



mibuso.com

CLIENT ADD-INS BLACK BELT: BRINGING .NET AND JAVASCRIPT TOGETHER

Vjekoslav Babić
(Fortempo - Business Consulting)

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



WTF?!

ABOUT ME

Vjekoslav Babić

consultant, trainer, blogger, author



Vjeko.com
ideas in the cloud

Twitter: @vjekob

Mibuso: Vjeko

Blog: vjeko.com

Author of many How Do I... videos for MSDN and PartnerSource for NAV 2013, NAV 2013 R2, and NAV 2015

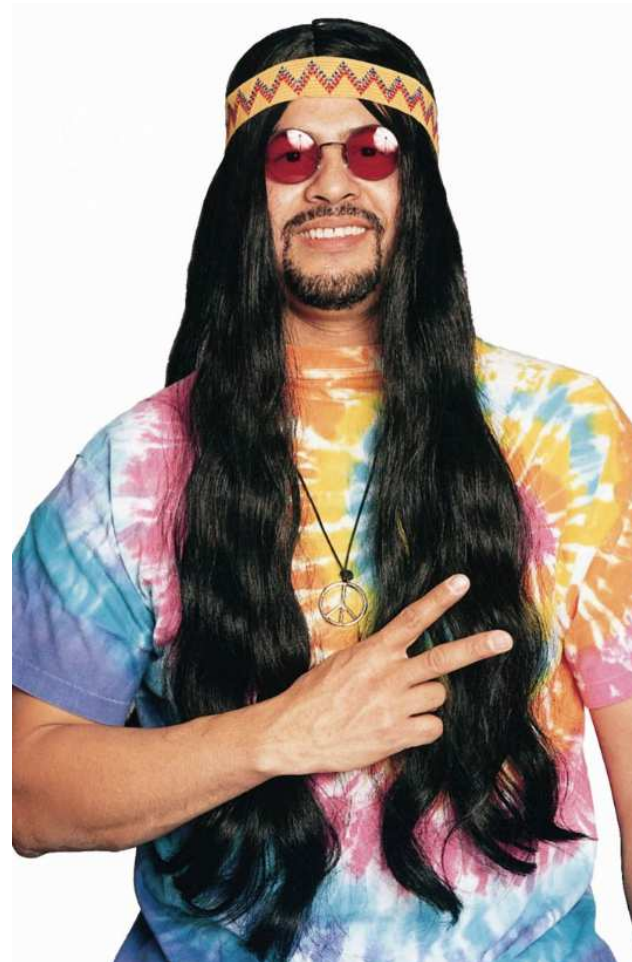
Co-author of "Implementing Microsoft Dynamics NAV 2009" book

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com





C/AL



JavaScript


```
b=Type.getInstanceType(a);return this.isAssignableFrom(b));Type.prototype.isAssignableFrom=function(c){if(this==Object||this==c)return true;
(this.__class){var a=c.__baseType;while(a){if(this==a)return true;a=a.__baseType}}else if(this.__interface){var b=c.__interfaces;if(b)
return true;var a=c.__baseType;while(a){b=a.__interfaces;if(b&&b.contains(this))return true;a=a.__baseType}}return false};Type.isClass=function(a){
a.__class==true};Type.isEnum=function(a){return a.__enum==true};Type.isFlags=function(a){return a.__enum==true&&a.__flags==true};Type.isInterface=
(a){return a.__interface==true};Type.isNamespace=function(a){return a.__namespace==true};Type.canCast=function(a,b){return b.isInstanceOfType(a)};
Type.safeCast=function(a,b){return b.isInstanceOfType(a)?a:null};Type.getInstanceType=function(b){var a=null;try{a=b.constructor}catch(e){}
!a.__typeName)a=Object;return a};Type.getType=function(a){if(!a)return null;if(!Type.__typeCache)Type.__typeCache={};var b=Type.__typeCache[a];
if(b)return b;b=eval(a);Type.__typeCache[a]=b;return b};Type.parse=function(a){return Type.getType(a)};ss.Enum=function(){};ss.Enum.registerClass=
function(c){ss.Enum.parse=function(d,i){var a=d.prototype;if(!d.__flags){for(var b in a)if(b===i)return a[b]}else{for(var g=i.split("|"),h=0,e=t
c--){var j=g[c].trim(),f=false;for(var b in a)if(b===j){h|=a[b];f=true;break}if(!f){e=false;break}}if(e)return h}throw"Invalid Enumeration Value"};
ss.Enum._enumToString=function(d){var b=this.prototype;if(!this.__flags||d===0){for(var c in b)if(b[c]===d)return c;throw"Invalid Enumeration Value"};
{var a=[];for(var c in b)if(b[c]&d){a.length&&a.add(" | ");a.add(c)}if(!a.length)throw"Invalid Enumeration Value";return a.join("")}};ss.Delegate.registerClass("Delegate");
ss.Delegate.Null=function(){};ss.Delegate._contains=function(b,d,c){for(var a=0;a<b.length;a+=1)if(b[a]===c)return true;return false};ss.Delegate._create=
function(a){var b=function(){if(a.length==2)return a[1].apply(a[0],arguments)};b._targets=a;return b};ss.Delegate.create=function(b,a){if(!b){b._targets=a;
return b};if(!b._targets){b._targets=a;return b};if(!b._targets){b._targets=a;return b};if(!b._targets){b._targets=a;return b};if(!b._targets){b._targets=a;return b};
return ss.Delegate._create(c.concat(d))};ss.Delegate.remove=function(c,a){if(!c||c===a)return null;if(!a)return c;var b=c._targets,f=a._targets[0];e=a._targets[1];
else e=a;for(var d=0;d<b.length;d+=2)if(b[d]===f&&b[d+1]===e){if(b.length==2)return null;b.splice(d,2)}return b};ss.Delegate._create(b)}return c};ss.Delegate.createExport=
function(b,c){var a="__"+(new Date).valueOf();ss.Delegate[a]=function(){!c.apply(b,arguments)};ss.Delegate.deleteExport(a);b.apply(null,arguments)};return a};
ss.Delegate.deleteExport=function(a){if(ss.Delegate[a])delete ss.Delegate[a]};ss.Delegate.clearExport=function(a){if(ss.Delegate[a])ss.Delegate[a]=Delegate.Null};
ss.CultureInfo=function(c,a,b){this.name=c;this.dateFormat=b};ss.CultureInfo.registerClass("CultureInfo");ss.CultureInfo.InvariantCulture=new ss.CultureInfo("en-US",
{naSymbol:"N/A",negativeSign:"-",positiveSign:"+",negativeInfinityText:"-Infinity",positiveInfinityText:"Infinity",percentSymbol:"%",percentGroupSizes:[3],
percentDecimalDigits:2,percentDecimalSeparator:".",percentGroupSeparator:",",percentPositivePattern:"{0} %",percentNegativePattern:"- {0} %",currencySymbol:"$",
currencyGroupSizes:[3],currencyDecimalDigits:2,currencyDecimalSeparator:".",currencyGroupSeparator:",",currencyPositivePattern:"${0}",numberGroupSizes:[3],
numberDecimalDigits:2,numberDecimalSeparator:".",numberGroupSeparator:",",},{amDe
```



Hands up...

How many control add-ins have you created?

10++

Think of this...

How many control add-ins can solve all your problems?

1



WHERE

IS THAT CONTROL?

Demo

A mean and lean all-purpose control add-in

TODAY'S AGENDA

Crossing page
boundaries

JSON

Deployment

Resolving
assemblies

User Interface

Call
synchronization

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com

NAV
TECH
DAYS
2015
mibuso.com



mibuso.com

PART 1

Crossing page boundaries:
Talking to the control add-in from other objects

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



BRAVE CLAIM #1

Code does not belong in pages.

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



... IF CODE DOES NOT BELONG IN PAGES, THEN...

The screenshot illustrates the integration of code between a page and a control add-in in the SAP NetWeaver IDE. It features three main windows: the Page Designer, the C/AL Editor for the page, and the C/AL Editor for the control add-in.

Page Designer: Shows a table with columns for Type, SubType, SourceExpr, Name, and Caption. The table contains two rows: one for 'Container' (SubType: ContentArea) and one for 'Field' (SubType: ContentArea). The 'ControlAddIn' control is highlighted in the table.

Page 66001 Hello World - C/AL Editor: Contains the following code:

```
27 ControlAddIn - OnValidate()  
28  
29 ControlAddIn - OnLookup(VAR Text : Text) : Boolean  
30  
31 ControlAddIn - OnDrillDown()  
32  
33 ControlAddIn - OnAssistEdit()  
34  
35 ControlAddIn - OnControlAddIn(Index : Integer;Data :  
36  
37 ControlAddIn::ControlReady()  
38 MyCodeunit.SetControlAddIn(CurrPage.ControlAddIn);  
39 MyCodeunit.SayHello('Hello, World!');  
40
```

ControlAddIn - Properties: Shows the properties of the control add-in. The 'ControlAddIn' property is highlighted with the value 'My Control Add-In;PublicKeyToken=ea9604bef2a38877'.

Codeunit 66001 Hello World - C/AL Editor: Contains the following code:

```
1 Documentation()  
2  
3 OnRun()  
4  
5 SetControlAddIn ControlAddInIn : DotNet "Vjeko.Com.ControlAddInTemplate.IControlAddIn"  
6 ControlAddIn := ControlAddInIn;  
7  
8 SayHello(Name : Text)  
9 ControlAddIn.SayHello(Name);  
10
```

Red arrows indicate the flow of data and control between the page and the control add-in. A green arrow points from the 'SayHello' call in the page code to the 'SayHello' method in the control add-in code.

Demo #1

... but this simply does not work.

IT ALL MAKES SENSE...

- Control add-in is an interface:

```
[ControlAddInExport("My Control Add-In")]
```

0 references

```
public interface IControlAddIn
```

```
{
```

```
    [ApplicationVisible]
```

```
    event EmptyEventHandler ControlReady;
```

```
    [ApplicationVisible]
```

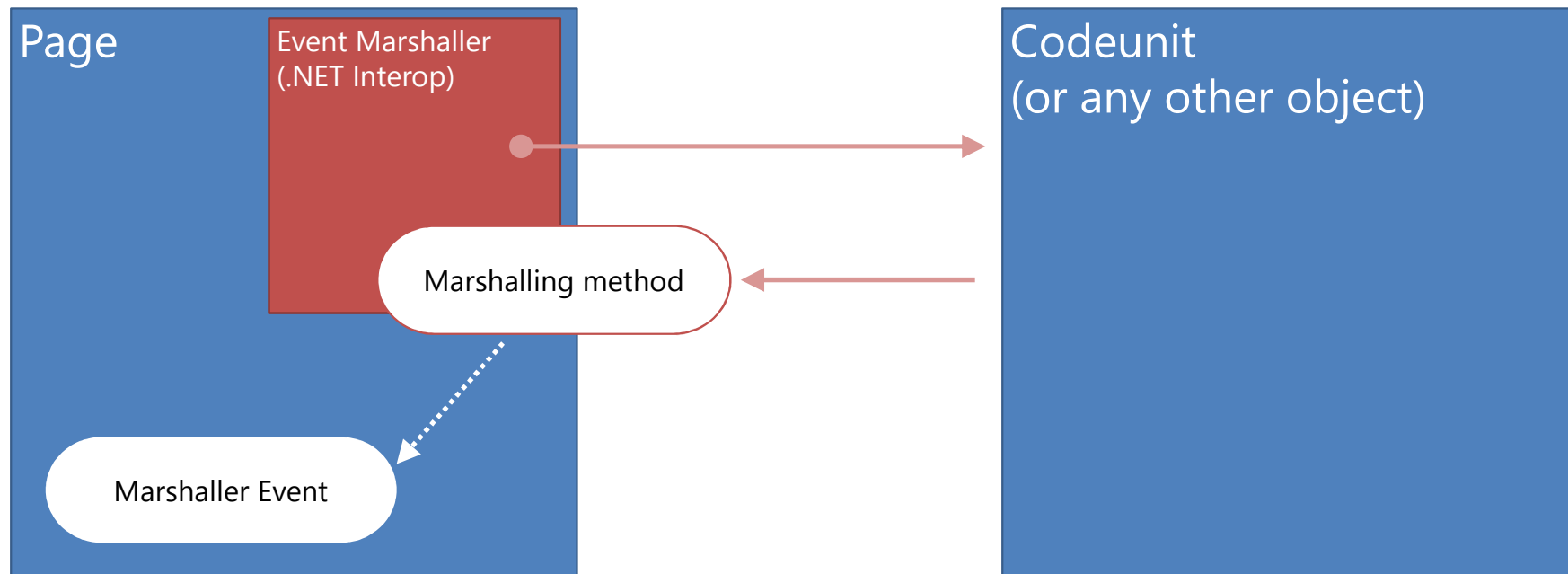
0 references

```
    void SayHello(string s);
```

```
}
```

- ... and interfaces don't have instances, unless they are implemented in a class.

SOLVING THE PUZZLE



Demo #2

Communication through an event marshaller



mibuso.com

PART 2

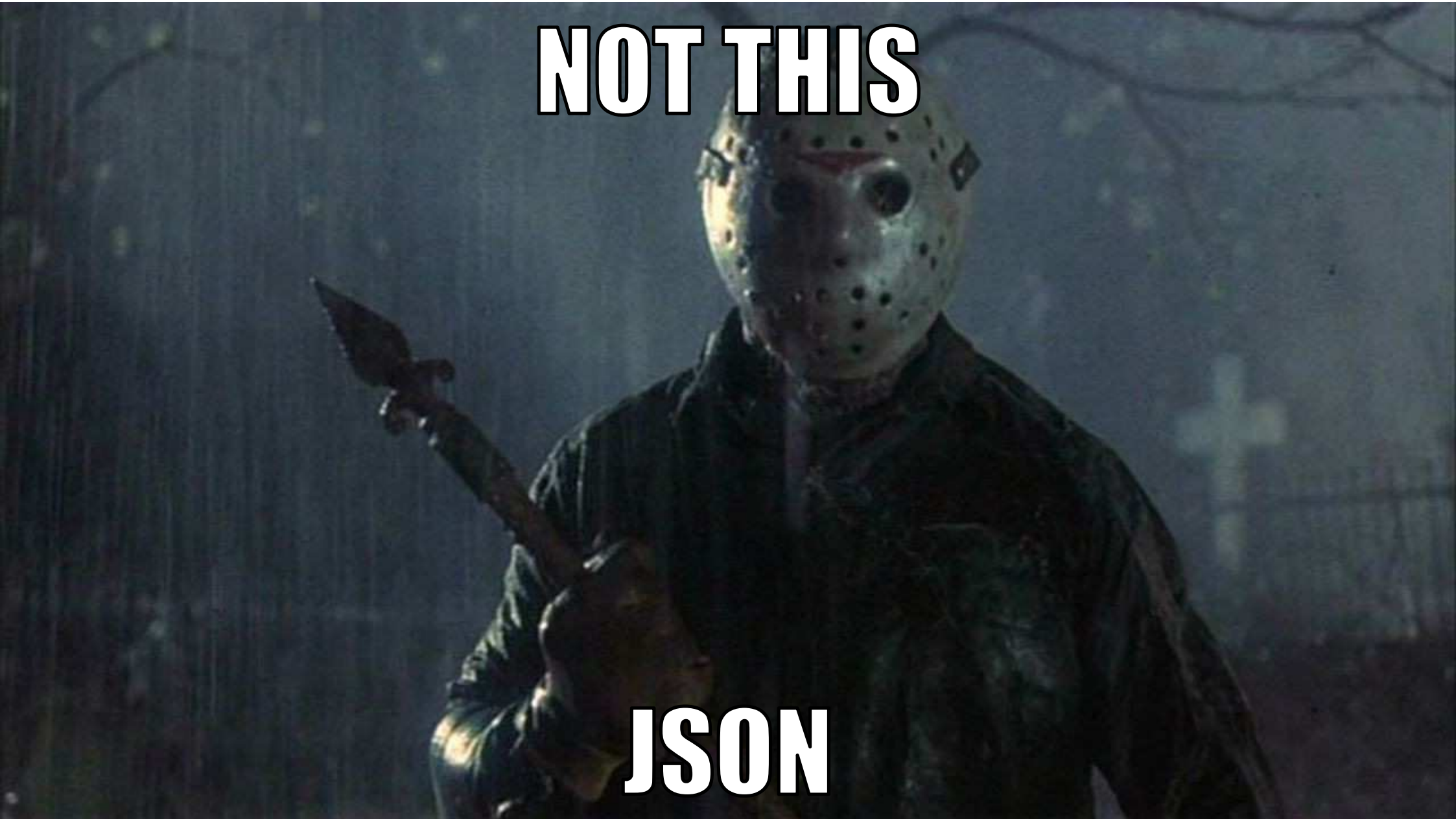
JSON

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



NOT THIS

JSON

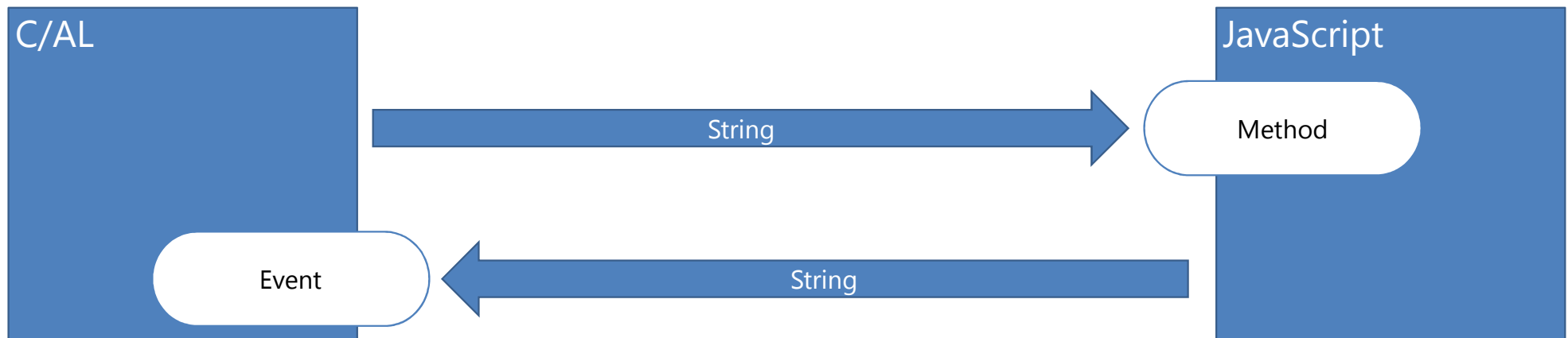


WHAT IS JSON?

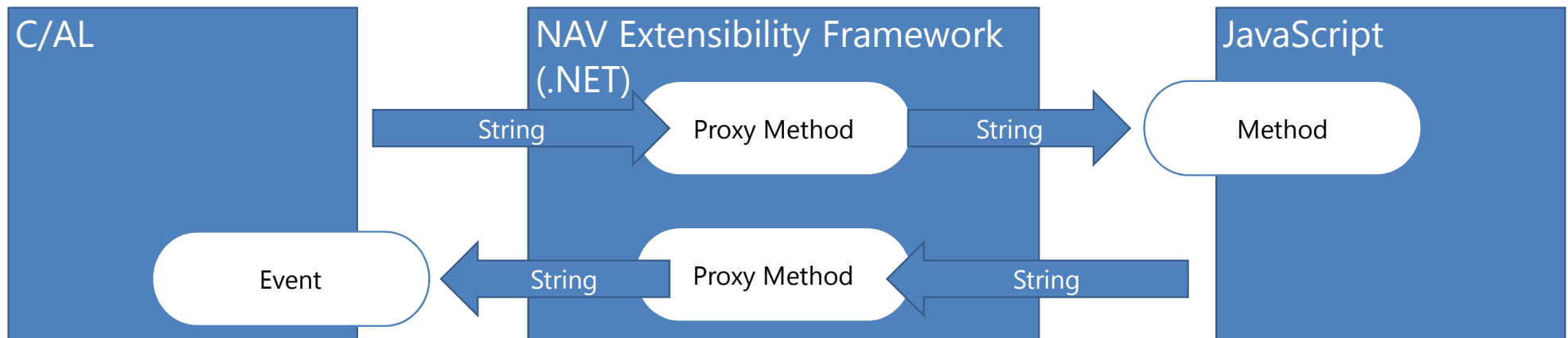
- JSON is JavaScript Object Notation
- JavaScript objects can be represented as JSON
- JavaScript can:
 - Convert an object to JSON string
 - Convert JSON string to an object
 - Instantiate an object from JSON

```
var j = {  
  Type: "Conference",  
  Location: "Antwerp",  
  Name: "NAV TechDays",  
  Year: 2015,  
  Version: 5.0,  
  Rocks: true  
}
```

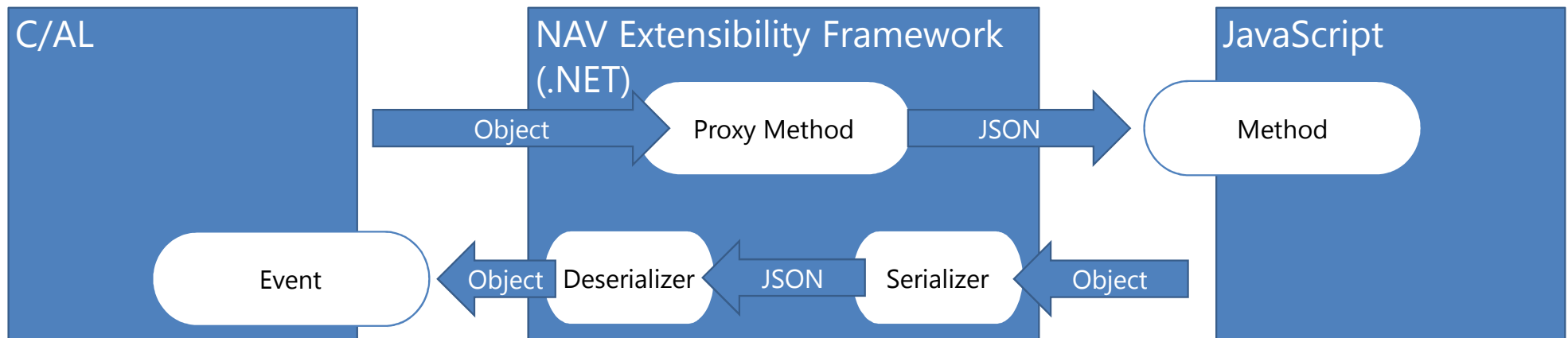
HOW DOES C/AL TALK TO JAVASCRIPT



HOW DOES C/AL TALK TO JAVASCRIPT



HOW DOES C/AL TALK TO JAVASCRIPT



PASSING OBJECTS BETWEEN C/AL AND JAVASCRIPT

- Objects of any serializable .NET type can be passed to JavaScript
- Any JavaScript object can be passed to C/AL
- NAV runtime handles serialization (conversion) between objects using JSON

C#	JavaScript (JSON)
<pre>public class Person { 1 reference public string FirstName { get; set; } 1 reference public string LastName { get; set; } 1 reference public int Age { get; set; } } Person p = new Person { FirstName = "John", LastName = "Doe", Age = 42 };</pre>	<pre>var p = { FirstName: 'John', LastName: 'Doe', Age: 42 }</pre>

PASSING OBJECTS BETWEEN C/AL AND JAVASCRIPT

```
public delegate void MessageEventHandler(Message msg);

[ControlAddInExport("My Control Add-In")]
0 references
public interface IControlAddIn
{
    [ApplicationVisible]
    event EmptyEventHandler ControlReady;

    [ApplicationVisible]
    event MessageEventHandler Message;

    [ApplicationVisible]
    0 references
    void SendRequest(Request request);
}
```


Demo #3

Passing objects back and forth

PASSING DYNAMIC STRUCTURES

Dictionary<,> is always mapped to a valid JSON object of dynamic structure:

- Dictionary instance becomes one object
- Every key becomes property
- Every value becomes property value

Receiving objects from JavaScript into Dictionary<,> is not directly supported.

Demo #4

Mapping Dictionary<,> to JSON



mibuso.com

PART 3

Deploying during development

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



MANUAL DEPLOYMENT

1. Retrieve the public-key token from the assembly
2. Create control add-in record
3. Copy the assembly to the Add-Ins folder
4. Create the resource zip file
5. Import the resource zip file

*If it can be automated,
it should be automated.*

(ancient Chinese proverb)



DEPLOYING DURING DEVELOPMENT

- Deploying new functionality costs time (as much as a couple minutes per build, or change)
- Building during development is a frequent process
- All deployment steps can be fully automated

AUTOMATING DEPLOYMENT USING POWERSHELL

Load all satellite assemblies

Search types for ControlAddInExport attribute

For every interface with attribute, create Resource folder structure

Read public key token from the assembly

Zip the resource file

Copy files to client and server add-ins

If necessary, restart NST

Import or modify Control Add-in records in NAV

Demo #5

Deploying from Visual Studio



mibuso.com

PART 4

Automatically Resolving .NET Interoperability Assembly Dependencies

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



DEPLOYING ASSEMBLIES

Copying
assembly .dll
into Add-ins
folder

Installing
into GAC

- We can automate this, but:
 - Only per environment
 - Moving to a new environment requires re-deployment

CLIENT-SIDE DEPLOYMENT IN NAV 2015

NAV 2015 and newer:

- Assemblies deployed to *Service\Add-ins* will be automatically deployed to clients
 - If the assembly is not already present in *RoleTailored Client\Add-ins*
 - If the assembly is not already present in the GAC
 - If the assembly wasn't already automatically deployed
- Assemblies are not automatically deployed to *RoleTailored Client\Add-ins* but to temporary folder in user's profile
- Redeploying an assembly to the server will only update it to clients if the Version property changes

CLIENT AND SERVER DEPLOYMENT IN NAV 2016

NAV 2016 and newer:

- Assemblies deployed into the Add-ins table will be automatically deployed at runtime from the database:
 - If Add-in Name matches the assembly name
 - If assembly is not already deployed to *Service\Add-ins* folder
 - If assembly is not already automatically deployed from the database
- Issues:
 - Ignores the fully qualified assembly name
 - Requires assembly files to be individually zipped and imported
 - Re-deploying an assembly requires version change

REINVENTING THE (BETTER) WHEEL

Goals:

- Backward compatibility all the way back to NAV 2013
- Both client-side and server-side
- Redeploy even if version does not change
- Must have zero-footprint on the outside environment: **must require no external assemblies**

ASSEMBLYRESOLVE EVENT

.NET applications automatically resolve assemblies:

- By looking into GAC
- By looking into the same folder where the application (entry assembly) resides

When the assembly cannot be found there .NET runtime calls System.AppDomain.AssemblyResolve event:

- An application uses this event to look in additional places
- Microsoft Dynamics NAV looks into server or client Add-ins folders and their subfolders

Subscribing to this event is not possible using C/AL – you must use a wrapper class

ZERO-FOOTPRINT ASSEMBLY RESOLVER

- Assemblies stored in the database
- Assemblies identified using their fully qualified name
- Assembly resolver wrapper class built at on-the-fly at runtime – no deployment needed either
- Wrapper class publishes C/AL-compatible event to resolve assemblies
- Resolved assemblies are not retained between sessions

Demo #6

Resolving assemblies from database



mibuso.com

PART 5

Managing User Interface

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



WHAT THEY TEACH US IN SCHOOL...

To create user interface

- Create UI through DOM manipulation
- Embed a stylesheet into resource
- Embed images into resource

To manage client-side logic

- Embed scripts into resource
- Load scripts from URL

WHAT THEY DON'T TEACH US IN SCHOOL...

Nothing needs to be embedded in resource.

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



Demo #7

Dynamic HTML, CSS, and JavaScript

Demo #8

Owning the playground
(and take AngularJS angle at it all...)

WARNING



DO NOT

PEE



ON THE

ELECTRIC FENCE

Demo #9

Owning some more playground
(by editing .aspx files)

SOME IDEAS...

Store HTML, CSS,
JavaScript in the
database (BLOB)

Load and feed to
the client on
demand

Construct HTML,
CSS
programmatically

Construct
JavaScript
programmatically

You own the
show, be
creative...



mibuso.com

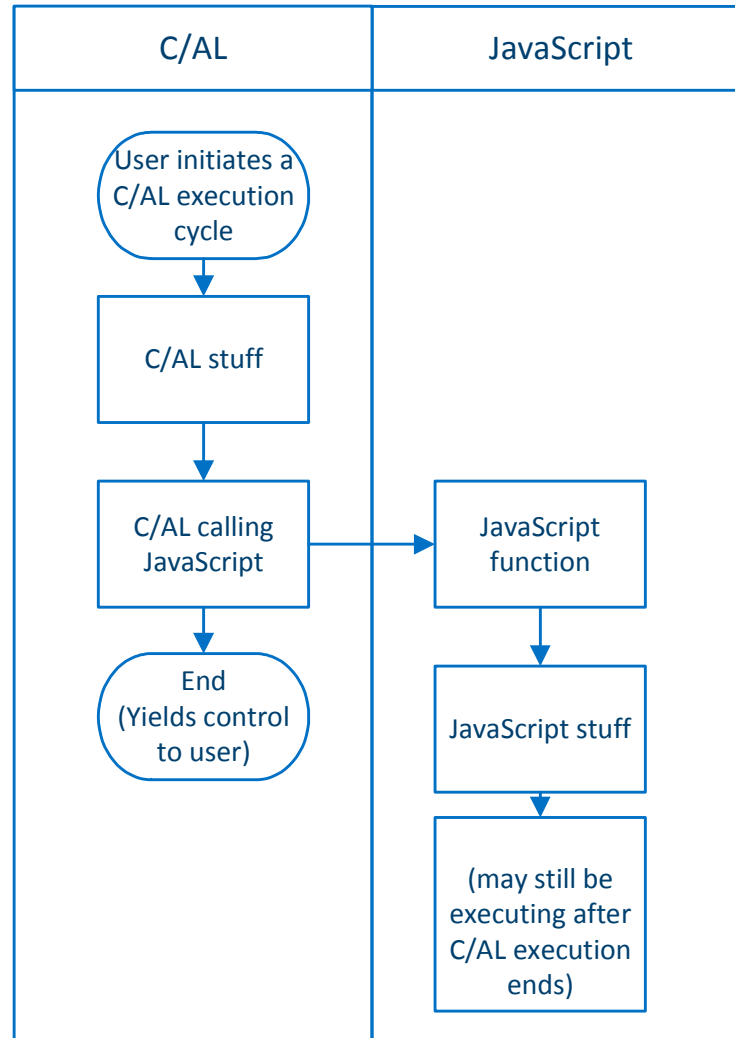
PART 6

Synchronizing Calls between JavaScript and C/AL

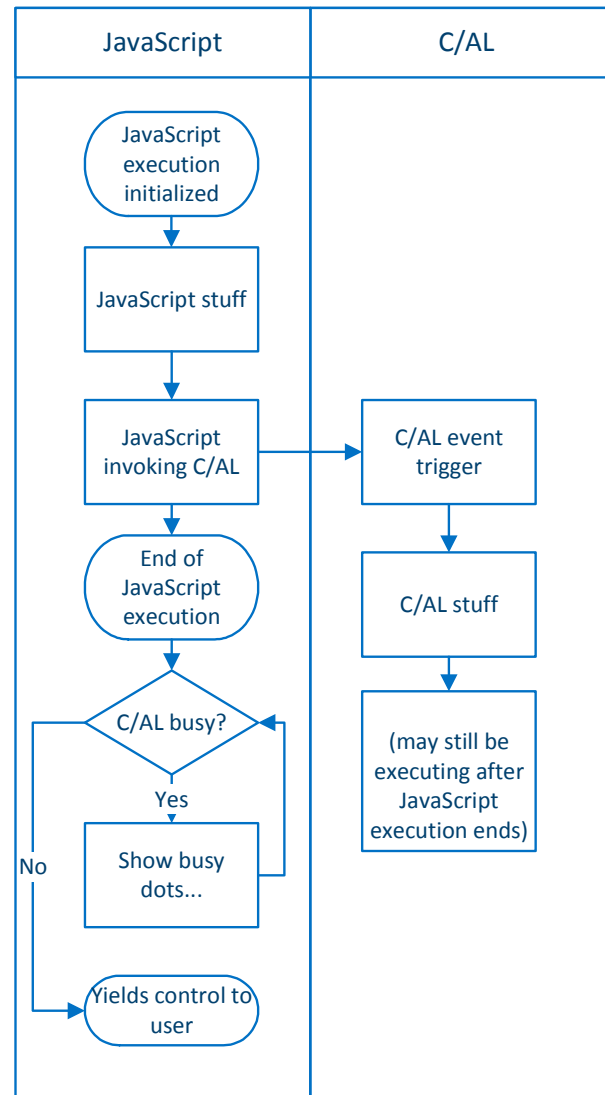
WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



C/AL CALLING JAVASCRIPT



JAVASCRIPT CALLING C/AL



Why the difference?

CONSEQUENCE OF ASYNCHRONOUS CALLING

Negative

No return values on methods or events

C/AL not aware if JavaScript is busy

Possible performance issues or crashes

Positive

Code becomes more modular

Architectures become event-driven

Better separation between UI and business logic

Demo #10

Issues due to incorrect synchronization

SYNCHRONIZING CALLS

- No option to synchronize calls from C/AL to JavaScript – C/AL simply goes on
- To synchronize calls from JavaScript to C/AL, we have (in NAV 2015 and newer):

Environment object

- Busy property
- OnBusyChanged event

InvokeExtensibilityMethod parameters

- skipIfBusy
- callback

Demos #10 and #11

Synchronizing calls
(two different approaches)

TAKEAWAYS...

Web Clients allow you to own the show. Totally.

Do not build hardcoded control add-ins. You only need one.

Combine the power of .NET with flexibility of JavaScript.

... AND SOME MORE TAKEAWAYS...



vjeko.com/categories/control-add-ins

20+ articles, examples, tips and tricks about control add-ins

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



Any Questions?

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com



THANK YOU

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com





DINNER

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com

