# Challenges of Managing CoreFX repo

DotNext in Moscow, RU (2017/11/13)

Karel Zikmund (🐦 @ziki_cz)

# CoreFX repo

[https://github.com/dotnet/**corefx**](https://github.com/dotnet/corefx)

- Core Framework of .NET Core
    - BCL (Base class libraries) – 50 namespaces:
        - System.Collections
        - System.IO
        - System.Linq
        - System.Data
        - System.Net
        - System.Reflection
        - System.Runtime
        - System.Xml
        - and more …
- Not included:
    - ASP.NET Core
    - WCF, EF
    - Namespaces, which are not part of .NET Core (e.g. UI / System.Xaml)

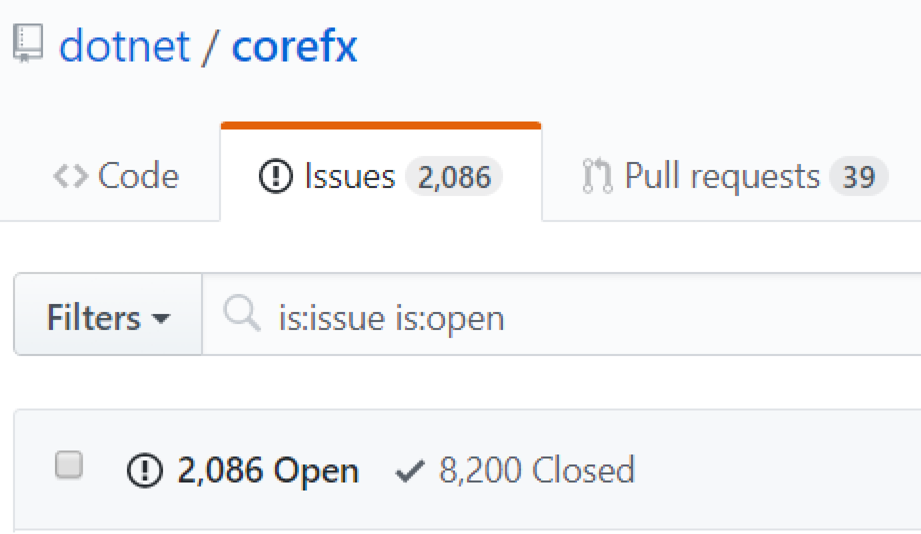
# People behind CoreFX repo

- ~40 active Microsoft engineers
  https://github.com/dotnet/corefx/graphs/contributors

- Ownership/expertise per technology
- Beyond .NET Core:
  - .NET Framework
  - .NET Native (UWP)

- More teams part of end-to-end:
  - Runtime/JIT, infrastructure, CLI, SDK, project system, NuGet, VS, etc.

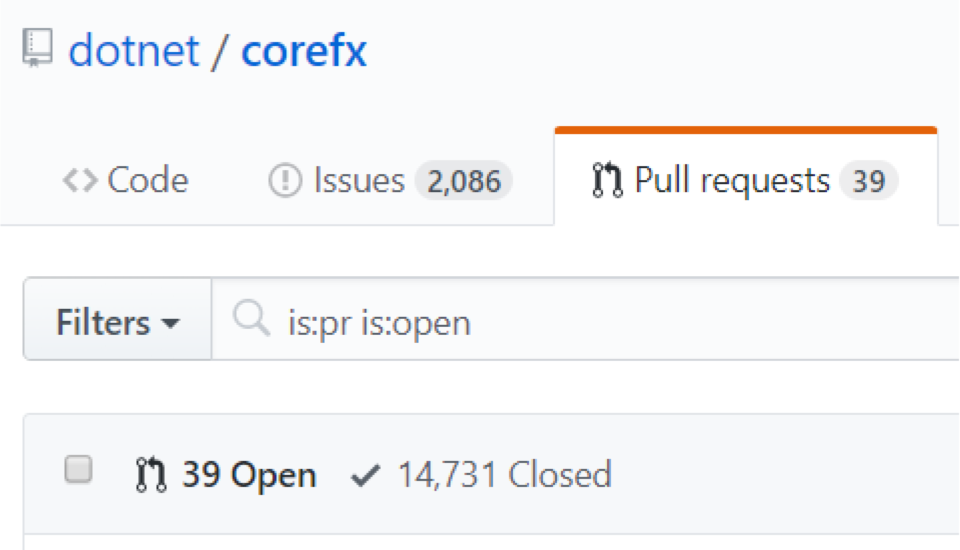
# Issues and PRs



dotnet / corefx

is:issue is:open

□ 2,086 Open ✓ 8,200 Closed

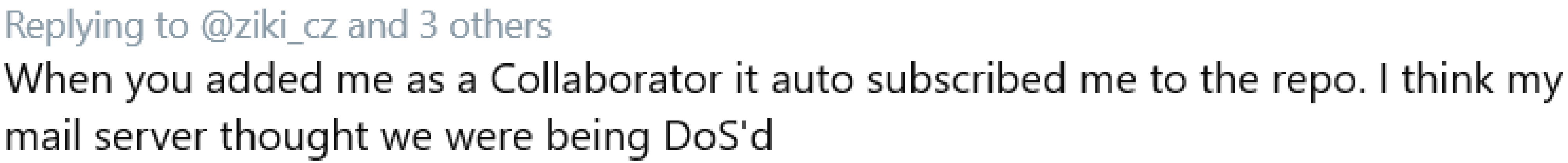


dotnet / corefx

is:pr is:open

□ 39 Open ✓ 14,731 Closed

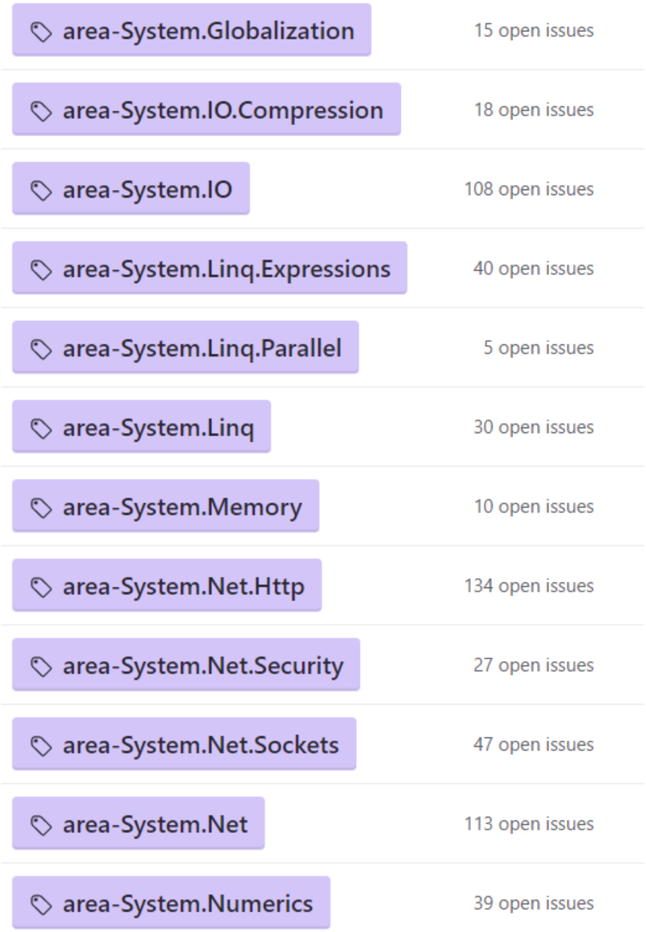
- Incoming monthly: 500-600 issues, 700 PRs
- Notifications: 400-500 per day



Replying to @ziki_cz and 3 others

When you added me as a Collaborator it auto subscribed me to the repo. I think my mail server thought we were being DoS'd

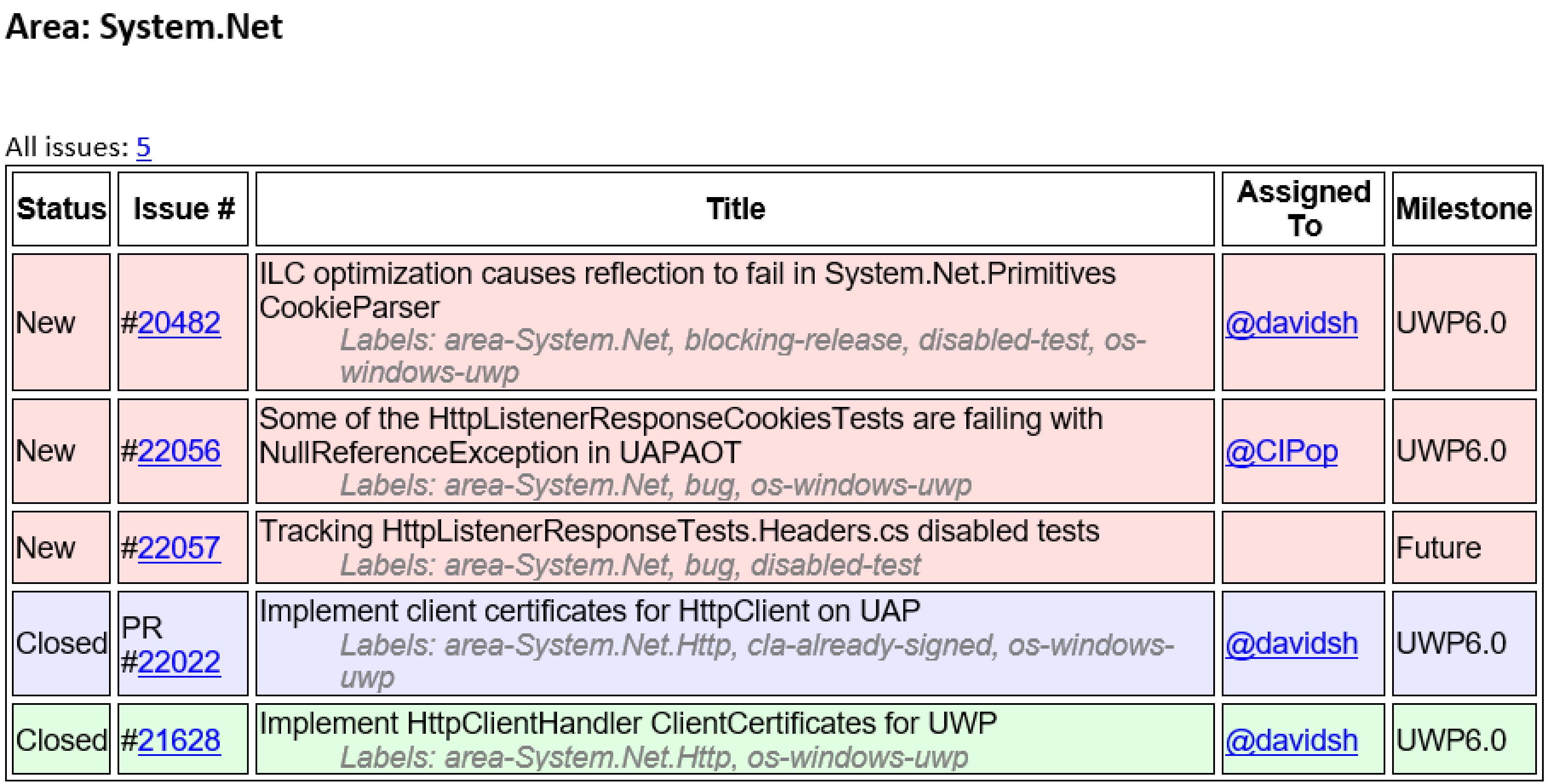
# Slicing large repo



| | |
|---|---|
| area-System.Globalization | 15 open issues |
| area-System.IO.Compression | 18 open issues |
| area-System.IO | 108 open issues |
| area-System.Linq.Expressions | 40 open issues |
| area-System.Linq.Parallel | 5 open issues |
| area-System.Linq | 30 open issues |
| area-System.Memory | 10 open issues |
| area-System.Net.Http | 134 open issues |
| area-System.Net.Security | 27 open issues |
| area-System.Net.Sockets | 47 open issues |
| area-System.Net | 113 open issues |
| area-System.Numerics | 39 open issues |

- Area labels
- Tooling – incoming / outgoing
  - Per area / query
  - Limitations (no replies, user-configuration, etc.)
- Similar problem for community:
  - New issues/PRs into Twitter feeds @corefxissues

# Tooling – Alert email

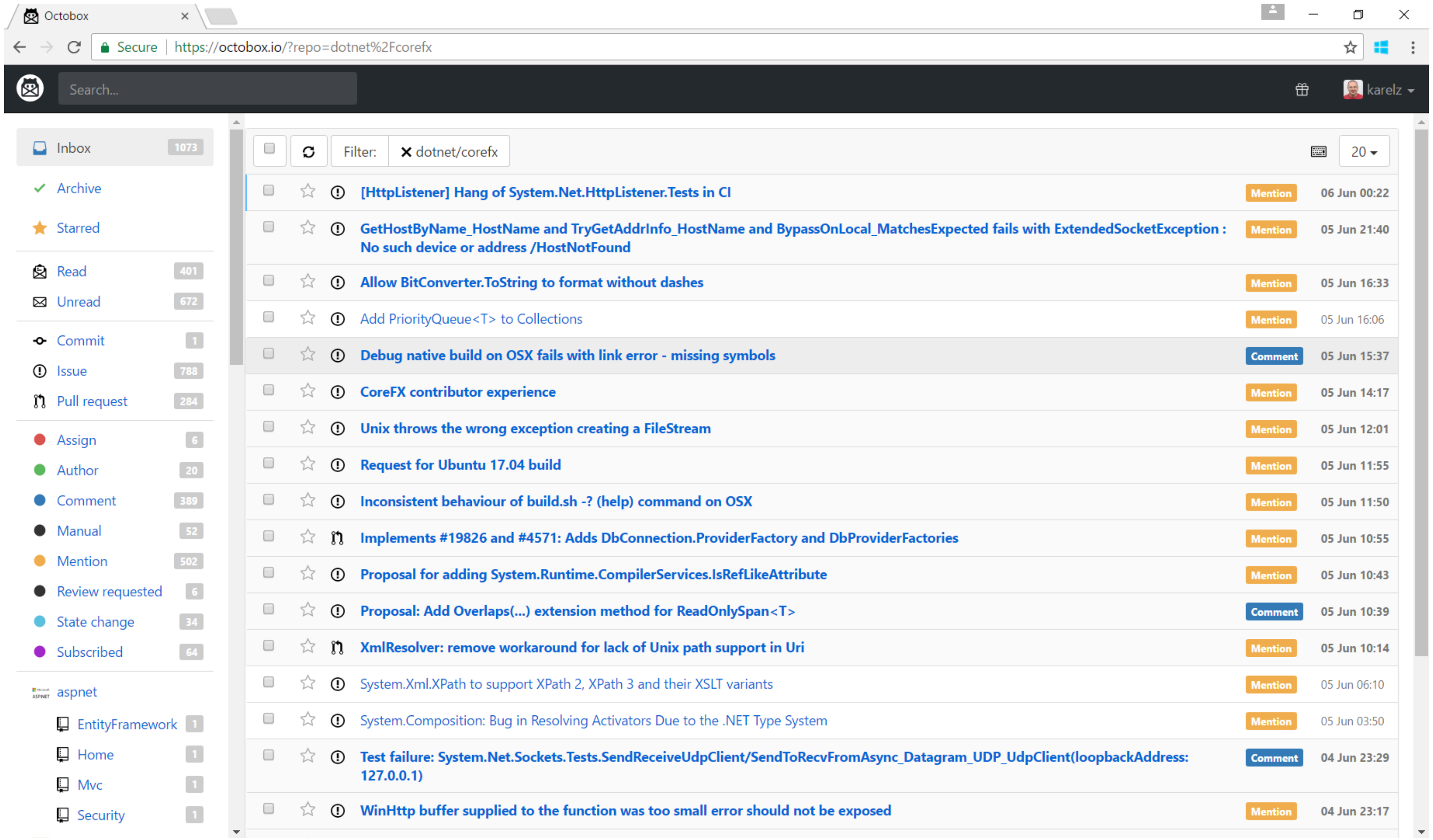


**Area: System.Net**

All issues: 5

| Status | Issue # | Title | Assigned To | Milestone |
|---|---|---|---|---|
| New | #20482 | ILC optimization causes reflection to fail in System.Net.Primitives CookieParser<br>*Labels: area-System.Net, blocking-release, disabled-test, os-windows-uwp* | @davidsh | UWP6.0 |
| New | #22056 | Some of the HttpListenerResponseCookiesTests are failing with NullReferenceException in UAPAOT<br>*Labels: area-System.Net, bug, os-windows-uwp* | @CIPop | UWP6.0 |
| New | #22057 | Tracking HttpListenerResponseTests.Headers.cs disabled tests<br>*Labels: area-System.Net, bug, disabled-test* | | Future |
| Closed | PR #22022 | Implement client certificates for HttpClient on UAP<br>*Labels: area-System.Net.Http, cla-already-signed, os-windows-uwp* | @davidsh | UWP6.0 |
| Closed | #21628 | Implement HttpClientHandler ClientCertificates for UWP<br>*Labels: area-System.Net.Http, os-windows-uwp* | @davidsh | UWP6.0 |

# https://octobox.io



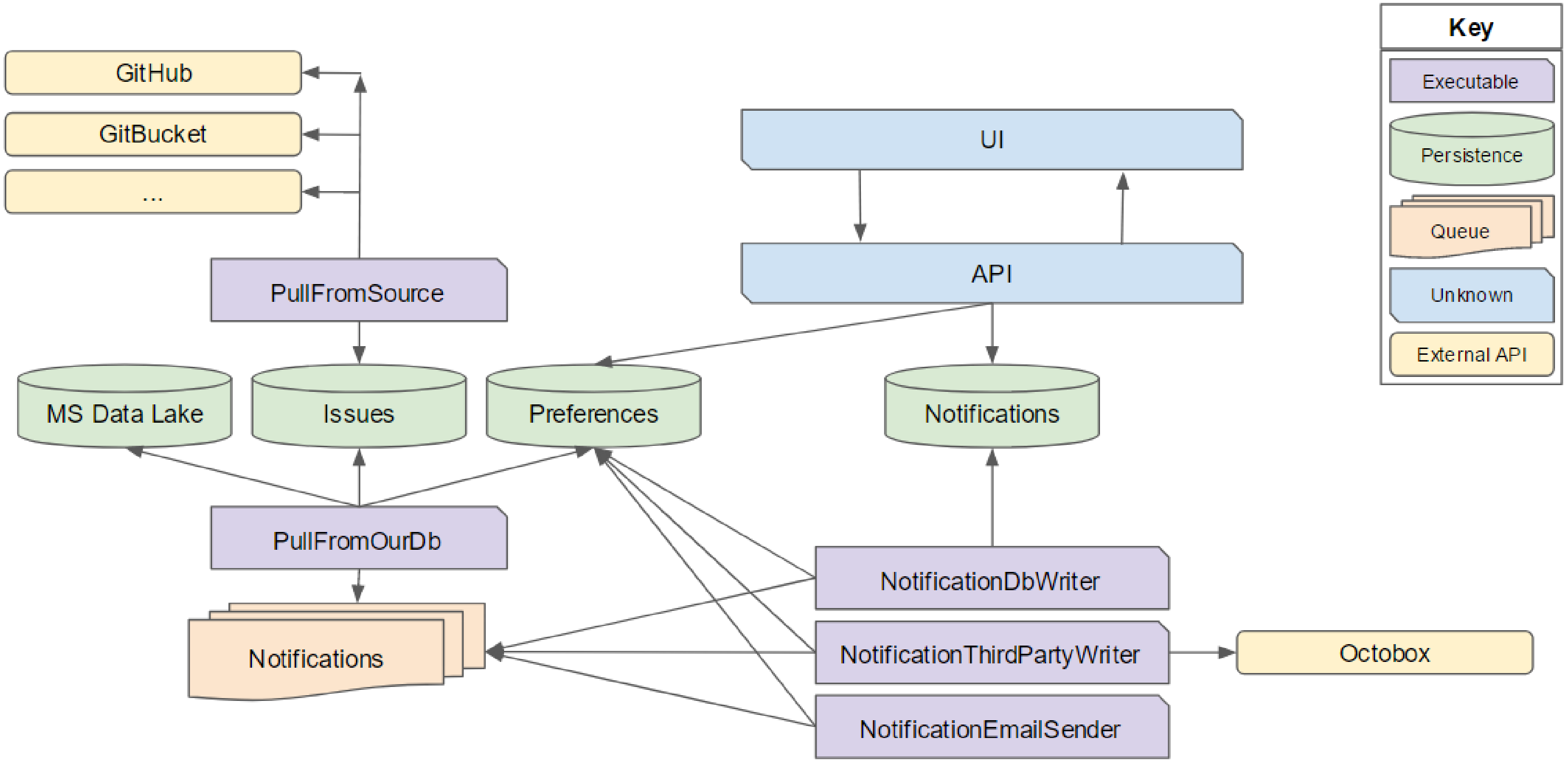
- Inbox 1073
- Archive
- Starred
- Read 401
- Unread 672
- Commit 1
- Issue 788
- Pull request 284
- Assign 6
- Author 20
- Comment 389
- Manual 52
- Mention 502
- Review requested 6
- State change 34
- Subscribed 64
- aspnet
  - EntityFramework 1
  - Home 1
  - Mvc 1
  - Security 1

Filter: ✕ dotnet/corefx    20

| | | |
|---|---|---|
| [HttpListener] Hang of System.Net.HttpListener.Tests in CI | Mention | 06 Jun 00:22 |
| GetHostByName_HostName and TryGetAddrInfo_HostName and BypassOnLocal_MatchesExpected fails with ExtendedSocketException : No such device or address /HostNotFound | Mention | 05 Jun 21:40 |
| Allow BitConverter.ToString to format without dashes | Mention | 05 Jun 16:33 |
| Add PriorityQueue<T> to Collections | Mention | 05 Jun 16:06 |
| Debug native build on OSX fails with link error - missing symbols | Comment | 05 Jun 15:37 |
| CoreFX contributor experience | Mention | 05 Jun 14:17 |
| Unix throws the wrong exception creating a FileStream | Mention | 05 Jun 12:01 |
| Request for Ubuntu 17.04 build | Mention | 05 Jun 11:55 |
| Inconsistent behaviour of build.sh -? (help) command on OSX | Mention | 05 Jun 11:50 |
| Implements #19826 and #4571: Adds DbConnection.ProviderFactory and DbProviderFactories | Mention | 05 Jun 10:55 |
| Proposal for adding System.Runtime.CompilerServices.IsRefLikeAttribute | Mention | 05 Jun 10:43 |
| Proposal: Add Overlaps(...) extension method for ReadOnlySpan<T> | Comment | 05 Jun 10:39 |
| XmlResolver: remove workaround for lack of Unix path support in Uri | Mention | 05 Jun 10:14 |
| System.Xml.XPath to support XPath 2, XPath 3 and their XSLT variants | Mention | 05 Jun 06:10 |
| System.Composition: Bug in Resolving Activators Due to the .NET Type System | Mention | 05 Jun 03:50 |
| Test failure: System.Net.Sockets.Tests.SendReceiveUdpClient/SendToRecvFromAsync_Datagram_UDP_UdpClient(loopbackAddress: 127.0.0.1) | Comment | 04 Jun 23:29 |
| WinHttp buffer supplied to the function was too small error should not be exposed | Mention | 04 Jun 23:17 |

# Next gen tooling

https://github.com/karelz/GitHubIssues



# Triage rules

1. One **area**-* label
2. Empty **Assignee**
3. Use **up-for-grabs** with next steps & complexity description
4. Set **Milestone** as 95% commitment
5. One **issue type** label
   - bug / enhancement / **api**-needs-work / documentation, etc.

6. Don't be afraid to say no, or close issues – just explain why and be polite
7. Don't be afraid to be wrong – just be flexible when new information appears

# Adding new APIs

- ~100 unique APIs in .NET Core now
  - Dictionary.GetValueOrDefault

- Decision what can go into CoreFX:
  - Is it existing BCL type?
  - Will it be used by CoreFX itself?
  - Can it live somewhere else? (e.g. PowerCollections)

- API approval process is slow by design
  - APIs are forever

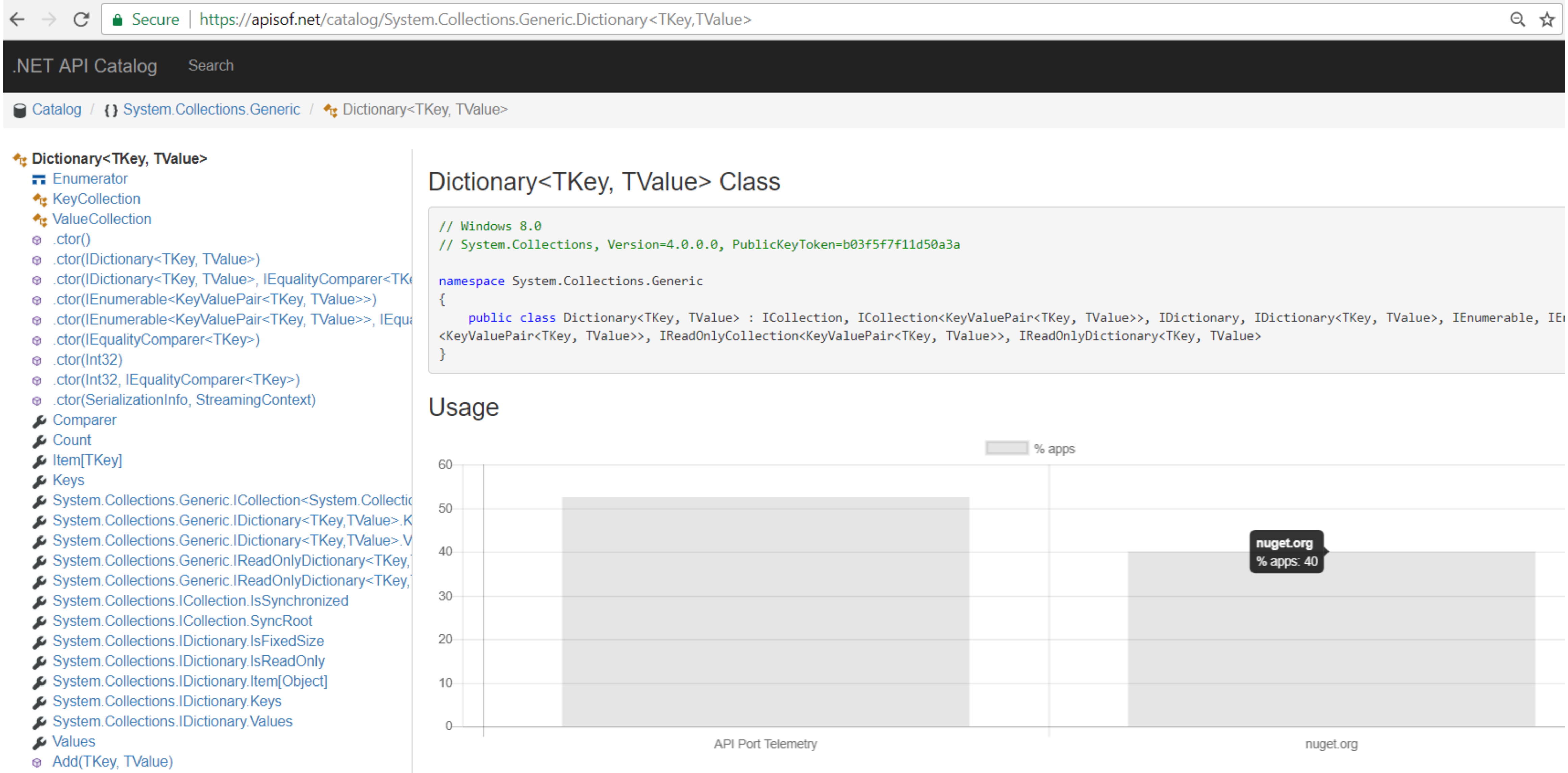
# Which code change to take?

- Performance changes
  - Performance trade-offs
  - Code complexity vs. value

- Compatibility breaking changes
  - vs. 1.x or .NET Framework / UWP / Mono
  - Is the value worth the risk someone will get hurt?
  - API usage

- Obsolete APIs
  - TreatAsWarnings = breaking change

# API usage information / telemetry



Secure | https://apisof.net/catalog/System.Collections.Generic.Dictionary<TKey,TValue>

.NET API Catalog    Search

Catalog / { } System.Collections.Generic / Dictionary<TKey, TValue>

**Dictionary<TKey, TValue>**
- Enumerator
- KeyCollection
- ValueCollection
- .ctor()
- .ctor(IDictionary<TKey, TValue>)
- .ctor(IDictionary<TKey, TValue>, IEqualityComparer<TKe
- .ctor(IEnumerable<KeyValuePair<TKey, TValue>>)
- .ctor(IEnumerable<KeyValuePair<TKey, TValue>>, IEqua
- .ctor(IEqualityComparer<TKey>)
- .ctor(Int32)
- .ctor(Int32, IEqualityComparer<TKey>)
- .ctor(SerializationInfo, StreamingContext)
- Comparer
- Count
- Item[TKey]
- Keys
- System.Collections.Generic.ICollection<System.Collecti
- System.Collections.Generic.IDictionary<TKey,TValue>.K
- System.Collections.Generic.IDictionary<TKey,TValue>.V
- System.Collections.Generic.IReadOnlyDictionary<TKey,
- System.Collections.Generic.IReadOnlyDictionary<TKey,
- System.Collections.ICollection.IsSynchronized
- System.Collections.ICollection.SyncRoot
- System.Collections.IDictionary.IsFixedSize
- System.Collections.IDictionary.IsReadOnly
- System.Collections.IDictionary.Item[Object]
- System.Collections.IDictionary.Keys
- System.Collections.IDictionary.Values
- Values
- Add(TKey, TValue)

## Dictionary<TKey, TValue> Class

```
// Windows 8.0
// System.Collections, Version=4.0.0.0, PublicKeyToken=b03f5f7f11d50a3a

namespace System.Collections.Generic
{
    public class Dictionary<TKey, TValue> : ICollection, ICollection<KeyValuePair<TKey, TValue>>, IDictionary, IDictionary<TKey, TValue>, IEnumerable, IEn
<KeyValuePair<TKey, TValue>>, IReadOnlyCollection<KeyValuePair<TKey, TValue>>, IReadOnlyDictionary<TKey, TValue>
}
```

## Usage

# API Compat tooling – Demo

Motivation:

- Obsolete APIs
- Differences between platforms

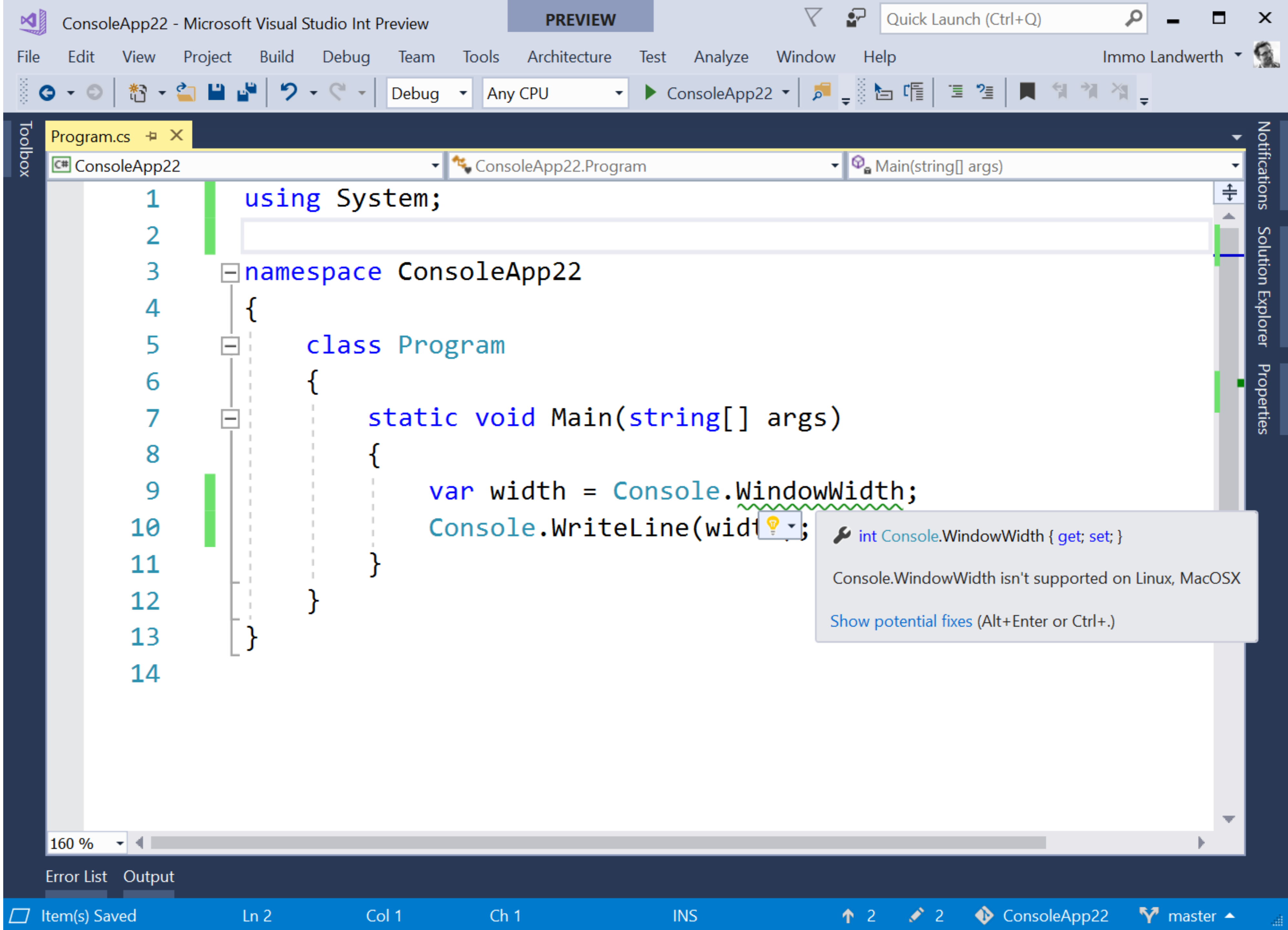
https://github.com/dotnet/platform-compat

Announced on .NET Blog 10/31 – Introducing API Analyzer

- Incl. 5 min video
- Seeking feedback

APIs that throw PNSE

Platform Not Supported Exception



```
using System;

namespace ConsoleApp22
{
    class Program
    {
        static void Main(string[] args)
        {
            var width = Console.WindowWidth;
            Console.WriteLine(widt   ;
        }
    }
}
```

int Console.WindowWidth { get; set; }

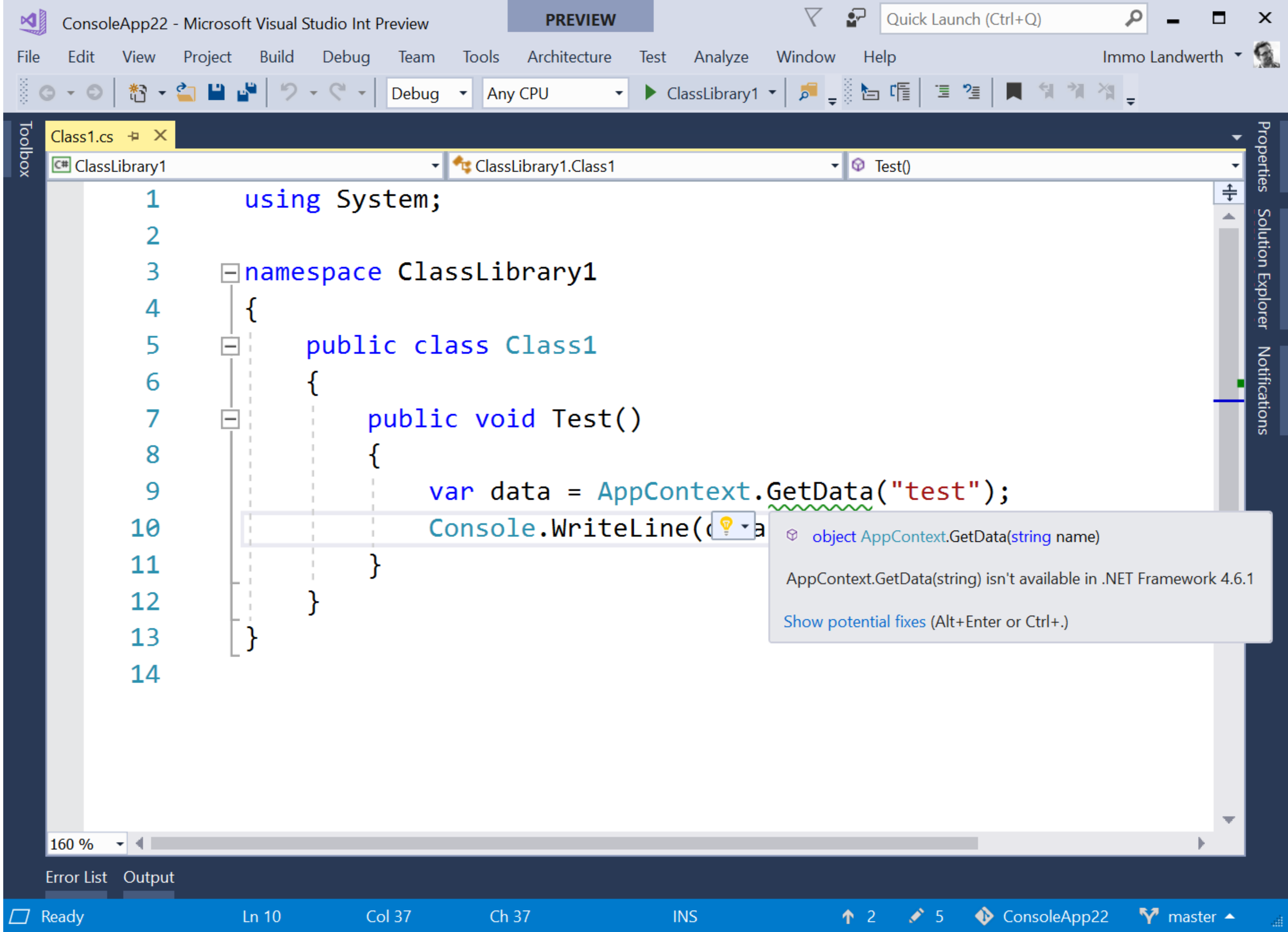
Console.WindowWidth isn't supported on Linux, MacOSX

Show potential fixes (Alt+Enter or Ctrl+.)

.NET Standard 2.0 APIs

missing from

.NET Framework 4.6.1



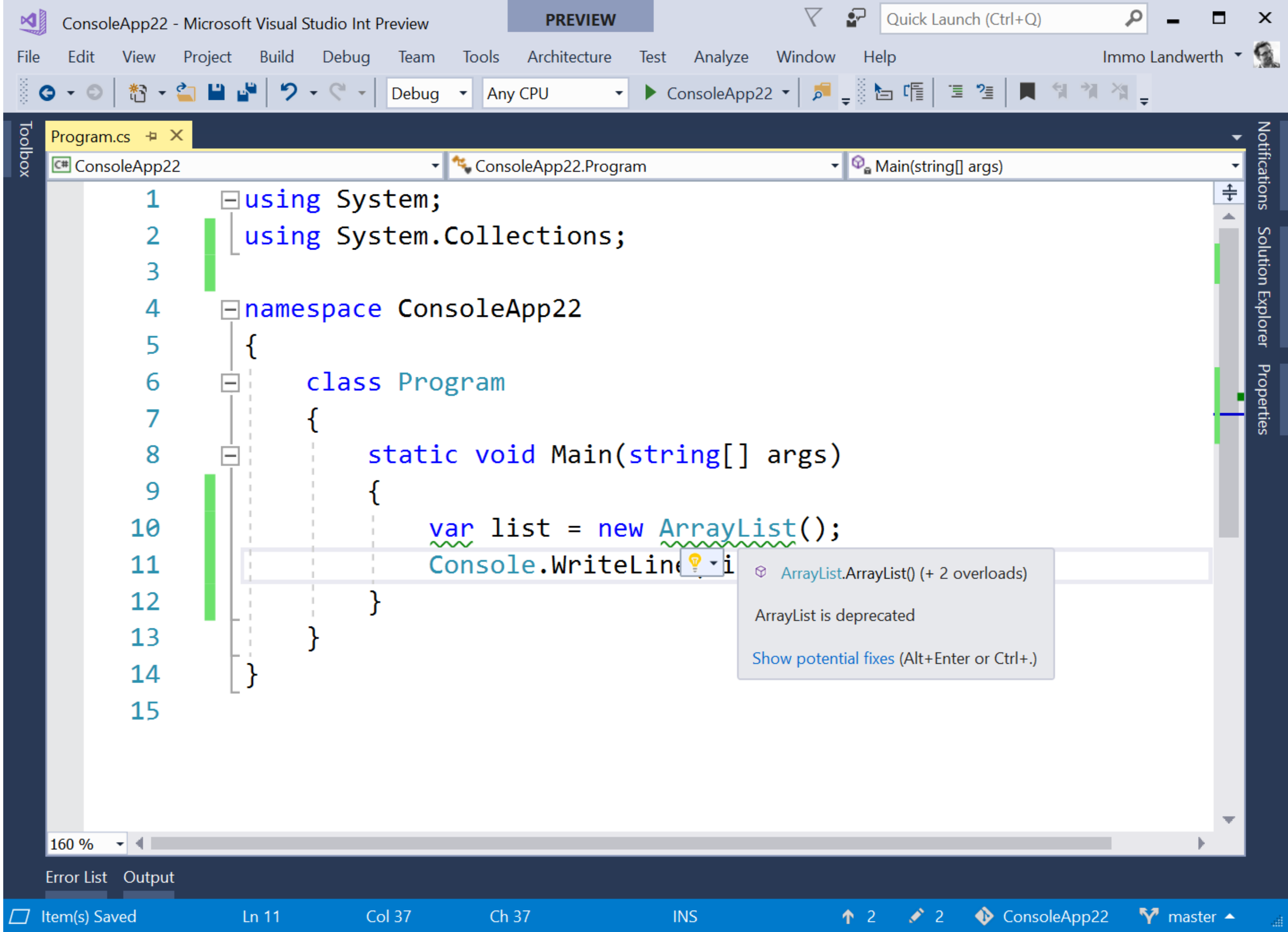
```
using System;

namespace ClassLibrary1
{
    public class Class1
    {
        public void Test()
        {
            var data = AppContext.GetData("test");
            Console.WriteLine(data
        }
    }
}
```

object AppContext.GetData(string name)

AppContext.GetData(string) isn't available in .NET Framework 4.6.1

Show potential fixes (Alt+Enter or Ctrl+.)

# Usage of deprecated APIs



```
using System;
using System.Collections;

namespace ConsoleApp22
{
    class Program
    {
        static void Main(string[] args)
        {
            var list = new ArrayList();
            Console.WriteLine i
        }
    }
}
```

ArrayList.ArrayList() (+ 2 overloads)

ArrayList is deprecated

Show potential fixes (Alt+Enter or Ctrl+.)

# Team Culture changes & challenges

- Discussions in the open
- Onboarding new contributors (docs, tooling, attitude)
- Communication – being polite, welcoming, patient; making it constructive
    - Unclear language (e.g. ESL = English as Second Language)
    - Missing/incomplete/complex repro steps
    - Frustration
    - Abuse (aggressiveness)
    - Having your own opinion vs. driving consensus

# Involving Contributors & Community

- CoreFX Contributors standups - https://github.com/dotnet/corefx-standup
  - Goals:
    - Bi-directional discussion
    - Identify blind spots
  - Mechanics:
    - Live streaming, chat open to everyone
    - Voting on topics upfront
    - ~monthly cadence
- API design reviews - https://github.com/dotnet/apireviews
  - Weekly cadence
  - Live streaming, chat open to everyone
- Future:
  - .NET Technology Q&A sessions
  - Monthly alpha versions for close community
  - More ideas: https://karelz.github.io

# Thank you

Seeking:

- Feedback
- Alternative / existing solutions
- Ideas (incl. above & beyond CoreFX, Microsoft)

Contacts:

karelz@microsoft.com

@karelz

@ziki_cz