The .NET Execution Environment (DNX) and its Ecosystem

Darren Hale

Platinum Sponsors

SyllogisTeks

OAKWOOD
Powering Transformation. Together.

POLARIS SOLUTIONS

Maritz CX

EMERSON
Climate Technologies

Gold Sponsors

World Wide Technology
asynchrony labs

CONCERO TECHNOLOGY GROUP

Sovereign Technologies

VantageLinks

Daugherty
BUSINESS SOLUTIONS

cosentry

signature CONSULTANTS

UNISYS

KEYHOLE SOFTWARE

WORKFORCE CENTER
SAINT LOUIS UNIVERSITY

stackify

ADAPTIVE SOLUTIONS GROUP

ArchitectNow
Designing Tomorrow's Solutions Today

SWANK MOTION PICTURES, INC.
Tim Swank, Chairman

Robert Half Technology

Perficient

USI Ungerboeck Software
INTERNATIONAL

Microsoft

Byrne Software
TECHNOLOGIES, INC.

mortgage re:turns

pluralsight
hardcore dev and IT training

EQUIFAX

twilio

Covenant
TECHNOLOGY PARTNERS

TEKsystems
Our people make IT possible

Silver Sponsors

Amitech Solutions

LogicNP Software
The power of Components

STL TECH TALK

jacobson staffing

WITSINC

redgate
ingeniously simple

ansira

Who Am I?

Who: Darren Hale

Where: Clearent

What: Manage software development team working on back-end processing.

How: @darrenhale

darren@clearent.com

Soccer Fan

Why Are We Here?

Analyze the upcoming game:

USMNT vs. St. Vincent and the Grenadines

Why Are We REALLY Here?

New and Improved ASP.NET 5 and DNX

Disclaimer

I'm really excited about it.

I'm NOT an expert!

Why Re-Invent ASP?

Trying to make ASP.NET leaner, modular, and cross-platform.

Making it easier to go to the cloud.

Competing with other frameworks.

Competition

Why compete with other stacks?

Market Share

Create Better Product for Devs

Better compete in cloud space.

OSS

What is "Cloud Enablement"?

Disparate apps can run in the same environment.

It's all about isolation.

Virtualization isolates servers.

Application isolation necessary for efficient use of resources.

Why Cloud-Enablement Matters

Scale

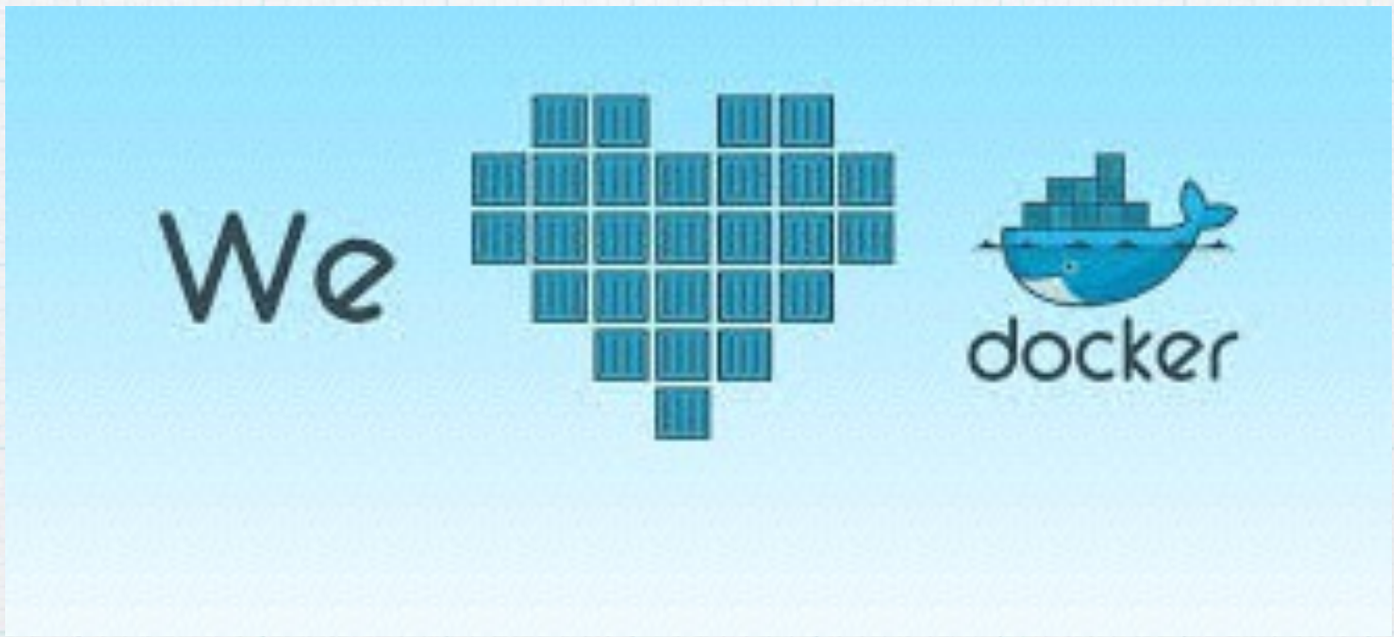
Security

Cost

Improved DevOps

Continuous Delivery

Why Do I Care?



What is DNX?

The DNX (.NET Execution Environment) contains the code required to bootstrap and run an application, including the compilation system, SDK tools, and the native CLR hosts.

Analogy

AngularJS

ExpressJS

NodeJS

MongoDB

AngularJS

MVC*

DNX

Mono

CoreCLR

SQL Server

* MVC
Razor
WebAPI
etc.

Ecosystem

SignalR

Kestrel

MVC

Razor

DNX

EntityFramework

Dependency Injection

DNVM

Web Hooks

Scaffolding

Logging

WebSockets

Identity

Configuration

homebrew-dnx

CORS

Options

Caching

aspnet-docker

Pay as You Go

Keep deployment size small.

Limit maintenance burden.

Reduce attack surface.

Side-by-Side Deployment

DNX enables apps on different versions to be deployed side-by-side.

Facilitated by dependencies being broken out.

Eliminates underlying environment as part of app deployment.

Main Tools

DNVM - .NET Version Manager

Utility used to manage DNX versions installed.

DNU - .NET Utility

Responsible for package maintenance and "compilation".

DNX - .NET Execution

Runs commands defined for each project.

Kestrel - Hosting Engine

Executes web processes for ASP.NET 5 applications.

Getting Started

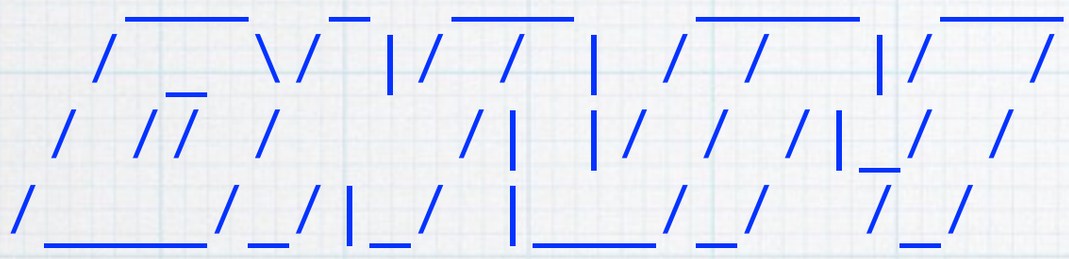
BEWARE!

Working with this software can hurt. Breaking changes are made often! Apps will break moving from one release to the next. THIS IS BLEEDING EDGE!

Easy Windows Way

Install Visual Studio 2015!

Non-Windows Way



Install DNVM!

* General installation instructions: https://github.com/aspnet/home

* Installation on OS X: https://docs.asp.net/en/latest/getting-started/installing-on-mac.html

* Installation on Linux: https://github.com/aspnet/Home/blob/dev/GettingStartedDeb.md (Navigate to "Getting Started with ASP.NET 5 and Linux")

* Can be accessed by installing Visual Studio 2015 and the ASP.NET 5 Templates.

Quick Aside: Yeoman

"The web's scaffolding tool for modern web apps."

Utility to help kickstart new projects, much like File > New > Project in Visual Studio. It uses generators to produce file templates.

The community has embraced Yeoman as a way to provide application templates to make it easier to get up and going with ASP.NET 5.

Demo

project.json

* This is the "definition" of a project.
* It is a plain old Javascript file (JSON).
* It is NOT tied to Visual Studio. Visual Studio can work with these, but they can be modified outside of Visual Studio. (This is in stark contrast to the .csproj files.)
* Defines everything for the project, including dependencies.
* Defines "commands" for interacting with the project.
* Can define scripts to run and when to run them (i.e. pre-publish).
* Lists supported frameworks.
* Much more. Full spec at https://github.com/aspnet/Home/wiki/Project.json-file

Project File Comparison

project.json

```
{
  "webroot": "wwwroot",
  "version": "1.0.0-*",

  "dependencies": {
    "Microsoft.AspNet.Server.IIS": "1.0.0-beta7",
    "Microsoft.AspNet.Server.WebListener": "1.0.0-beta7",
    "Microsoft.AspNet.Server.Kestrel": "1.0.0-beta7",
    "Microsoft.AspNet.Mvc": "6.0.0-beta7"
  },

  "commands": {
    "kestrel": "Microsoft.AspNet.Hosting --server
Microsoft.AspNet.Server.Kestrel --config hosting.ini",
    "web": "Microsoft.AspNet.Hosting --server
Microsoft.AspNet.Server.WebListener --config hosting.ini"
  },

  "frameworks": {
    "dnx451": {},
    "dnxcore50": {}
  },

  "publishExclude": [
    "node_modules",
    "bower_components",
    "**.xproj",
    "**.user",
    "**.vspscc"
  ],
  "exclude": [
    "wwwroot",
    "node_modules",
    "bower_components"
  ]
}
```

csproj

```
<?xml version="1.0" encoding="utf-8"?>
<Project DefaultTargets="Build" xmlns="http://
schemas.microsoft.com/developer/msbuild/2003"
ToolsVersion="4.0">
  <PropertyGroup>
    <Configuration Condition=" '$(Configuration)' == ''
">Debug</Configuration>
    <Platform Condition=" '$(Platform)' == '' ">AnyCPU</
Platform>
    <ProductVersion>9.0.30729</ProductVersion>
    <SchemaVersion>2.0</SchemaVersion>
    <ProjectGuid>{2A77F79C-435D-41BD-BA47-B61633CB3B6F}</
ProjectGuid>
    <OutputType>Library</OutputType>
    <AppDesignerFolder>Properties</AppDesignerFolder>
    <RootNamespace>Clearent.Common.ClearBillSettle</
RootNamespace>
    <AssemblyName>Clearent.ClearBillSettle.DataObjects</
AssemblyName>
    <TargetFrameworkVersion>v3.5</TargetFrameworkVersion>
    <FileUpgradeFlags>
    </FileUpgradeFlags>
    <UpgradeBackupLocation>
    </UpgradeBackupLocation>
    <OldToolsVersion>3.5</OldToolsVersion>
  </PropertyGroup>
  <PropertyGroup Condition=" '$(Configuration)|$(Platform)'
== 'Debug|AnyCPU' ">
    <DebugSymbols>true</DebugSymbols>
    <DebugType>full</DebugType>
    <Optimize>false</Optimize>
    <OutputPath>bin\Debug\</OutputPath>
      <HintPath>..\References\FileHelpers\FileHelpers.dll</
HintPath>
    </Reference>
    <Reference Include="log4net, Version=1.2.10.0,
Culture=neutral, PublicKeyToken=1b44e1d426115821,
processorArchitecture=MSIL">
```

Commands

"A command is a named execution of a .NET entry point with specific arguments."

Commands are short-hand for specifying the command assembly and its arguments directly to DNX.

Can define commands at the project level or globally under a user profile.

What I've Learned

* Text-based dependency management is cool! (project.json)
* Did I mention this stuff changes often?
* Manage DNX version dependencies with start scripts.

Community

* Active discussion threads at https://jabbr.net.
* Yeoman generators (https://github.com/OmniSharp/generator-aspnet)
* Omnisharp (https://github.com/OmniSharp) is a series of projects that enable VS-like editing in other editors.

Community (cont'd.)

* The Community Stand-up (https://live.asp.net) helps to stay informed.
* People to follow:
  * Damian Edwards
  * David Fowler
  * Scott Hanselman
  * Jon Galloway

Where Does it Live?

https://github.com/aspnet

https://asp.net

https:// jabbr.net

DNX Schedule

* Kept at https://github.com/aspnet/Home/wiki/Roadmap.
* Beta8 Released 10/15/2015.
* RC1 slated for November 2015.
* 1.0.0 Release slated for Q1 2016.

This Material

* https://github.com/haled/presentations
* Ping me on Twitter @darrenhale or email darren@clearent.com

World Cup Soccer

vs.

Thank You!

Darren Hale

darren@clearent.com

@darrenhale

https://github.com/haled/presentations