**Как создавать данные метрик кода**

* 02.11.2018
* Чтение занимает 6 мин
  + [https://github.com/Mikejo5000.png?size=32](https://github.com/Mikejo5000)

Данные метрик кода можно создавать тремя способами:

* Включив [анализаторы качества кода .NET](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/how-to-generate-code-metrics-data?view=vs-2019#net-code-quality-analyzers-code-metrics-rules) и включив в него четыре метрики кода (сопровождаемость).
* Выбрав команду меню [**анализ** > **метрики кода**](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/how-to-generate-code-metrics-data?view=vs-2019#calculate-code-metrics-menu-command) в Visual Studio.
* Из [командной строки](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/how-to-generate-code-metrics-data?view=vs-2019#command-line-code-metrics) для проектов C# и Visual Basic.

**Правила метрик кода для анализаторов качества кода .NET**

Анализаторы качества кода .NET включают несколько правил [анализатора](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/roslyn-analyzers-overview?view=vs-2019) кода метрик:

* [CA1501](https://docs.microsoft.com/ru-ru/dotnet/fundamentals/code-analysis/quality-rules/ca1501)
* [CA1502](https://docs.microsoft.com/ru-ru/dotnet/fundamentals/code-analysis/quality-rules/ca1502)
* [CA1505](https://docs.microsoft.com/ru-ru/dotnet/fundamentals/code-analysis/quality-rules/ca1505)
* [CA1506](https://docs.microsoft.com/ru-ru/dotnet/fundamentals/code-analysis/quality-rules/ca1506)

Эти правила по умолчанию отключены, но их можно включить в [**Обозреватель решений**](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/use-roslyn-analyzers?view=vs-2019#set-rule-severity-from-solution-explorer) или в файле [EditorConfig](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/use-roslyn-analyzers?view=vs-2019#set-rule-severity-in-an-editorconfig-file) . Например, чтобы включить правило CA1502 в виде предупреждения, файл EditorConfig будет содержать следующую запись:

C#Копировать

dotnet\_diagnostic.CA1502.severity = warning

**Конфигурация**

Вы можете настроить пороговые значения, при которых будут срабатывать правила метрики кода.

1. Создание текстового файла. Например, можно присвоить имя *CodeMetricsConfig.txt*.
2. Добавьте необходимые пороговые значения в текстовый файл в следующем формате:

txtКопировать

CA1502: 10

В этом примере правило [CA1502](https://docs.microsoft.com/ru-ru/dotnet/fundamentals/code-analysis/quality-rules/ca1502) настраивается на срабатывание, когда сложность сложностью организации циклов метода больше 10.

1. В окне **Свойства** Visual Studio или в файле проекта пометьте действие сборки файла конфигурации как [**аддитионалфилес**](https://docs.microsoft.com/ru-ru/visualstudio/ide/build-actions?view=vs-2019#build-action-values). Пример:

XMLКопировать

<ItemGroup>

<AdditionalFiles Include="CodeMetricsConfig.txt" />

</ItemGroup>

**Команда меню «вычислить метрики кода»**

Создайте метрики кода для одного или всех открытых проектов в интегрированной среде разработки с помощью меню **анализ** > **метрик кода** .

**Создание результатов метрик кода для всего решения**

Вы можете создавать Результаты метрик кода для всего решения одним из следующих способов:

* В строке меню выберите **анализ** > **рассчитать метрики кода** > **для решения**.
* В **Обозреватель решений** щелкните правой кнопкой мыши решение и выберите пункт **рассчитать метрики кода**.
* В окне **Результаты метрик кода** выберите кнопку **вычислить метрики кода для решения** .

Результаты будут сформированы, и откроется окно **результаты метрики кода** . Чтобы просмотреть подробные сведения о результатах, разверните дерево в столбце **Иерархия** .

**Создание результатов метрик кода для одного или нескольких проектов**

1. В **Обозреватель решений** выберите один или несколько проектов.
2. В строке меню выберите **анализ** > **рассчитать метрики кода** > **для выбранных проектов**.

Результаты будут сформированы, и откроется окно **результаты метрики кода** . Чтобы просмотреть подробные сведения о результатах, разверните дерево в **иерархии**.

**Метрики кода командной строки**

Вы можете создавать данные метрик кода из командной строки для проектов C# и Visual Basic для .NET Framework, .NET Core и .NET Standard приложений. Чтобы запустить метрики кода из командной строки, установите [пакет NuGet Microsoft. CodeAnalysis. Metrics](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/how-to-generate-code-metrics-data?view=vs-2019#microsoftcodeanalysismetrics-nuget-package) или [Metrics.exe](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/how-to-generate-code-metrics-data?view=vs-2019#metricsexe) Создайте исполняемый файл самостоятельно.

**Пакет NuGet Microsoft. CodeAnalysis. Metrics**

Самый простой способ создать данные метрик кода из командной строки — установить пакет NuGet [Microsoft. CodeAnalysis. Metrics](https://www.nuget.org/packages/Microsoft.CodeAnalysis.Metrics/) . После установки пакета запустите msbuild /t:Metrics из каталога, содержащего файл проекта. Пример:

shellКопировать

C:\source\repos\ClassLibrary3\ClassLibrary3>msbuild /t:Metrics

Microsoft (R) Build Engine version 16.0.360-preview+g9781d96883 for .NET Framework

Copyright (C) Microsoft Corporation. All rights reserved.

Build started 1/22/2019 4:29:57 PM.

Project "C:\source\repos\ClassLibrary3\ClassLibrary3\ClassLibrary3.csproj" on node 1 (Metrics target(s))

.

Metrics:

C:\source\repos\ClassLibrary3\packages\Microsoft.CodeMetrics.2.6.4-ci\build\\..\Metrics\Metrics.exe /project:C:\source\repos\ClassLibrary3\ClassLibrary3\ClassLibrary3.csproj /out:ClassLibrary3.Metrics.xml

Loading ClassLibrary3.csproj...

Computing code metrics for ClassLibrary3.csproj...

Writing output to 'ClassLibrary3.Metrics.xml'...

Completed Successfully.

Done Building Project "C:\source\repos\ClassLibrary3\ClassLibrary3\ClassLibrary3.csproj" (Metrics target(s)).

Build succeeded.

0 Warning(s)

0 Error(s)

Можно переопределить имя выходного файла, указав /p:MetricsOutputFile=<filename> . Вы также можете получить данные метрик кода в [стиле прежних версий](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/how-to-generate-code-metrics-data?view=vs-2019#previous-versions) , указав /p:LEGACY\_CODE\_METRICS\_MODE=true . Пример:

shellКопировать

C:\source\repos\ClassLibrary3\ClassLibrary3>msbuild /t:Metrics /p:LEGACY\_CODE\_METRICS\_MODE=true /p:MetricsOutputFile="Legacy.xml"

Microsoft (R) Build Engine version 16.0.360-preview+g9781d96883 for .NET Framework

Copyright (C) Microsoft Corporation. All rights reserved.

Build started 1/22/2019 4:31:00 PM.

The "MetricsOutputFile" property is a global property, and cannot be modified.

Project "C:\source\repos\ClassLibrary3\ClassLibrary3\ClassLibrary3.csproj" on node 1 (Metrics target(s))

.

Metrics:

C:\source\repos\ClassLibrary3\packages\Microsoft.CodeMetrics.2.6.4-ci\build\\..\Metrics.Legacy\Metrics.Legacy.exe /project:C:\source\repos\ClassLibrary3\ClassLibrary3\ClassLibrary3.csproj /out:Legacy.xml

Loading ClassLibrary3.csproj...

Computing code metrics for ClassLibrary3.csproj...

Writing output to 'Legacy.xml'...

Completed Successfully.

Done Building Project "C:\source\repos\ClassLibrary3\ClassLibrary3\ClassLibrary3.csproj" (Metrics target(s)).

Build succeeded.

0 Warning(s)

0 Error(s)

**Выходные данные метрик кода**

Созданный выход XML имеет следующий формат:

XMLКопировать

<?xml version="1.0" encoding="utf-8"?>

<CodeMetricsReport Version="1.0">

<Targets>

<Target Name="ConsoleApp20.csproj">

<Assembly Name="ConsoleApp20, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null">

<Metrics>

<Metric Name="MaintainabilityIndex" Value="100" />

<Metric Name="CyclomaticComplexity" Value="1" />

<Metric Name="ClassCoupling" Value="1" />

<Metric Name="DepthOfInheritance" Value="1" />

<Metric Name="SourceLines" Value="11" />

<Metric Name="ExecutableLines" Value="1" />

</Metrics>

<Namespaces>

<Namespace Name="ConsoleApp20">

<Metrics>

<Metric Name="MaintainabilityIndex" Value="100" />

<Metric Name="CyclomaticComplexity" Value="1" />

<Metric Name="ClassCoupling" Value="1" />

<Metric Name="DepthOfInheritance" Value="1" />

<Metric Name="SourceLines" Value="11" />

<Metric Name="ExecutableLines" Value="1" />

</Metrics>

<Types>

<NamedType Name="Program">

<Metrics>

<Metric Name="MaintainabilityIndex" Value="100" />

<Metric Name="CyclomaticComplexity" Value="1" />

<Metric Name="ClassCoupling" Value="1" />

<Metric Name="DepthOfInheritance" Value="1" />

<Metric Name="SourceLines" Value="7" />

<Metric Name="ExecutableLines" Value="1" />

</Metrics>

<Members>

<Method Name="void Program.Main(string[] args)" File="C:\source\repos\ConsoleApp20\ConsoleApp20\Program.cs" Line="7">

<Metrics>

<Metric Name="MaintainabilityIndex" Value="100" />

<Metric Name="CyclomaticComplexity" Value="1" />

<Metric Name="ClassCoupling" Value="1" />

<Metric Name="SourceLines" Value="4" />

<Metric Name="ExecutableLines" Value="1" />

</Metrics>

</Method>

</Members>

</NamedType>

</Types>

</Namespace>

</Namespaces>

</Assembly>

</Target>

</Targets>

</CodeMetricsReport>

**Metrics.exe**

Если вы не хотите устанавливать пакет NuGet, вы можете создать и использовать *Metrics.exe* исполняемый файл напрямую. Чтобы создать *Metrics.exe* исполняемый файл:

1. Клонировать репозиторий [DotNet/Roslyn-Analyzers](https://github.com/dotnet/roslyn-analyzers) .
2. Откройте Командная строка разработчика для Visual Studio с правами администратора.
3. В корне репозитория **Roslyn-Analyzers** выполните следующую команду: Restore.cmd
4. Перейдите в каталог *срк\тулс*.
5. Выполните следующую команду, чтобы выполнить сборку проекта **метрик. csproj** :

shellКопировать

msbuild /m /v:m /p:Configuration=Release Metrics.csproj

Исполняемый файл с именем *Metrics.exe* создается в каталоге *артифактс\бин* в корне репозитория.

**Использование Metrics.exe**

Чтобы запустить *Metrics.exe*, укажите проект или решение и выходной XML-файл в качестве аргументов. Пример:

shellКопировать

C:\>Metrics.exe /project:ConsoleApp20.csproj /out:report.xml

Loading ConsoleApp20.csproj...

Computing code metrics for ConsoleApp20.csproj...

Writing output to 'report.xml'...

Completed Successfully.

**Устаревший режим**

Можно выбрать сборку *Metrics.exe* в *устаревшем режиме*. Версия средства в устаревшем режиме создает значения метрик, которые ближе к [более старым версиям созданного средства](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/how-to-generate-code-metrics-data?view=vs-2019#previous-versions). Кроме того, в устаревшем режиме *Metrics.exe* создает метрики кода для того же набора типов методов, для которого в предыдущих версиях средства сформированы метрики кода. Например, он не создает данные метрик кода для инициализаторов полей и свойств. Устаревший режим удобен для обеспечения обратной совместимости или при наличии шлюзов возврата кода на основе номеров метрик кода. Команда для сборки *Metrics.exe* в устаревшем режиме:

shellКопировать

msbuild /m /v:m /t:rebuild /p:LEGACY\_CODE\_METRICS\_MODE=true Metrics.csproj

Дополнительные сведения см. [в разделе Включение создания метрик кода в устаревшем режиме](https://github.com/dotnet/roslyn-analyzers/pull/1841).

**предыдущих версий**

В Visual Studio 2015 имеется средство метрик кода командной строки, которое также называлось *Metrics.exe*. Предыдущая версия средства выполняла двоичный анализ, то есть анализ на основе сборок. Более новая версия средства *Metrics.exe* анализирует исходный код. Поскольку более новая *Metrics.exe* средство является исходным кодом, результаты метрик кода командной строки могут отличаться от результатов, создаваемых интегрированной средой разработки Visual Studio и предыдущими версиями *Metrics.exe*. Начиная с Visual Studio 2019, интегрированная среда разработки Visual Studio анализирует исходный код, подобный средству командной строки, и результаты должны быть одинаковыми.

Новое средство метрик кода командной строки выдает метрики даже при наличии ошибок исходного кода, пока решение и проект могут быть загружены.

**Различия в значениях метрик**

Начиная с Visual Studio 2019 версии 16,4 и Microsoft. CodeAnalysis. метрики (2.9.5), SourceLines ExecutableLines замените предыдущую LinesOfCode метрику. Описание новых метрик см. в разделе [значения метрик кода](https://docs.microsoft.com/ru-ru/visualstudio/code-quality/code-metrics-values?view=vs-2019). LinesOfCodeМетрика доступна в устаревшем режиме.

Другие метрики, такие как CyclomaticComplexity и, MaintainabilityIndex используют те же формулы, что и предыдущие версии *Metrics.exe*, но новое средство подсчитывает количество IOperations (инструкции логического источника) вместо инструкций промежуточного языка (IL). Номера будут немного отличаться от тех, которые создаются в интегрированной среде разработки Visual Studio и предыдущих версиях *Metrics.exe*.

https://docs.microsoft.com/ru-ru/visualstudio/code-quality/how-to-generate-code-metrics-data?view=vs-2019