Characterizing the Workload Patterns of Web Applications

Roozbeh Aghili, Heng Li, Foutse Khomh

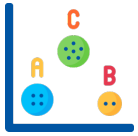
A little of context!

Web application

Workload

Workload pattern

Trace



Time-series clustering

A little of context!

Web application

Web applications are software services or applications accessible over the internet through web browsers [1].

Workload

Workload pattern

Trace

Time-series clustering

A little of context!

Web application

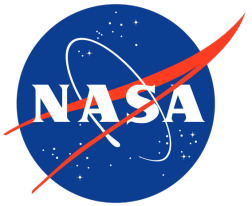
Workload

Workload pattern

Trace

Time-series clustering

Web applications are software services or applications accessible over the internet through web browsers [1].



A little of context!

Web application

Workload

Workload pattern

Trace

Time-series clustering

Workloads refer to tasks, processes, or operations managed by a computer system or a server or a cloud environment at a given time [2].

A little of context!

Web application

Workload

Workload pattern

Trace

Time-series clustering

Workloads refer to tasks, processes, or operations managed by a computer system or a server or a cloud environment at a given time [2].

Examples:

- Running web servers
- Managing database queries
- **User requests**

A little of context!

Web application

Workload

Workload pattern

Trace

Time-series clustering

Workload patterns are recurring characteristics or behaviors exhibited by an application's workloads within a specific time interval [3].

A little of context!

Web application

Workload

Workload pattern

Trace

Time-series clustering

Workload patterns are recurring characteristics or behaviors exhibited by an application's workloads within a specific time interval [3].

Important for tasks such as:

- Capacity planning
- Resource allocation
- Performance optimization

A little of context!

Web application

Workload

Workload pattern

Trace

Time-series clustering

A trace is a systematic and detailed record of events, actions, or data in a specific system used for various purposes, such as debugging and system optimization [4,5].

A little of context!

Web application

Workload

Workload pattern

Trace

Time-series clustering

A trace is a systematic and detailed record of events, actions, or data in a specific system used for various purposes, such as debugging and system optimization [4,5].

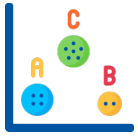
A little of context!

Web application

Workload

Workload pattern

Trace



Time-series clustering

Time-series clustering groups time-series data based on similarities and patterns [6]. Time-series is naturally high-dimensional and large in data size [7].

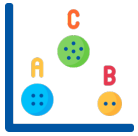
A little of context!

Web application

Workload

Workload pattern

Trace



Time-series clustering

Time-series clustering groups time-series data based on similarities and patterns [6]. Time-series is naturally high-dimensional and large in data size [7].

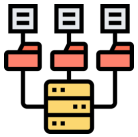
Improving:

- Decision-making
- Pattern recognition

Characterizing the Workload Patterns of Web Applications

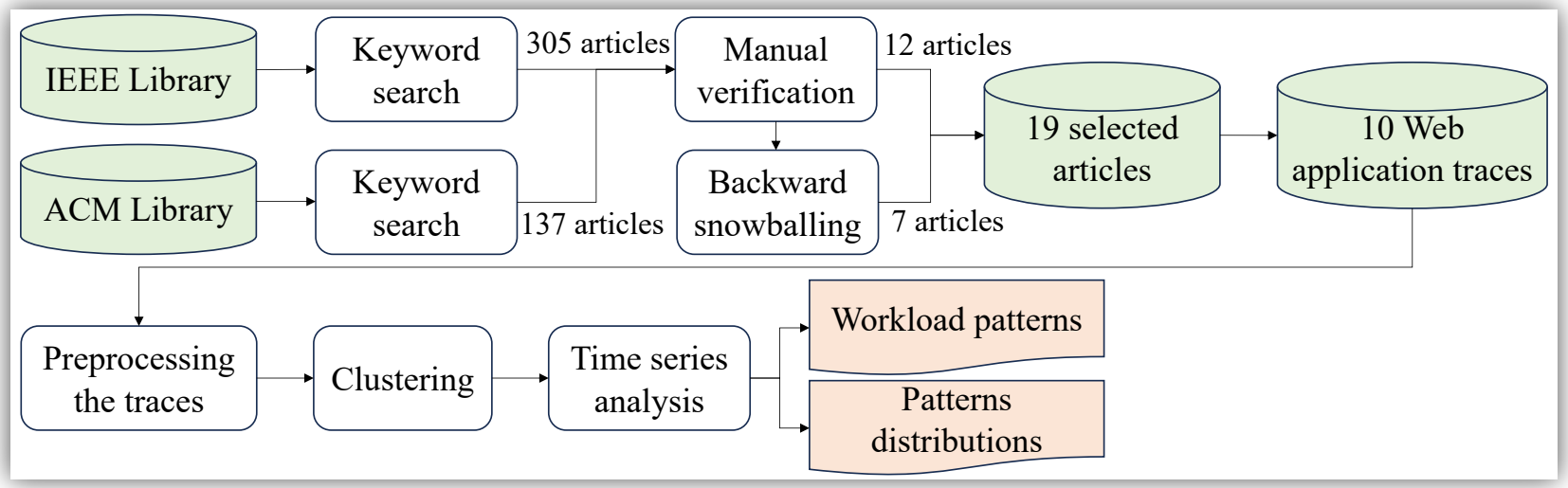
Characterizing the Workload Patterns of Web Applications

RQ1. What are the existing workload patterns in web application traces?

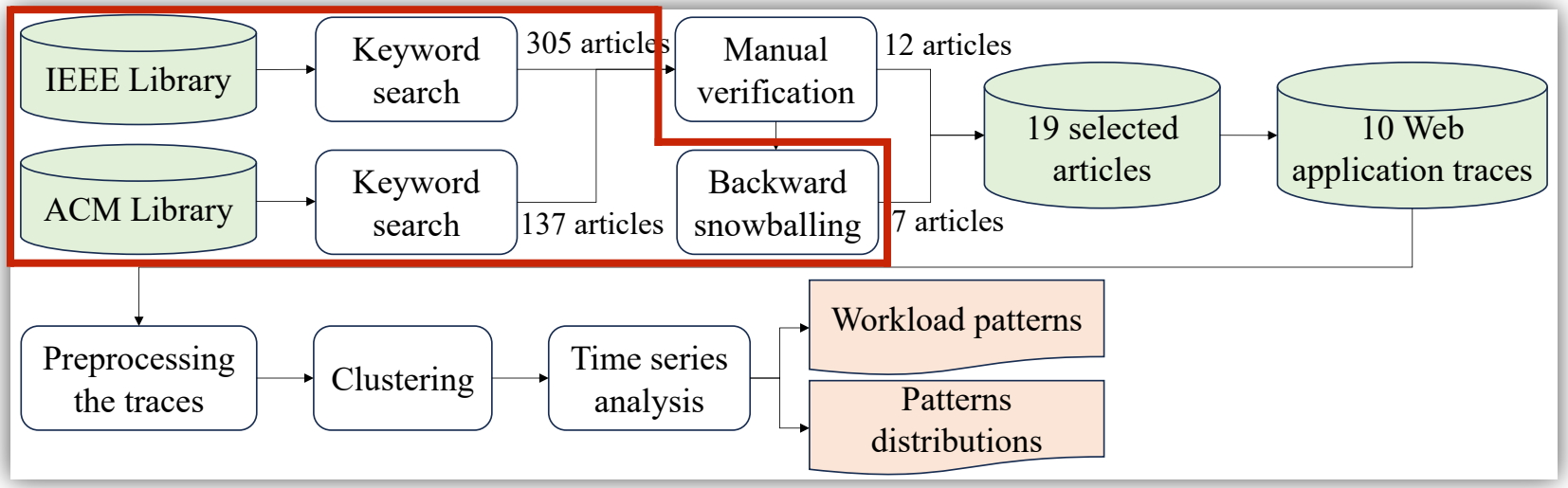


RQ2. How are different workload patterns distributed in web application traces?

The overview of our study

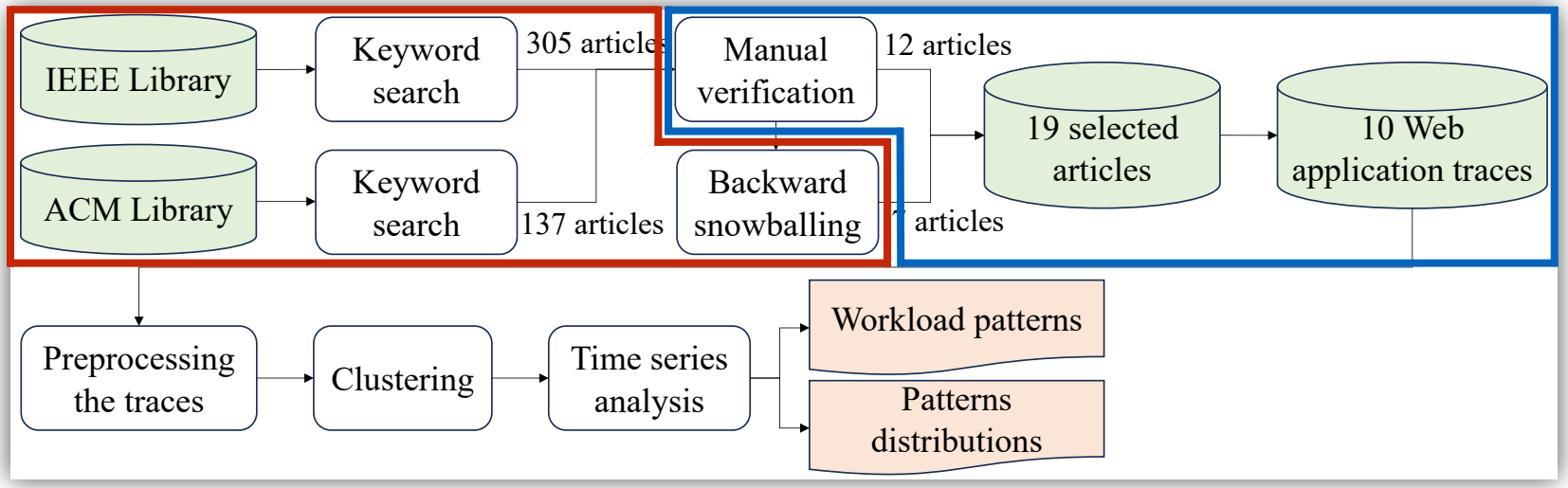


The overview of our study



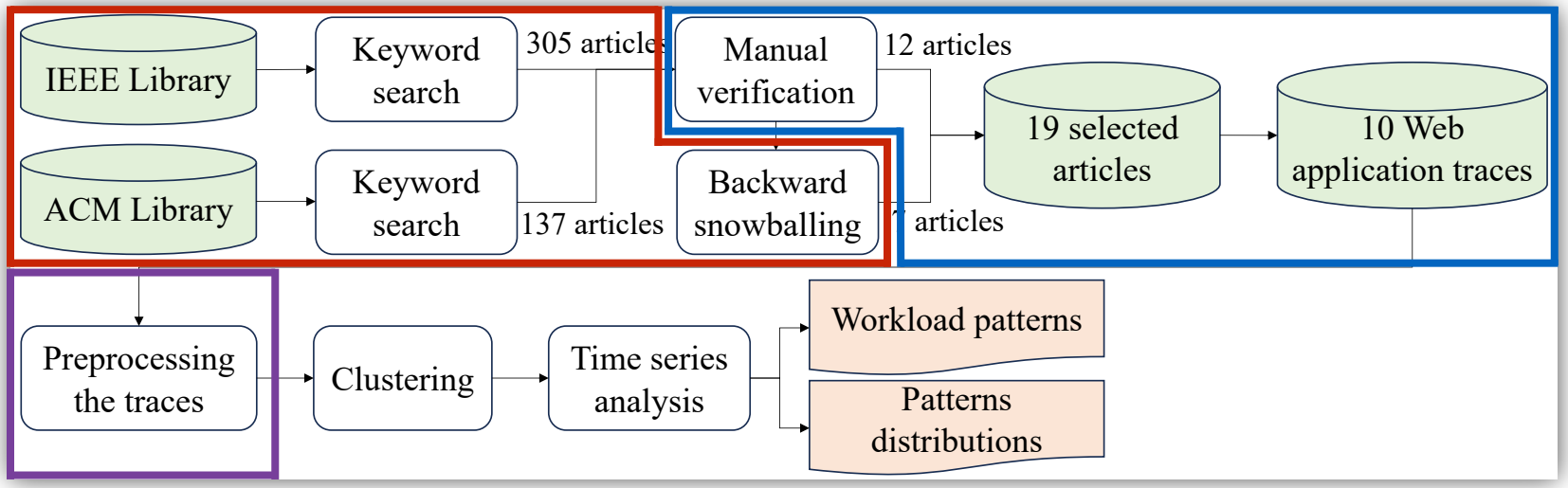
1. Systematic literature search

The overview of our study



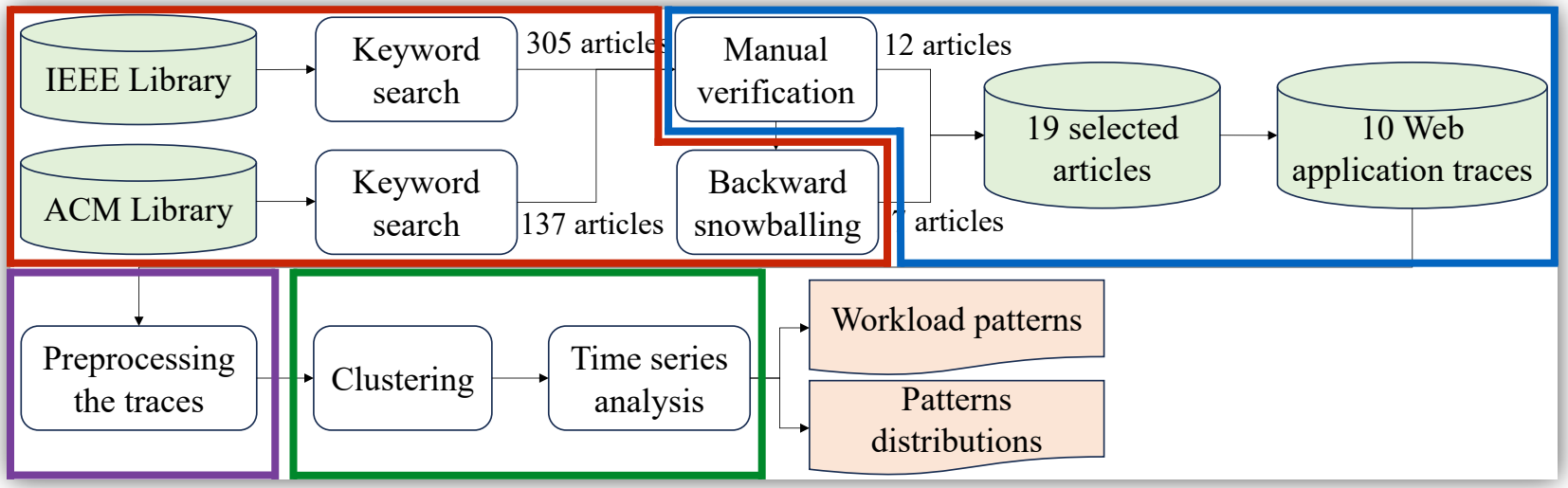
1. **Systematic literature search**
2. **Selecting web application traces**

The overview of our study



1. **Systematic literature search**
2. **Selecting web application traces**
3. **Preprocessing the traces**

The overview of our study



1. **Systematic literature search**
2. **Selecting web application traces**
3. **Preprocessing the traces**
4. **Time-series clustering and analysis**

List of web application traces

Trace	Duration	# Trace Instances
Wikipedia	5.5 years	1.0 T
WorldCup98	2 months	1.3 B
NASA	2 months	3.5 M
Saskatchewan	7 months	2.4 M
Calgary	1 year	727 K
EPA	1 day	47.7 K
Clarknet	14 days	3.3 M
Retailrocket	4 months	2.8 M
Boston	6 months	1.1 M
SDSC	1 day	28.3 K

Preprocessing the traces

Preprocessing the traces

Raw data

Wikipedia

en.m Cristiano_Júnior 7 0
en.m Cristiano_Lucarelli 5 0
en.m Cristiano_Ronaldo 4888 0

WorldCup98

2705258 - - [13/Jul/1998:22:00:01 +0000] "GET/images/102378.gif HTTP/1.0" 200 1658
1630377 - - [13/Jul/1998:22:00:01 +0000] "GET/images/hm_score_up_line03.gif HTTP/1.0" 200 90
917 - - [13/Jul/1998:22:00:01 +0000] "GET /images/s102377.gif HTTP/1.0" 200 173

Preprocessing the traces

Two granularities

Daily workloads provide detailed insights into user engagement, system load, and operations peaks over a 24-hour cycle. Useful for short-term trends, such as hourly spikes in traffic.

Weekly workloads offer a broader perspective, capturing trends and variations that span across different days of the week. Useful for longer-term trends, such as system load along different weeks of the year.

Preprocessing the traces

Processed data

Trace	Day	0	1	2	…	21	22	23
Boston	1994-11-21	0	0	0	…	0	0	0
Boston	1994-11-22	0	0	0	…	132	40	0
Boston	1994-11-23	143	664	763	…	0	0	0

Trace	Week	M	Tu	W	Th	F	Sa	Su
Boston	1994-11-27	100	2984	4481	1460	819	0	328
Boston	1994-12-04	499	304	774	299	523	561	246
Boston	1994-12-11	398	1135	623	21	611	19	733

Time-series clustering

1. **Aggregation of traces into time intervals (

& 

)
2. **Normalizing**
3. **Clustering**
    - K-means
    - Silhouette score
    - K ranging from 1 to 20

RQ1. What are the existing workload patterns in web application traces?

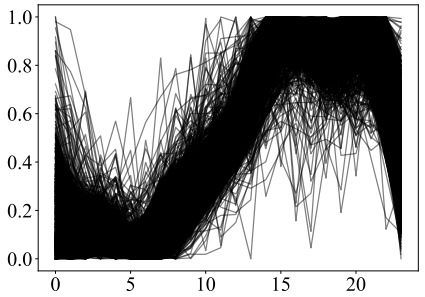
RQ1. What are the existing workload patterns in web application traces?

Metrics

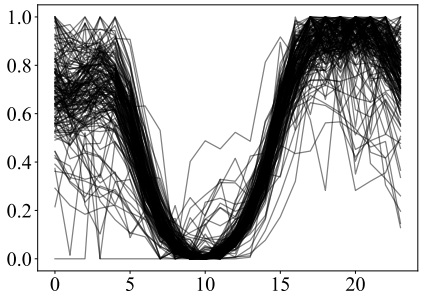
Metric	Definition
Skewness	Is a metric of the asymmetry of a distribution. A distribution is asymmetrical when its left and right sides are not mirror images [8].
Kurtosis	Is a statistical metric that quantifies how the tails of a distribution deviate from the tails of a normal distribution [8].
Active time	Is a metric employed to discover the duration in which the workloads remain active.
Climbing speed	Provides insights into the temporal characteristics of centroids. Quantifies the duration required for the centroid to transit from the first quartile (q1) to the third quartile (q3).
Descending speed	Another metric for quantifying the centroids. Quantifies the duration required for the centroid to transit from q3 to q1.

RQ1. What are the existing workload patterns in web application traces?

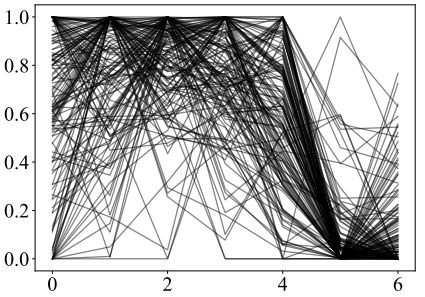
Our clustering reveals 4 daily workload patterns and 2 weekly workload patterns.



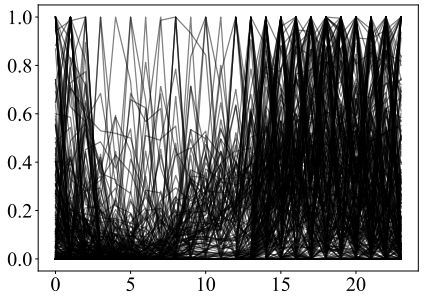
D1 (2164 instances)



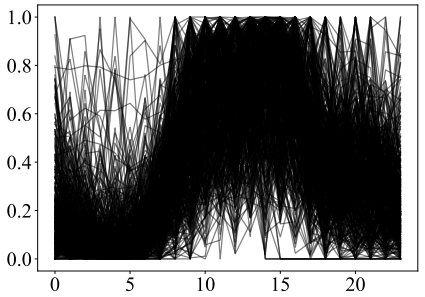
D2 (156 instances)



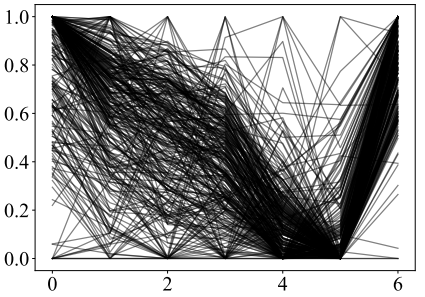
W1 (152 instances)



D3 (296 instances)



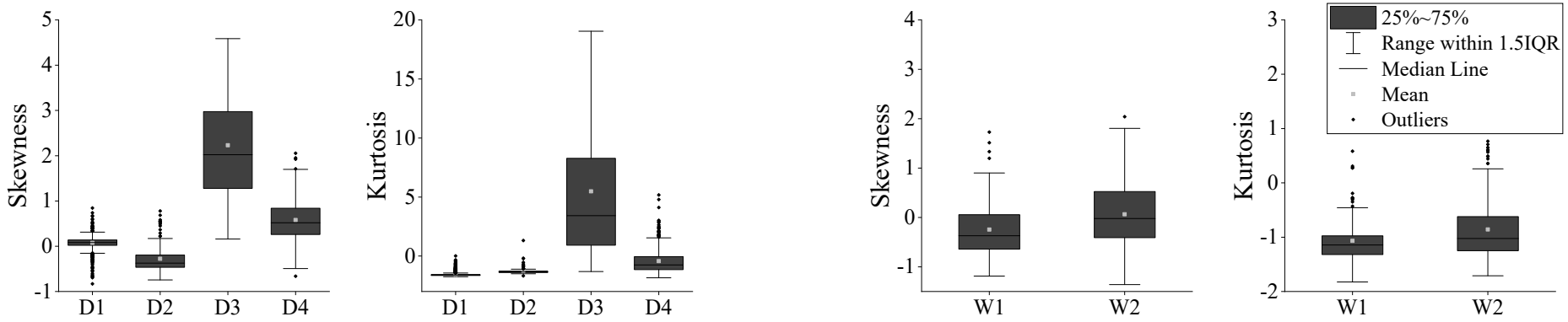
D4 (502 instances)



W2 (298 instances)

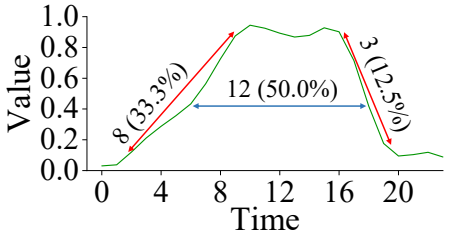
RQ1. What are the existing workload patterns in web application traces?

The majority of the workload patterns exhibit a stable, near-normal distribution.

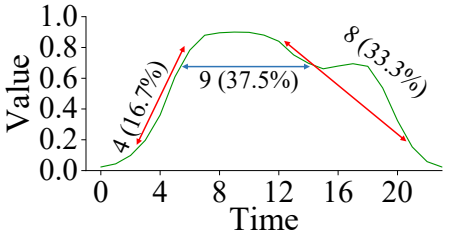


RQ1. What are the existing workload patterns in web application traces?

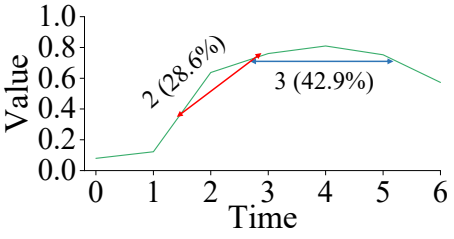
The workload patterns typically follow an inactive -> climbing -> active -> descending sequence, with different variations.



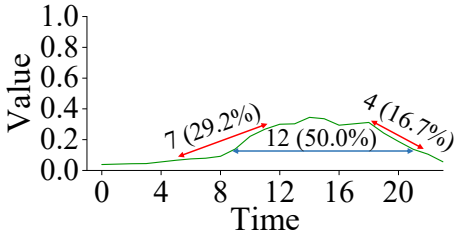
D1



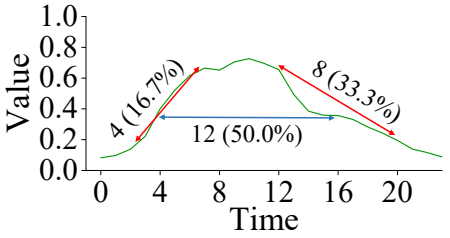
D2



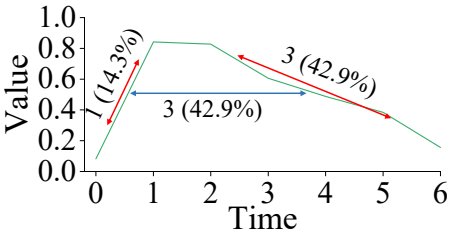
W1



D3



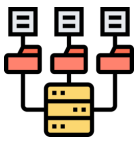
D4



W2

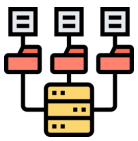
RQ1. What are the existing workload patterns in web application traces?

Following our clustering approach, we have identified four and two workload patterns at the daily and weekly levels, respectively. Using statistical metrics such as skewness and active time, we find that the majority of the identified patterns exhibit a stable distribution and follow a similar sequence. However, these patterns possess unique and distinctive characteristics, showcasing notable variations.



RQ2. How are different workload patterns distributed in web application traces?

Co-existence **Time dependence** **Association**



RQ2. How are different workload patterns distributed in web application traces?

Co-existence

Time dependence

Association

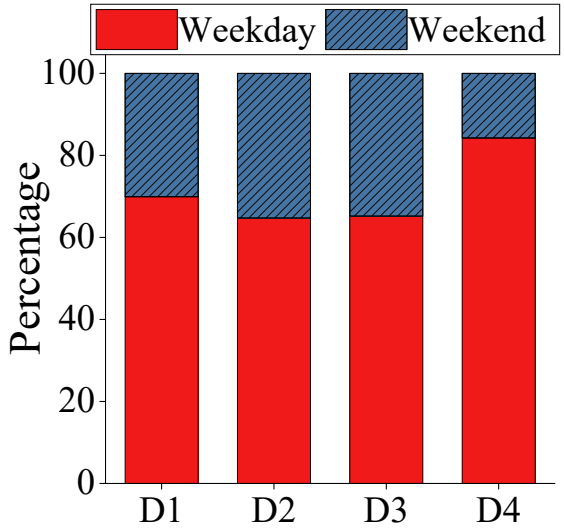
Most of the applications contain one or two workload patterns, while no daily or weekly pattern is exclusive to a single web application.

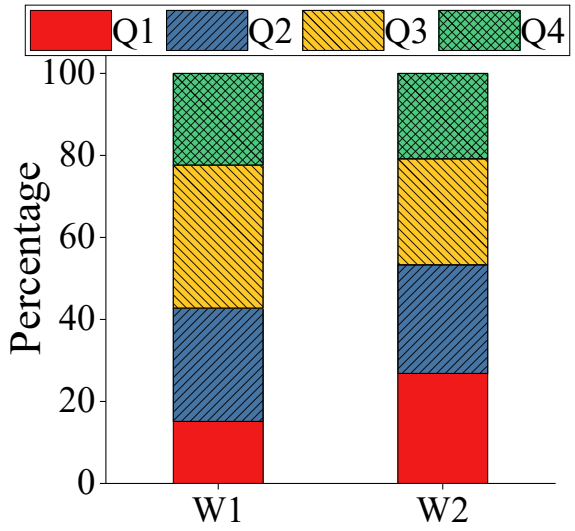
	D1	D2	D3	D4
Wikipedia	99.52	0.00	0.29	0.19
WorldCup98	45.45	18.18	32.95	3.42
NASA	23.73	0.00	1.69	74.58
Saskatchewan	15.89	0.00	0.00	84.11
Calgary	2.55	0.28	22.38	74.79
EPA	0.00	0.00	50.00	50.00
Clarknet	64.29	0.00	0.00	35.71
Retailrocket	0.00	99.28	0.72	0.00
Boston	0.00	0.56	99.44	0.00
SDSC	0.00	0.00	0.00	100.00

RQ2. How are different workload patterns distributed in web application traces?

Co-existence | **Time dependence** | Association

Different workload patterns showcase different temporal dependencies.

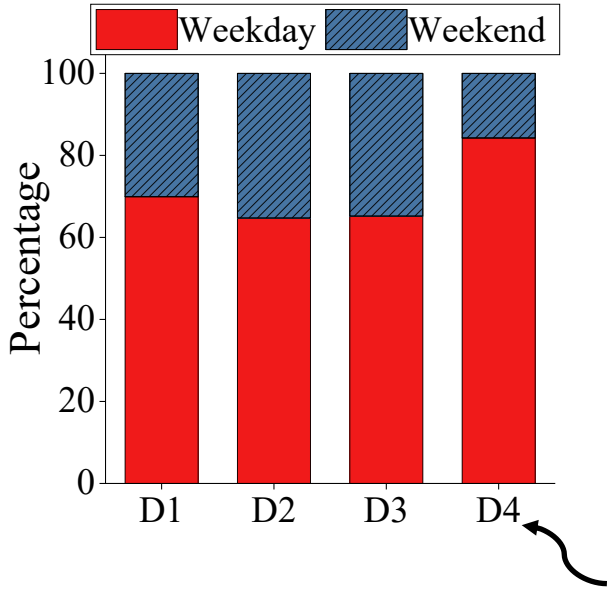




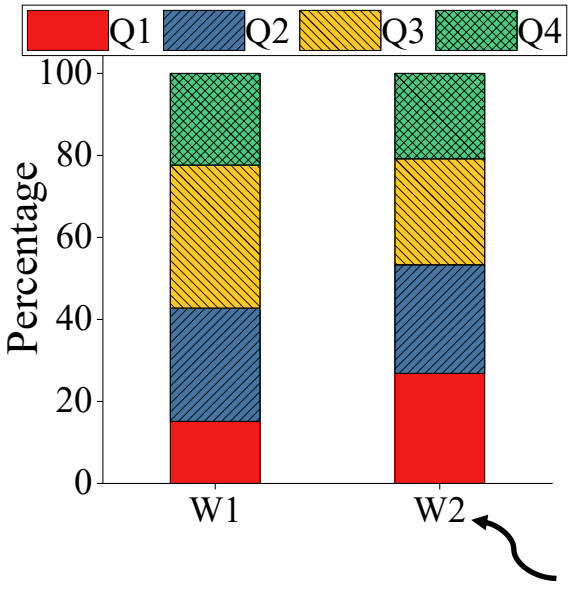
RQ2. How are different workload patterns distributed in web application traces?

Co-existence **Time dependence** Association

Different workload patterns showcase different temporal dependencies.



Calgary & Saskatchewan

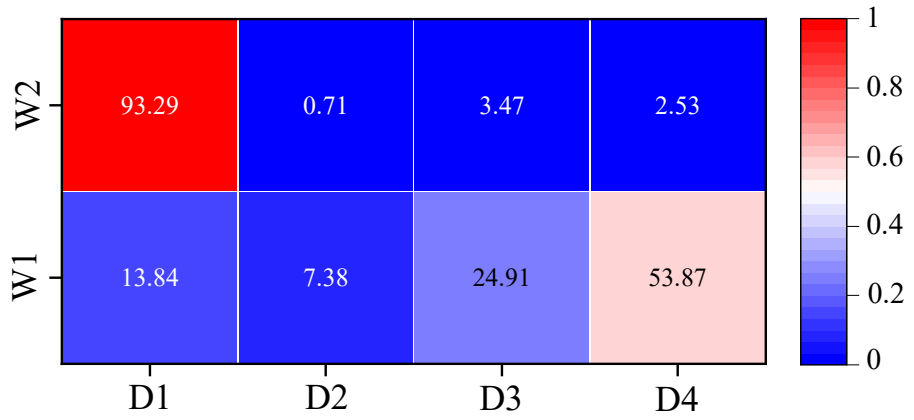


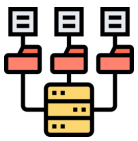
Wikipedia

RQ2. How are different workload patterns distributed in web application traces?

Co-existence Time dependence **Association**

Daily workload patterns have a strong association with weekly patterns.
Each weekly pattern has a dominant daily pattern.

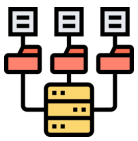




RQ2. How are different workload patterns distributed in web application traces?

Co-existence **Time dependence** **Association**

Daily and weekly patterns could be categorized as *global* and *local* meta-patterns.



RQ2. How are different workload patterns distributed in web application traces?

Our analysis indicates that most web applications have one or two workload patterns, whereas no daily or weekly patterns are exclusive to a web application. We also observe a strong association between weekly and daily patterns: each weekly pattern has a dominant daily pattern.

Implications of our study

Workload generation

- Future work should consider the daily and weekly patterns to design realistic workloads.
- Considering a mix of workload patterns and the associations between daily and weekly patterns can help in designing complex workload generation scenarios.

Resource allocation optimization

- Resource allocation strategies can leverage the daily and weekly workload patterns to obtain more cost-effective resource allocations.
- Resource allocation strategies should incorporate elasticity to effectively handle workload spikes.

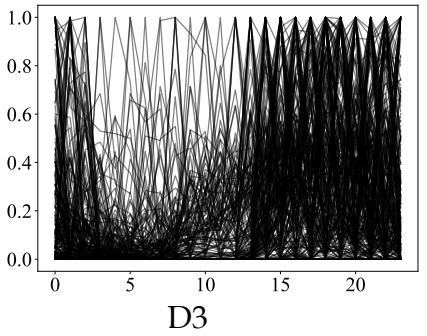
Implications of our study

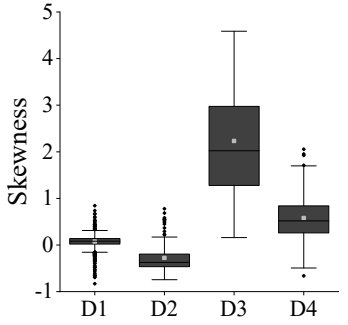
Workload generation

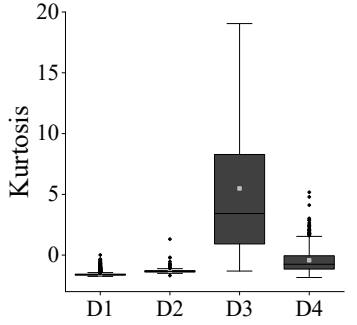
- Future work should consider the daily and weekly patterns to design realistic workloads.
- Considering a mix of workload patterns and the associations between daily and weekly patterns can help in designing complex workload generation scenarios.

Resource allocation optimization

- Resource allocation strategies can leverage the daily and weekly workload patterns to obtain more cost-effective resource allocations.
- **Resource allocation strategies should incorporate elasticity to effectively handle workload spikes.**





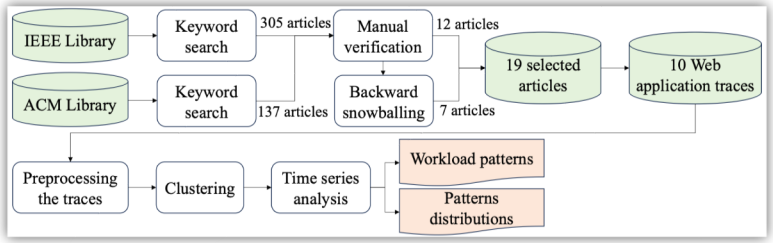


Characterizing the Workload Patterns of Web Applications

RQ1. What are the existing workload patterns in web application traces?

RQ2. How are different workload patterns distributed in web application traces?

The overview of our study



RQ1. What are the existing workload patterns in web application traces?

Following our clustering approach, we have identified four and two workload patterns at the daily and weekly levels, respectively. Using statistical metrics such as skewness and active time, we find that the majority of the identified patterns exhibit a stable distribution and follow a similar sequence. However, these patterns possess unique and distinctive characteristics, showcasing notable variations.

RQ2. How are different workload patterns distributed in web application traces?

Our analysis indicates that most web applications have one or two workload patterns, whereas no daily or weekly patterns are exclusive to a web application. We also observe a strong association between weekly and daily patterns: each weekly pattern has a dominant daily pattern.

References

1. Mehdi Jazayeri. 2007. Some trends in web application development. In Future of Software Engineering (FOSE'07). IEEE, 199–213.

2. Domenico Ferrari, Giuseppe Serazzi, and Alessandro Zeigner. 1983. Measurement and tuning of computer systems. (No Title) (1983).

3. Waheed Iqbal, Abdelkarim Erradi, and Arif Mahmood. 2018. Dynamic workload patterns prediction for proactive auto-scaling of web applications. Journal of Network and Computer Applications 124 (2018), 94–107.

4. Jing Guo, Zihao Chang, Sa Wang, Haiyang Ding, Yihui Feng, Liang Mao, and Yungang Bao. 2019. Who limits the resource efficiency of my datacenter: An analysis of alibaba datacenter traces. In Proceedings of the International Symposium on Quality of Service. 1–10.

5. Charles Reiss, Alexey Tumanov, Gregory R Ganger, Randy H Katz, and Michael A Kozuch. 2012. Heterogeneity and dynamicity of clouds at scale: Google trace analysis. In Proceedings of the third ACM symposium on cloud computing. 1–13.

6. Saeed Aghabozorgi, Ali Seyed Shirkhorshidi, and Teh Ying Wah. 2015. Time-series clustering–a decade review. Information systems 53 (2015), 16–38.

7. Eamonn Keogh and Shruti Kasetty. 2002. On the need for time series data mining benchmarks: a survey and empirical demonstration. In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. 102–111.

8. Daniel Zwillinger and Stephen Kokoska. 1999. CRC standard probability and statistics tables and formulae. Crc Press.