

# Изучение структуры таблиц

## Общая информация о таблицах:

- Таблица issues: 51,955 строк
- Таблица resolutions: 176 строк

```
SELECT 'issues' as table_name, COUNT(*) as row_count FROM issues
UNION ALL
SELECT 'resolutions' as table_name, COUNT(*) as row_count FROM
resolutions;
```

	AZ table_name	123 row_count
1	issues	51 955
2	resolutions	176

```
SELECT * FROM issues LIMIT 10;
```

	123 created	AZ key	A-Z resolution	123 resolved	A-Z category
1	1 685 582 400 000	CRAZY-1787362	4f6b29cc3004e9443e453da7	1 685 630 400 000	Remote
2	1 685 582 400 000	CRAZY-1787363	4f6b29cc3004e9443e453da7	1 685 583 000 000	Remote
3	1 685 582 400 000	CRAZY-1787364	4f6b29cc3004e9443e453da7	1 685 583 000 000	Remote
4	1 685 584 700 000	CRAZY-1787366	4f6b29cc3004e9443e453da7	1 685 635 000 000	Remote
5	1 685 584 700 000	CRAZY-1787368	4f6b29cc3004e9443e453da7	1 685 635 000 000	Remote
6	1 685 584 700 000	CRAZY-1787365	4f6b29cc3004e9443e453da7	1 685 635 000 000	Remote
7	1 685 584 700 000	CRAZY-1787367	4f6b29cc3004e9443e453da7	1 685 635 000 000	Remote
8	1 685 584 900 000	CRAZY-1787369	4f6b29cc3004e9443e453da7	1 685 600 600 000	Remote
9	1 685 586 000 000	CRAZY-1787370	4f6b29cc3004e9443e453da7	1 685 586 600 000	Remote
10	1 685 586 000 000	CRAZY-1787371	4f6b29cc3004e9443e453da7	1 685 586 600 000	Remote

## Просмотр примеров данных

Из issues (первые 10 записей)

# Изучение структуры таблиц

## Просмотр примеров данных

Из resolutions (первые 10 записей)

```
SELECT *  
FROM resolutions  
LIMIT 10;
```

	A-Z id	A-Z key
1	4f6b29cc3004e9443e453da7	fixed
2	504eea3de4b0b9df6b280b73	won'tFix
3	504eea49e4b0b9df6b280b7d	can'tReproduce
4	506d8453e4b0840e65b4926b	duplicate
5	506d8582e4b0840e65b49298	invalid
6	506d9852e4b0840e65b49382	willNotFix
7	506d9f62e4b0840e65b494f5	incomplete
8	506eda09e4b06d25c5bb09d7	later
9	537c8e19e4b02fe867e8b76e	domainSuspended
n	5527c097e4b00931e465a35d	transfer

```
SELECT DISTINCT category FROM issues ORDER BY category;
```

```
SELECT  
    MIN(created) as min_created,  
    MAX(created) as max_created,  
    MIN(resolved) as min_resolved,  
    MAX(resolved) as max_resolved  
FROM issues;
```

	A-Z category
1	Local
2	New User
3	Remote
4	Software License

	123 min_created	123 max_created	123 min_resolved	123 max_resolved
1	16 856 076	1 688 169 000 000	1 685 583 000 000	1 685 623 700 000 000

## Структура столбцов issues:

- 4 уникальные категории: Local, New User, Remote, Software License
- Диапазон дат в миллисекундах

# Выводы из структурного анализа

Структура данных показывает:

- 📝 Относительно простую классификацию по 4 категориям
- ⌚ Упрощённую сегментацию и анализ
- 👤 Возможную избыточность справочника (5 типов резолюций, используются не все)



# Проверка качества данных

## Проверка NULL значений в таблице:

- Всего строк: 51,955
- Отсутствие даты создания: 3 записи (0.006%)
- Отсутствие даты решения: 2,766 записей (5.32%)

```
SELECT
    COUNT(*) as total_rows,
    COUNT(key) as has_key,
    COUNT(created) as has_created,
    COUNT(resolved) as has_resolved,
    COUNT(resolution) as has_resolution,
    COUNT(category) as has_category,
    COUNT(*) - COUNT(key) as missing_key,
    COUNT(*) - COUNT(created) as missing_created,
    COUNT(*) - COUNT(resolved) as missing_resolved,
    COUNT(*) - COUNT(resolution) as missing_resolution,
    COUNT(*) - COUNT(category) as missing_category
FROM issues;
```

	123 total_rows	123 has_key	123 has_created	123 has_resolved	123 has_resolution	123 has_category
1	51 955	51 955	51 952	49 189	51 955	51 955

	123 missing_key	123 missing_created	123 missing_resolved	123 missing_resolution	123 missing_category
	0	3	2 766	0	0

# Выявление аномалий в данных

## Классификация аномалий:

- Тикеты с `created = 1970-01-01 (epoch)`: 2 записи
- Тикеты где `resolved < created`: 0 записей
- Тикеты с обработкой > 1 года: 3 записи

Корректные решённые тикеты:  
44,889 запись

```
SELECT
    'Тикеты с created = 1970-01-01 (epoch)' as issue_type,
    COUNT(*) as count
FROM issues
WHERE created < 10000000000000

UNION ALL

SELECT
    'Тикеты где resolved < created' as issue_type,
    COUNT(*) as count
FROM issues
WHERE resolved IS NOT NULL AND resolved < created

UNION ALL
```

```
SELECT
    'Тикеты с обработкой > 1 года' as issue_type,
    COUNT(*) as count
FROM issues
WHERE resolved IS NOT NULL
    AND resolved > created
    AND (resolved - created) / 1000 / 60 / 60 / 24 > 365

UNION ALL

SELECT
    'Корректные решённые тикеты' as issue_type,
    COUNT(*) as count
FROM issues
WHERE resolved IS NOT NULL
    AND resolved > created
    AND created > 10000000000000
    AND (resolved - created) / 1000 / 60 / 60 / 24 <= 365;
```

issue_type	count
Корректные решённые тикеты	44 889
Тикеты с обработкой > 1 года	3
Тикеты где resolved < created	0
Тикеты с created = 1970-01-01 (epoch)	2

# Проверка связности данных (результаты)

## Анализ резолюций в таблице issues:

- Резолюции, которых нет в справочнике resolutions: 1 строка
- Всего резолюций в справочнике: 176
- Реально используемых резолюций: только 5
- Неиспользуемые резолюции: 171

	123 total_resolutions_in_dict	123 used_resolutions
1	176	5

```
-- Резолюции в issues, которых нет в resolutions
SELECT DISTINCT i.resolution
FROM issues i
LEFT JOIN resolutions r ON i.resolution = r.id
WHERE i.resolution IS NOT NULL
    AND r.id IS NULL;
```

```
-- Используемые резолюции
SELECT
    COUNT(DISTINCT r.id) as total_resolutions_in_dict,
    COUNT(DISTINCT i.resolution) as used_resolutions
FROM resolutions r
LEFT JOIN issues i ON r.id = i.resolution;
```

	A-Z resolution
1	
2	4f6b29cc3004e9003e453da7

# Итоги по качеству данных

## Выявленные проблемы:

1. Пропущенные значения: 0.006% без даты создания, 5.32% без даты решения

2. Временные аномалии: 2 тикета с некорректной датой (epoch), 3 тикета с обработкой > 1 года

3. Проблемы связности: 1 тикет на несуществующую резолюцию, 171 неиспользуемая резолюция

## Рекомендации:

- Исключить аномальные записи из анализа
- Очистить справочник резолюций
- Установить валидацию данных на уровне системы

# Создание и проверка чистого датасета (VIEW)

Создан VIEW "issues\_clean" с преобразованиями:

- Конвертация Unix timestamp в читаемый формат даты
- Расчёт времени обработки в днях
- Фильтрация аномальных записей
- Единая точка доступа для анализа

AZ key	created_dt	resolved_dt	created_raw	resolved_raw	AZ resolution	AZ category	days_to_resolve
1	2023-06-01 04:19:41.056 +0300	2023-06-01 17:39:13.408 +0300	1 685 582 400 000	1 685 630 400 000	4f6b29cc3004e9443e453da7	Remote	0.5552355556
2	2023-06-01 04:19:41.056 +0300	2023-06-01 04:30:36.416 +0300	1 685 582 400 000	1 685 583 000 000	4f6b29c3004e9443e453da7	Remote	0.0075851852
3	2023-06-01 04:19:41.056 +0300	2023-06-01 04:30:36.416 +0300	1 685 582 400 000	1 685 583 000 000	4f6b29c3004e9443e453da7	Remote	0.0075851852
4	2023-06-01 04:19:41.056 +0300	2023-06-01 04:30:36.416 +0300	1 685 582 400 000	1 685 583 000 000	4f6b29c3004e9443e453da7	Remote	0.0075851852
5	2023-06-01 04:59:00.352 +0300	2023-06-01 18:55:40.928 +0300	1 685 584 700 000	1 685 635 000 000	4f6b29c3004e9443e453da7	Remote	0.5810251852
6	2023-06-01 04:59:00.352 +0300	2023-06-01 18:55:40.928 +0300	1 685 584 700 000	1 685 635 000 000	4f6b29c3004e9443e453da7	Remote	0.5810251852
7	2023-06-01 04:59:00.352 +0300	2023-06-01 18:55:40.928 +0300	1 685 584 700 000	1 685 635 000 000	4f6b29c3004e9443e453da7	Remote	0.5810251852
8	2023-06-01 04:59:00.352 +0300	2023-06-01 09:23:20.064 +0300	1 685 584 900 000	1 685 600 600 000	4f6b29c3004e9443e453da7	Remote	0.1820444444
9	2023-06-01 05:20:51.072 +0300	2023-06-01 05:29:35.360 +0300	1 685 586 000 000	1 685 586 600 000	4f6b29c3004e9443e453da7	Remote	0.0060681481
10	2023-06-01 05:20:51.072 +0300	2023-06-01 05:29:35.360 +0300	1 685 586 000 000	1 685 586 600 000	4f6b29cc3004e9443e453da7	Remote	0.0060681481

	total_tickets	resolved_tickets	unresolved_tickets
1	51 950	49 184	2 766

```
CREATE VIEW issues_clean AS
SELECT
    key,
    TO_TIMESTAMP(created / 1000) as created_dt,
    TO_TIMESTAMP(resolved / 1000) as resolved_dt,
    created as created_raw,
    resolved as resolved_raw,
    resolution,
    category,
    CASE
        WHEN resolved IS NOT NULL
        THEN (resolved - created) / 1000 / 60 / 60 / 24
        ELSE NULL
    END as days_to_resolve
FROM issues
WHERE created > 100000000000; -- Исключаем ероч
аномалии
```

## Проверка созданного VIEW:

- Всего тикетов: 51,950 (исключены аномалии)
- Решённых тикетов: 49,184 (94.67%)
- Нерешённых тикетов: 2,766 (5.33%)

# Базовая описательная статистика

## Общая статистика по тикетам:

- Всего тикетов: 51,950
- Процент решённых: 94.67%
- Период анализа: от первого до последнего тикета
- Уникальные категории: 4

```
SELECT
    COUNT(*) as total_tickets,
    COUNT(CASE WHEN resolved_dt IS NOT NULL THEN 1 END) as resolved,
    COUNT(CASE WHEN resolved_dt IS NULL THEN 1 END) as unresolved,
    ROUND(COUNT(CASE WHEN resolved_dt IS NOT NULL THEN 1 END) * 100.0 / COUNT(*), 2) as resolution_rate,
    MIN(created_dt) as first_ticket,
    MAX(created_dt) as last_ticket,
    COUNT(DISTINCT category) as unique_categories
FROM issues_clean;
```

	total_tickets	resolved	unresolved	resolution_rate	first_ticket	last_ticket	unique_categories
1	51 950	49 184	2 766	94,68	2023-06-01 04:19:41.056 +0300	2023-07-01 02:49:15.904 +0300	4

```
SELECT
    category,
    COUNT(*) as ticket_count,
    ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM issues_clean), 2) as percentage
FROM issues_clean GROUP BY category ORDER BY ticket_count DESC;
```

	A-Z category	ticket_count	percentage
1	Local	26 366	50,75
2	Remote	18 167	34,97
3	New User	6 144	11,83
4	Software License	1 273	2,45

## Распределение по категориям:

- Local: 26,366 тикетов (50.75%)
- Remote: 18,167 тикетов (34.97%)
- New User: 6,144 тикетов (11.83%)
- Software License: 1,273 тикетов (2.45%)

# Интерпретация распределения по категориям

Анализ показывает неравномерность распределения:

- Local + Remote: 85.72% (преобладание технических проблем)
- New User: 11.83% (недостаточная документация?)
- Software License: 2.45% (специфичная категория)

Выводы для планирования ресурсов:

- Приоритет: Local и Remote категории
- Необходимость улучшить onboarding для New User
- Стандартизировать процесс для Software License

# Доля тикетов решенных в тот же месяц

Общий показатель решения в месяц создания:

- Решено в тот же месяц: 49,184 из 49,963 тикетов
- Процент: 95.48%
- Это высокий показатель эффективности поддержки

```
SELECT
    COUNT(CASE
        WHEN DATE_TRUNC('month', created_dt) =
DATE_TRUNC('month', resolved_dt)
        THEN 1
    END) as same_month_resolved,
    COUNT(*) as total_resolved,
    ROUND(
        COUNT(CASE
            WHEN DATE_TRUNC('month', created_dt) =
DATE_TRUNC('month', resolved_dt)
            THEN 1
        END) * 100.0 / COUNT(*),
        2
    ) as percentage_same_month
FROM issues_clean
WHERE resolved_dt IS NOT NULL;
```

	123 same_month_resolved	123 total_resolved	123 percentage_same_month
1	46 963	49 184	95.48

# Распределение по категориям

- Software License: ~100% (практически все в месяц)
- Remote: ~95.35%
- New User: ~96.29%
- Local: ~95.16%

	AZ category	123 total_resolved	123 same_month_count	123 same_month_percentage
1	Software License	1 273	1 273	100
2	New User	5 821	5 605	96,29
3	Remote	17 122	16 326	95,35
4	Local	24 968	23 759	95,16

```
SELECT
    category,
    COUNT(*) as total_resolved,
    COUNT(CASE
        WHEN DATE_TRUNC('month', created_dt) =
DATE_TRUNC('month', resolved_dt)
        THEN 1
    END) as same_month_count,
    ROUND(
        COUNT(CASE
            WHEN DATE_TRUNC('month', created_dt) =
DATE_TRUNC('month', resolved_dt)
            THEN 1
        END) * 100.0 / COUNT(*),
        2
    ) as same_month_percentage
FROM issues_clean
WHERE resolved_dt IS NOT NULL
GROUP BY category
ORDER BY same_month_percentage DESC;
```

# Анализ скорости обработки по резолюциям

## Распределение решённых тикетов по типам резолюций:

- Fixed: 46,808 тикетов (95.17%)
- Duplicate: 294 тикетов (0.6%)
- Wont Fix: 1251 тикетов (2.54%)
- resolvedByUser: 791 тикетов (1.61%)
- escalated: 39 тикетов (0.08%)

```
SELECT
    r.key as resolution_name,
    COUNT(*) as ticket_count,
    ROUND((COUNT(*) * 100.0 / (SELECT COUNT(*) FROM issues_clean WHERE resolved_dt IS NOT NULL))::numeric, 2)
        as percentage,
    ROUND(AVG(i.days_to_resolve)::numeric, 2) as
    avg_days_to_resolve,
    ROUND(MIN(i.days_to_resolve)::numeric, 2) as min_days,
    ROUND(MAX(i.days_to_resolve)::numeric, 2) as max_days
FROM issues_clean i
INNER JOIN resolutions r ON i.resolution = r.id
WHERE i.resolved_dt IS NOT NULL
    AND i.days_to_resolve <= 365
GROUP BY r.key
ORDER BY ticket_count DESC;
```

```
SELECT
    r.key as resolution_name,
    COUNT(*) as ticket_count,
    ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM issues_clean
        WHERE resolved_dt IS NOT NULL), 2) as percentage
FROM issues_clean i
INNER JOIN resolutions r ON i.resolution = r.id
WHERE i.resolved_dt IS NOT NULL
GROUP BY r.key
ORDER BY ticket_count DESC;
```

	AZ resolution_name	ticket_count	percentage
1	fixed	46 808	95,17
2	won'tFix	1 251	2,54
3	resolvedByUser	791	1,61
4	duplicate	294	0,6
5	escalated	39	0,08

## Время обработки по типам резолюций:

- Средние/минимальные/максимальные дни для каждой резолюции

	AZ resolution_name	ticket_count	percentage	avg_days_to_resolve	min_days	max_days
1	fixed	46 807	95,17	1,61	0	36,02
2	won'tFix	1 251	2,54	1,69	0	33,98
3	resolvedByUser	791	1,61	2,22	0	25,8
4	duplicate	294	0,6	1,5	0	32,12
5	escalated	39	0,08	1,81	0	27,17

# Связь категорий с типами резолюций

## Перекрёстный анализ категорий и резолюций:

- Какие резолюции чаще встречаются в каждой категории
- Процентное распределение

SELECT

```
r.key as resolution_name,  
i.category,  
COUNT(*) as ticket_count,  
ROUND(COUNT(*) * 100.0 / SUM(COUNT(*)) OVER (PARTITION  
BY r.key), 2) as pct_within_resolution  
FROM issues_clean i  
INNER JOIN resolutions r ON i.resolution = r.id  
WHERE i.resolved_dt IS NOT NULL  
GROUP BY r.key, i.category  
ORDER BY r.key, ticket_count DESC;
```

	A-Z resolution_name	A-Z category	123 ticket_count	123 pct_within_resolution
1	duplicate	Local	143	48,64
2	duplicate	Remote	112	38,1
3	duplicate	New User	37	12,59
4	duplicate	Software License	2	0,68
5	escalated	Local	20	51,28
6	escalated	Remote	17	43,59
7	escalated	New User	2	5,13
8	fixed	Local	23 763	50,77
9	fixed	Remote	16 223	34,66
10	fixed	New User	5 554	11,87
11	fixed	Software License	1 268	2,71
12	resolvedByUser	Local	366	46,27
13	resolvedByUser	Remote	334	42,23
14	resolvedByUser	New User	91	11,5
15	won'tFix	Local	675	53,96
16	won'tFix	Remote	436	34,85
17	won'tFix	New User	137	10,95
	won'tFix	Software License	3	0,2

# ТОП-3 резолюции по каждой категории

## Ранжирование резолюций по категориям:

- Для каждой категории показаны три наиболее частые резолюции
- Процент от всех решений в категории

	A-Z category	A-Z resolution_name	123 ticket_count	123 percentage
1	Local	fixed	23 763	95,18
2	Local	won'tFix	675	2,7
3	Local	resolvedByUser	366	1,47
4	New User	fixed	5 554	95,41
5	New User	won'tFix	137	2,35
6	New User	resolvedByUser	91	1,56
7	Remote	fixed	16 223	94,75
8	Remote	won'tFix	436	2,55
9	Remote	resolvedByUser	334	1,95
10	Software License	fixed	1 268	99,61
11	Software License	won'tFix	3	0,24
12	Software License	duplicate	2	0,16

```
WITH ranked AS (
  SELECT
    i.category,
    r.key as resolution_name,
    COUNT(*) as ticket_count,
    ROUND((COUNT(*) * 100.0 / SUM(COUNT(*))) OVER
      (PARTITION BY i.category))::numeric, 2) as percentage,
    ROW_NUMBER() OVER (PARTITION BY i.category ORDER BY
      COUNT(*) DESC) as rank
  FROM issues_clean i
  INNER JOIN resolutions r ON i.resolution = r.id
  WHERE i.resolved_dt IS NOT NULL
  GROUP BY i.category, r.key
)
SELECT
  category,
  resolution_name,
  ticket_count,
  percentage
FROM ranked
WHERE rank <= 3
ORDER BY category, rank;
```

# Процент решения в тот же месяц по категориям

## Детальный анализ скорости решения:

- Software License: 100%
- Local: 95.16%
- Remote: 95.35%
- New User: 96.29%

```
SELECT
    category,
    COUNT(*) as total_resolved,
    COUNT(CASE
        WHEN DATE_TRUNC('month', created_dt) =
DATE_TRUNC('month', resolved_dt)
        THEN 1
    END) as same_month_count,
    ROUND((COUNT(CASE
        WHEN DATE_TRUNC('month', created_dt) =
DATE_TRUNC('month', resolved_dt)
        THEN 1
    END) * 100.0 / COUNT(*))::numeric, 2) as
same_month_percentage
FROM issues_clean
WHERE resolved_dt IS NOT NULL
GROUP BY category
ORDER BY same_month_percentage DESC;
```

	AZ category	123 total_resolved	123 same_month_count	123 same_month_percentage
1	Software License	1 273	1 273	100
2	New User	5 821	5 605	96,29
3	Remote	17 122	16 326	95,35
4	Local	24 968	23 759	95,16

# Распределение по дням недели

## Анализ паттернов обращений по дням:

- Ежедневное распределение тикетов
- Выявление пиков нагрузки
- Процентное распределение по категориям

```
SELECT
    EXTRACT(DOW FROM created_dt)::int as day_number,
    TO_CHAR(created_dt, 'Day') as day_name,
    category,
    COUNT(*) as ticket_count,
    ROUND((COUNT(*) * 100.0 / SUM(COUNT(*))) OVER
        (PARTITION BY category))::numeric, 2) as pct_in_category
FROM issues_clean
GROUP BY day_number, day_name, category
ORDER BY day_number, category;
```

day_number	day_name	category	ticket_count	pct_in_category
0	Sunday	Local	1 141	4,33
0	Sunday	Remote	645	3,55
0	Sunday	Software License	26	2,04
1	Monday	Local	5 956	22,59
1	Monday	Remote	2 759	15,19
1	Monday	Software License	210	1,65
2	Tuesday	Local	1	0
2	Tuesday	New User	6 144	100
2	Tuesday	Remote	3 239	17,83
3	Wednesday	Local	6 280	23,82
3	Wednesday	Remote	3 414	18,79
3	Wednesday	Software License	365	2,87
4	Thursday	Local	6 557	24,87
4	Thursday	Remote	3 899	21,46
4	Thursday	Software License	376	2,95
5	Friday	Local	5 709	21,65
5	Friday	Remote	3 477	19,14
5	Friday	Software License	280	2,2
6	Saturday	Local	722	2,74
6	Saturday	Remote	734	4,04
6	Saturday	Software License	16	1,21

# Комплексная сравнительная таблица по категориям

## Сводная статистика по всем метрикам:

- Всего тиков
- Процент от всех
- Количество решённых
- Процент решения
- Среднее время обработки
- Процент решения в месяц

A2 category	123 total_tickets	123 pct_of_all	123 resolved	123 resolution_rate	123 avg_days_to_resolve	123 same_month_pct
1 Local	26 366	50,75	24 968	94,7	1,64	95,16
2 Remote	18 167	34,97	17 122	94,25	1,73	95,35
3 New User	6 144	11,83	5 821	94,74	1,6	96,29
4 Software License	1 273	2,45	1 273	100	0	100

```
SELECT
    category,
    COUNT(*) as total_tickets,
    ROUND((COUNT(*) * 100.0 / (SELECT COUNT(*) FROM
issues_clean))::numeric, 2) as pct_of_all,
    COUNT(CASE WHEN resolved_dt IS NOT NULL THEN 1 END) as
resolved,
    ROUND((COUNT(CASE WHEN resolved_dt IS NOT NULL THEN 1
END) * 100.0 / COUNT(*))::numeric, 2) as resolution_rate,
    ROUND(AVG(CASE
        WHEN resolved_dt IS NOT NULL AND days_to_resolve <=
365
        THEN days_to_resolve
    END)::numeric, 2) as avg_days_to_resolve,
    ROUND((COUNT(CASE
        WHEN resolved_dt IS NOT NULL
        AND DATE_TRUNC('month', created_dt) =
DATE_TRUNC('month', resolved_dt)
        THEN 1
    END) * 100.0 / COUNT(CASE WHEN resolved_dt IS NOT NULL
THEN 1 END))::numeric, 2) as same_month_pct
FROM issues_clean
GROUP BY category
ORDER BY total_tickets DESC;
```

# Ключевые закономерности распределения категорий

## Объёмные характеристики:

-  Local и Remote доминируют (85.72%)
-  Стабильное распределение во времени
-  Нет резких всплесков нагрузки

## Скорость обработки:

-  Software License: самая быстрая
-  New User: требует индивидуального подхода
-  Local/Remote: вариативность по сложности

# Паттерны решения и временные тренды

## Паттерны закрытия:

-  Все категории закрываются преимущественно резолюцией "Fixed"
-  Software License: наиболее однородное распределение

## Временные паттерны:

-  Нет корреляции между днём недели и количеством обращений
-  Равномерное распределение по месяцам
-  Стабильная нагрузка на поддержку

# Сильные стороны системы и области для улучшения

## Положительные результаты:

- 1 Высокий показатель решения тикетов
- 2 Большая часть тикетов решаются в месяц создания
- 3 Преобладание резолюции "Fixed" говорит об эффективной работе поддержки
- 4 Стабильная нагрузка без резких всплесков

## Выявленные проблемы:

- 🚫 2,766 незакрытых тикетов (5.33%)
- ❗ Необходима процедура работы с долгоживущими тикетами
- ❗❗ Справочник резолюций нуждается в оптимизации (171 неиспользуемая)