

Вдосконалення інструменту автоматичної оцінки стабільності та зрозумілості програмного коду Swift з інтеграцією в середовище розробки

Слайди до доповіді

Виконав:

Студент БП-4 “Інженерія програмного забезпечення”

Суліменко А.А.

Науковий керівник:

Старший викладач

Франків О.О.

Мета

Розширити та оптимізувати існуючий аналізатор для автоматизованої оцінки Swift-коду, забезпечивши його гнучку конфігурацію та тісну інтеграцію в середовище розробки Xcode і CI/CD процеси.

Результат

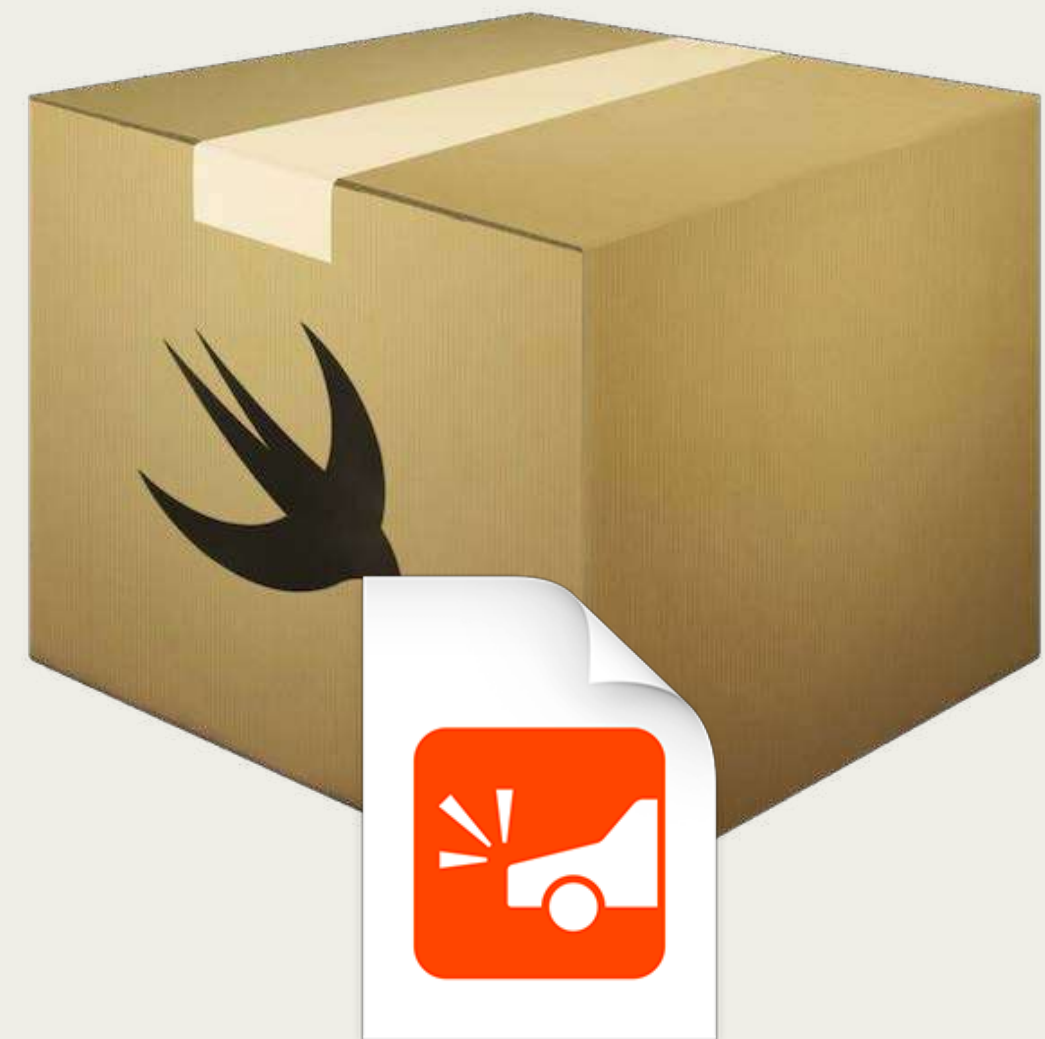
Вдосконалений SPM-інструмент – Stability Assurance Tool з інтеграцією через Build Phases та SPM Command Plugins, YAML-конфігурацією, оптимізованим ядром аналізу та можливістю виведення результатів в IDE, що підвищує ефективність контролю якості Swift-проектів.

Stability Assurance Tool

Актуальність та передумови

- Зростання складності Swift-проектів вимагає ефективних інструментів контролю якості.
- Динамічний розвиток мови Swift та її екосистеми створює нові виклики для аналізу стабільності.
- Потреба в комплексній оцінці, що поєднує різні аспекти якості та надає розробникам чіткий, дієвий зворотний зв'язок

Проблема: Необхідність глибшої інтеграції аналізу в Xcode, автоматизації в CI/CD та гнучкого налаштування для практичного застосування.



Основні напрямки вдосконалень

Покращення Ядра Аналізу:

- Оптимізація парсингу Swift-коду (з використанням SwiftSyntax).
- Впровадження паралельних обчислень метрик (Swift Concurrency Task).
- Перехід на гнучку протокольно-орієнтовану архітектуру для метрик.
- Розширення та адаптація метрик програмного забезпечення.

Тісна Інтеграція в Середовище Розробки:

- Через Build Phases Scripts для автоматичного аналізу під час збірки.
- Реалізація SPM Command Plugins для ручного запуску.

Гнучка Конфігурація:

- Налаштування правил, вагових коефіцієнтів, порогових значень та рівнів критичності через YAML-файли.

Покращена Візуалізація Результатів:

- Виведення попереджень та помилок безпосередньо в Issue Navigator Xcode.

Структура Stability Assurance Tool

Core Structure



Swift Command Line Tool



Swift Syntax



Swift Argument Parser



Code Analyser

Metrics



Stability Assurance Tool struct



Product stability report



```
StabilityAssuranceTool - zsh - 98x34
andreysulimenko@mbp--andrii StabilityAssuranceTool % swift run sat evaluate /Users/andreysulimenko/
/Desktop/TestDirectory
Building for debugging...
Build complete! (0.16s)
Trying to apply metrics to evaluate stability for: /Users/andreysulimenko/Desktop/TestDirectory

PRODUCT STABILITY EVALUATION REPORT
-----
System analyzed: Swift
-----
Project Overview
  Project directory: /Users/andreysulimenko/Desktop/TestDirectory
  Number of classes: 8
  Lines of code: 104
  Project scale: small
-----
Metrics Summary
WMC ('Weighted Method per Class'): 0.25
- mark: accepted
RFC ('Response for Class'): 0.88
- mark: good
NOC ('Number of Children'): 0.0
- mark: good
Overall mark: Good
NOTE: The evaluated product demonstrates stability,
aligning with stability metrics that indicate a
satisfactory level of stability.
```

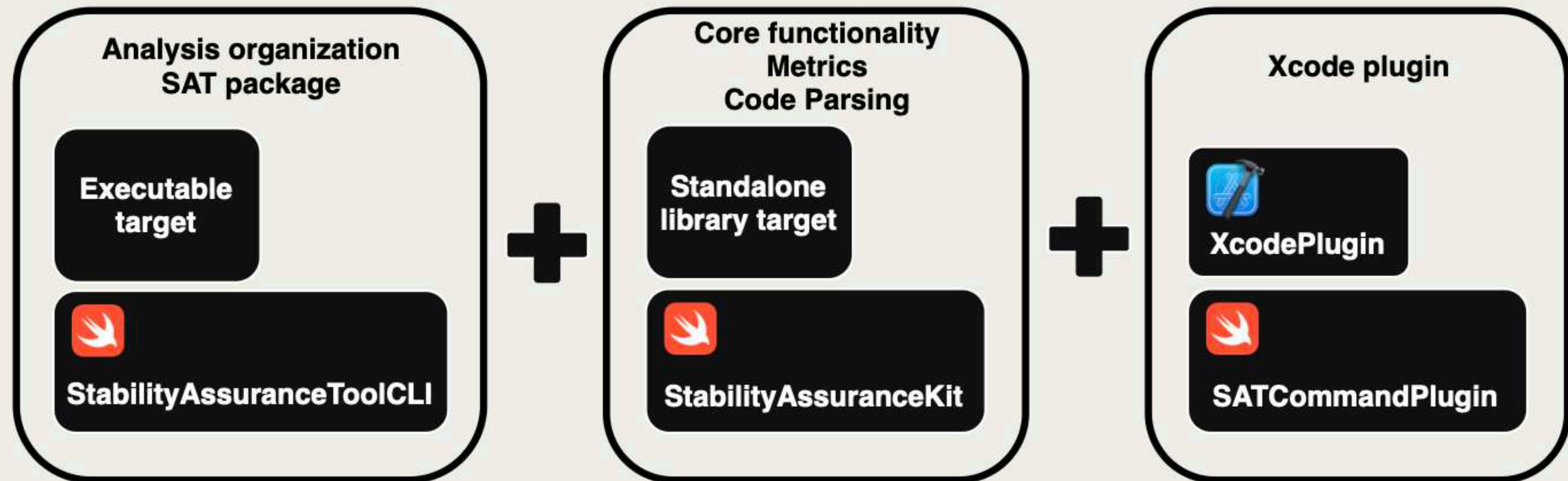
Stability Assurance Tool | адаптація метрик

База: Дослідження Chidamber, S. R., & Kemerer “A metrics suite for object oriented design.”

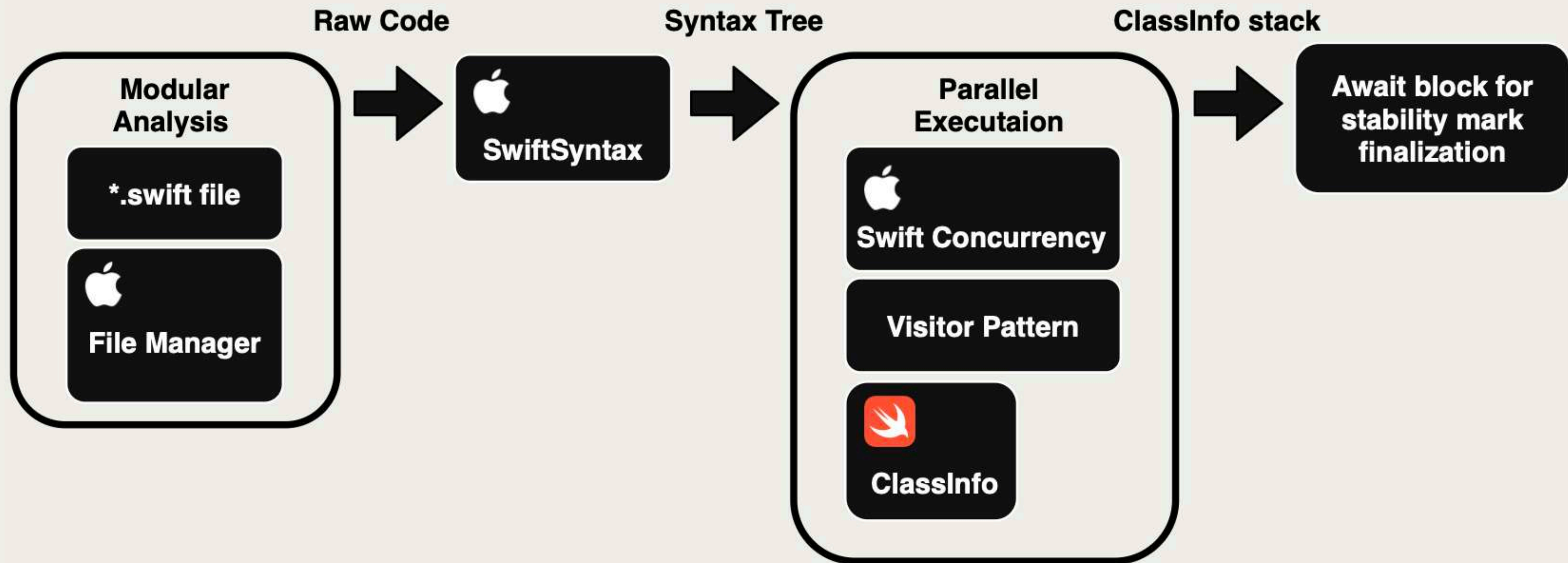
Запропонована адаптація:

- WMC - зважений метод на клас
- NOC - кількість нащадків
- RFC - реакція на клас
- LOC - розмірність модуля програмного забезпечення
- LOCM - показник недостатньої зчепленості методів

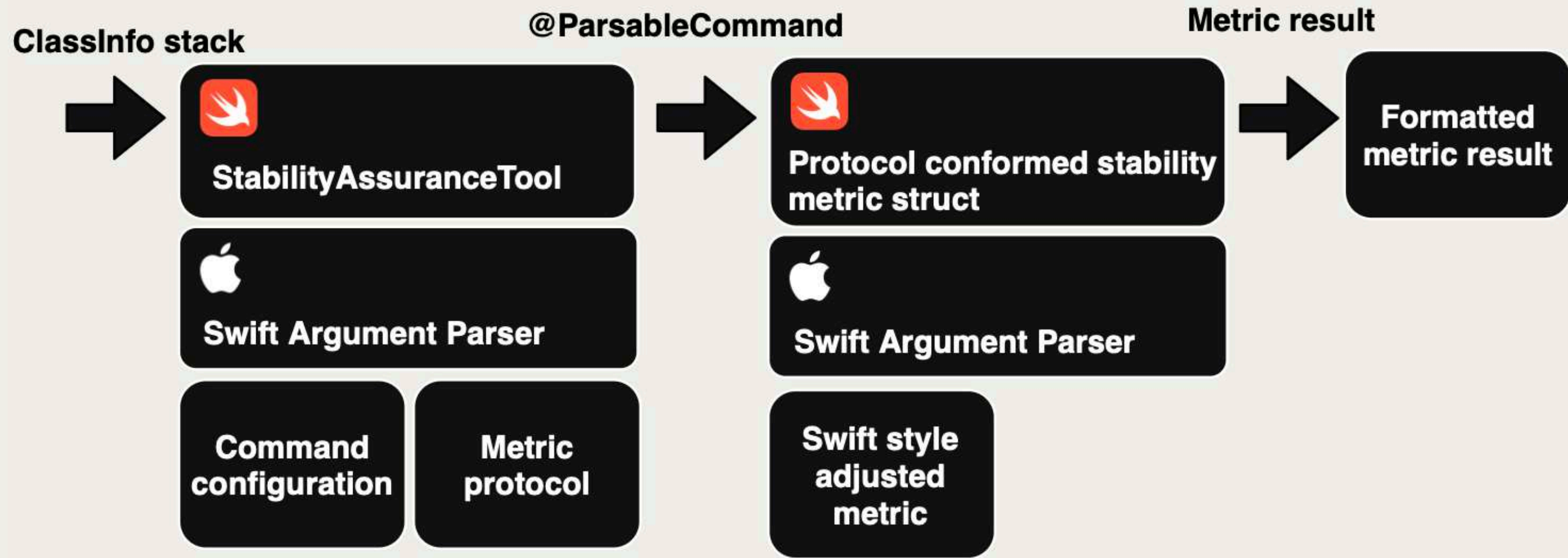
Архітектурний Дизайн SPM-Інструменту



Stability Assurance Tool | Core Module Development



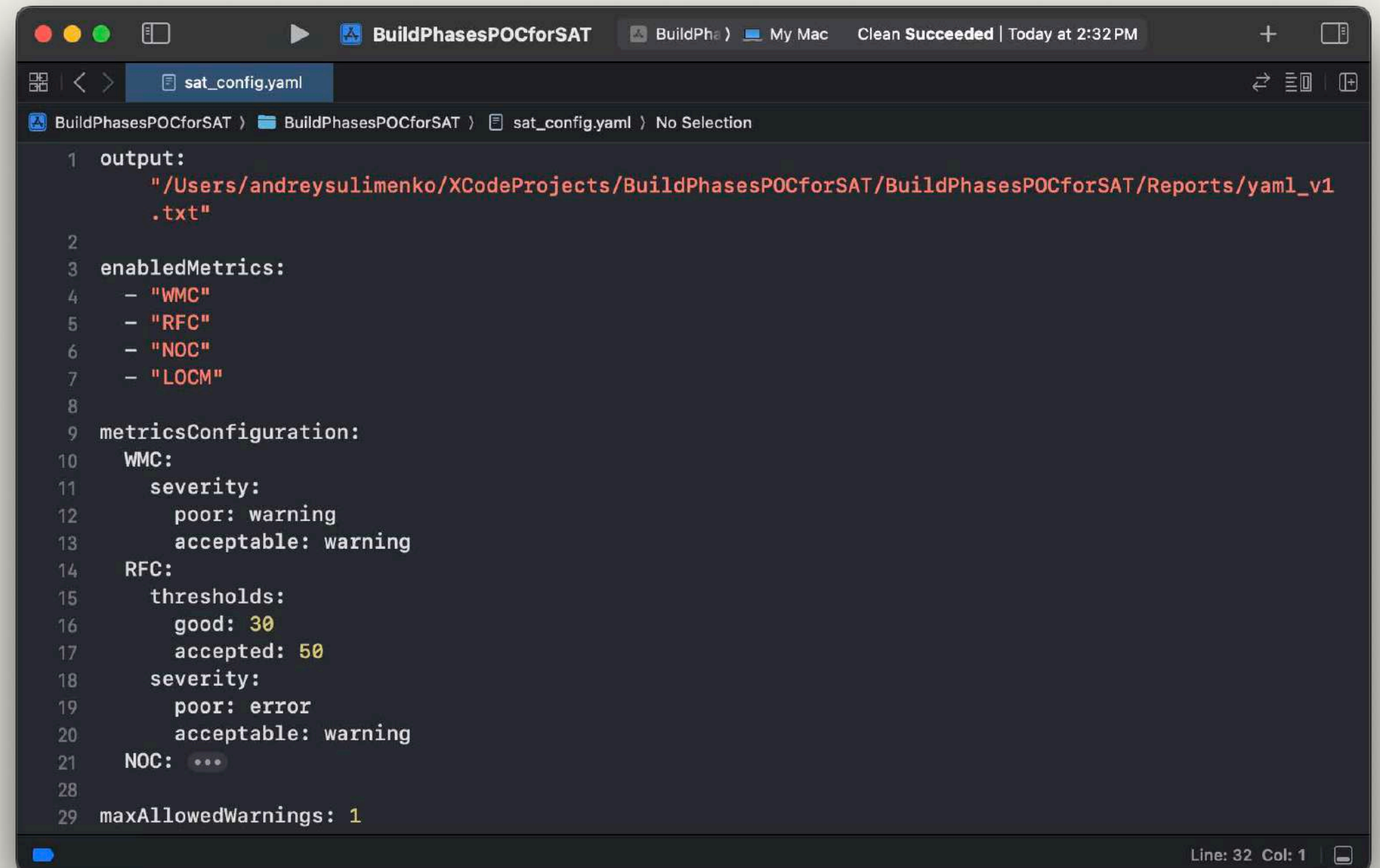
Stability Assurance Tool



Конфігураційні зміни

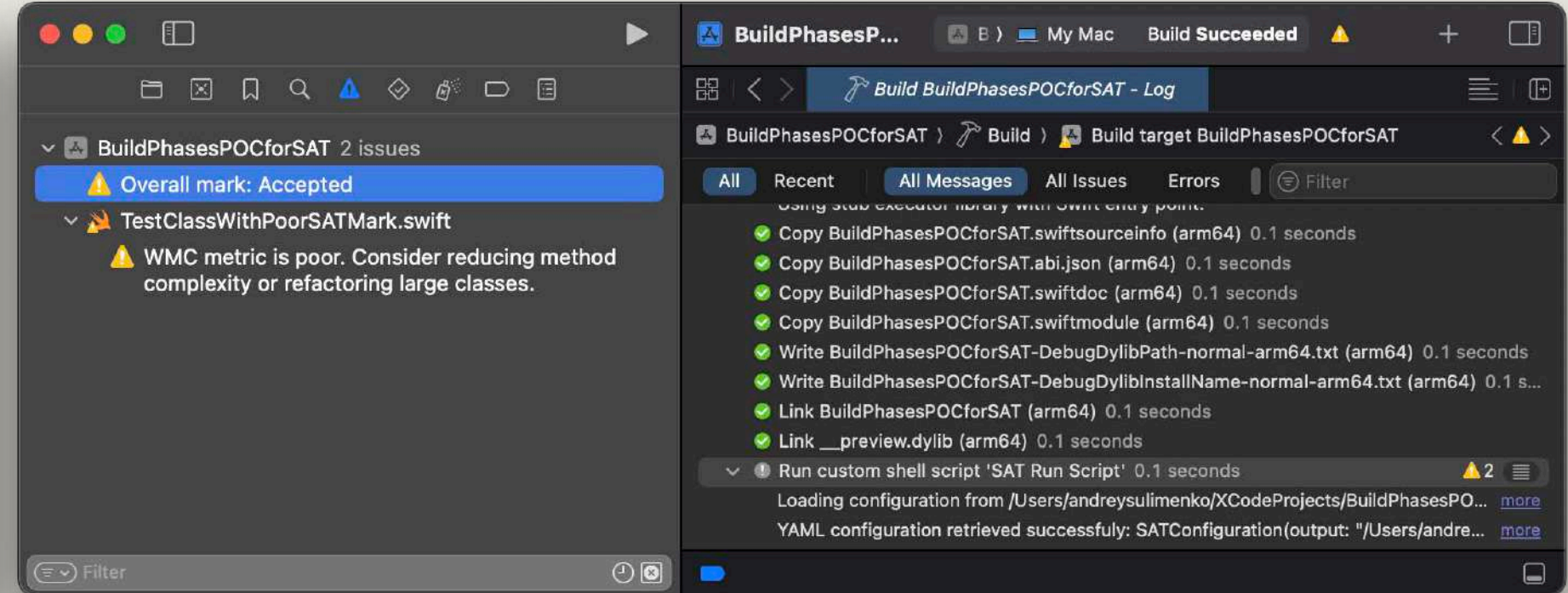
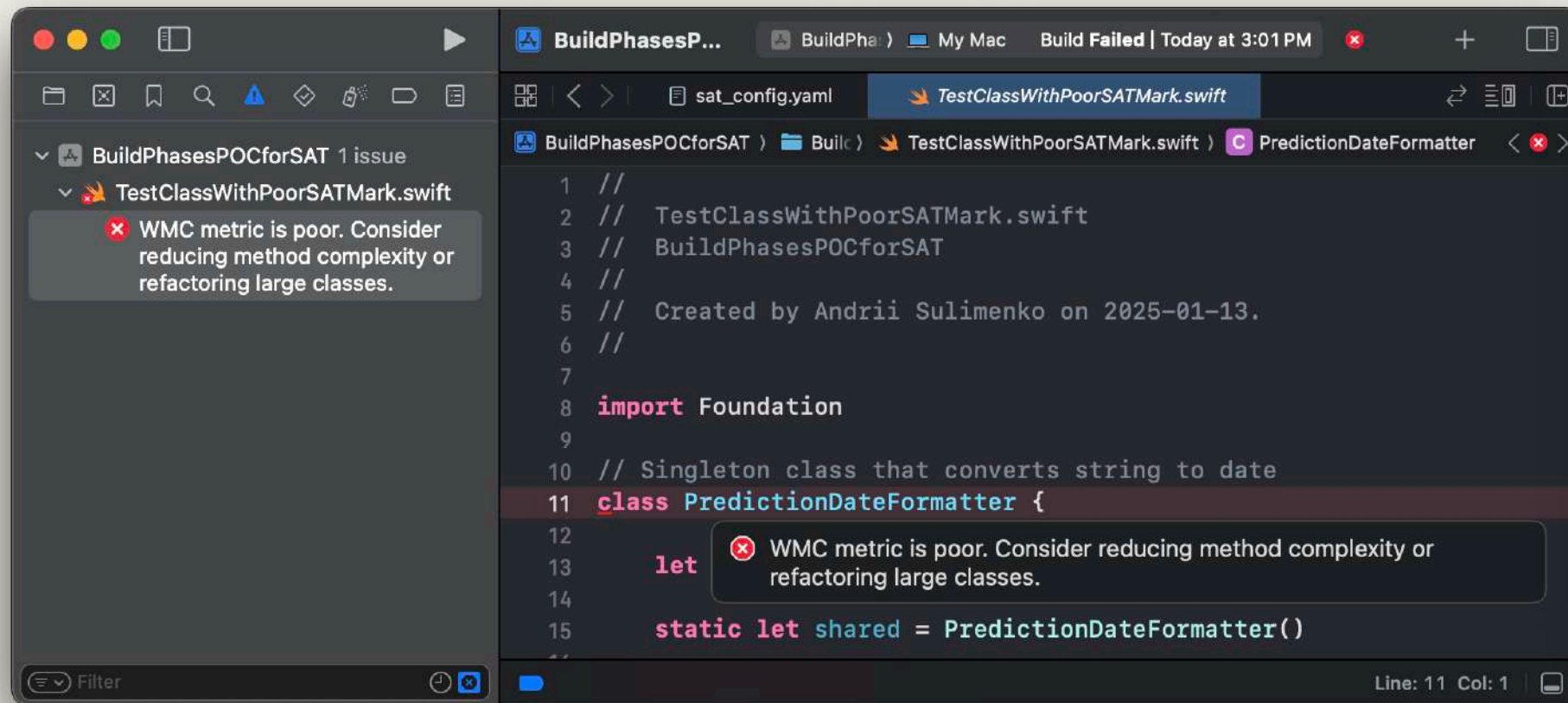
Що конфігурується:

- Увімкнення/вимкнення правил/метрик
- Порогові значення для метрик
- Вагові коефіцієнти
- Виключення файлів
- Рівні критичності (Severity Levels) та їх вплив на оцінку/Quality Gates

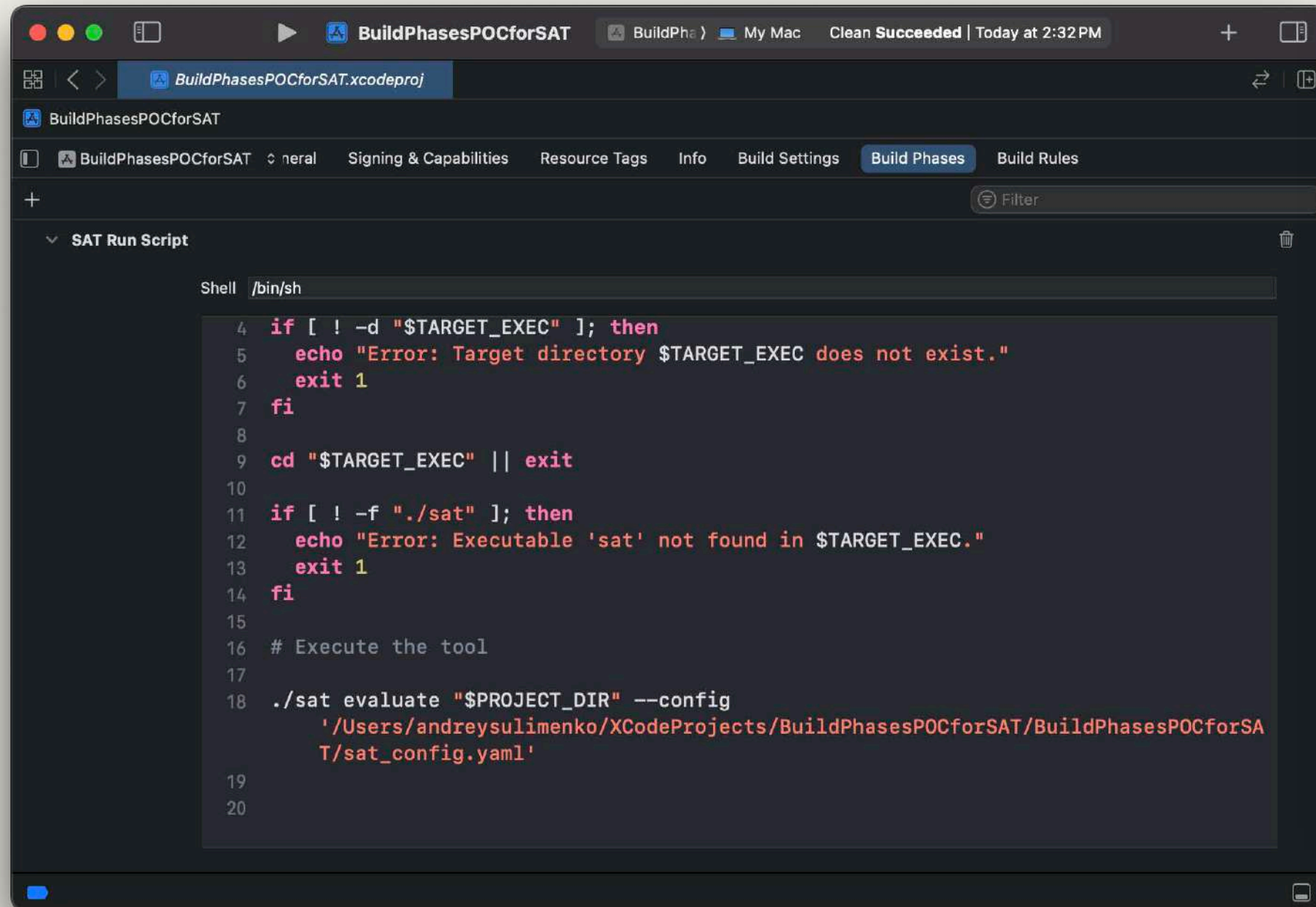


```
1 output:
2   "/Users/andreysulimenko/XCodeProjects/BuildPhasesPOCforSAT/BuildPhasesPOCforSAT/Reports/yaml_v1
3   .txt"
4
5 enabledMetrics:
6   - "WMC"
7   - "RFC"
8   - "NOC"
9   - "LOCM"
10
11 metricsConfiguration:
12   WMC:
13     severity:
14       poor: warning
15       acceptable: warning
16   RFC:
17     thresholds:
18       good: 30
19       accepted: 50
20     severity:
21       poor: error
22       acceptable: warning
23   NOC: ...
24
25
26
27
28
29 maxAllowedWarnings: 1
```

Впровадження Загальної Оцінки та CI/CD

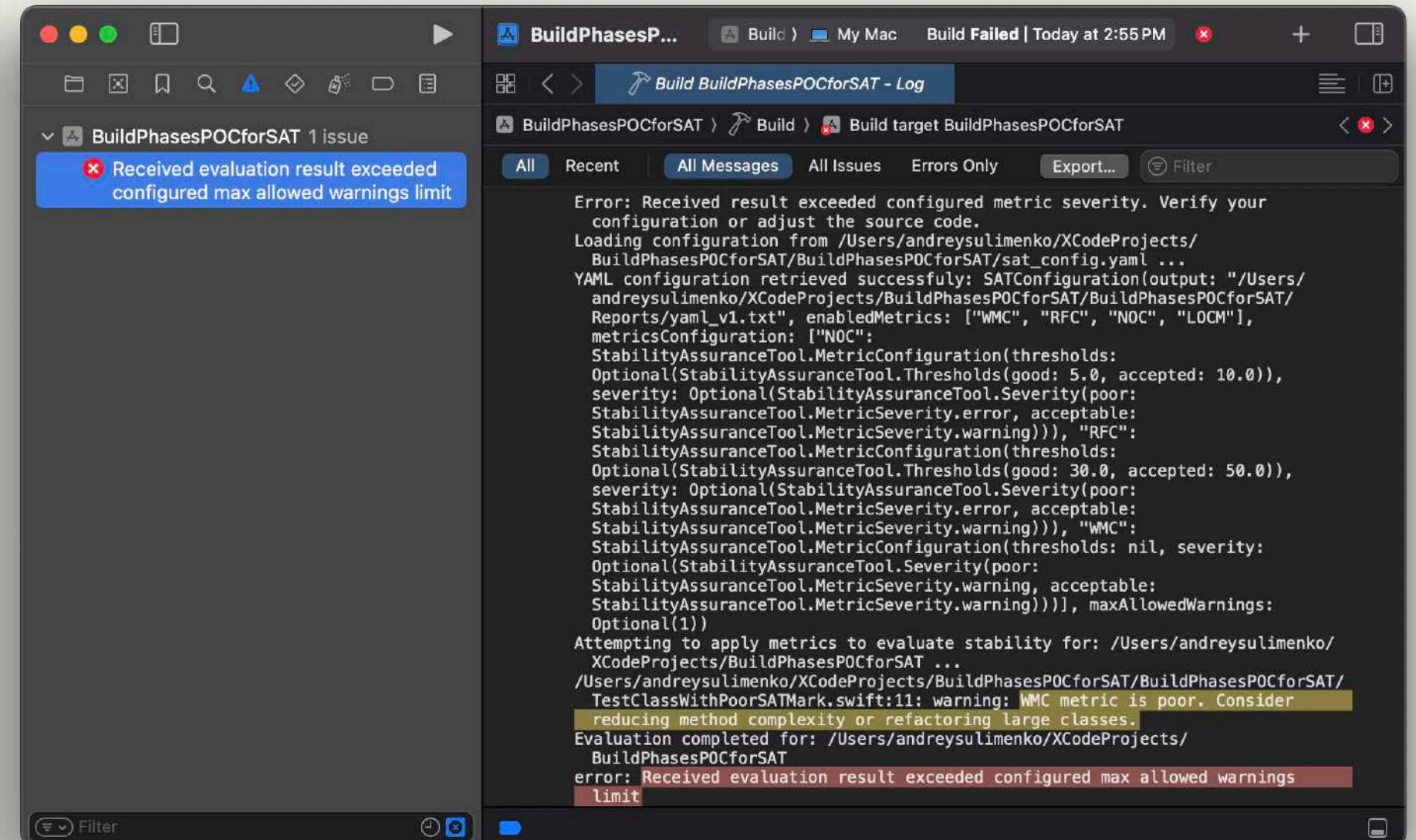


Інтеграція з Xcode: Build Phases та Виведення Результатів



The screenshot shows the Xcode interface with the 'Build Phases' tab selected. A 'SAT Run Script' phase is added to the 'Build' phase. The script content is as follows:

```
Shell /bin/sh
4 if [ ! -d "$TARGET_EXEC" ]; then
5     echo "Error: Target directory $TARGET_EXEC does not exist."
6     exit 1
7 fi
8
9 cd "$TARGET_EXEC" || exit
10
11 if [ ! -f "./sat" ]; then
12     echo "Error: Executable 'sat' not found in $TARGET_EXEC."
13     exit 1
14 fi
15
16 # Execute the tool
17
18 ./sat evaluate "$PROJECT_DIR" --config
19     '/Users/andreysulimenko/XCodeProjects/BuildPhasesPOCforSAT/BuildPhasesPOCforSA
20     T/sat_config.yaml'
```



The screenshot shows the Xcode Build Log for a failed build. A blue notification banner at the top reads: "Received evaluation result exceeded configured max allowed warnings limit". The log content is as follows:

```
Build target BuildPhasesPOCforSAT
All Messages All Issues Errors Only Export... Filter
Error: Received result exceeded configured metric severity. Verify your
configuration or adjust the source code.
Loading configuration from /Users/andreysulimenko/XCodeProjects/
BuildPhasesPOCforSAT/BuildPhasesPOCforSAT/sat_config.yaml ...
YAML configuration retrieved successfully: SATConfiguration(output: "/Users/
andreysulimenko/XCodeProjects/BuildPhasesPOCforSAT/BuildPhasesPOCforSAT/
Reports/yaml_v1.txt", enabledMetrics: ["WMC", "RFC", "NOC", "LOCM"],
metricsConfiguration: ["NOC":
StabilityAssuranceTool.MetricConfiguration(thresholds:
Optional(StabilityAssuranceTool.Thresholds(good: 5.0, accepted: 10.0)),
severity: Optional(StabilityAssuranceTool.Severity(poor:
StabilityAssuranceTool.MetricSeverity.error, acceptable:
StabilityAssuranceTool.MetricSeverity.warning))), "RFC":
StabilityAssuranceTool.MetricConfiguration(thresholds:
Optional(StabilityAssuranceTool.Thresholds(good: 30.0, accepted: 50.0)),
severity: Optional(StabilityAssuranceTool.Severity(poor:
StabilityAssuranceTool.MetricSeverity.error, acceptable:
StabilityAssuranceTool.MetricSeverity.warning))), "WMC":
StabilityAssuranceTool.MetricConfiguration(thresholds: nil, severity:
Optional(StabilityAssuranceTool.Severity(poor:
StabilityAssuranceTool.MetricSeverity.warning, acceptable:
StabilityAssuranceTool.MetricSeverity.warning))), maxAllowedWarnings:
Optional(1))
Attempting to apply metrics to evaluate stability for: /Users/andreysulimenko/
XCodeProjects/BuildPhasesPOCforSAT ...
/Users/andreysulimenko/XCodeProjects/BuildPhasesPOCforSAT/BuildPhasesPOCforSAT/
TestClassWithPoorSATMark.swift:11: warning: WMC metric is poor. Consider
reducing method complexity or refactoring large classes.
Evaluation completed for: /Users/andreysulimenko/XCodeProjects/
BuildPhasesPOCforSAT
error: Received evaluation result exceeded configured max allowed warnings
limit
```

В и с н о в к и

- В ході дослідження було детально описано опрацьовані розширення та модернізації архітектури інструменту Stability Assurance Tool.
- Визначено особливості аналізу архітектури програмних модулів Swift
- Реалізовано гнучкий механізм конфігурації аналізу програмного забезпечення.
- Оптимізовано ядро аналізатора та впроваджено механізм формування загальної оцінки якості з візуалізацією результатів в IDE.

Дякую за увагу!
