

Test task JetBrains

Questions:

1. Как дела у CLion в плане выручки и какая доля на рынке C++ разработчиков;
2. Какую выручку ожидать в следующем периоде;
3. Какие тренды среди C++ разработчиков наблюдаются сейчас;
4. Что пишут пользователи CLion на StackOverflow, с какими трудностями они сталкиваются при использовании CLion.

Для исследования этих вопросов в вашем распоряжении есть 3 источника данных:

- Данные продаж (sales.csv);
- Опросные данные Developer Ecosystem 2019, 2020;
- Текстовые данные вопросов / ответов на StackOverflow, которые доступны публично

Analysis Results

This report provides an analysis of CLion's performance, trends among C++ developers, and insights from discussions about CLion on Stack Overflow. The work is structured into three main sections.

The first section (CLion Revenue Overview) focuses on analyzing CLion's financial performance, market share and forecasting of CLion's future revenue, based on historical trends and predictive modeling. The second section examines the technological trends within the C++ developer community. The final section explores discussions related to CLion on Stack Overflow.

1. CLion Revenue overview

Sequence of analysis:

1. Description of data
2. General revenue characteristics
3. CLion's market share among C++ developers
4. CLion's Revenue Forecast

General information about the data:

- 1) period: from 2019-01-01 to 2020-12-31 (2 years),
- 2) analyzed 123,056 purchases by 48,059 unique clients (including 41,301 CLion product purchases by 29,749 unique clients),
- 3) the sample includes 3 product types: CL, WS, II,
- 4) 2 license types: New, Renew,
- 5) 2 methods of purchase price calculation: initial license cost, final cost (with discount).

1.1. General revenue characteristics

CLion revenue in 2020 compared to 2019 increased by \$26.2 million (+32.8%). Since the license price is stable, revenue growth is driven by an increase in the number of unique clients (the average number of licenses purchased by a client remained unchanged).

The median monthly revenue (final price) for the entire period **is \$7.75 million** (calculated at the stock price - \$15.4 million, therefore monthly discount amount was \$7.66 million). On average, month-to-month revenue grows by 5%; however, in 2020, **the growth rate slowed to 4.35% per month.**

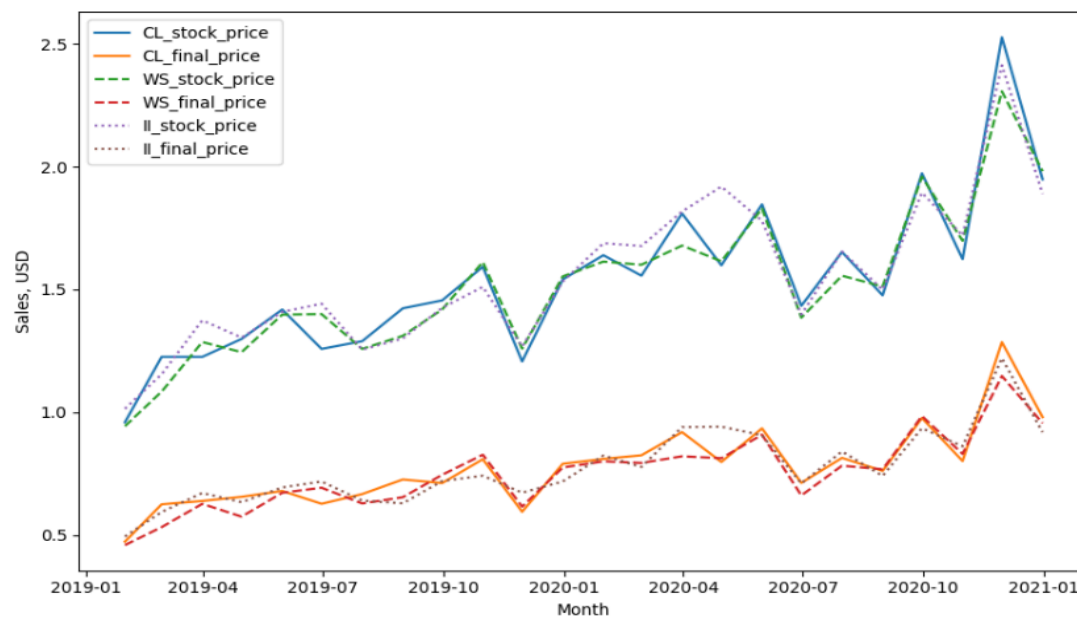
To assess revenue stability across months, a variation coefficient of 24.6% was calculated. This indicates that the **data is generally homogenous**, but the monthly revenue deviation from the average revenue for the period is significant. A more detailed comparison is in *Table 1*.

Table 1 - CLion sales results for 2019, 2020

Year	Annual Revenue (stock price, USD)	Annual Revenue (final price, USD)	Avg Monthly Revenue (final price, USD)	Avg. Growth Rate (%)	Unique Clients	Revenue per Client, USD	Licenses per Client	Share among other two products (%)
2019	158 935 450	79 913 226	6 659 435	+ 5,43	12 888	6201	70	33,7
2020	210 880 406	106 123 518	8 843 626	+ 4,35	17 067	6218	70	33,7
Change (units)	+ 51944956 \$	+ 26 210 291 \$	+ 2 184 191 \$	- 1,07 п.п.	+ 4179	+ 17	0	0
Growth (%)	+ 32.7	+ 32.8	+ 32.8		+ 32.4	+ 0,27	0	0

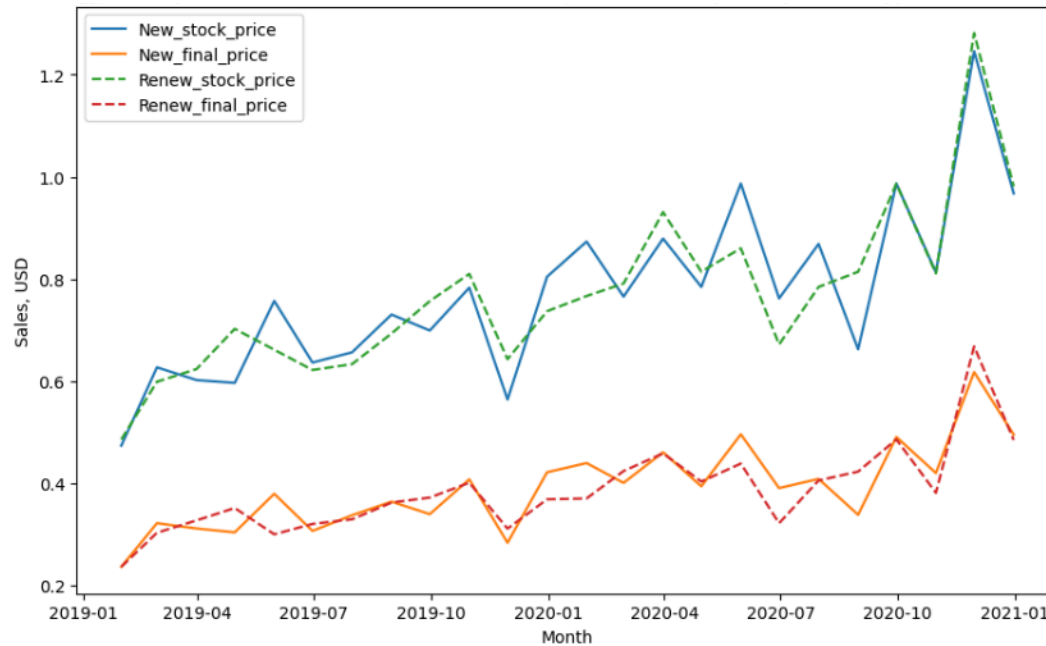
A closer look at revenue from CLion licenses shows that the revenue volume and its dynamics match those of the other two products (WS and II). *Chart 1* illustrates that all three products show a steady growth in revenue throughout the period. The gap between stock and final price remains relatively consistent across all products, suggesting a similar discounting strategy. The sharp rise in late 2020 may be attributed to seasonal factors and increased demand. If the spike in November 2022 is excluded, the average monthly growth in 2020 would be only 1.4%.

Chart 1 - Monthly revenue dynamics by product and pricing method, 2019-2020.



For CLion New and Renew licenses, the sales volume and dynamics align, demonstrating stability in both product segments, *Chart 2*.

Chart 2 - CLion revenue dynamics by month, by license and pricing method, 2019-2020.



1.2. CLion's market share among C++ developers

Based on the JetBrains survey data, CLion holds the largest share of the C++ market among IDEs, demonstrating steady growth. In 2023, its share increased by 1.5 percentage points, rising from 29.9% in 2022 to 31.4%. This highlights CLion's strong position and growing popularity among C++ developers.

It is important to note that the survey was conducted by JetBrains, which may introduce a sampling bias, as the respondents are likely to include a higher proportion of JetBrains customers. To take this bias into account, we can calculate the confidence interval for the share.

We do not know the size of the population, but we can say that the population is much larger than the sample, so we will use the formula for calculating the average error of the share:

$$\mu_p = \sqrt{\frac{w(1-w)}{n}}$$

We will calculate the marginal error at the 95% confidence level. Thus, the marginal error of the sample for the share of CLion in both 2022 and 2023 was 1.8 p.p.

As a result, with 95% probability, the share of C++ developers who choose CLion was:

in 2022: from 28.1 to 31.8,

in 2023: from 29.6 to 33.3%.

Further insights into C++ developers' IDE preferences are provided in *Table 2*, offering a more detailed comparison of market shares across different tools.

Table 2 - IDE Market Share in 2022 and 2023

IDE	2022			2023			Share change	margin_error	
	lower limit	share	upper limit	lower limit	share	upperlimit		2022	2023
CLion	28,1	29,9	31,8	29,6	31,4	33,3	1,5	1,8	1,8
Visual Studio Code	21,8	23,5	25,2	23,6	25,3	27	1,8	1,7	1,7
Visual Studio	25,2	26,9	28,7	22,2	23,9	25,6	-3	1,8	1,7
Other	5	6	6,9	5,7	6,6	7,6	0,7	0,9	1
Vi/Vim	3,6	4,4	5,2	3,3	4	4,8	-0,3	0,8	0,8
QtCreator	2,3	3	3,7	2	2,6	3,2	-0,4	0,7	0,6
Xcode	1	1,4	1,9	1	1,4	1,9	0	0,5	0,5

Eclipse CDT	0,7	1,1	1,5	0,7	1,1	1,5	0	0,4	0,4
Emacs	0,9	1,4	1,9	0,7	1,1	1,5	-0,3	0,5	0,4
Android Studio	0,7	1,1	1,6	0,6	1	1,4	-0,2	0,4	0,4
Sublime	0,8	1,3	1,7	0,5	0,9	1,2	-0,4	0,4	0,4

1.3. Revenue Forecast

As previously noted, license sales revenue shows an upward trend, but monthly dynamics are quite unstable (that is why seasonality and fluctuations must be taken into account). Since sales trends and volumes for different license types are similar, the model will be built using combined data for both licenses.

Forecasting Approach:

1. Monthly forecast based on trend and seasonality components (classical time series forecast).
2. Daily forecast accounting for deeper time correlations (Prophet model).

Description of Classical Forecast Construction:

The classical time series forecast was created by modeling a linear trend using Linear Regression (sklearn library). The independent variable is time. Two different models were built: for the total and final cost. Models quality metrics indicate significance at a 0.05 level, with an R^2 value:

- for revenue based on stock price: 0.639 (the model explains 63.9% of the target variance),
- for revenue based on final price: 0.605 (the model explains 60.5% of the target variance).

The seasonal (monthly) component was identified to measure how much revenue deviates from the trend. Forecast values based on the linear model were adjusted for the seasonal component, improving the R^2 value to 0.68 for stock price, to 0.66 for final price.

Table - Quality metrics of a seasonally adjusted regression model for stock price

Metric	Value	Interpretation
ME	-71632	The average monthly deviation is -71 K, which means that on average, the actual revenue is lower than the predicted value
MAE	818788	On average, the predicted value deviates by +- 819 K per month
RMSE	1237180	Standard deviation, on average, the predicted value deviates by 1.2 M per month
MPE	-1,181	The average monthly deviation is -1.2% from the actual value, which means that on average, the actual revenue is lower than the predicted

Metric	Value	Interpretation
		value
MAPE	6,102	On average, the predicted value deviates by +- 6% from the actual
R2	68,2	The forecast curve describes 68% of the revenue variation

Table - Quality metrics of a seasonally adjusted regression model for final price

Metric	Value	Interpretation
ME	-36664	The average deviation is -37 K, which means that on average, the actual revenue is lower than the predicted value
MAE	447102	On average, the predicted value deviates by +- 447 K
RMSE	651177	Standard deviation, on average, the predicted value deviates by 651 K
MPE	-1,299	The average deviation is -1.3% from the actual value, which means that on average, the actual revenue is lower than the predicted value
MAPE	6,616	On average, the predicted value deviates by +- 6.6% from the actual
R2	66,6	The forecast curve describes 66,6% of the revenue variation

Forecast for the Next Period:

A forecast with a horizon equal to one-third of the training dataset length is considered appropriate, so the focus will be on the first half of 2021. *Table 3* shows the forecast for 2 types of prices. The trend remains positive, with average monthly revenue growing by 20% compared to the first half of 2020.

Table 3 - Forecast Revenue for CLion in H1 2021 (Classical Forecast)

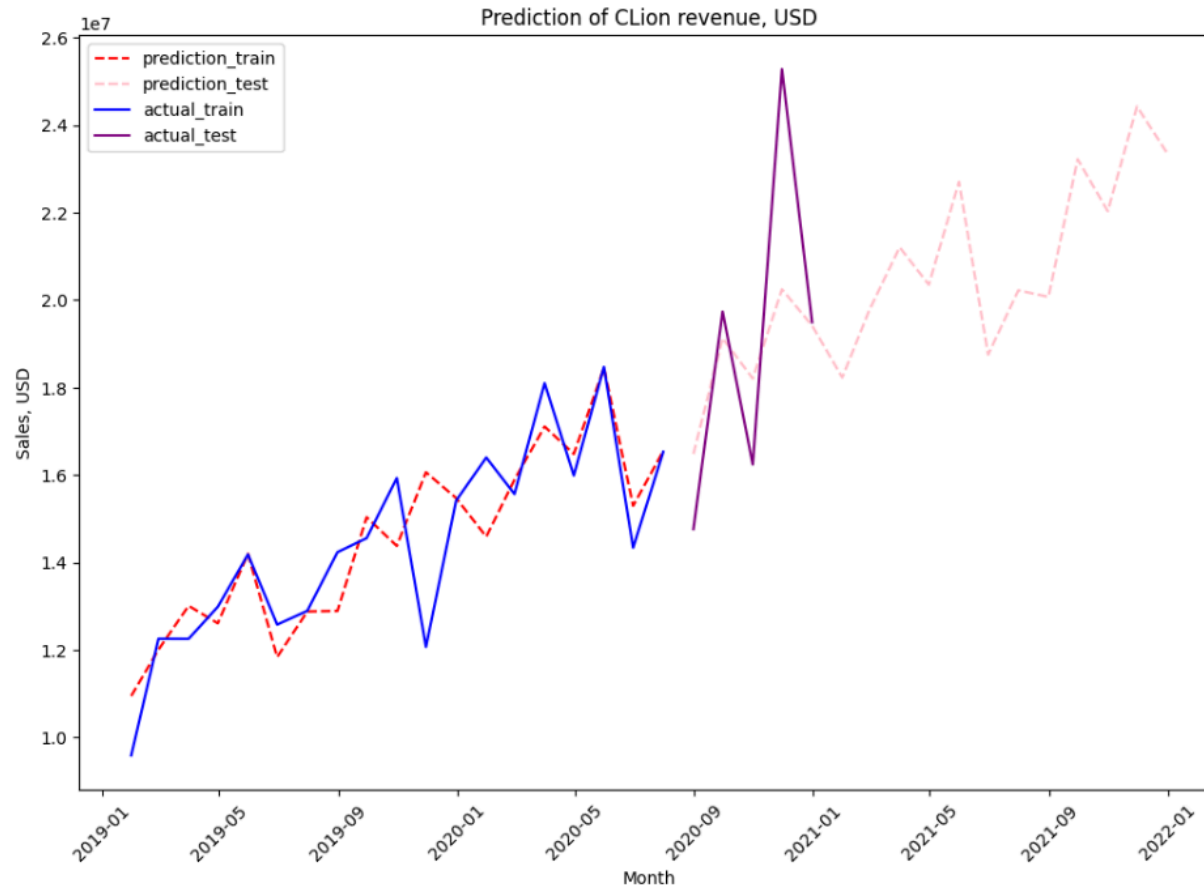
Period	Stock price		Final Price	
	Total Revenue, USD	YoY Change (%)	Total Revenue, USD	YoY Change (%)
H1 2021	119 213 964	98 850 529 (+ 20,6)	60 300 704	+10 344 965 (+20,7%)
Average per month	19 980 515	16 475 088 (+ 21,8)	10 050 117	+1 724 160 (+20,7%)

The dynamics of the real revenue and predicted one can be seen in detail in the *Chart 3*.

Predictions (pink dashed line) align well with actual data (purple solid line), although some discrepancies occur in areas with sharp spikes or dips. A significant spike in revenue is observed around late 2020, followed by a steady growth trend.

There is an overall upward trend in revenue across the entire period, indicating growth in CLion sales. Seasonal variations are visible. The model performs well overall, though it slightly struggles with abrupt spikes in the forecasting period. This chart indicates that the prediction model is robust, capturing the general revenue patterns while leaving some room for improvement in predicting extreme changes.

Chart 3 - Forecast of revenue dynamics (at stock cost) by month (classic forecast)



Prophet Model Forecast

For more accurate revenue forecasts, another time series forecast model was used. Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. The forecast was calculated using daily aggregated revenue.

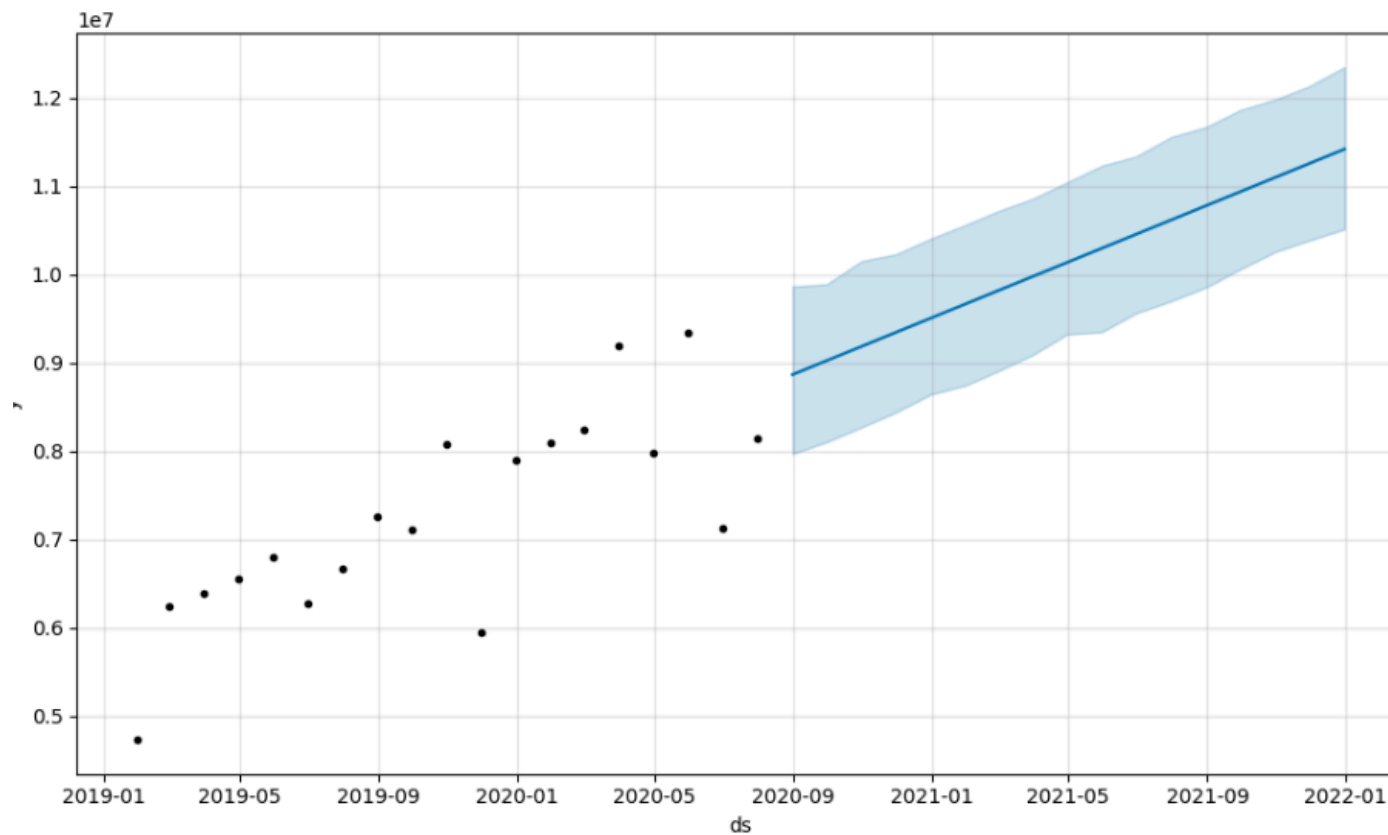
The advantage of this model is that it draws the lower and upper boundaries of the possible forecast. *Table 4* shows the main revenue volumes for the coming half of the year. The minimum percentage increase in monthly revenue will be 10%, the maximum - 32%

Table 4 - Forecast Revenue for CLion in H1 2021 (Prophet Model)

Period	Stock price		Final Price	
	Lower Bound (USD/%)	Upper Bound (USD/%)	Lower Bound (USD/%)	Upper Bound (USD/%)
H1 2021	110138792 (+ 11,4 %)	130505849 (+ 32,0%)	54967894 (+ 10,0%)	65743569 (+ 31,6%)
Average per month	19351267 (+ 17,5 %)	22703073 (+ 37,8%)	9644625 (+ 15,8%)	11441196 (+ 37,4%)

Chart 4 shows the model prediction by month. The steady upward trend and the upper and lower forecast boundaries are also displayed.

Chart 4 - Forecast of revenue dynamics (stock price) by month (according to the Prophet model)



We can expect that in 2021 revenue will have a positive trend. Average monthly revenue at the final price will at least grow by 15.8% and will be around \$9.6 mln, at most it will grow by 37.4% and will be around \$11.4 mln.

As part of improving the revenue prediction, more could be done:

- instead of a linear trend, a logistic curve can be used, which reflects a slowing rate of revenue growth.
- if additional data is available, the accuracy of the model can be increased, for example, information about license terms and payment sequences can be used.

2. Trends Among C++ Developers

Sequence of analysis:

1. Description of data, determination of significance level, definition of C++ developers
2. Description of a sample of C++ developers
3. Main trends of C++ developers

4. Key Changes Compared to 2022

General information about the data

Data from a survey conducted by JetBrains in 2022 and 2023. Each dataset contains about 26,000 responses.

2. 1 Determination of significance level, definition of C++ developers

To understand that our sample is sufficient for analysis, we need to calculate what the sample should be if the general population is extremely large. 2 approximate samples with assumptions were calculated.

1. With the highest accuracy (confidence level of 0.99) and the lowest selection bias of 1%, the sample size should be 13572 observations.
2. With weaker accuracy (confidence level of 0.95) and maximum acceptable selection bias of 4%, the sample size should be 600 observations.

Accordingly, a sample of 26K observations can be considered representative. At the same time, questions with fewer than 600 answers should be excluded from the analysis.

To analyze C++ developers, we need to select them from the dataset. The questionnaire has 2 questions about the programming languages: “What are your primary programming languages?”, “What programming languages have you used in the last 12 months?”. In the following analysis, we will consider the answers of those respondents who used C++ in the last 12 months (5809 respondents). However, it is interesting to note that in 2023:

- 10,3% of respondents had C++ as their primary programming language,
- 22,0% of respondents had used C++ in the last 12 months (only 1.3% of respondents had written exclusively in C++ in the last 12 months, the remaining 98,7% combined it with other programming languages),
- 3,6% of respondents were planning to learn C++.

2.2 Respondents' overview (demographics and background of C++ developers)

Age and Gender: The majority of C++ developers are aged 21-29 (47.7%) and predominantly male (91.8%).

Experience: Nearly half (44.3%) have been programming for 1-5 years.

Employment and Education:

- Most are fully employed (48.9%) or students (37.5%).
- 37.6% hold a bachelor's degree.

Global Distribution: The largest groups of respondents are from:

- China (11.3%),
- The US (11.2%),
- Germany (7.7%),
- India (6.3%).

Workplace: A significant proportion (25.9%) work for mid-sized companies (51-500 employees).

Workplace Language: English is the dominant language, used by 80.2% of respondents.

Primary Development Focus: Developers are involved in:

- Website development (44%),
- Utilities (small apps for specific tasks) (42.3%),
- System software (30.1%),
- Database/Data storage solutions (29.5%).

2.3 Trends and Popular Technologies Among C++ Developers***Key Changes in 2023 Compared to 2022***

The Chi-square homogeneity test was used to determine significant changes in the structure of responses. The proportions of responses to each question were submitted for calculation if the p-value for chi-square is less than 0.05, then the answers are significantly different.

C++ Standards:

- 46.9% use the C++17 standard, followed by 34.0% using C++20.
- Earlier standards like C++11 and C++14 also remain popular.
- Among C++17 users, 37.1% plan to transition to C++20. However, 46.3% of C++98 users do not plan to adopt newer standards.

Future Adoption:

- In the next 12 months, 52.6% plan to use C++20 concepts (e.g., template constraints, `requires`).
- 46.8% plan to use C++20 modules (e.g., `export`, `import`).
- C++/CLI: Respondents are evenly split on whether they use C++/CLI.

Main Communication Tools:

- Instant messaging and video calling platforms like Microsoft Teams, Slack, and Skype are used by 86.3%.
- Email services like Gmail and Microsoft Mail Server are utilized by 84.5%.

Agile Frameworks:

- Scrum is the most popular framework, used by 47.3%.
- Kanban is also widely adopted (26.4%).

AI tools:

- Text generation tools (like ChatGPT, information retrieval, and question answering) are used by 76.2%.
- 64.9% use code generation tools (like GitHub Copilot).

2.4 Key Changes in 2023 Compared to 2022

The Chi-square homogeneity test was used to determine significant changes in the structure of responses. The proportions of responses to each question were submitted for calculation if p-value for chi-square is less than 0.05, then the answers are significantly different in 2023 compared to 2022.

The attitude of C++ developers towards security has changed significantly. The answer to the question "Do you follow a secure software development life cycle?" began to be "Yes" - 54.9% in 2023, while in 2022 the share of positive answers was only 29%.

DS MPP tools preferences have changed. Lost share: Azure Data Explorer (-1 p.p.), ClickHouse (-0.8 p.p.); increased in share: Greenplum (+2.4 p.p.), Spanner (+1.1 p.p)

In 2023, fewer respondents answered "prefer not to answer" to the question about salary satisfaction, so the results are more reliable. As a result, 29.9% of C++ developers answered that they were rather satisfied with their salary, and 24.9% that they were mostly satisfied. At the same time, there remains a fairly large portion of respondents (18.9%) who answered that they were rather dissatisfied with their salary.

In conclusion, the steady transition to newer C++ standards and the rapid integration of AI tools reflect the adaptability of the C++ developer community to emerging trends and technologies. The significant increase in secure development practices demonstrates the growing importance of security as a core aspect of modern software engineering.

3. Overview of CLion on StackOverflow

Sequence of analysis:

1. For understanding: analyze the dynamics of posts
2. Identify the most popular posts, the most recent posts
3. Determine the main topics and problems that are posted on Stack OverFlow

General information about the data

Data contains StackOverflow posts about CLion. The search for relevant posts was performed by tag “CLion”, as a result, information about 2,339 posts was downloaded. For further analysis of problematic topics, only posts in English were taken. Period: 2014.09.12 - 2023.01.26

3.1 Posts dynamics (*Chart 5*)

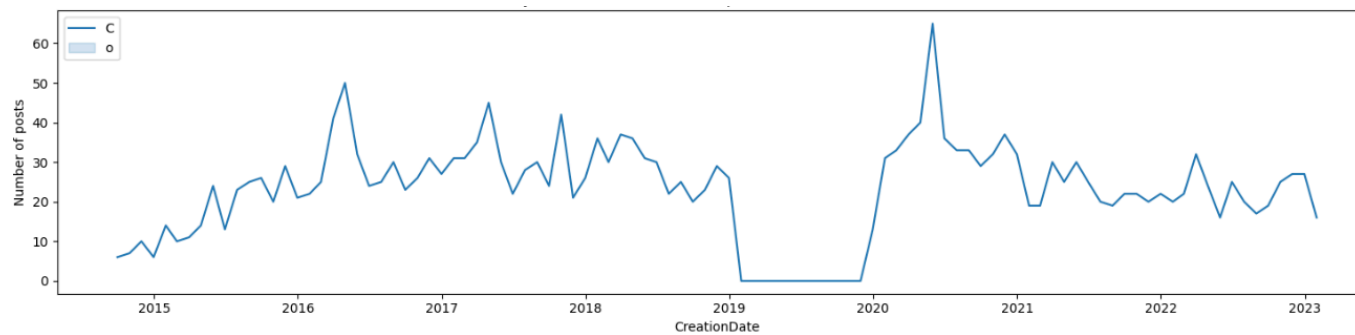
From 2015 to early 2020, the number of posts about CLion generally increased, indicating growing interest or usage of the tool.

The number of posts peaked in May 2020, followed by fluctuations and a slight decline from 2021 onwards. In 2019, there is no information on posts in the database, I assume that either the tags were changed, or it is not pulled from the database.

After 2021 the number of posts per month is at the level of 15-20. The decline after 2021 might suggest:

- fewer new issues or challenges faced by users as CLion matured;
- a more stable user base with fewer discussions or queries;
- the rise of alternative tools or shifting interests in the developer community.

Chart 5 - Dynamics of the number of posts on Stack OverFlow about CLion



3.2 Popular / recent posts

These are 7 posts that have about 100 thousand or more views. For information, the titles of these posts are given in the *Table 5*. These posts are mostly from 2015 - 2018. However, the last column shows the day of the last activity and it is 2022 - 2023. The most popular post is about how to install CLion.

Table 5 - The most popular posts about CLion on StackOverFlow

CreationDate	Title	ViewCount	LastActivityDate
21.10.2015	How do I <i>set up CLion</i> to compile and run?	181856	24.12.2022
04.05.2017	CMake - Creating a <i>static library</i>	140787	18.09.2023
15.09.2017	In CLion, header only library: file "does not belong to any project target, code insight features might not work properly"	138854	11.05.2023
13.11.2014	Using <i>local makefile</i> for CLion instead of CMake	121446	19.09.2022
09.01.2018	CMake was unable to find a build program corresponding to " <i>Unix Makefiles</i> "	116215	12.07.2023
27.02.2015	How to include <i>external library</i> (boost) into CLion C++ project with CMake?	83114	07.06.2020
14.05.2015	How to configure CLion IDE for <i>Qt Framework</i> ?	76348	25.12.2020

Table 6 contains the 4 most recent posts (January 2023), the number of views in them varies from 376 - 1718, indicating their popularity. These questions also mention Makefile and external libraries.

Table 6 - The most recent posts about CLion on StackOverFlow

CreationDate	Title	ViewCount
20.01.2023	Build problems with test Project trying to use zeromq	781

21.01.2023	Adding include paths to a CLion Makefile project	1718
22.01.2023	CMake is unable to locate include files	449
26.01.2023	Linker error with Libtorch, Cmake and CLion	376

3.3 The main topics and problems that are posted on Stack OverFlow

To identify the main problems that developers face when using Clion, a ***Latent Dirichlet allocation (LDA) model*** was used. LDA is a topic modeling technique for uncovering the central topics and their distributions across a set of documents. By tracking frequency of co-occurrence, LDA assumes ***words that occur together are likely part of similar topics***. The algorithm then assigns document-topic distributions based on the clusters of words that appear in the given document.

The output of the model is keywords that characterize the main topic. By analyzing these words, we can identify several key issues that CLion users face.

In the *Table 7* below in the column *Keywords* presents the 8 main words for the topic determined by the model. The *Topic name* and *Possible issues* are written based on these keywords.

Table 7 - Explanation of the main topics obtained as a result of constructing the model

№	Topic name	Keywords (main 8 words)	Possible Issues
1	Project setup, CMake, and library management	code, project, cmake, clion, sdl, library, file, include	1. Users face difficulties configuring and managing projects with CMake. 2. Questions about linking libraries (e.g., SDL) and handling header files (include). 3. Errors while integrating libraries into projects.
2	Errors in C++ code, related to basic operations and data types.	code, int, return, cout, string, size, printf.	1. Problems related to C++ syntax (e.g., cout, printf). 2. Errors with data types (int, string) or their size (size). 3. Difficulties writing and debugging basic C++ code.
3	Working with CLion tools and build errors	cmakefiles, mingw, opencv, error, make_exe, bin.	1. Errors using CMake (cmakefiles) and configuring MinGW. 2. Integration challenges with third-party libraries like OpenCV. 3. Build errors (error, make_exe, bin) caused by improper environment settings.
4	Working with external resources, including images and links.	quot, png, https, img_src, alt, rel, nofollow_noreferrer, href	1. Managing resources such as images (png, img_src) in projects. 2. Problems with relative and absolute links (href, rel). 3. Errors when working with network or external resources.
5	Integrating OpenGL, Boost, and MySQL	boost, std, glfw, opengl, glew, mysql, undefined_reference, log	1. OpenGL and related libraries integration (glfw, glew). 2. Using Boost and MySQL. 3. Linking errors (undefined_reference) and logging issues (log).

Also there was built another model that provides deeper insights into the nature of issues CLion users face. Below is a breakdown of topics and issues.

Main topics:

- CMake configuration and environment setup.
- Library integrations (OpenGL, SDL, Boost, OpenCV).
- Linking, build, and configuration errors (especially with MinGW).
- Logging, POSIX, multithreading, and Python integration.

In conclusion of this question, it can be said that the number of posts/issues published on the Stack Overflow has been decreasing since 2021. However, posts published in 2015-2018 are still relevant. Based on the LDA model, the following common issues were identified:

- ***challenges with external dependency configuration.***
- ***errors integrating graphical and system libraries.***
- ***compatibility issues with tools like Visual Studio and MinGW.***
- ***CLion users are working on advanced projects that demand complex tool setups and library integrations.***

As part of improving the posts analysis and a more in-depth result, more could be done:

- use a different model and compare the results,
- choose different learning parameters,
- analyze not only English posts,
- deal with the lack of posts in 2019.

Conclusion

The analysis reveals that CLion experienced a substantial revenue growth of 32.8% in 2020, driven by an increase in the number of unique clients. The monthly revenue, despite an overall upward trend, showed fluctuations. Revenue forecasts indicate continued growth. In the first half of 2021, the average monthly revenue at the final price is projected to increase by a minimum of 15.8%, reaching \$9.6 million, and at a maximum by 37.4%, reaching \$11.4 million.

Trends among C++ developers highlight their shifting focus toward newer standards and technologies. The adoption of C++20 concepts (52.6%) and modules (46.8%) demonstrates a willingness to embrace innovation, while AI tools like ChatGPT and GitHub Copilot are becoming integral to development workflows. Additionally, there is a growing emphasis on secure software practices, with a sharp rise in adherence to secure development life cycles, reaching 54.9% in 2023 compared to 29% in 2022.

On Stack Overflow, the analysis of posts related to CLion underscores the primary challenges faced by its users, including configuration issues, library integration, and build errors. Despite these problems, the dynamics of posts indicate a consistent engagement with the IDE, reflecting its

popularity and utility within the developer community. Addressing the frequently discussed challenges on platforms like Stack Overflow could further enhance user satisfaction and consolidate CLion's position as a leading IDE for C++.