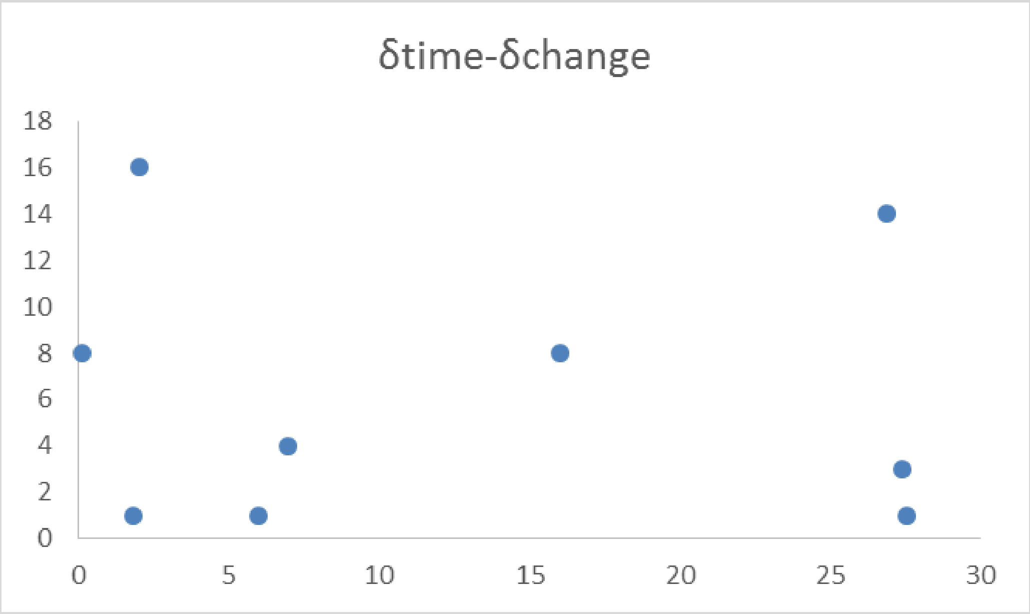


WC:0.0, WT:1.0, PPC:ON, PPT:ON

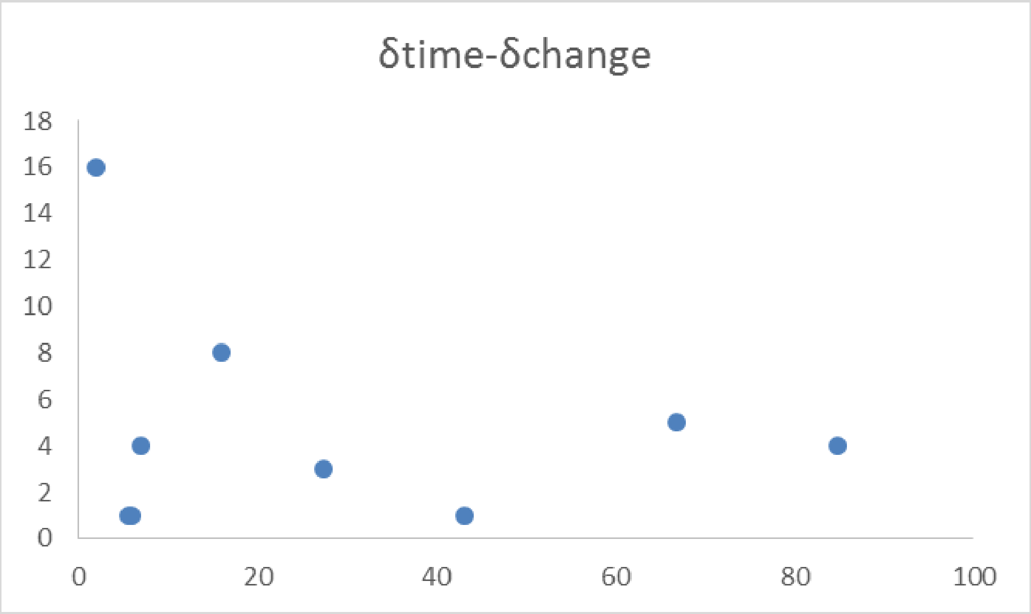


Computing a segmentation of the history into phases: Assessment via spread in the time x change space (4/7)

WC:0.5, WT:0.5, PPC:OFF, PPT:OFF



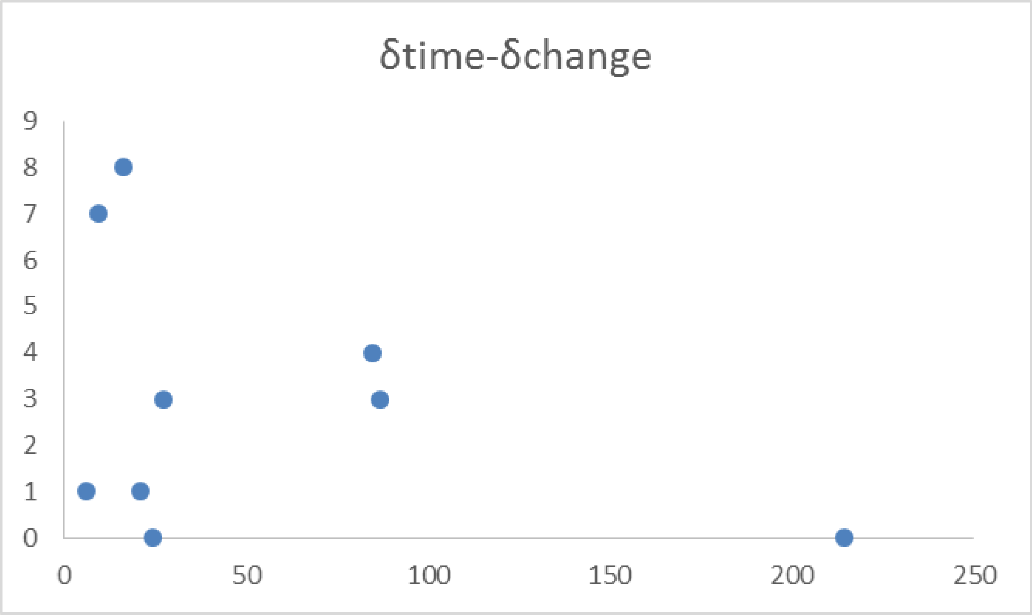
WC:0.5, WT:0.5, PPC:ON, PPT:OFF

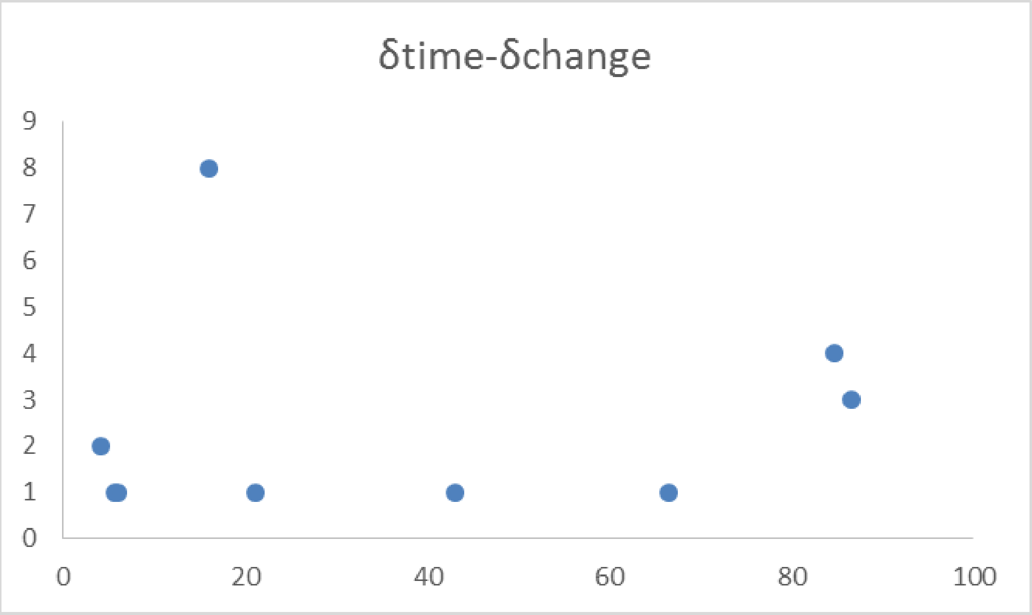


Computing a segmentation of the history into phases: Assessment via spread in the time x change space (5/7)

WC:0.5, WT:0.5, PPC:OFF, PPT:ON

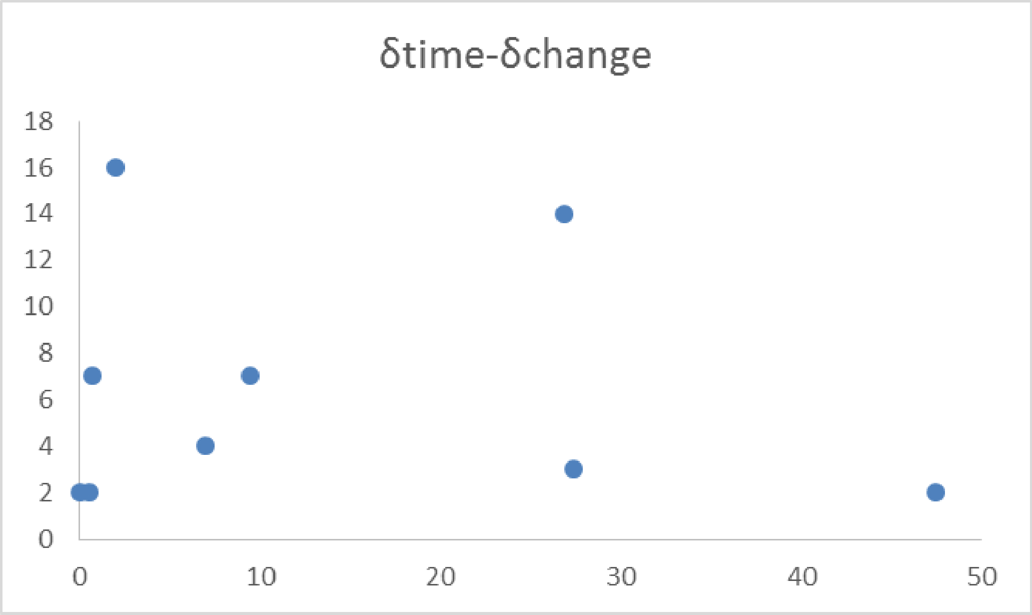
WC:0.5, WT:0.5, PPC:ON, PPT:ON



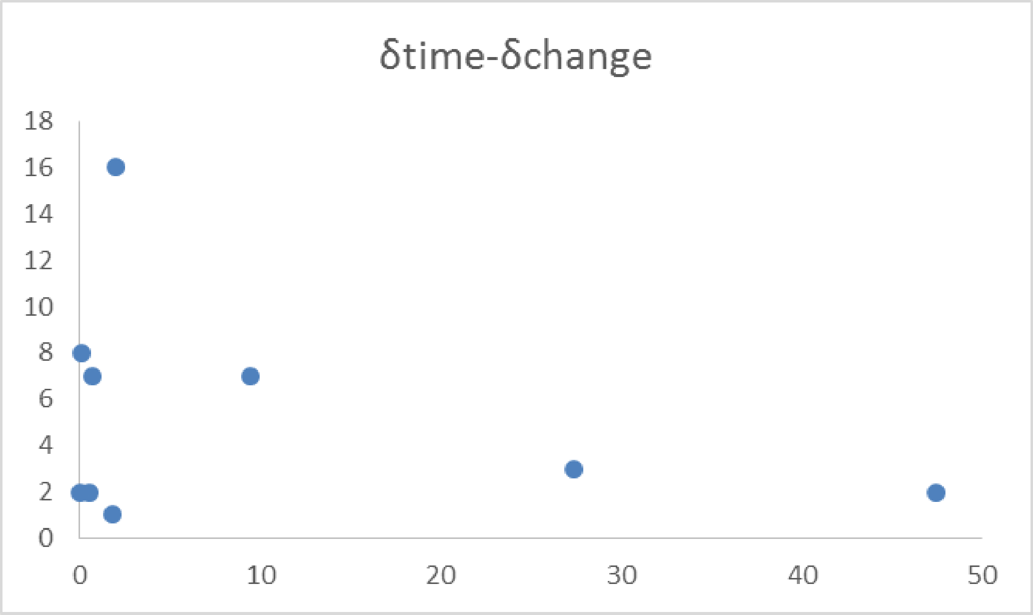


Computing a segmentation of the history into phases: Assessment via spread in the time x change space (6/7)

WC:1.0, WT:0.0, PPC:OFF, PPT:OFF



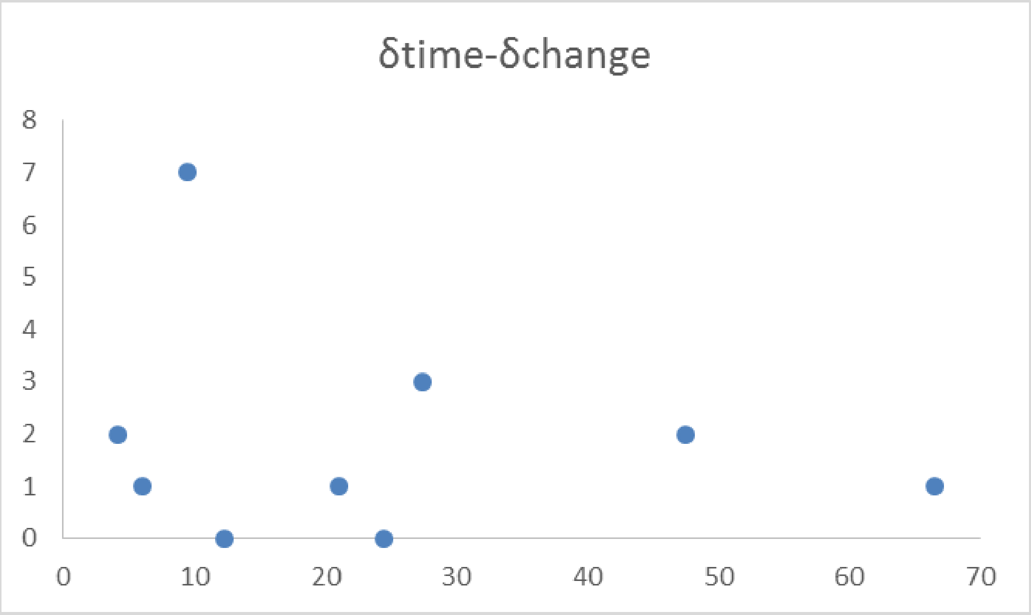
WC:1.0, WT:0.0, PPC:ON, PPT:OFF

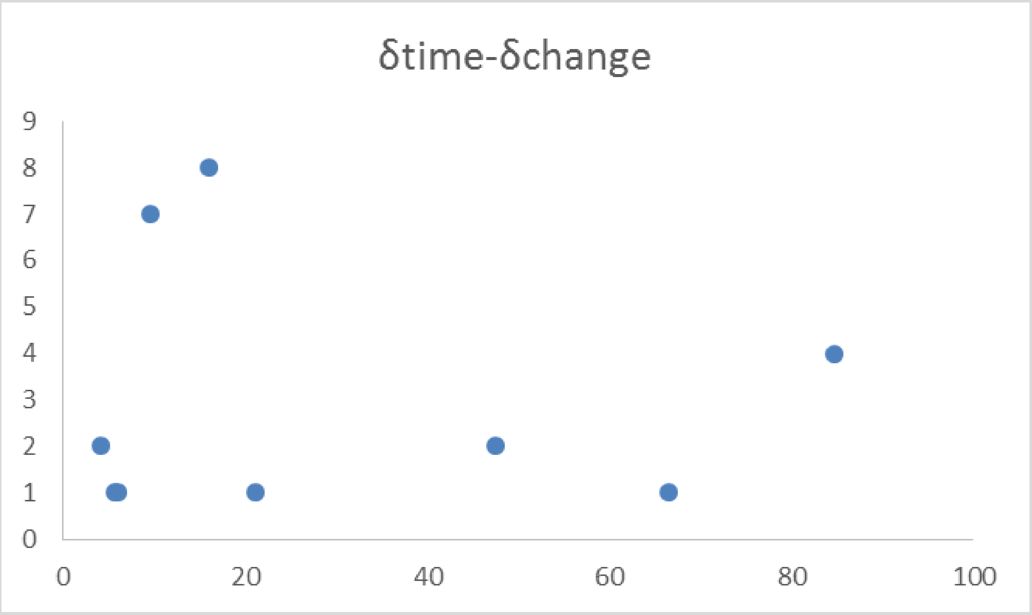


Computing a segmentation of the history into phases: Assessment via spread in the time x change space (7/7)

WC:1.0, WT:0.0, PPC:OFF, PPT:ON

WC:1.0, WT:0.0, PPC:ON, PPT:ON





Term	Description	Formula	
$\delta(cluster_A, cluster_B)$	Total distance between two clusters		
w_b	The weight that will be assigned to the distance that is related with the birth date		
$\delta_{birth}(c_A, c_B)$	The distance between birth dates of the two compared clusters	Plain $c_A.birth - c_B.birth$	Normalized $\frac{\delta_{birth}(c_A, c_B)}{DB\ duration}$
w_d	The weight that will be assigned to the distance that is related with the death date		
$\delta_{death}(c_A, c_B)$	The distance between death dates of the two compared clusters	Plain $\begin{cases} \emptyset, \boldsymbol{if\ both\ alive} \\ c_A.death - c_B.death, \boldsymbol{else} \end{cases}$	Normalized $\frac{\delta_{death}(c_A, c_B)}{DB\ duration}$
w_c	The weight that will be assigned to the distance that is related with the total changes		
$\delta_{change}(c_A, c_B)$	The distance between the total changes that have been committed to the two compared clusters	Plain $c_A.changes - c_Bchanges$	Normalized $\frac{\lvert Ch(A)\rvert - \lvert Ch(B)\rvert}{\lvert Ch(A)\rvert + \lvert Ch(B)\rvert}$ where Ch is the total number of changes

Grouping tables into clusters: Assessment via External evaluation – Results

w_b	w_d	w_c	Entropy (e)
0.333	0.333	0.333	0.40
0	1	0	0.45
0	0.5	0.5	0.51
0	0	1	1.14
0.5	0.5	0	0.32
0.5	0	0.5	0.50
1	0	0	**0.30**

Atlas

w_b	w_d	w_c	Entropy (e)
0.333	0.333	0.333	1.13
0	1	0	0.79
0	0.5	0.5	1.06
0	0	1	1.14
0.5	0.5	0	**0.00**
0.5	0	0.5	0.57
1	0	0	0.52

bioSQL

Grouping tables into clusters: Assessment via External evaluation – Results

w_b	w_d	w_c	Entropy (e)
0.333	0.333	0.333	0.38
0	1	0	0.19
0	0.5	0.5	0.38
0	0	1	0.38
0.5	0.5	0	0.19
0.5	0	0.5	0.60
1	0	0	**0.00**

Coppermine

w_b	w_d	w_c	Entropy (e)
0.333	0.333	0.333	0.13
0	1	0	0.94
0	0.5	0.5	0.28
0	0	1	0.28
0.5	0.5	0	**0.00**
0.5	0	0.5	0.20
1	0	0	0.26

phpBB

Grouping tables into clusters: Assessment via Internal evaluation - Results

Atlas

Wb	Wd	Wc	Cohesion	Separation
0.00	**1.00**	**0.00**	1323.69	2115.12
0.33	**0.33**	**0.33**	650.45	2598.62
0.50	**0.50**	**0.00**	331.31	2797.45
0.50	**0.00**	**0.50**	**1383.74**	**2049.81**
1.00	**0.00**	**0.00**	1271.30	2314.82

Coppermine

Wb	Wd	Wc	Cohesion	Separation
0.00	**1.00**	**0.00**	**211.33**	**362.10**
0.33	**0.33**	**0.33**	35.62	421.41
0.50	**0.50**	**0.00**	40.16	418.65
0.50	**0.00**	**0.50**	35.62	421.41
1.00	**0.00**	**0.00**	40.16	418.65

Grouping tables into clusters: Assessment via Internal evaluation - Results

bioSQL

Wb	Wd	Wc	Cohesion	Separation
0.00	1.00	0.00	320.45	781.45
0.33	0.33	0.33	159.73	890.94
0.50	0.50	0.00	122.00	886.21
0.50	0.00	0.50	**473.46**	**668.98**
1.00	0.00	0.00	253.59	827.05

Ensembl

Wb	Wd	Wc	Cohesion	Separation
0.00	1.00	0.00	11167.50	30780.37
0.33	0.33	0.33	**21301.49**	**21566.66**
0.50	0.50	0.00	5289.72	33661.45
0.50	0.00	0.50	19684.44	24562.84
1.00	0.00	0.00	14182.14	27347.17

Grouping tables into clusters: Assessment via Internal evaluation - Results

mwiki

Wb	Wd	Wc	Cohesion	Separation
0.00	1.00	0.00	4653.55	**6882.07**
0.33	0.33	0.33	1752.76	9397.74
0.50	0.50	0.00	1033.57	9561.50
0.50	0.00	0.50	**5349.92**	7390.00
1.00	0.00	0.00	4775.46	7740.74

Opencart

Wb	Wd	Wc	Cohesion	Separation
0.00	1.00	0.00	3924.06	15890.54
0.33	0.33	0.33	3359.07	16089.72
0.50	0.50	0.00	2366.92	16189.32
0.50	0.00	0.50	**7604.85**	**13317.76**
1.00	0.00	0.00	3202.38	16068.17

Grouping tables into clusters: Assessment via Internal evaluation - Results

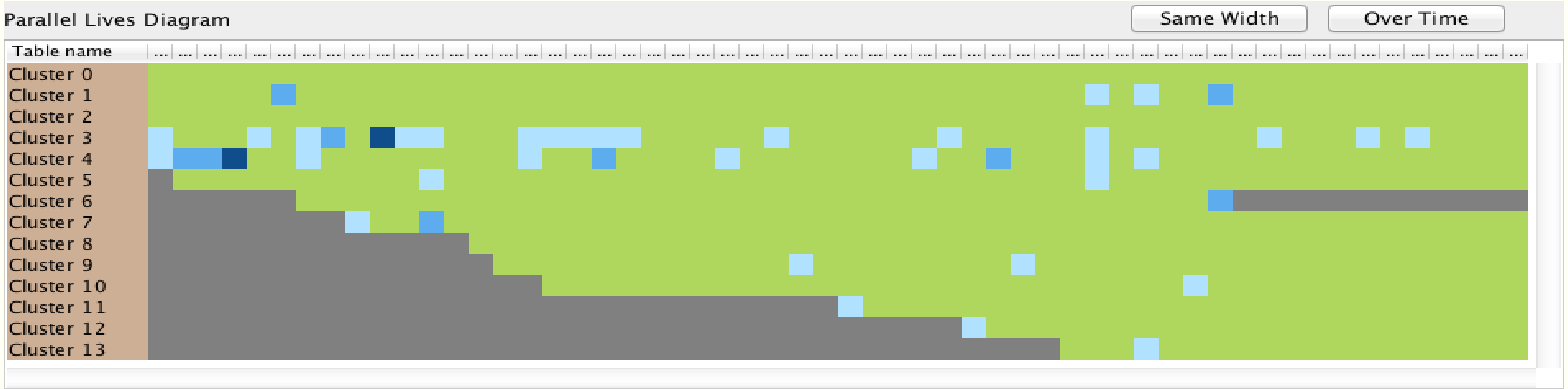
phpBB

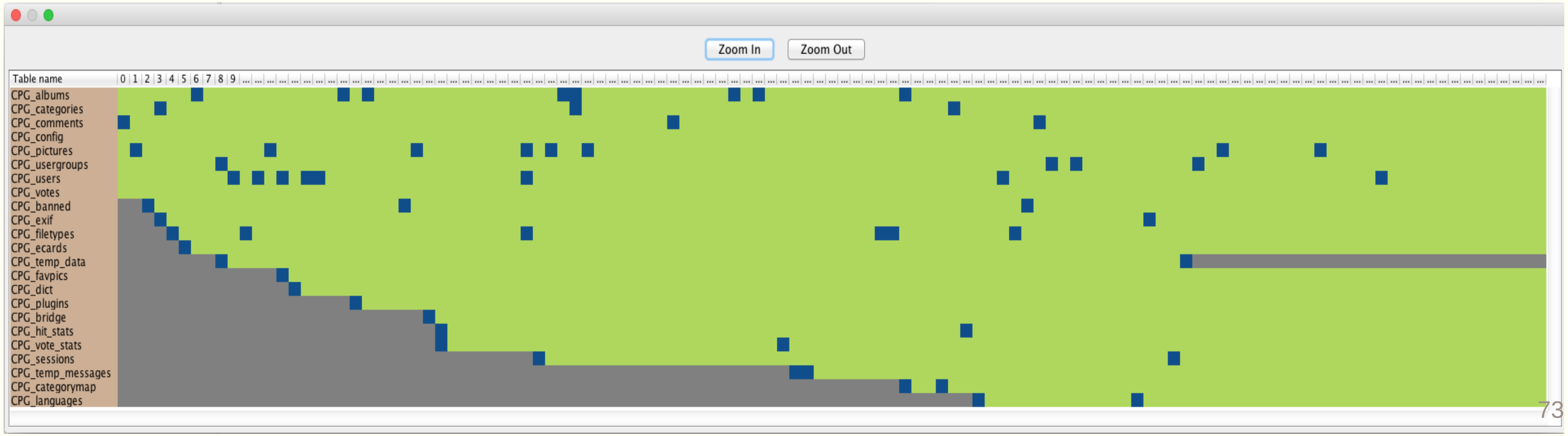
Wb	Wd	Wc	Cohesion	Separation
0.00	**1.00**	**0.00**	766.54	2053.20
0.33	**0.33**	**0.33**	**2243.29**	**512.37**
0.50	**0.50**	**0.00**	506.25	2196.72
0.50	**0.00**	**0.50**	2104.33	565.81
1.00	**0.00**	**0.00**	506.25	2196.72

typo3

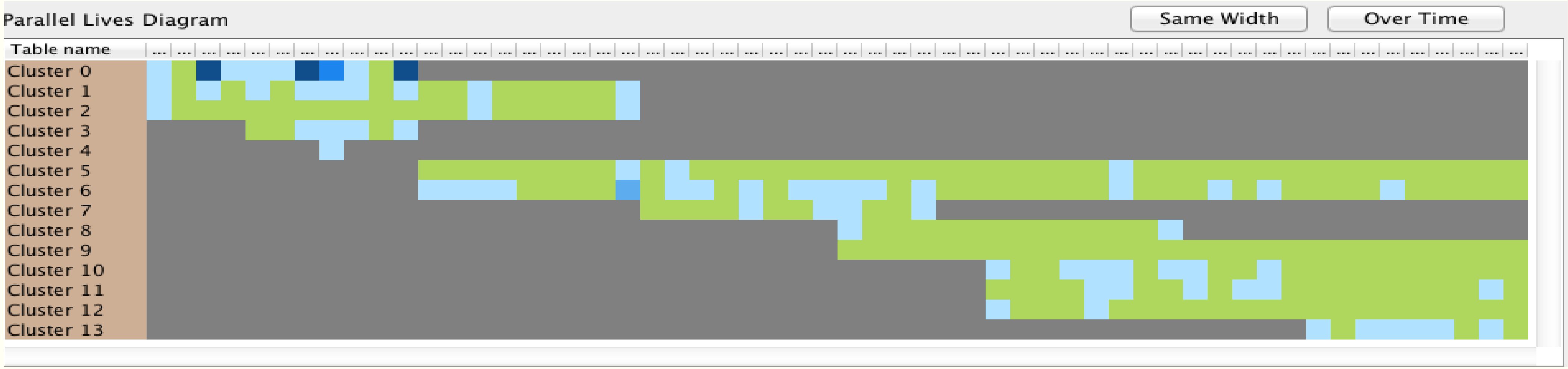
Wb	Wd	Wc	Cohesion	Separation
0.00	**1.00**	**0.00**	**414.14**	**1096.70**
0.33	**0.33**	**0.33**	192.07	1240.47
0.50	**0.50**	**0.00**	239.91	1213.34
0.50	**0.00**	**0.50**	208.57	1212.90
1.00	**0.00**	**0.00**	277.97	1200.29

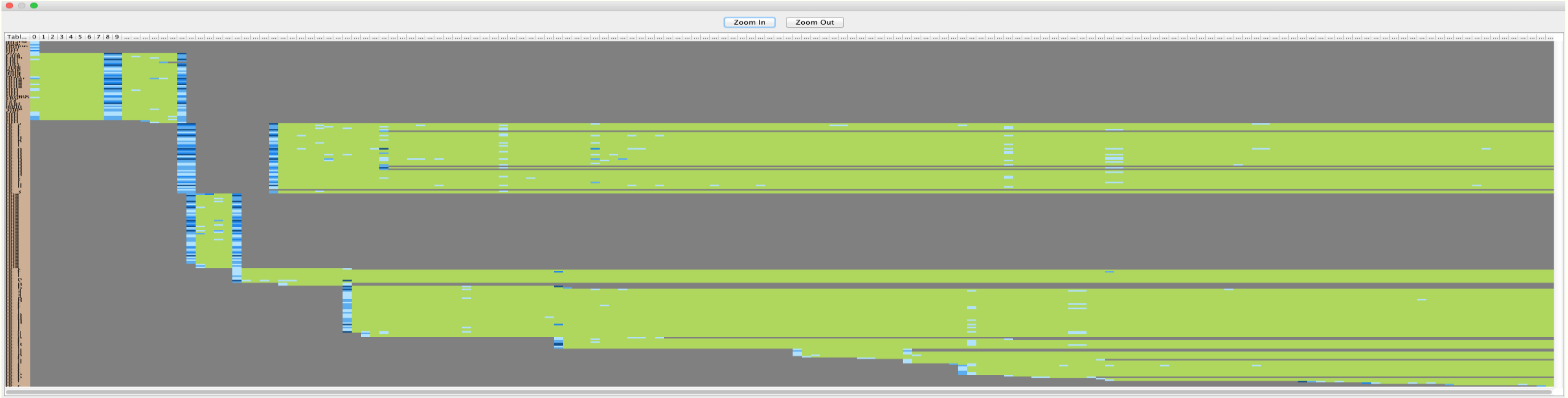
Overview vs PLD (Coppermine)



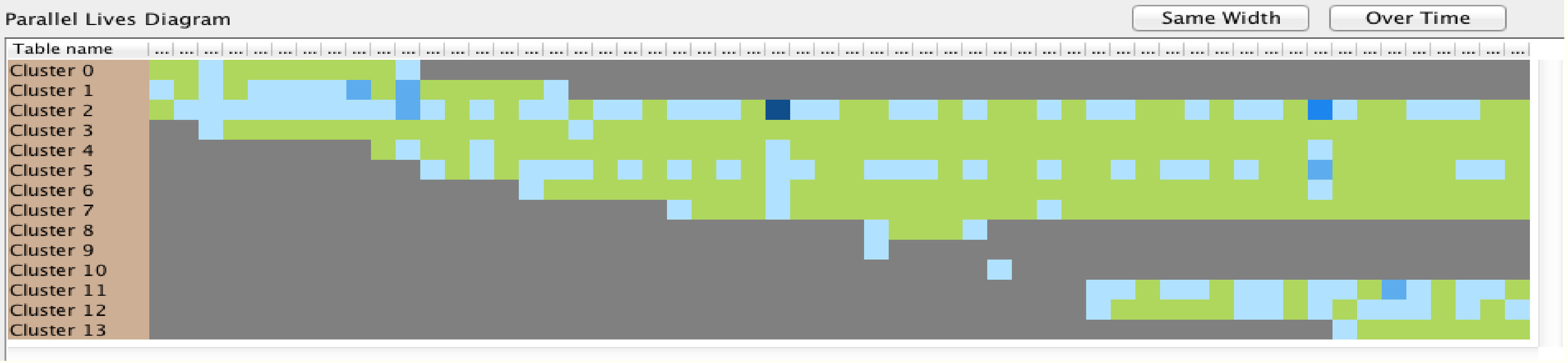


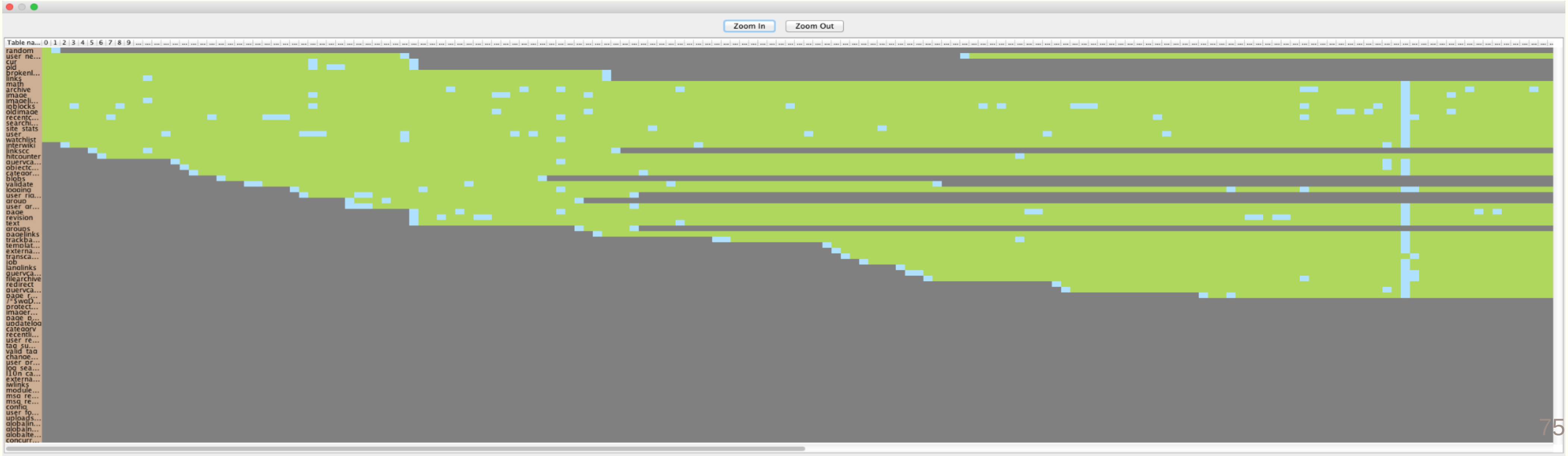
Overview vs PLD (Opencart)



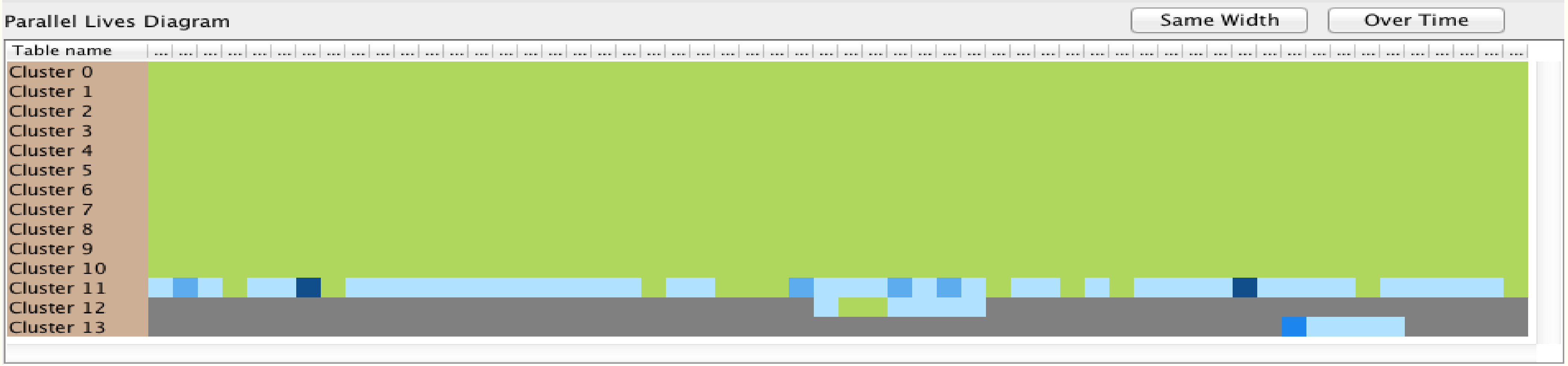


Overview vs PLD (mediaWiki)





Overview vs PLD (phpBB)



Parallel Lives Diagram

Table name

Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10
Cluster 11
Cluster 12
Cluster 13

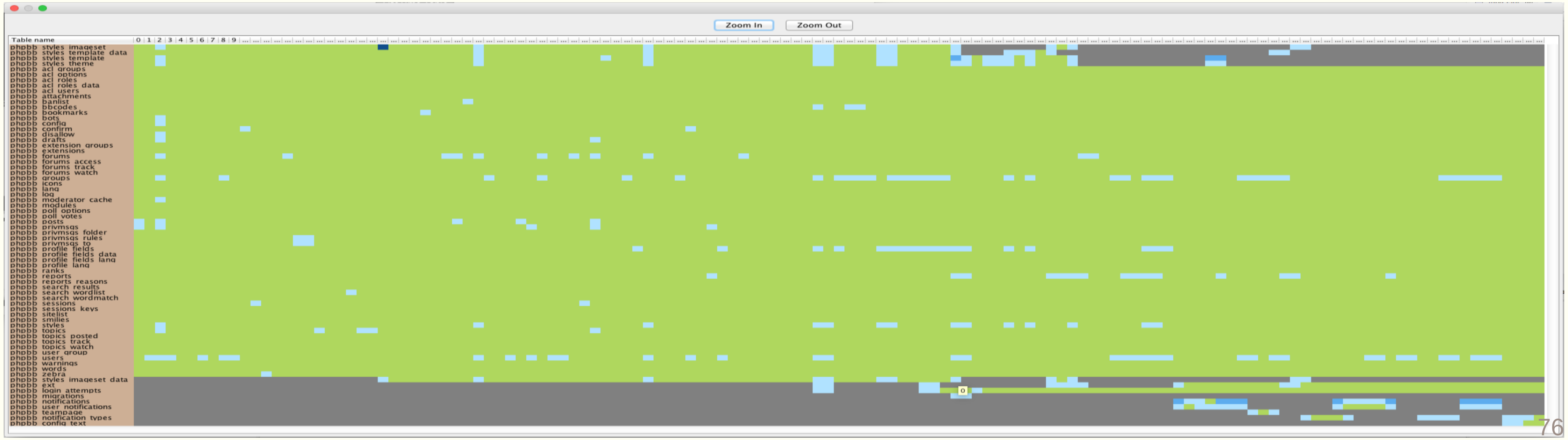
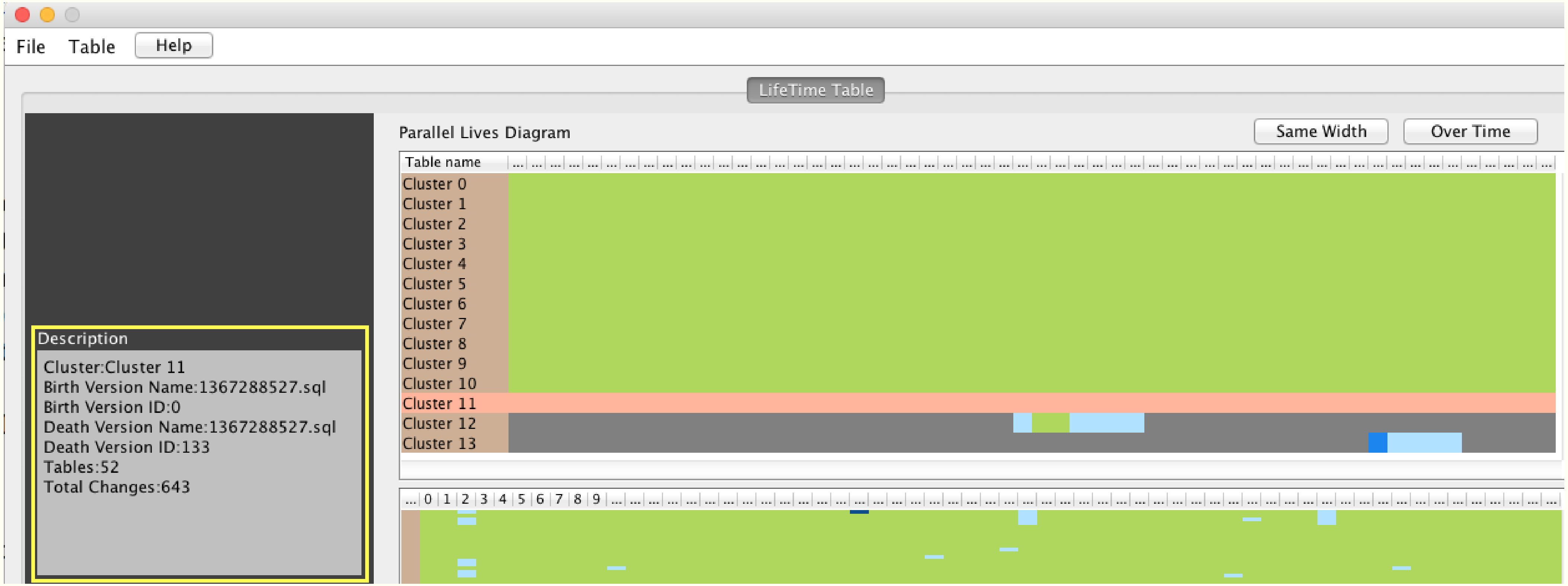


Table name

phpbb styles imageset
phpbb styles template data
phpbb styles template
phpbb styles theme
phpbb acl groups
phpbb acl options
phpbb acl roles
phpbb acl roles data
phpbb acl users
phpbb attachments
phpbb banlist
phpbb bbcodes
phpbb bookmarks
phpbb bots
phpbb config
phpbb confirm
phpbb disallow
phpbb drafts
phpbb extension groups
phpbb extensions
phpbb forums
phpbb forums access
phpbb forums track
phpbb forums watch
phpbb groups
phpbb icons
phpbb lang
phpbb log
phpbb moderator cache
phpbb modules
phpbb poll options
phpbb poll votes
phpbb posts
phpbb privmsgs
phpbb privmsgs folder
phpbb privmsgs rules
phpbb privmsgs to
phpbb profile fields
phpbb profile fields data
phpbb profile fields lang
phpbb profile lang
phpbb ranks
phpbb reports
phpbb reports reasons
phpbb search results
phpbb search wordlist
phpbb search wordmatch
phpbb sessions
phpbb sessions keys
phpbb sitelist
phpbb smilies
phpbb styles
phpbb topics
phpbb topics posted
phpbb topics track
phpbb topics watch
phpbb user group
phpbb users
phpbb warnings
phpbb words
phpbb zebra
phpbb styles imageset data
phpbb ext
phpbb login attempts
phpbb migrations
phpbb notifications
phpbb user notifications
phpbb teampage
phpbb notification types
phpbb config text

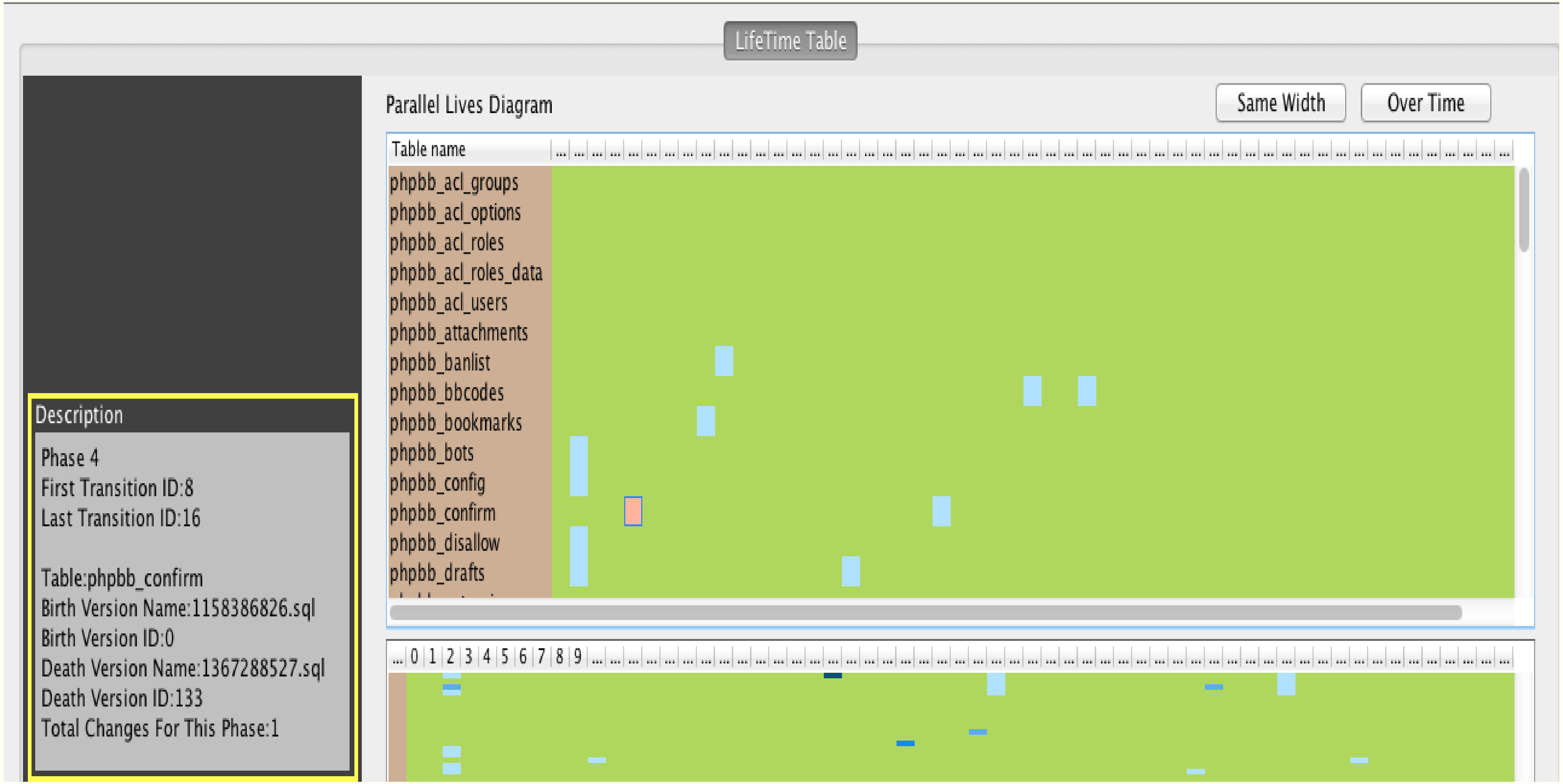
Details on demand for selected cluster



Description

Cluster:Cluster 11
Birth Version Name:1367288527.sql
Birth Version ID:0
Death Version Name:1367288527.sql
Death Version ID:133
Tables:52
Total Changes:643

Details on demand for selected cell



Description

Phase 4
First Transition ID:8
Last Transition ID:16

Table:phpbb_confirm
Birth Version Name:1158386826.sql
Birth Version ID:0
Death Version Name:1367288527.sql
Death Version ID:133
Total Changes For This Phase:1

References

[ECLI15]	Eclipse IDE. Available at https://eclipse.org/downloads/. Last accessed 2015-09-30.
[HECA15]	Hecate. Available at https://github.com/daintiness-group/hecate . Last accessed 2015-09-30.
[PPL15]	Plutarch's Parallel Lives at https://github.com/daintiness-group/plutarch_parallel_lives. Last accessed 2015-09-30
[Shne96]	Shneiderman, Ben. "The eyes have it: A task by data type taxonomy for information visualizations." Visual Languages, 1996. Proceedings., IEEE Symposium on. IEEE, 1996.
[SkVZ14]	Skoulis, Ioannis, Panos Vassiliadis, and Apostolos Zarras. "Open-Source Databases: Within, Outside, or Beyond Lehman's Laws of Software Evolution?."Advanced Information Systems Engineering. Springer International Publishing, 2014.
[TaSK05]	Pang-Ning Tan, Michael Steinbach and Vipin Kumar. Introduction to Data Mining. 1st ed. Pearson, 2005.
[TaTT06]	Mielikäinen, Taneli, Evimaria Terzi, and Panayiotis Tsaparas. "Aggregating time partitions." Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2006.
[TeTs06]	Terzi, Evimaria, and Panayiotis Tsaparas. "Efficient Algorithms for Sequence Segmentation." SDM. 2006.
[ZhSt05]	Xing, Zhenchang, and Eleni Stroulia. "Analyzing the evolutionary history of the logical design of object-oriented software." Software Engineering, IEEE Transactions on 31.10 (2005): 850-868.