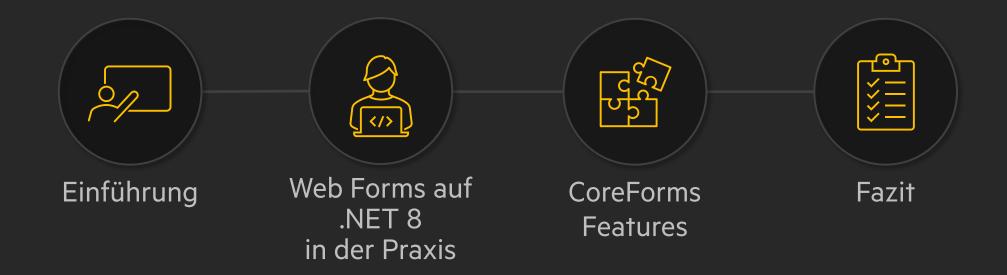




Retargeting einer Anwendung mit ASP.NET-Web Forms auf .NET 8



Agenda





Einführung Herausforderung Web Forms

Am Anfang stand das .NET Framework

Ausgangslage

- 2000 2015: Viele Fachanwendungen wurden mit Web Forms auf dem .NET Framework gebaut
- 2015 dato: .NET löst das .NET Framework als bevorzugte Entwicklungsplattform ab
- Status Quo: ASP.NET Web Forms ist auf .NET 8 nicht verfügbar



Web Forms Applikationen komplex & organisch gewachsen

Der Anwendungs-Code ...

- ... ist mit Web Forms stark verflochten
- ... erfüllt ein gewachsenes Set an komplexen Anforderungen

Jahrzehnte von Entwicklungsarbeit und Millionen Codezeilen sind hier im Einsatz!



Wie könnte eine Portierung aussehen? Neuer Web-Stack & Laufzeitumgebung?

Big Bang Migration?

- Hohe Investionen an Zeit und Geld notwendig
- Know-How und Expertise notwendig
- Erfolgsaussichten dennoch ungewiss

Eine Portierung auf eigene Faust wird daher viele schlaflose Nächte verursachen!



Web Forms-Portierung Ist das denn überhaupt notwendig?

Gründe für Portierung

- Komplexe Anwendungen stehen auf dem Spiel
- Entwickler:innen lieben die neueste Technologie
- Web Forms ist aber veraltet

Es wird immer schwieriger, junge Entwickler:innen für Web Forms zu begeistern!



Eine einfache Frage

Wieso kein Web Forms für .NET 8?

Seriously!!!

Wieso kein Web Forms für .NET 8?

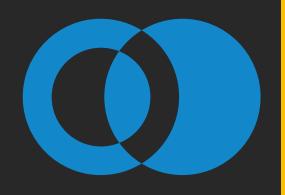
Es geht also doch: Web Forms für .Net 8!

Facts

- 2019 fragte man sich bei RUBICON: "Wieso kein Web Forms für .NET 8?"
- Mehrere Jahre Entwicklungsarbeit später ...
- 2023 hat RUBICON schließlich eine Antwort: "Web Forms läuft auf .NET 8!!!"

Die Lösung heißt CoreForms und ist voll funktionsfähig!





Web Forms auf .NET 8 in der Praxis

Running IISExpress...

Nur drei Schritte Web Forms auf .NET 8 ausführen

Drei Schritte zum Erfolg

- Abhängigkeiten aktualisieren:
 So viele Abhängigkeiten wie möglich aktualisieren
- 2. Projektdateien abändern: Abändern der csproj-Datei & web.config
- 3. Starten:
 Anwendung auf .NET 8 ausführen



Beispiel-Demo (Wingtip Toys): https://github.com/CoreForms/WingtipToys-Demo



Nur drei Schritte <u>Web Forms auf .NET 8 ausführen</u>

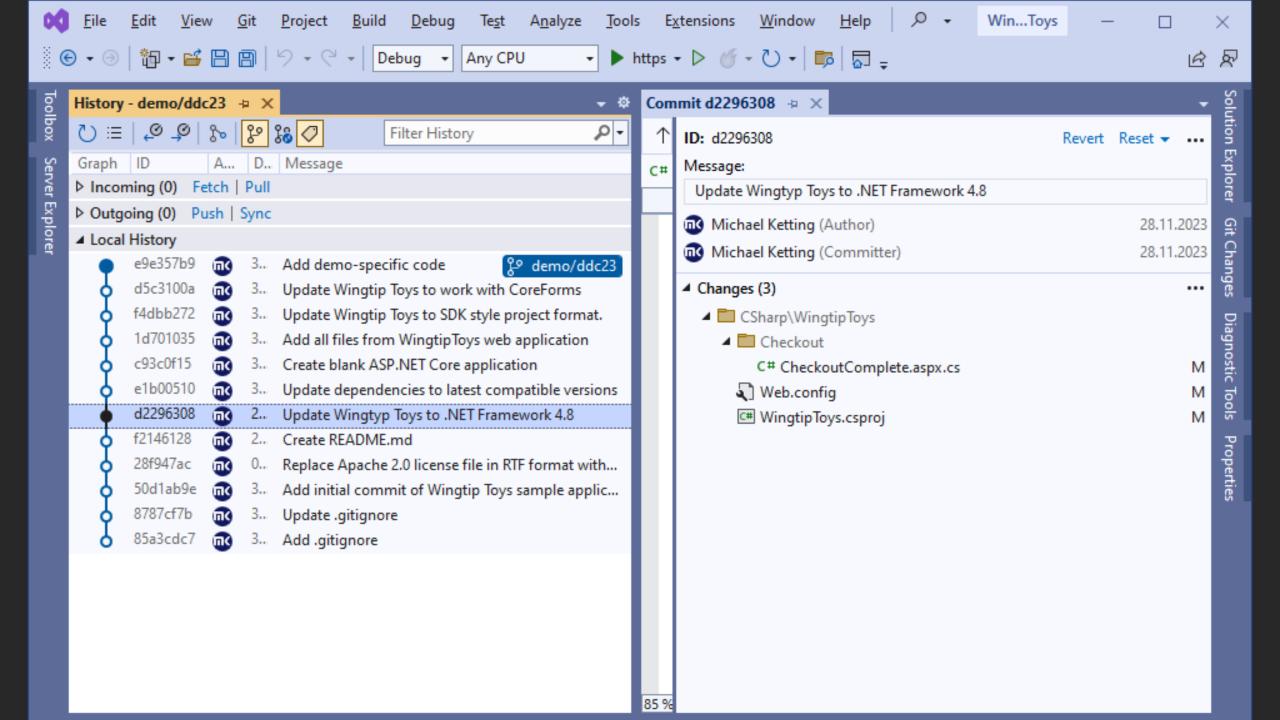
Drei Schritte zum Erfolg

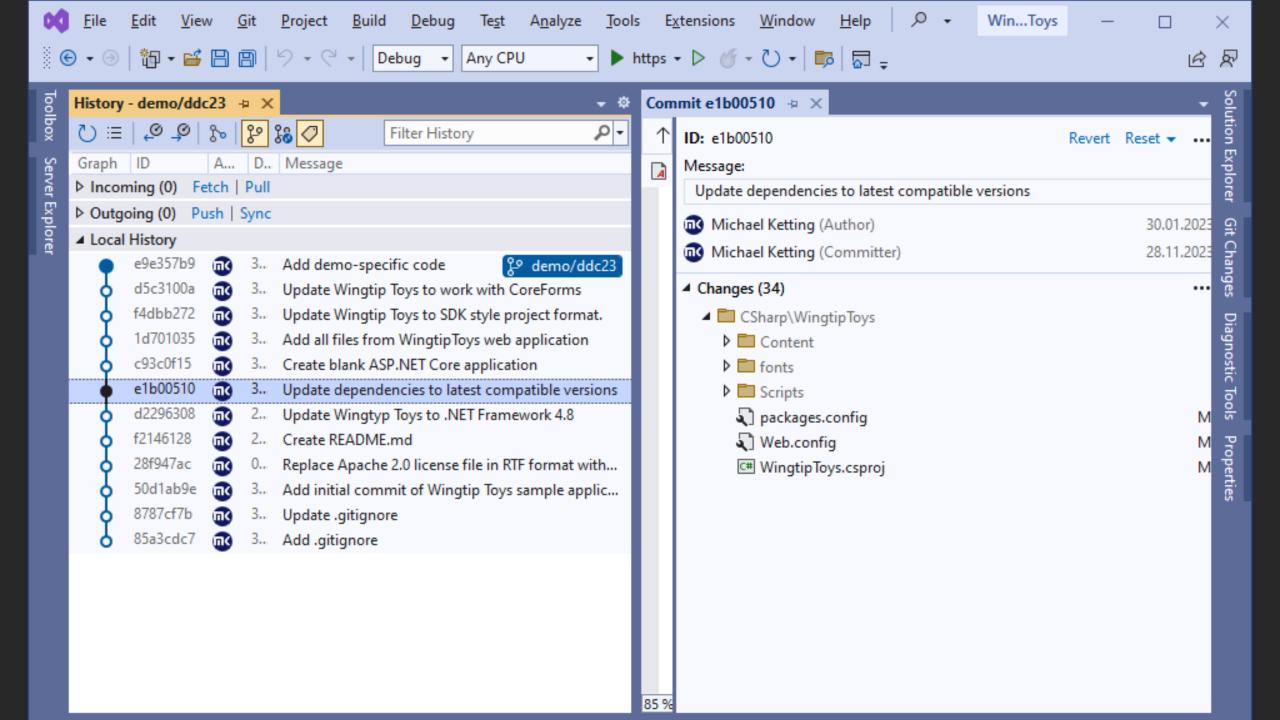
1. Abhängigkeiten aktualisieren:

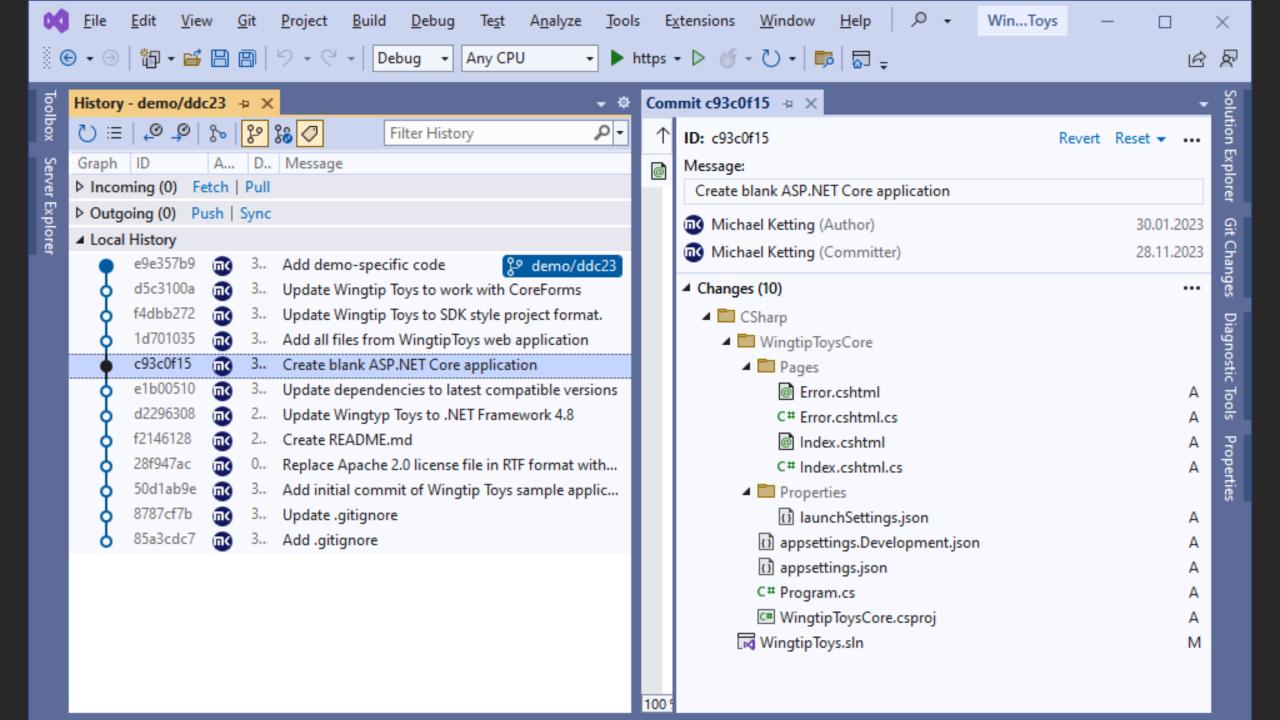


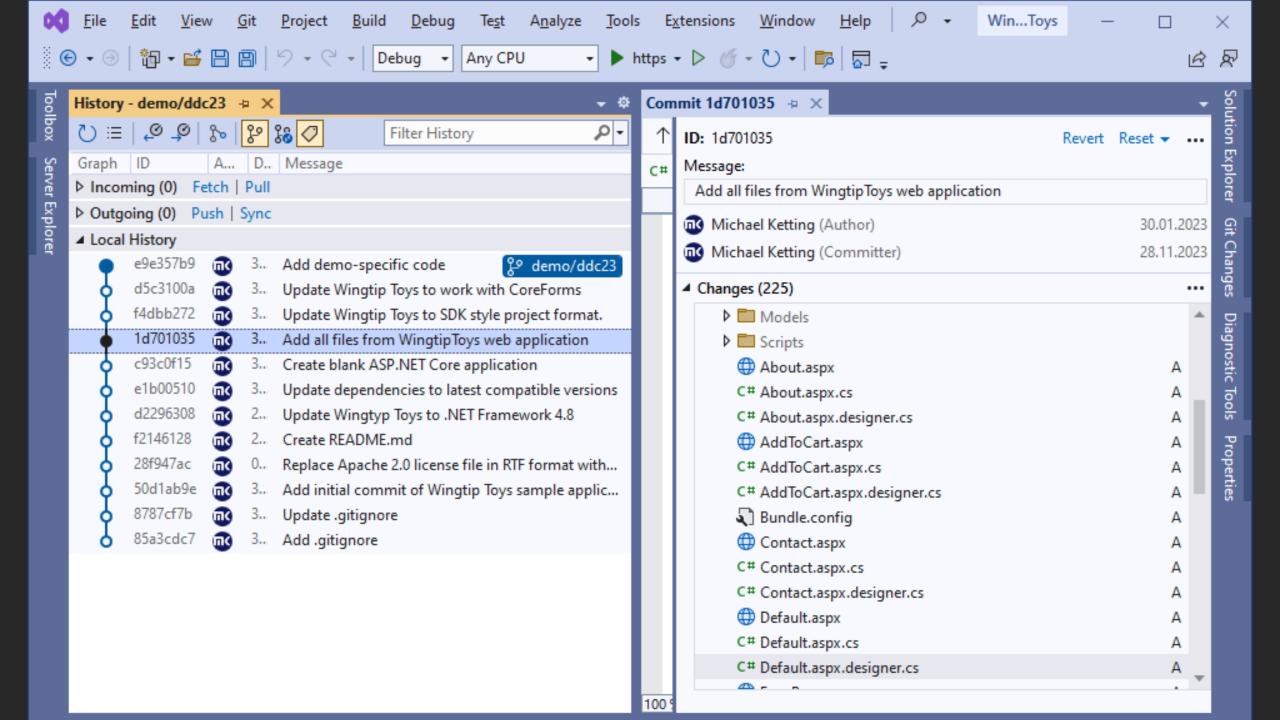
Beispiel-Demo (Wingtip Toys):

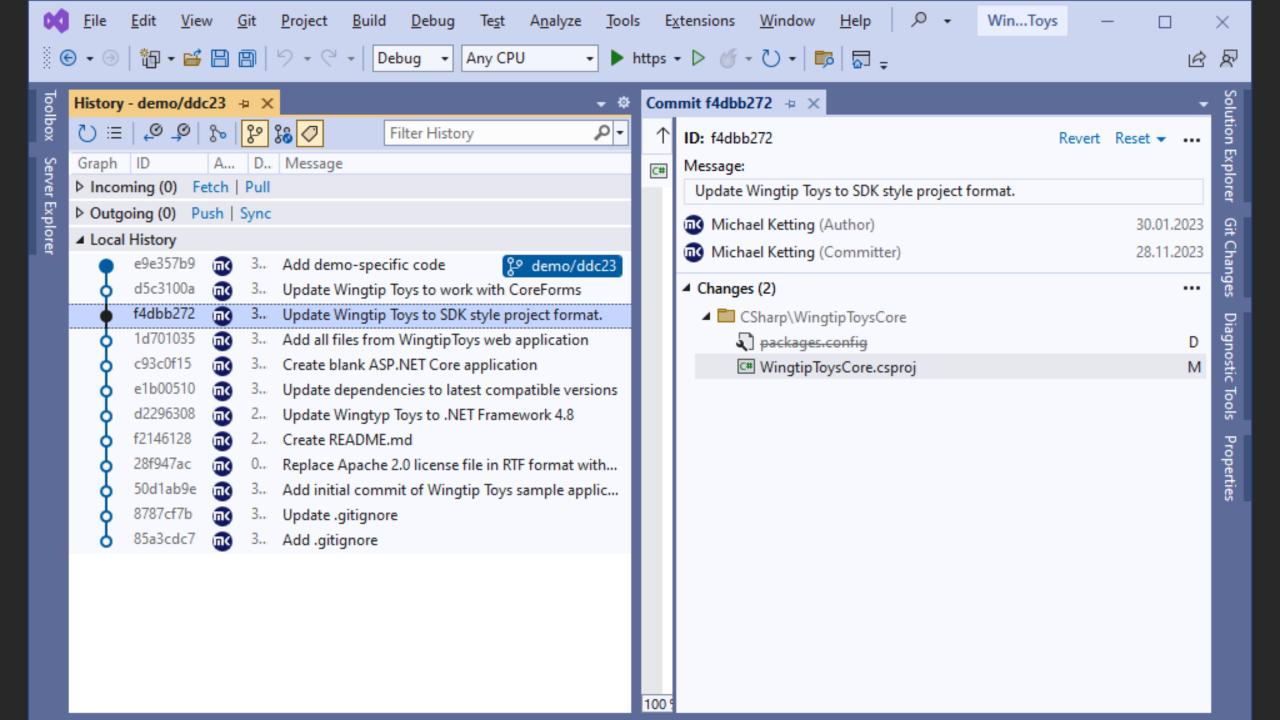
https://github.com/CoreForms/WingtipToys-Demo

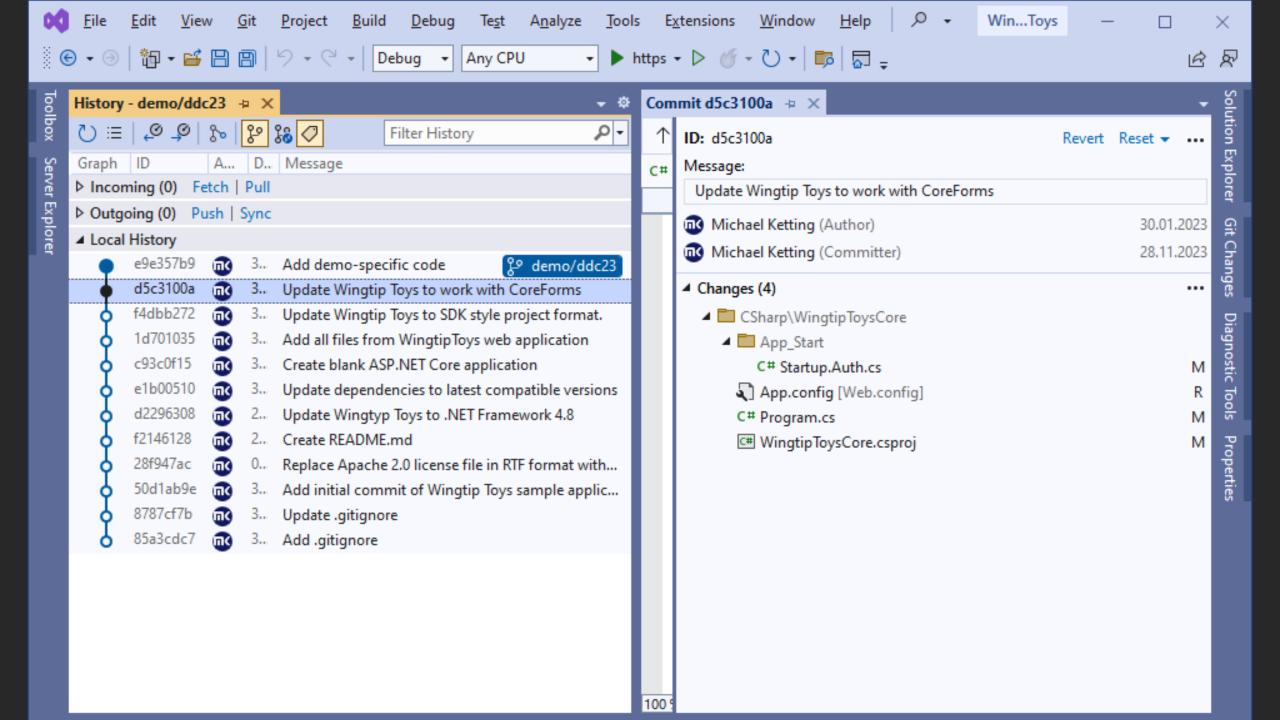


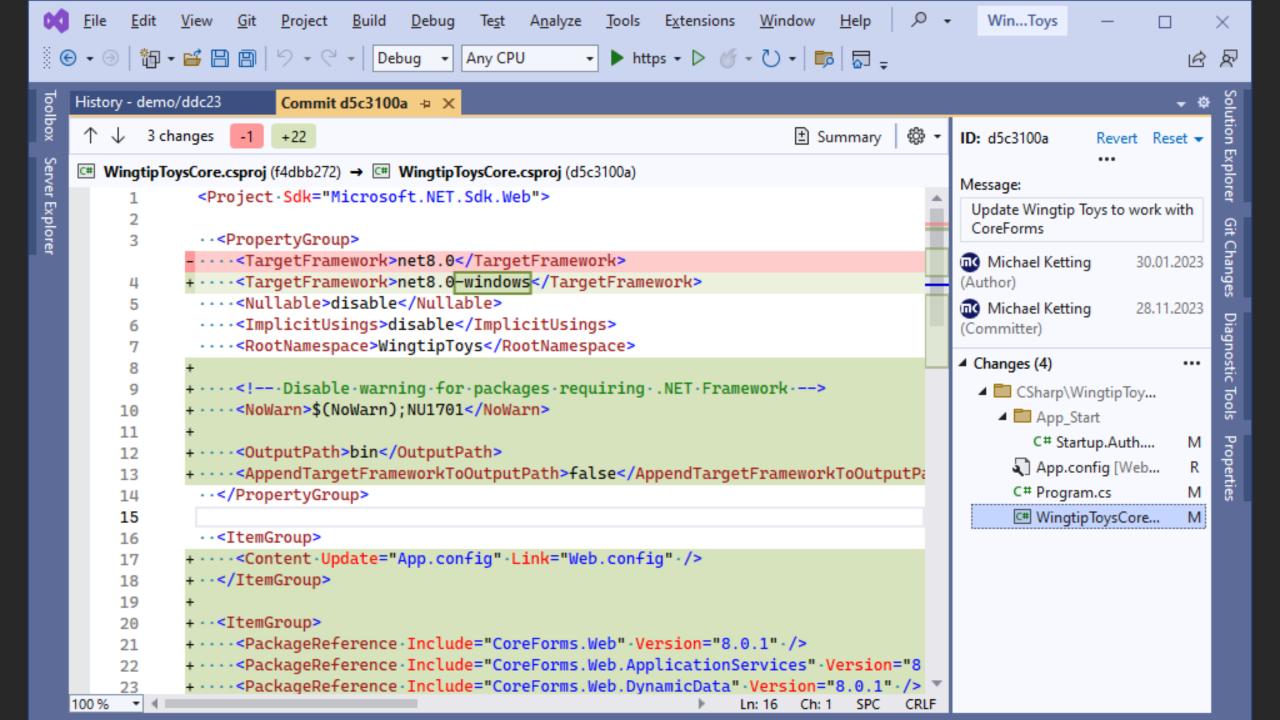


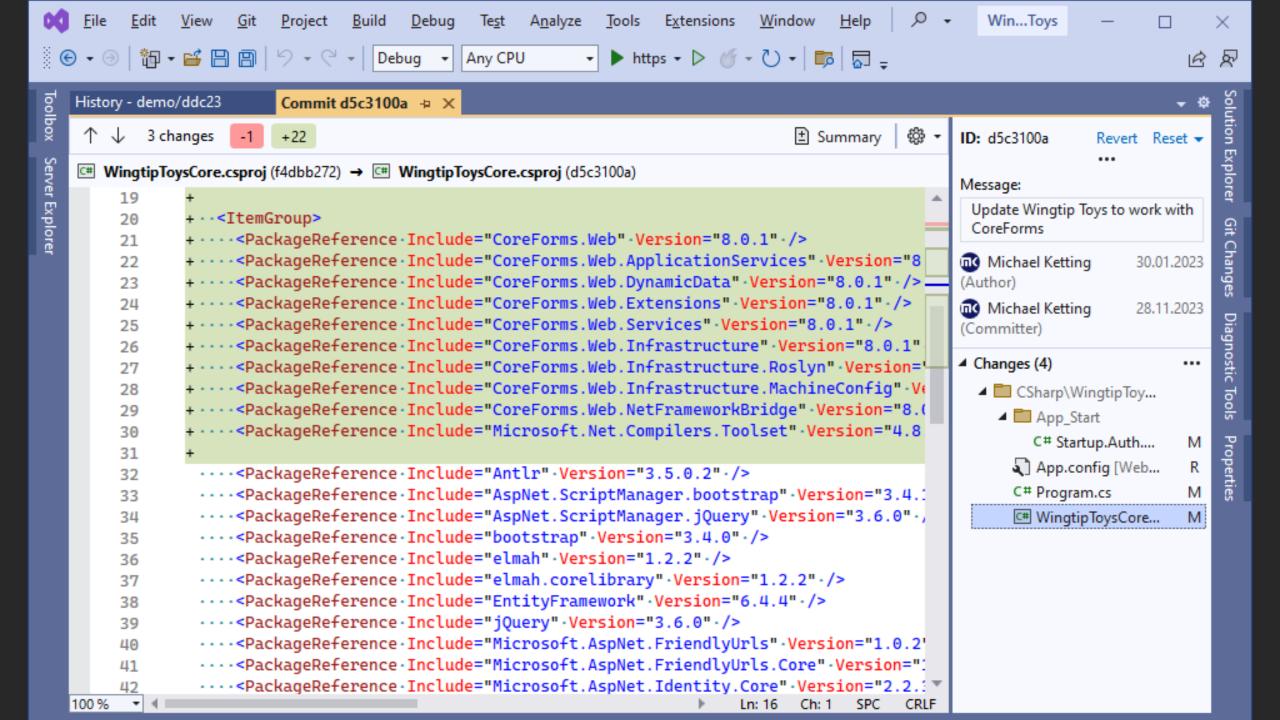


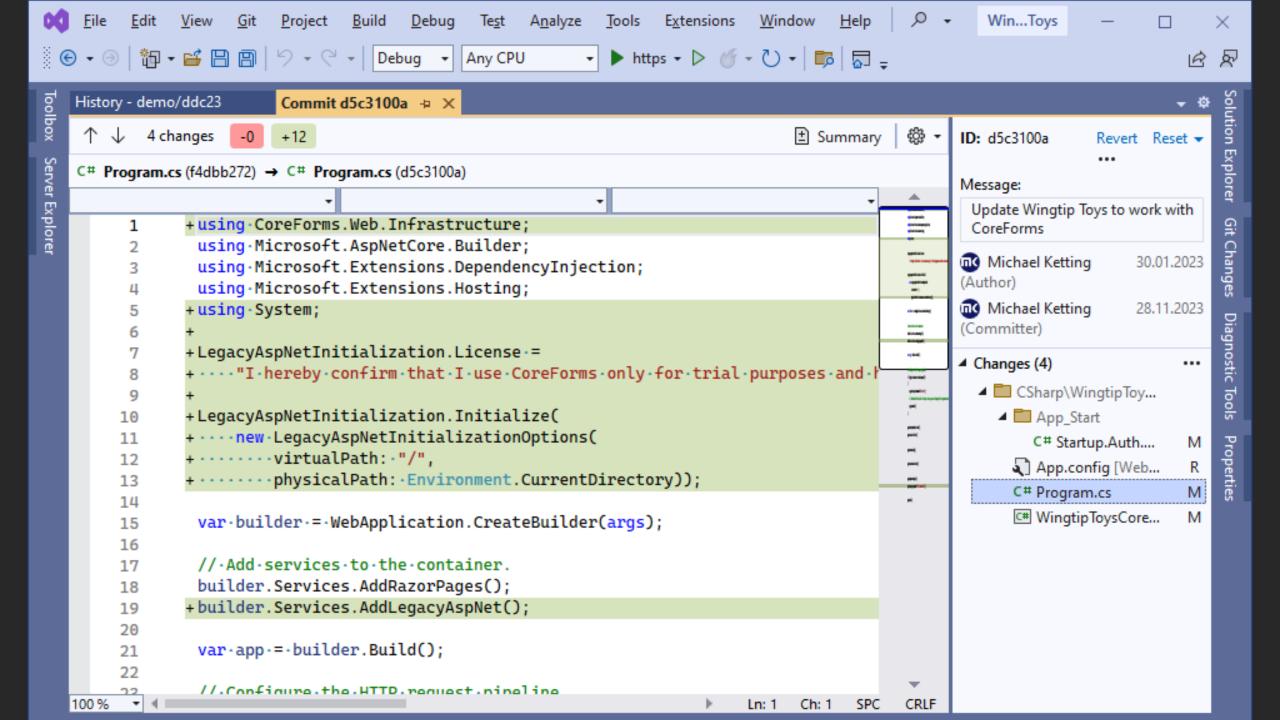


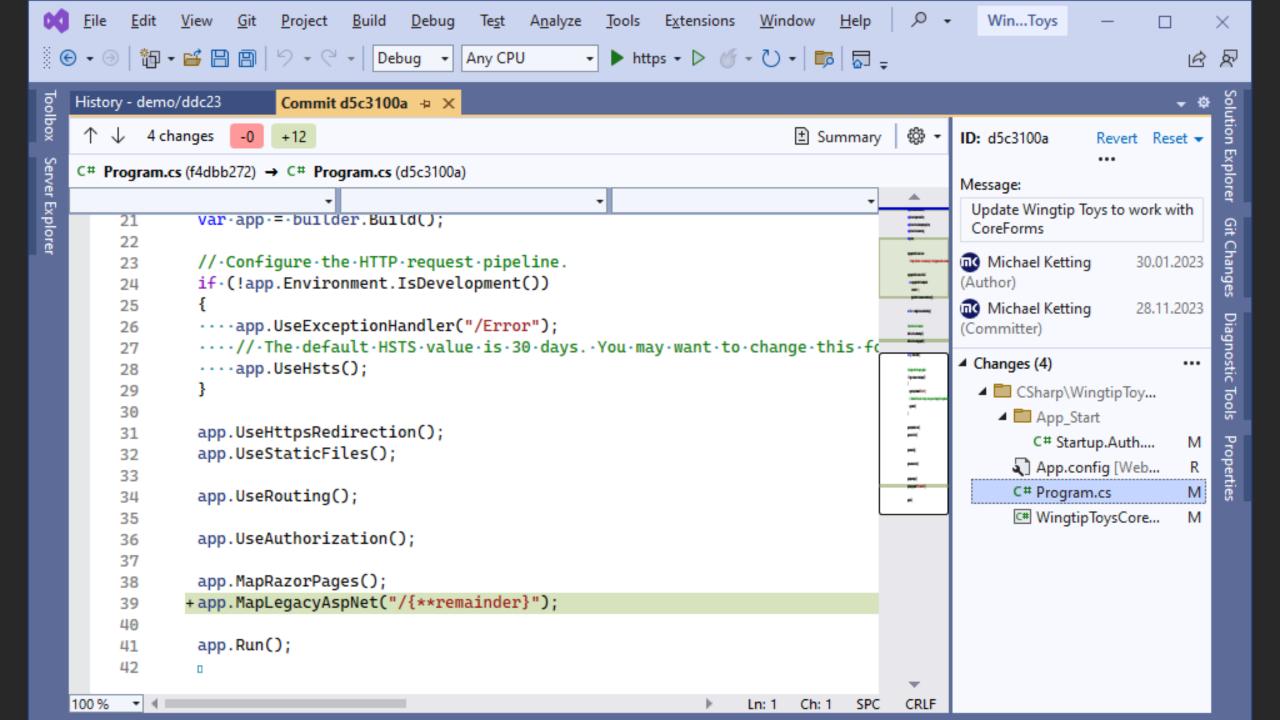


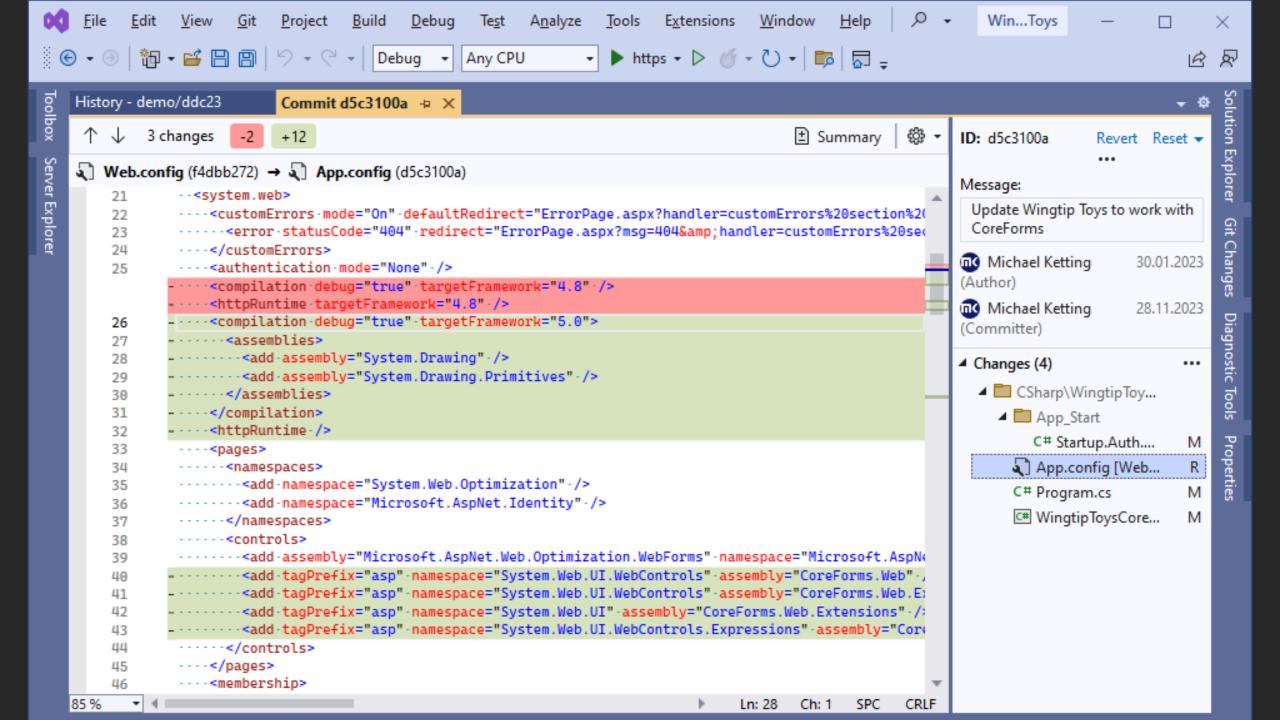


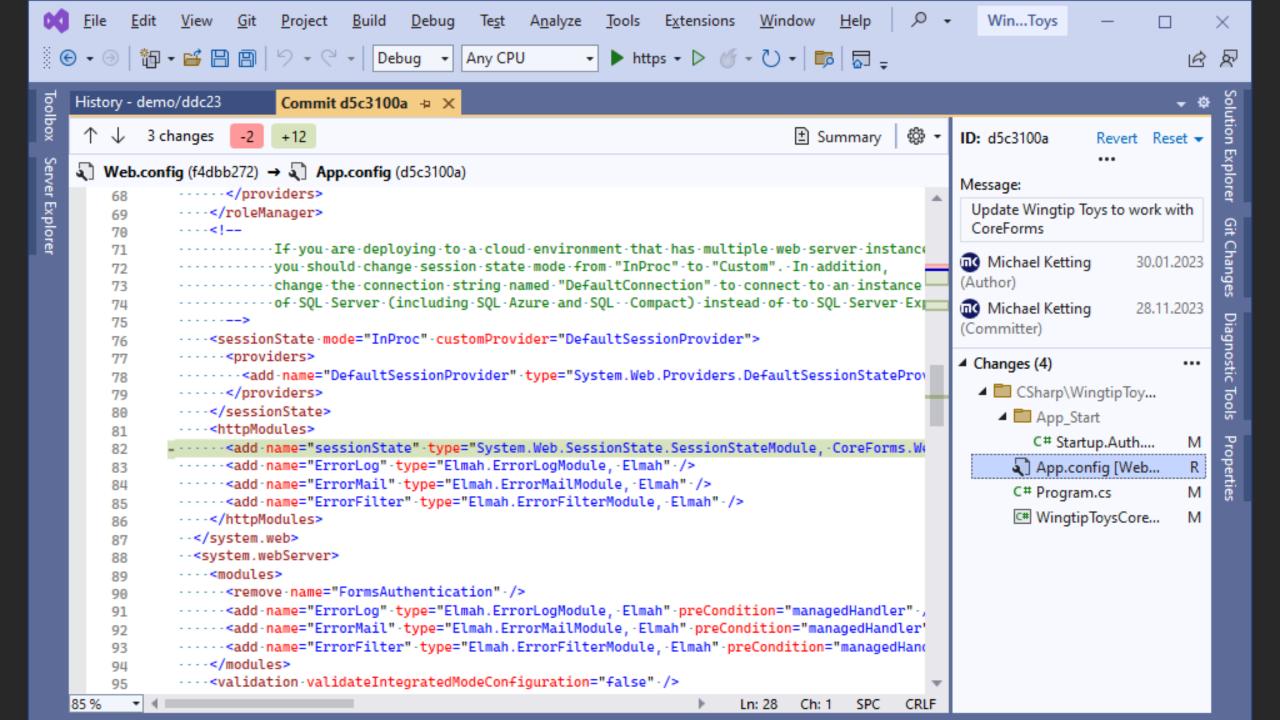












Running Kestrel...

Web Forms auf .NET 8 ausführen

CoreForms Pakete in der csproj-Datei eintragen: (Verfügbar auf NuGet.org)

Web Forms auf .NET 8 ausführen

web.config ändern:

- Auf app.config umbenennen
- Ziel des ASP.NET Compiler auf "5.0" einstellen

```
<compilation targetFramework="5.0">
```

Notwendige Tag-Präfixe hinzufügen

Session-State-Modul registrieren

```
<add name="sessionState"

type="System.Web.SessionState.SessionStateModule, CoreForms.Web" />
```

Web Forms auf .NET 8 ausführen

Start-Logik erweitern:

CoreForms konfigurieren

```
LegacyAspNetInitialization.License = "...";
LegacyAspNetInitialization.Initialize(
    new ("/", Environment.CurrentDirectory));
```

CoreForms in ASP.NET Core registrieren

```
var builder = WebApplication.CreateBuilder();
builder.Services.AddLegacyAspNet();
```

ASPX-Dateien registrieren

```
var app = builder.Build();
app.MapLegacyAspNet("/{**remainder}");
```



CoreForms Easy to use

Portierung leicht gemacht CoreForms bietet zahlreiche Features



Session Management

Das Teilen von Session-Informationen leicht gemacht



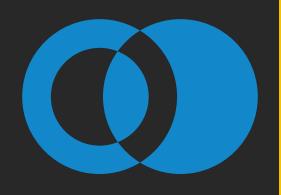
Einheitliche Authentifizierung

Drei Optionen für das Arbeiten mit Authentication-Cookies



Unterstützung von Drittanbietern

Einfache Verwendung von Drittanbieter-Komponenten



Session Management

CoreForms Features Session Management

Überblick

- CoreForms integriert die Session-Lösung von ASP.NET Web Forms
- Zur einfacheren Migration einer Anwendung können die Session-Inhalte auch in ASP.NET Core verwendet werden.



CoreForms Features ObjectSession

Gemeinsam genutzte Session-Inhalte

```
var builder = WebApplication.CreateBuilder();
builder.Services.AddSession();
builder.Services.AddInMemoryObjectSession();
var app = builder.Build();
app.UseSession();
app.UseRouting();
app.UseObjectSession();
```



Beispiel-Demo (ObjectSession):

https://github.com/CoreForms/demo (Project: Demo.ObjectSession)

CoreForms Features ObjectSession

Gemeinsam genutzte Session-Inhalte



Beispiel-Demo (ObjectSession):

https://github.com/CoreForms/demo

(Project: Demo.ObjectSession)

(Project: Demo.ObjectSession)

Running Sessions...



Einheitliche Authentifizierung

CoreForms Features Einheitliche Authentifizierung

Ablauf einer typischen Authentifizierung

- Zugangsdaten werden an die Anwendung übertragen
- 2. Authentifizierungs-Cookie wird ausgestellt, um den Benutzer bei nachfolgenden Anfragen zu identifizieren
- 3. Benutzer-Prinzipal wird auf Basis des Authentifizierungs-Cookies erstellt



CoreForms Features Verwendung des Authentifizierungs-Cookies

Drei Optionen

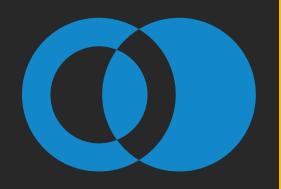
- Isolated (Standard)
 Getrennte Authentifizierung in ASP.NET und ASP.NET Core
- Shared
 Einheitliche Authentifizierung in ASP.NET und ASP.NET Core
- 3. Shared Read-Only
 Der Benutzer-Prinzipal kann nur in ASP.NET Core geändert werden



CoreForms Features Konfiguration der Authentifizierung

Beispiel für die einheitliche Authentifizierung

```
builder.Services.AddLegacyAspNet (
b => {
   b.AuthenticationSharing =
        AspNetAuthenticationSharing.Shared;
});
```



Unterstützung von Drittanbietern

CoreForms Features Unterstützung von Drittanbietern

Kompatibilität mit Hilfe von NuGet-Paket

- Drittanbieter-Komponenten gibt es nur für ASP.NET Web Forms
- CoreForms.Web.NetFrameworkBridge ermöglicht die Kompatibilität mit CoreForms



CoreForms Features Unterstützung von Drittanbietern

Kompatibilität mit Hilfe von NuGet-Paket

```
<PackageReference Include="CoreForms.Web.NetFrameworkBridge"</pre>
                   Version="..." />
```



Bar Beispiel-Demo (Telerik Components): https://github.com/CoreForms/demo (Project: Demo.Telerik)

CoreForms Features Unterstützung von Drittanbietern

Kompatibilität mit Hilfe von NuGet-Paket



Beispiel-Demo (Telerik Components):

https://github.com/CoreForms/demo

(Project: Demo.Telerik)

Running Telerik...



Fazit CoreForms ist die Lösung

Fazit CoreForms- funktioniert das denn?

Status Quo

- Produktiv
 CoreForms ist aktuell bereits vielfach produktiv im Einsatz
- 2. Bewährt
 Anwendungen mit tausenden Code-Zeilen
 nutzen CoreForms bereits erfolgreich
- 3. Zukunftssicher RUBICON Expert:innen arbeiten aktiv an offenen Fragen oder Spezialthemen



Fazit CoreForms, der smarte Weg für Web Forms

Mehr Informationen



Website Besuche https://coreforms.net



Trial & Demo
Verfügbar auf nuget.org https://www.nuget.org/packages/CoreForms.Web/

Fazit CoreForms, der smarte Weg für Web Forms

Mehr Informationen



Website

https://coreforms.net

Fazit CoreForms, der smarte Weg für Web Forms

Mehr Informationen



Trial & Demo:

https://www.nuget.org/packages/CoreForms.Web/

Ressourcen auf GitHub



即黎 Dokumentation

https://github.com/CoreForms/docs

Demo



陽里 Link 1:

https://github.com/CoreForms/demo



https://github.com/CoreForms/WingtipToys-Demo

Ressourcen auf GitHub



Dokumentation https://github.com/CoreForms/docs

Ressourcen auf GitHub



Demo 1

https://github.com/CoreForms/demo

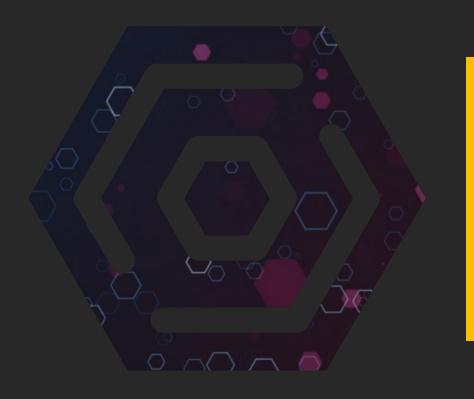
Ressourcen auf GitHub



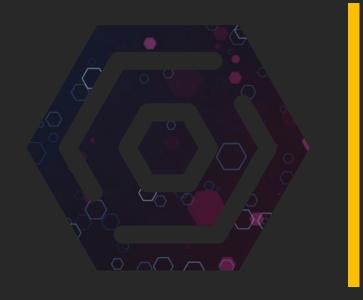
Demo 2 https://github.com/CoreForms/WingtipToys-Demo



Alle Links auf einen Blick: https://coreforms.net/#event-ddc23



Fragen,
Wünsche,
Anregungen?





Ihre persönlichen Ansprechpartner, kontaktieren Sie uns!