



AROWWAI INDUSTRIES

# OPERATION ANALYTICS & INVESTIGATING METRIC SPIKE

ANALYZING OPERATIONAL DATA TO IDENTIFY  
TRENDS AND RESOLVE SPIKES

PRESENTED BY: BHAWNA KUMARI



# PROJECT OVERVIEW



## DOMAIN

Operations & Performance Analytics

## GOAL

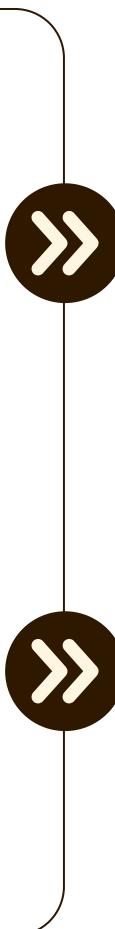
Analyze end-to-end job performance data and investigate metric spikes

## KEY FOCUS

Daily/weekly spikes, user engagement, device-based usage trends

# CASE STUDY 1: INVESTIGATING METRIC SPIKES

## A. FOR WEEKLY USER ENGAGEMENT



Week	Weekly_User_engagement
17	740
18	1260
19	1287
20	1351
21	1299
22	1381
23	1446
24	1471
25	1459
26	1509
27	1573
28	1577
29	1607
30	1706
31	1514
32	1454
33	1438
34	1443
35	118

```
SELECT  
    EXTRACT(WEEK FROM occurred_at) AS 'week',  
    COUNT(DISTINCT user_id) AS 'weekly_user_engagement'  
FROM  
    events  
GROUP BY week(occurred_at);
```

### FINDINGS:

USER ENGAGEMENT SAW AN INCREASE FROM WEEK 18 TO WEEK 30, AFTER WHICH IT BEGAN TO DECLINE.



# CASE STUDY 1: INVESTIGATING METRIC SPIKES

## B. USER GROWTH ANALYSIS

```
• set @g :=0;  
• select a.no_of_users, a.date,  
    (@g := @g + a.no_of_users ) as User_growth  
from  
(select count (user_id) as no_of_users,  
date(created_at) as date  
from users where state = "active"  
group by date (created_at))a;
```

no_of_users	date	user_growth
7	2013-01-01	7
7	2013-01-02	14
6	2013-01-03	20
1	2013-01-04	21
2	2013-01-05	23
3	2013-01-06	26
4	2013-01-07	30
2	2013-01-08	32
6	2013-01-09	38
6	2013-01-10	44
6	2013-01-11	50
3	2013-01-12	53



FINDINGS : USER GROWTH VARIED SIGNIFICANTLY POST WEEK 31

# CASE STUDY 1: INVESTIGATING METRIC SPIKES

## C. WEEKLY ENGAGEMENT PER DEVICE

```
• SELECT week(occurred_at) as Weeks,  
    device,  
    count(distinct user_id)as User_engagement FROM events  
GROUP BY device,  
    week(occurred_at)  
ORDER BY week(occurred_at);
```

»»  
**DEVICE INSIGHTS: MACBOOK &  
IPHONE USERS MOST ACTIVE**

Weeks	device	User_engagement
17	acer aspire desktop	12
17	acer aspire notebook	23
17	amazon fire phone	4
17	asus chromebook	23
17	dell inspiron desktop	20
17	dell inspiron notebook	48
17	hp pavilion desktop	17
17	htc one	19
17	ipad air	29
17	ipad mini	19
17	iphone 4s	27
17	iphone 5	69
17	iphone 5s	47
17	kindle fire	6
17	lenovo thinkpad	94
17	mac mini	7
17	macbook air	61
17	macbook pro	154



# CASE STUDY 1: INVESTIGATING METRIC SPIKES

## D. EMAIL ENGAGEMENT ANALYSIS

```
• SELECT  
    WEEK(occurred_at) AS Week,  
    COUNT(DISTINCT (CASE  
        WHEN action = 'sent_weekly_digest' THEN user_id  
        END)) AS weekly_digest,  
    COUNT(DISTINCT (CASE  
        WHEN action = 'sent_reengagement_email' THEN user_id  
        END)) AS reengagement_mail,  
    COUNT(DISTINCT (CASE  
        WHEN action = 'email_open' THEN user_id  
        END)) AS opened_email,  
    COUNT(DISTINCT (CASE  
        WHEN action = 'email_clickthrough' THEN user_id  
        END)) AS email_clickthrough  
  
FROM  
    emails  
GROUP BY WEEK(occurred_at)  
ORDER BY WEEK(occurred_at);
```

### FINDINGS:

- THE EMAIL OPENING RATE IS AROUND 34% AND EMAIL CLICKING RATE IS AROUND 15%.
- THE USERS ARE ENGAGING WITH THE EMAIL SERVICE WHICH IS GOOD FOR THE COMPANY TO EXPAND.

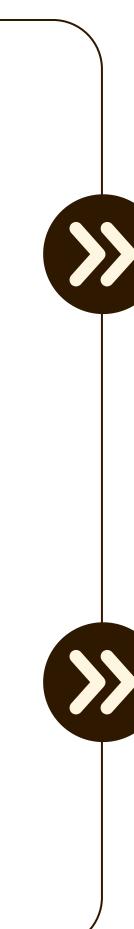
Week	weekly_digest	reengagement_mail	opened_email	email_clickthrough
17	908	73	310	166
18	2602	157	900	425
19	2665	173	961	476
20	2733	191	989	501
21	2822	164	996	436
22	2911	192	965	478
23	3003	197	1057	529
24	3105	226	1136	549
25	3207	196	1084	524
26	3302	219	1149	550
27	3399	213	1207	613
28	3499	213	1228	594
29	3592	213	1201	583
30	3706	231	1363	625
31	3793	222	1338	444



# CASE STUDY 2: JOB DATA ANALYSIS

## A. JOBS REVIEWED OVER TIME

```
select ds as dates, round((count(job_id)/sum(time_spent))*3600) as "jobs reviewed per hour"  
from job_data  
where ds between '2020-11-01' and '2020-11-30'  
group by ds;
```



	dates	jobs reviewed per hour
▶	2020-11-30	180
	2020-11-29	180
	2020-11-28	218
	2020-11-27	35
	2020-11-26	64
	2020-11-25	80

JOBS REVIEWED: MAX ON  
NOVEMBER 28, 2020 (218 JOBS)



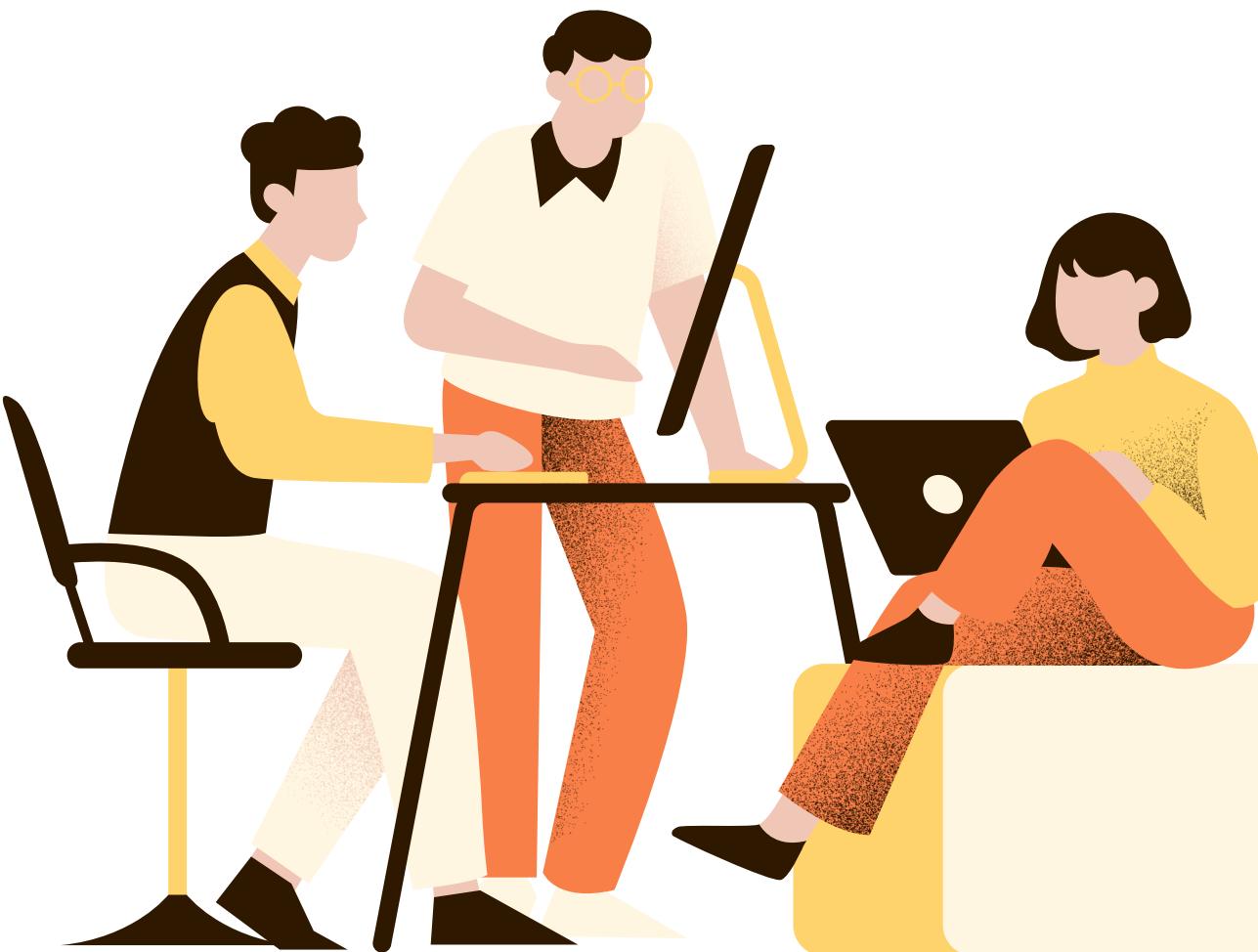
# CASE STUDY 2: JOB DATA ANALYSIS

## B. 1:7 DAY ROLLING AVERAGE OF THOUGHPUT

```
select round(count(event)/sum(time_spent),2) as "weekly throughput"  
from job_data;
```



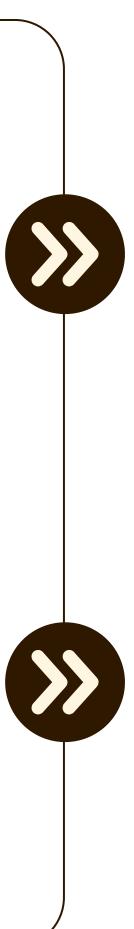
**FINDINGS:**  
THROUGHPUT:  
AVG = 0.03,



# CASE STUDY 2: JOB DATA ANALYSIS

## B.2 : AVERAGE DAILY METRIC THROUGHPUT.

```
select ds as dates, round(count(event)/sum(time_spent),2) as "daily throughput"  
from job_data  
group by ds  
order by ds;
```



	dates	daily throughput
▶	2020-11-25	0.02
▶	2020-11-26	0.02
▶	2020-11-27	0.01
▶	2020-11-28	0.06
▶	2020-11-29	0.05
▶	2020-11-30	0.05

### FINDINGS:

ON 2020-11-28 THE THROUGHPUT IS HIGHEST 0.06.



# CASE STUDY 2: JOB DATA ANALYSIS

## C. LANGUAGE SHARE ANALYSIS

```
select language,  
       count(job_id) as no_ofjobs,  
       count(job_id)*100/sum(count(*))over() as percentage  
  from job_data  
 where ds between '2020-11-01' and '2020-11-30'  
 group by language;
```

	language	no_ofjobs	percentage
▶	English	1	12.5000
▶	Arabic	1	12.5000
▶	Persian	3	37.5000
▶	Hindi	1	12.5000
▶	French	1	12.5000
▶	Italian	1	12.5000

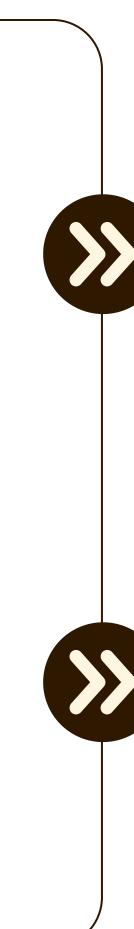
**FINDINGS:**  
**PERSIAN LANGUAGE IS THE HIGHEST WITH 37.5% TOTAL.**



# CASE STUDY 2: JOB DATA ANALYSIS

## D. DUPLICATION ROWS DETECTION

```
SELECT ds, job_id, actor_id, event, language, time_spent, org,  
CASE when duplicates = 1 then "No Duplicate" else "Duplicate" end as Duplicate  
FROM  
( SELECT *, row_number() OVER (partition by ds, job_id, actor_id, event, language, time_spent, org)  
as duplicates FROM job_data ) as a ;
```



	ds	job_id	actor_id	event	language	time_spent	org	Duplicate
▶	2020-11-25	20	1003	transfer	Italian	45	C	No Duplicate
	2020-11-26	23	1004	skip	Persian	56	A	No Duplicate
	2020-11-27	11	1007	decision	French	104	D	No Duplicate
	2020-11-28	23	1005	transfer	Persian	22	D	No Duplicate
	2020-11-28	25	1002	decision	Hindi	11	B	No Duplicate
	2020-11-29	23	1003	decision	Persian	20	C	No Duplicate
	2020-11-30	21	1001	skip	English	15	A	No Duplicate
	2020-11-30	22	1006	transfer	Arabic	25	B	No Duplicate



FINDINGS: NONE FOUND – ALL ROWS WERE UNIQUE

# KEY INSIGHTS

- Rolling metrics are better to smooth out daily fluctuations
- Metric spikes often correlate with user behavior patterns
- Device preference insights help optimize app experience
- Email engagement shows opportunity for CRM expansion



# LEARNINGS

- Mastered writing **efficient SQL** for daily/weekly metrics
- Understood importance of **metric context** and smoothing
- Gained experience in **root cause analysis** through spikes
- Strengthened **operational visibility** through **structured queries**



# BUSINESS IMPACT



- Helps proactively detect system failures or performance drops
- Enables real-time tracking of operational KPIs
- Prevents downtime by early anomaly detection
- Optimizes decision-making with spike awareness



# CONTACT ME:

E-MAIL

bhawnak0313@gmail.com

LINKEDIN

[linkedin.com/in/bhawna-kumari-data-analyst/](https://linkedin.com/in/bhawna-kumari-data-analyst/)

GITHUB

<https://github.com/Bhawna013>

123 Anywhere St., Any City

# THANK YOU

