



mibuso.com

JAVASCRIPT ARCHITECTURE: TURNING PAIN INTO GAIN

VJEKOSLAV BABIĆ – CLOUD READY SOFTWARE

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



About me



Vjekoslav Babić

consultant, trainer, blogger, author



Vjeko.com
ideas in the cloud

Twitter: @vjekob
Mibuso: Vjeko
Blog: vjeko.com

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

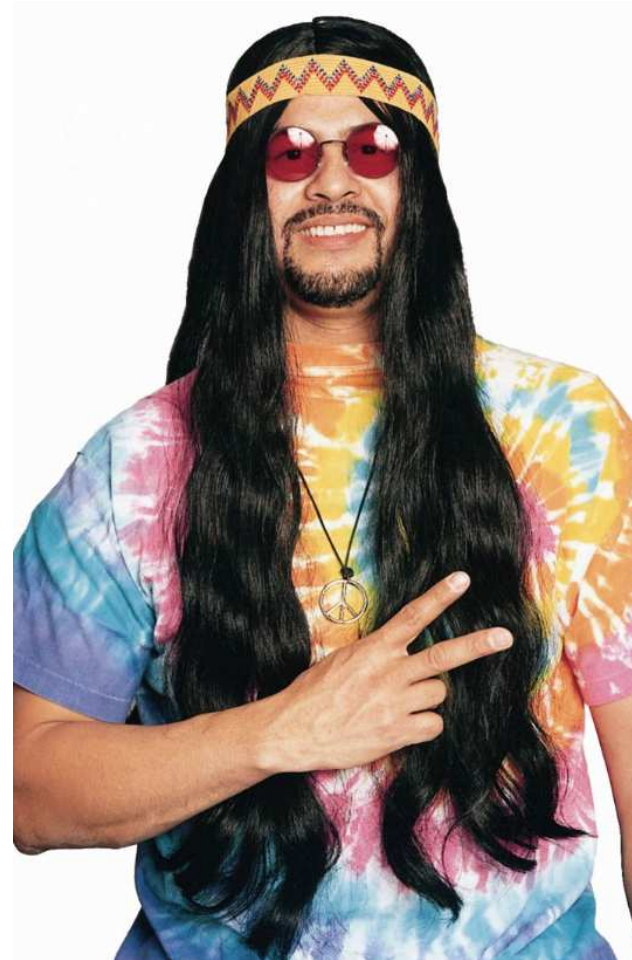
Co-author of "Implementing Microsoft Dynamics NAV 2009" book



400



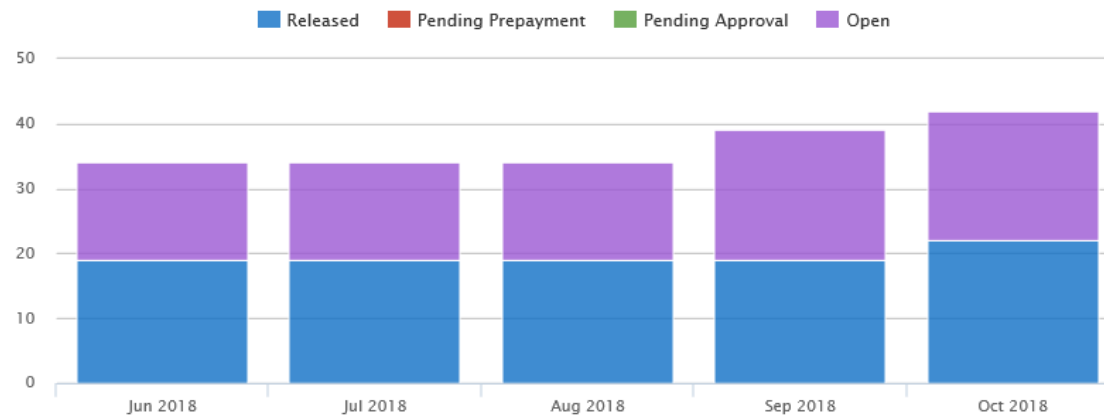
C/AL



JavaScript











Why Architecture?

All Orders|Month|No. of Orders|. (Updated at 03:37:05.)





Description	Quantity	Unit Price	Amount Including VAT
Large Draft Beer	1,00	45,00	45,00
Rosé Wine Glass	1,00	45,00	45,00


SUBTOTAL 90,00


 White Wine Glass	 Red Wine Regular Bottle			
 White Wine Small Bottle	 Rosé Wine Glass			
 White Wine Regular Bottle	 Rosé Wine Small Bottle			
 Red Wine Glass	 Rosé Wine Regular Bottle			
 Red Wine Small Bottle				 Back


Item No.



Delete Line


Cancel Sale


Terminal -not in use


Quantity



Other Functions



Go To Payment


Rosé Wine Glass


No. 40016


Unit Price 45,00



Gift Voucher Sales



Lookup



Save



Customer


Print


Retrieve Sale


Return


Open Register Drawer


Discount

CRONUS International Ltd.

20000 · Selangorian Ltd.

HOME

ACTIONS

NAVIGATE

REPORT

View

Edit

New

Delete

Sales Quote

Sales Invoice

Sales Order

Sales Credit Memo

Reminder

Send Approval Request

Cancel Approval Request

Approvals

Contact

Bank Accounts

Direct Debit Mandates

Ship-to Addresses

OneNote

Notes

Links

Refresh

Clear Filter

Go to

Previous

Next

Manage

New Document

Request Approval

Customer

Show Attached

Page

20000 · Selangorian Ltd.

General

Customer Picture

No.:20000

Name:Selangorian Ltd.

IC Partner Code:

Balance (LCY):96.049,99

Balance Due (LCY):96.049,99

Credit Limit (LCY):0,00

Blocked:

Salesperson Code:PS

Responsibility Center:

Service Zone Code:N

Document Sending Profile:

Total Sales:6.510,64

Costs (LCY):3.998,90

Profit (LCY):2.511,74

Profit %:38,6

Language Code:ENG

Last Date Modified:25.10.2016.

Show fewer fields

Address & Contact

Customer Statistics

Address

Address:153 Thomas Drive

Address 2:

Post Code:CV6 1GY

City:Coventry

Country/Region Code:GB

Show on Map

Contact

Primary Contact Code:CT000010

Contact Name:Mr. Mark McArthur

Phone No.:

Email:

Home Page:

Show more fields

Invoicing

Payments

Shippinn

DOMESTIC

DOMESTIC

RETAIL

14 DAYS

Partial

EXW

Balance (LCY):96.049,99

Sales

Outstanding Ord...9.351,34

Shipped Not Inv...0,00

Outstanding Inv...0,00

Service

Outstanding Ser...0,00

Serv Shipped No...0,00

Outstanding Ser...0,18

Total (LCY):105.401,...

Credit Limit (LCY):0,00

Overdue Amount...49.633,17

Total Sales (LCY):6.510,64

Invoiced Prepay...0,00

OK



Description	Quantity	Unit Price	Amount Including VAT	Item Use
Large Dark Beer	1.00	45.00	45.00	
Food Wine Glass	1.00	45.00	45.00	
				<div><div>Order Item</div><div>Cancel Item</div><div>Transfer Item to cart</div><div>Quantity</div><div>Clear Inventory</div><div>Go to Payment</div></div>
				<div>Food Wine Glass</div> <div>No.</div> <div>Unit Price</div> <div>45.00</div> <div>45.00</div>
				<div><div>DR-Menu-Likes</div><div>Like/Dislike</div><div>Item</div><div>Categories</div><div>Print</div><div>Reorder Item</div><div>Refresh</div><div>Open Register Drawer</div><div>Discount</div></div>
				<div>LAST DEMO</div> <div>DATE</div> <div>VERSION</div>

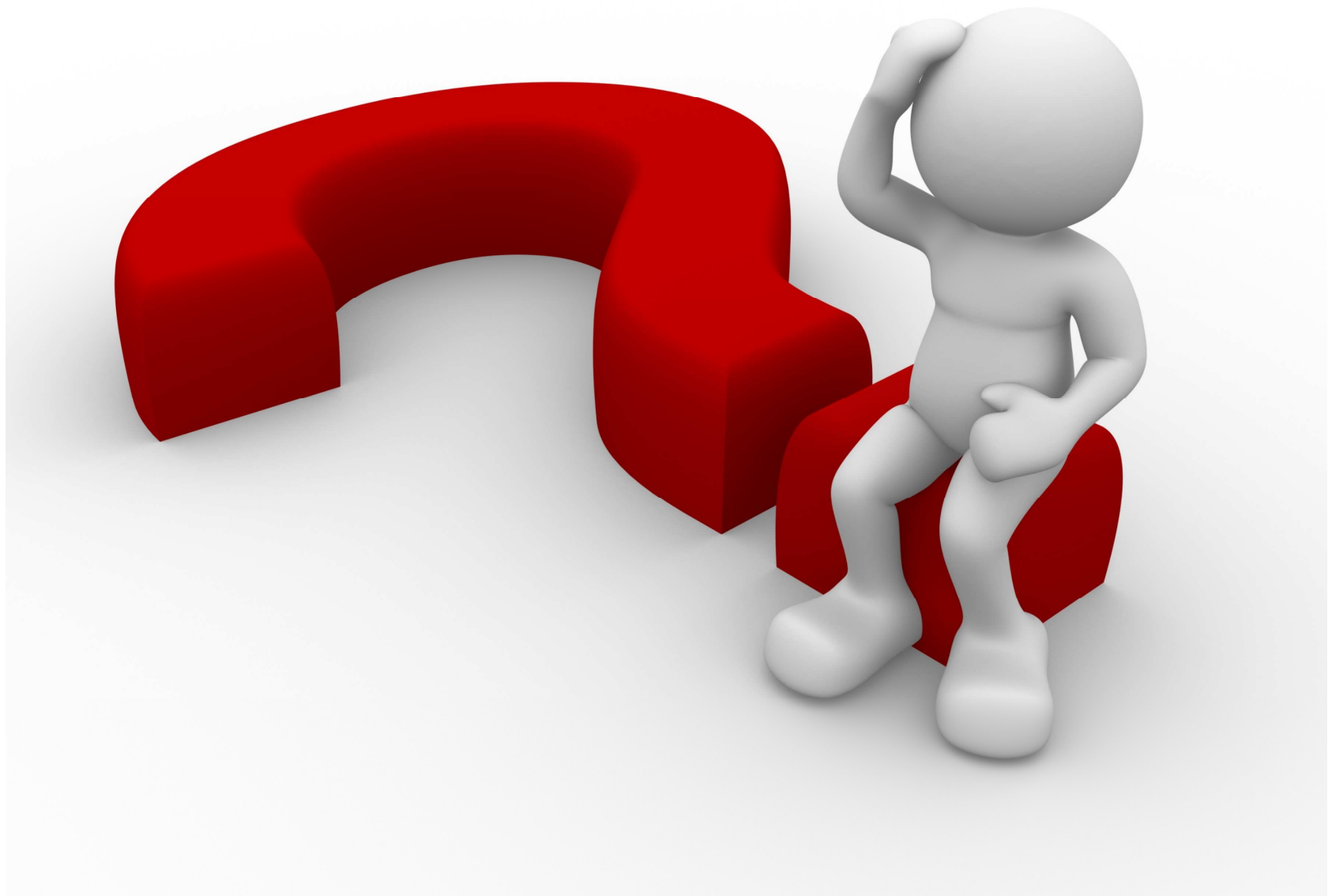
Page 60004 Demo 4 - Page Designer

E.. Type	SubType	SourceExpr	Name	Cap
Container	ContentArea		<Control1>	<<
Field			AddIn	<

Navigation icons: Left, Right, Up, Down, Preview, Help

```
1 NewPage()
2
3 ...
4 ...
5 ...
6 ...
7 ...
8 ...
9 ...
10 ...
11 ...
12 ...
13 ...
14 ...
15 ...
16 ...
17 ...
18 ...
19 ...
20 ...
21 ...
22 ...
23 ...
24 ...
25 ...
26 ...
27 ...
28 ...
29 ...
30 ...
31 ...
32 ...
33 ...
34 ...
35 ...
36 ...
37 ...
38 ...
39 ...
40 ...
41 ...
42 ...
43 ...
44 ...
45 ...
46 ...
47 ...
48 ...
49 ...
50 ...
51 ...
52 ...
53 ...
54 ...
55 ...
56 ...
57 ...
58 ...
59 ...
60 ...
61 ...
62 ...
63 ...
64 ...
65 ...
66 ...
67 ...
68 ...
69 ...
70 ...
71 ...
72 ...
73 ...
74 ...
75 ...
76 ...
77 ...
78 ...
79 ...
80 ...
81 ...
82 ...
83 ...
84 ...
85 ...
86 ...
87 ...
88 ...
89 ...
90 ...
91 ...
92 ...
93 ...
94 ...
95 ...
96 ...
97 ...
98 ...
99 ...
100 ...
```

```
3  OnInit()  
4  |  
5  OnOpenPage()  
6  | ERROR('Code does not belong here, Einstein!');  
7  |  
8  OnClosePage()  
9  |  
10 OnFindRecord(Which : Text) : Boolean  
11 |  
12 OnNextRecord(Steps : Integer) : Integer  
13 |  
14 OnAfterGetRecord()  
15 |  
16 OnNewRecord(BelowxRec : Boolean)  
17 |  
18 OnInsertRecord(BelowxRec : Boolean) : Boolean  
19 |  
20 OnModifyRecord() : Boolean  
21 |  
22 OnDeleteRecord() : Boolean  
23 |  
24 OnQueryClosePage(CloseAction : Action None) : Boolean
```

Visual Studio

For all my demos I use the control add-in template you can find here:

<http://vjeko.com/vstemplate>

Part 1

Optimizing Framework

Problem statement

Control Add-ins:

- Get heavy and can cause performance issues
- Rigid due to C# interface and C/AL structure
- Require redeployment after any change

```
1 using Microsoft.Dynamics.Framework.UI.Extensibility;
2
3 namespace AddIn
4 {
5     [ControlAddInExport("NAVTechDays.2016")]
6     public interface IControlAddIn
7     {
8         [ApplicationVisible]
9         void MethodCall(object call);
10
11         [ApplicationVisible]
12         event EmptyEventHandler ControlReady;
13
14         [ApplicationVisible]
15         event ObjectEventHandler ControlEvent;
16
17         [ApplicationVisible]
18         event ObjectEventHandler Require;
19
20         [ApplicationVisible]
21         event EmptyEventHandler CloseRequested;
```



Demo 1

Simple but fat control add-in

Step 1: Turning the control into a framework

1. Replace specific methods and events with generic ones
2. Move code from page to codeunit
3. Enable event-based communication
4. Move scripts, styles, and images from control add-in to setup

Demo 2

Generic methods and events

Problem statement

Business logic is blended with UI logic

Infrastructure logic (JSON management) is blended with both UI logic and business logic

Events could help!



Two flavors of event subscribers

Dynamic-Static

- Subscription is automatic
- No possibility of unsubscribing
- Every event is called in a fresh new instance of the subscriber codeunit

Manual

- Subscription is manual
- Subscriber can be unsubscribed
- All events are called in the same instance of the subscriber codeunit

C/AL statements:

```
BINDSUBSCRIPTION  
UNBINDSUBSCRIPTION
```

Demo 3

Decoupling through events

Problem statement

Depending on how page is accessed and navigated away, the OnClosePage event may not fire

Results:

- Multiplication of subscriber instances
- Subscribers remain subscribed
- No possibility to unsubscribe after the user navigates away from the page



Demo 4

Solving the page close problem

Problem statement

Loading of scripts is not optimal.

Reasons:

- Each control add-in lives in a separate IFRAME
- Scripts may be downloaded multiple times
- Especially scripts already loaded by NAV

Goals:

- Reduce script loading multiplication
- Optimize dependencies (scripts, styles, images)



Demo 5

Optimizing dependencies

Problem statement

This solution works only in the Web clients.

Dependencies are managed as hardcoded text.

Reason:

- Windows client has no “top” window to pull from.

Goal:

- Create a setup for storing dependency files.
- Create a framework for loading dependencies.



RequireAL – a RequireJS for C/AL

RequireJS is a module loader for JavaScript that can retrieve and load JavaScript modules on demand.

Modules are loaded from:

- Microsoft Dynamics NAV Web Client top window
- Back-end C/AL request/response exchange
- URL

Benefits:

- Control Add-in remains as lean as possible
- Scripts are loaded in the most efficient way possible
- Fallback mechanism ensures that scripts are always loaded
- Changes possible without redeploying add-in or modifying page code:
 - Upgrading existing functionality
 - Extending existing functionality with new features

Demo 6

Control Add-in Setup and RequireAL

Part 1

Optimizing Back End

Problem statement

Code is still tightly bound:

- Logic is removed from pages as much as possible; but
- UI logic, infrastructure logic and business logic are still tightly bound in a single codeunit

Solution:

- Applying design patterns



Polymorphism



Sad Truth About Polymorphism in C/AL

"There is no such thing as polymorphism in C/AL."
(some very astute C/AL developer)

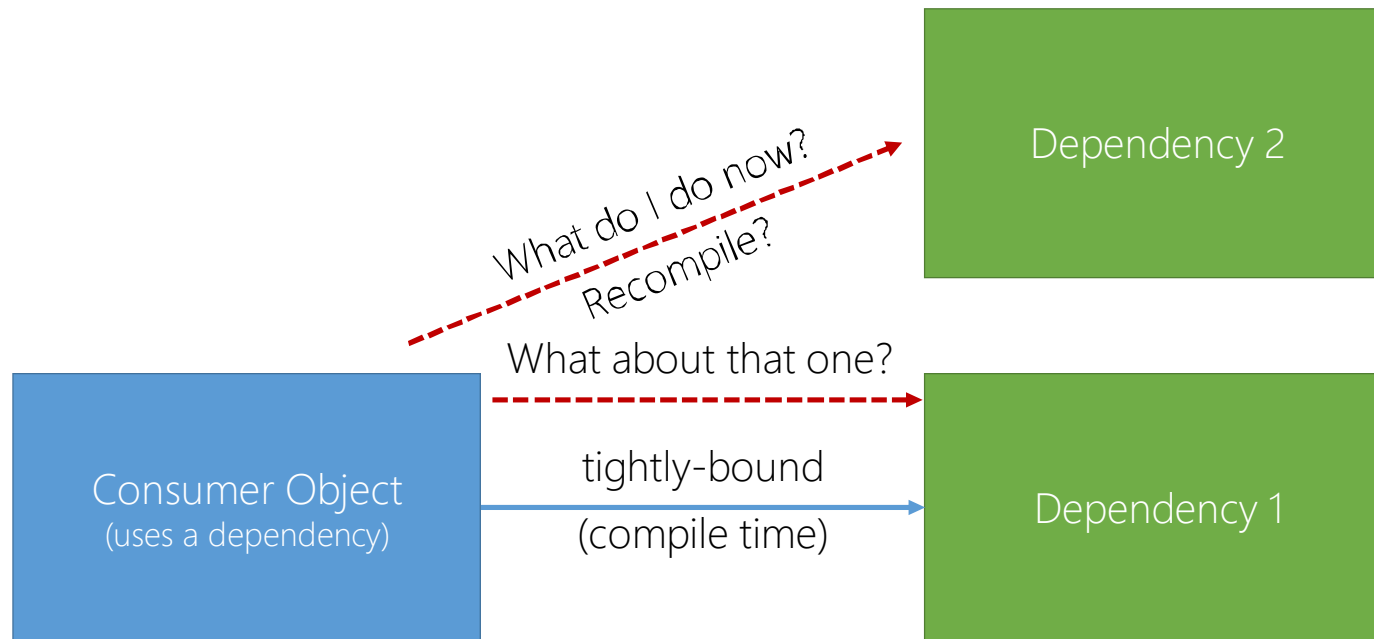
Polymorphic patterns

Strategy Command Producer-Consumer Delegation
Proxy Service Locator Prototype Twin Factory
Builder Provider Facade Dependency Injection
Composite Visitor Bridge Transfer Object
Repository Plug-in Servant Template Method
Mediator

Polymorphic design patterns (most common)

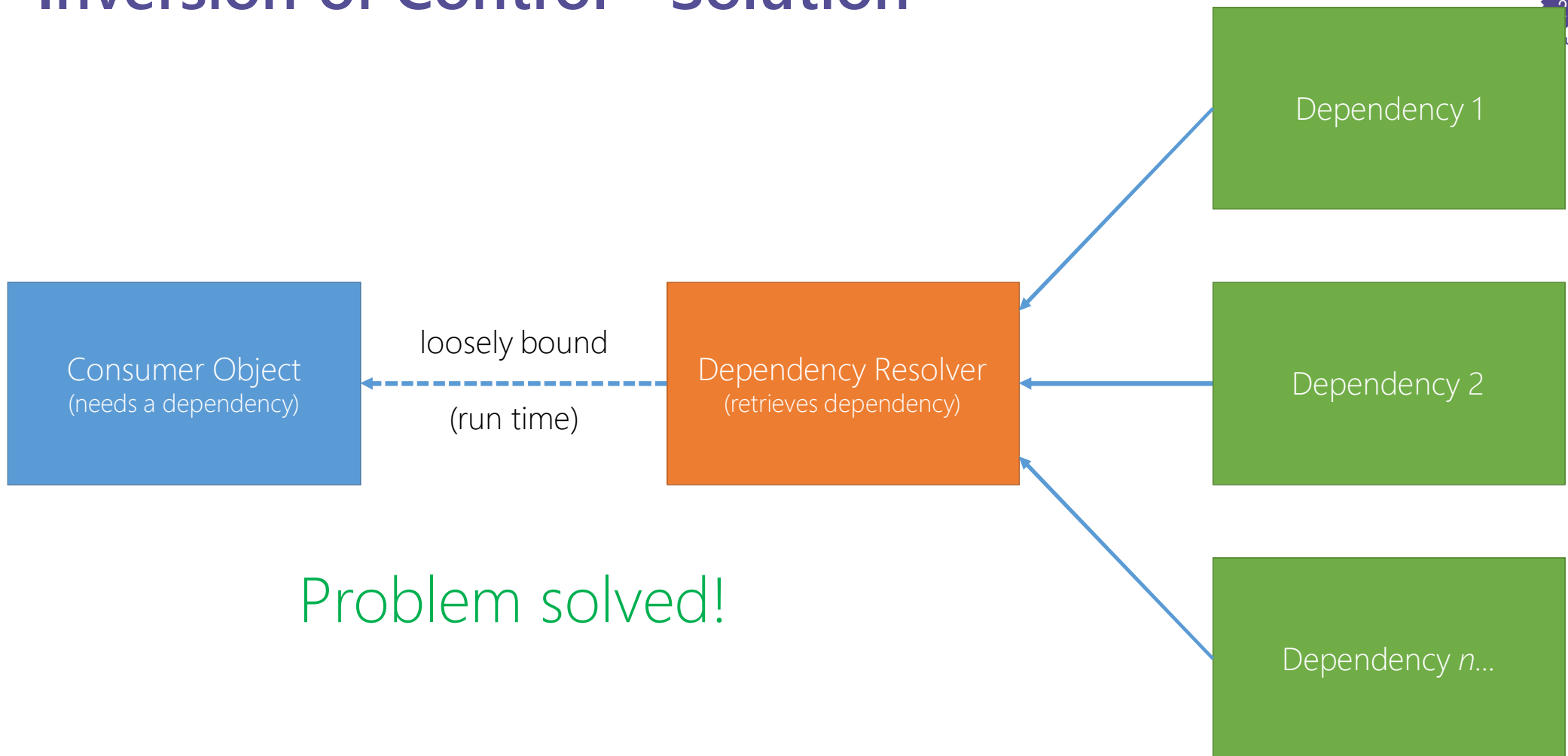
Inversion of Control

Inversion of Control - Problem



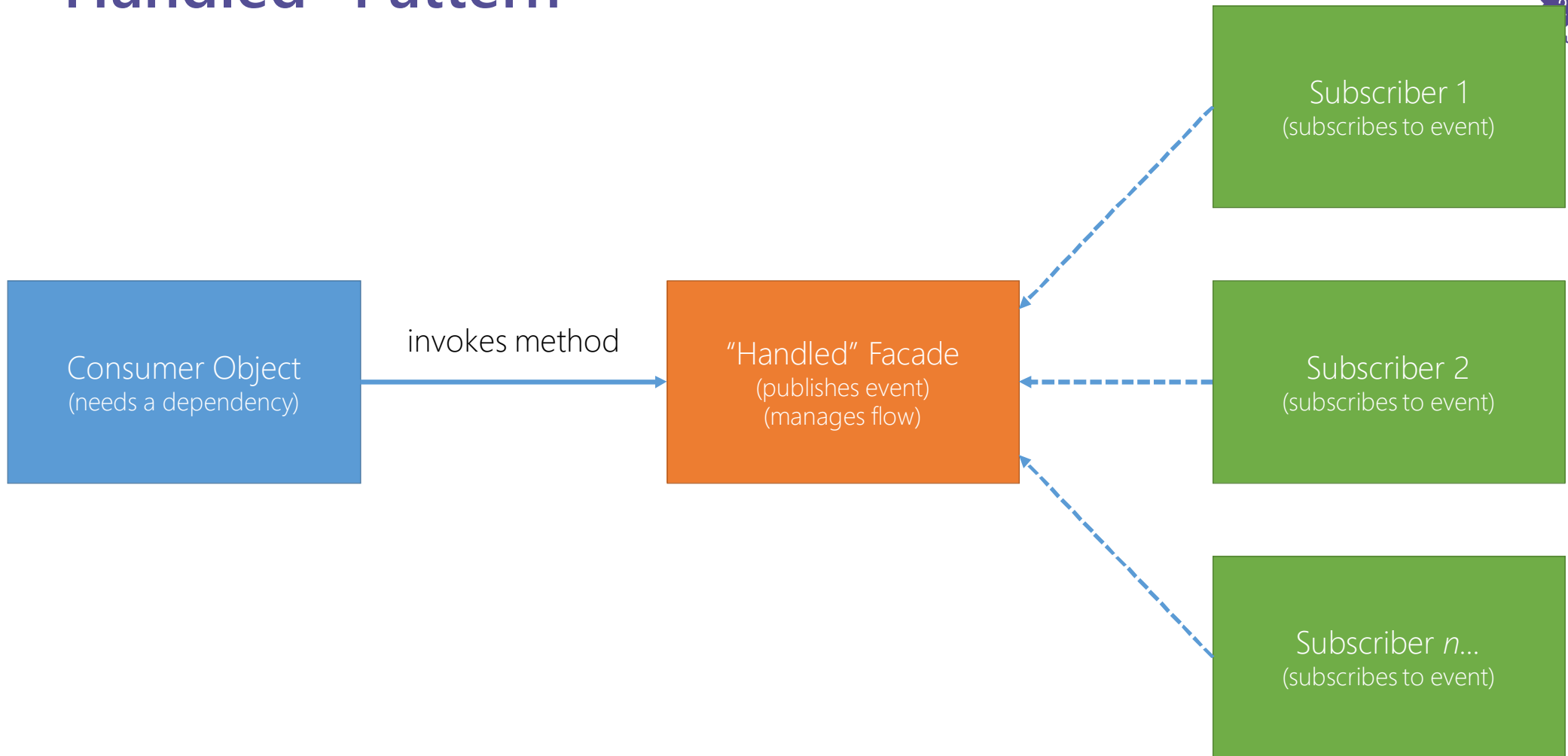
This is obviously a bad practice!

Inversion of Control - Solution



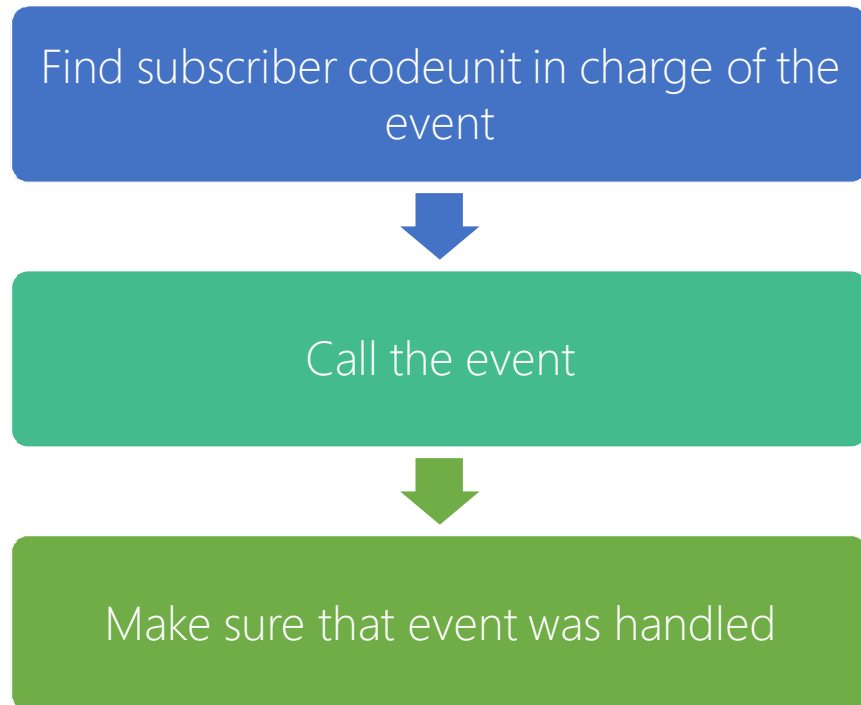
Problem solved!

"Handled" Pattern

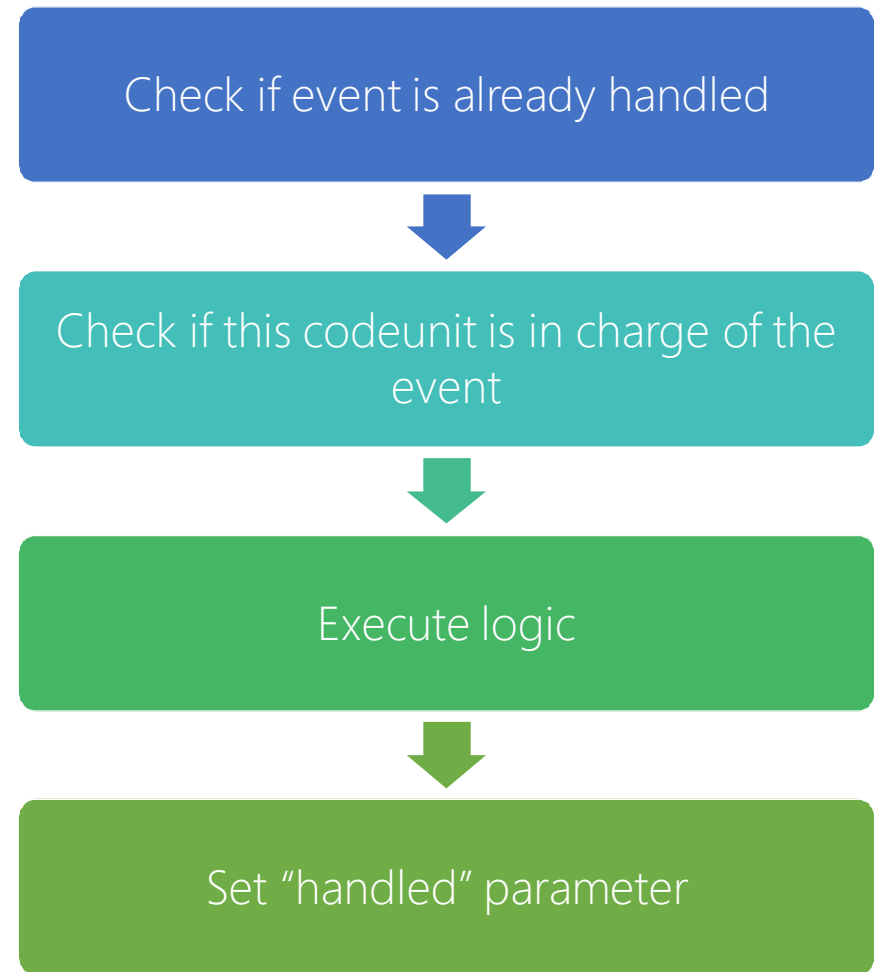


"Handled" Pattern Flow

Publisher



Subscriber



A Closer Look: "Handled" Pattern

```
Codeunit 50050 Event Log - Handled - C/AL Editor

5 LogEvent(VAR EventArgs : Record "Event Log Arguments")
6 // Retrieve handler identifier
7 GetHandler(CodeunitID);
8
9 // Call event
10 OnLogEvent(CodeunitID,EventArgs,Handled);
11
12 // Make sure event was handled
13 IF NOT Handled THEN
14     ERROR(ErrorNotHandled);
15
16 LOCAL [IntegrationEvent] OnLogEvent(HandlerCodeunit : Integer;VAR EventArgs : Record "Event Log Arguments";VAR Handled : Boolean)
17
18 LOCAL GetHandler(VAR CodeunitID : Integer)
19 IF EventLogSetup.GET AND (EventLogSetup."Event Log Handler Codeunit" <> 0) THEN
20     CodeunitID := EventLogSetup."Event Log Handler Codeunit";
```

```
Codeunit 50051 Event Log - Handled File - C/AL Editor

5 LOCAL [EventSubscriber] LogEvent(HandlerCodeunit : Integer;VAR EventArgs : Record "Event Log Arguments")
6 IF Handled OR (HandlerCodeunit <> CODEUNIT::"Event Log - Handled File") THEN
7     EXIT;
8
9 IF NOT Setup.GET OR (Setup."Log File Path" = '') THEN
10     EXIT;
11
12 File.AppendAllText(
13     Setup."Log File Path",
14     STRSUBSTNO(
15         EntryLine,
16         CURRENTDATETIME,
17         PADSTR(EventArgs.Source,20),
18         EventArgs.Message,
19         Environment.NewLine));
20
21 Handled := TRUE;
```

```
Codeunit 50052 Event Log - Handled Event Log - C/AL Editor

5 LOCAL [EventSubscriber] LogEvent(HandlerCodeunit : Integer;VAR EventArgs : Record "Event Log Arguments")
6 IF Handled OR (HandlerCodeunit <> CODEUNIT::"Event Log - Handled Event Log") THEN
7     EXIT;
8
9 EntryType := EntryType.Information;
10
11 CASE EventArgs.Type OF
12     EventArgs.Type::Warning: EntryType := EntryType.Warning;
13     EventArgs.Type::Error: EntryType := EntryType.Error;
14 END;
15
16 String := STRSUBSTNO(EventEntryText,EventArgs.Source,EventArgs.Message);
17 EventMessage := String.Replace('\',Env.NewLine);
18 EventLog := EventLog.EventLog();
19 EventLog.WriteEntry(EventLogSource,EventMessage,EntryType);
20
21 Handled := TRUE;
```

"Handled" Pitfalls

```
LOCAL [EventSubscriber] LogEvent(HandlerCodeunit : Integer;VAR EventArgs : Re
IF HandlerCodeunit < CODEUNIT::"Event Log - Handled File" THEN
    EXIT;

IF NOT Setup.GET OR (Setup."Log File Path" = '') THEN
    EXIT;

File.AppendAllText(
    Setup."Log File Path",
    STRSUBSTNO(
        EntryLine,
        CURRENTDATETIME,
        PADSTR(EventArgs.Source,20),
        EventArgs.Message,
        Environment.NewLine));

//Handled := TRUE;
```

erCodeunit : Integer;VAR EventArgs : Rec
UNIT::"Event Log - Handled File") THEN

Obviously, we need to reach for the manual subscribers.

“Handled” Pitfalls after Manual Subscriptions

Subscriber
identification

Multiple
active
subscribers

Cross-call
state
preservation

Infrastructure
separation

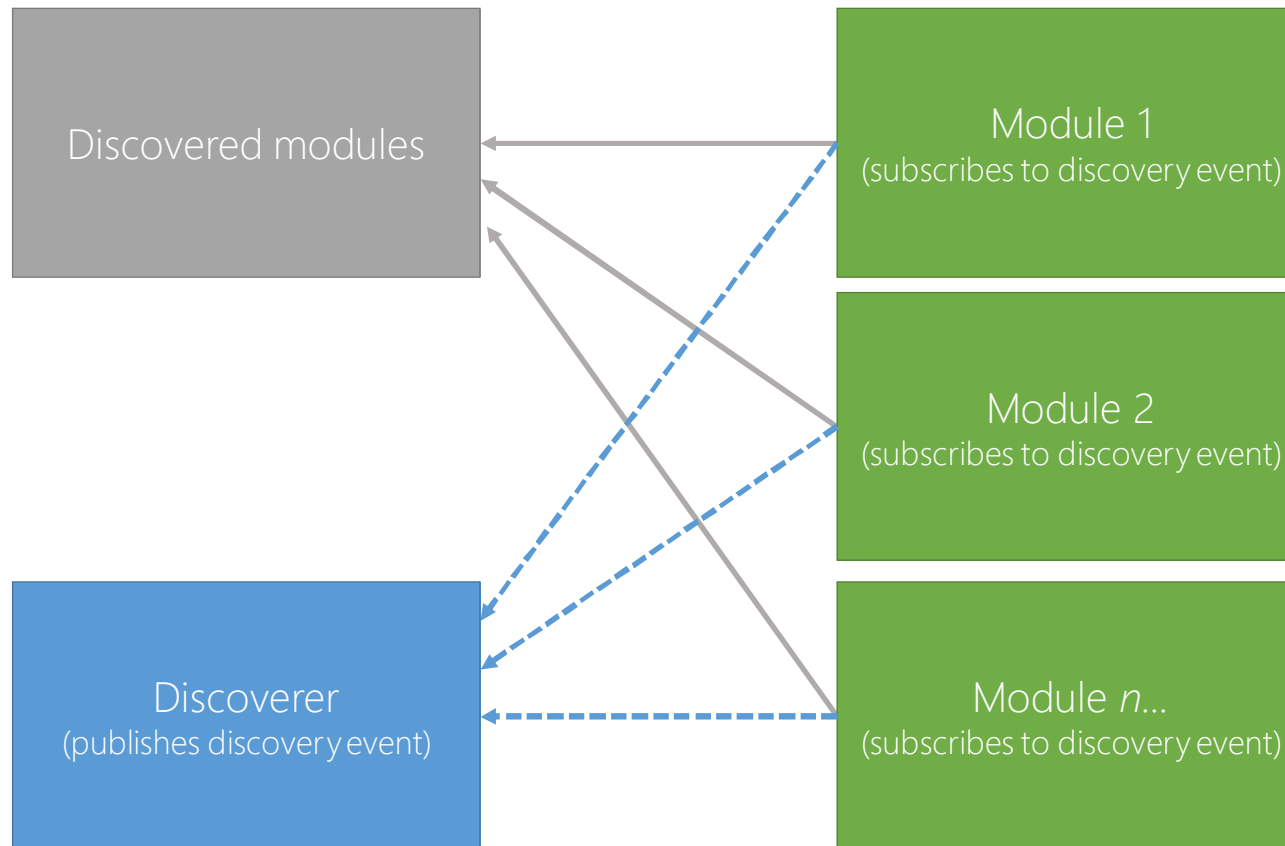
Infrastructure
fragility

Another event pattern can help...

Discovery Event

<https://community.dynamics.com/nav/w/designpatterns/271.discovery-event>

Discovery Event Pattern



Demo 7

Discovery Event Pattern

"Handled" Pitfalls After Discovery

Subscriber
identification

Multiple
active
subscribers

Cross-call
state
preservation

Infrastructure
separation

Infrastructure
fragility

Solution: Module Binder Pattern

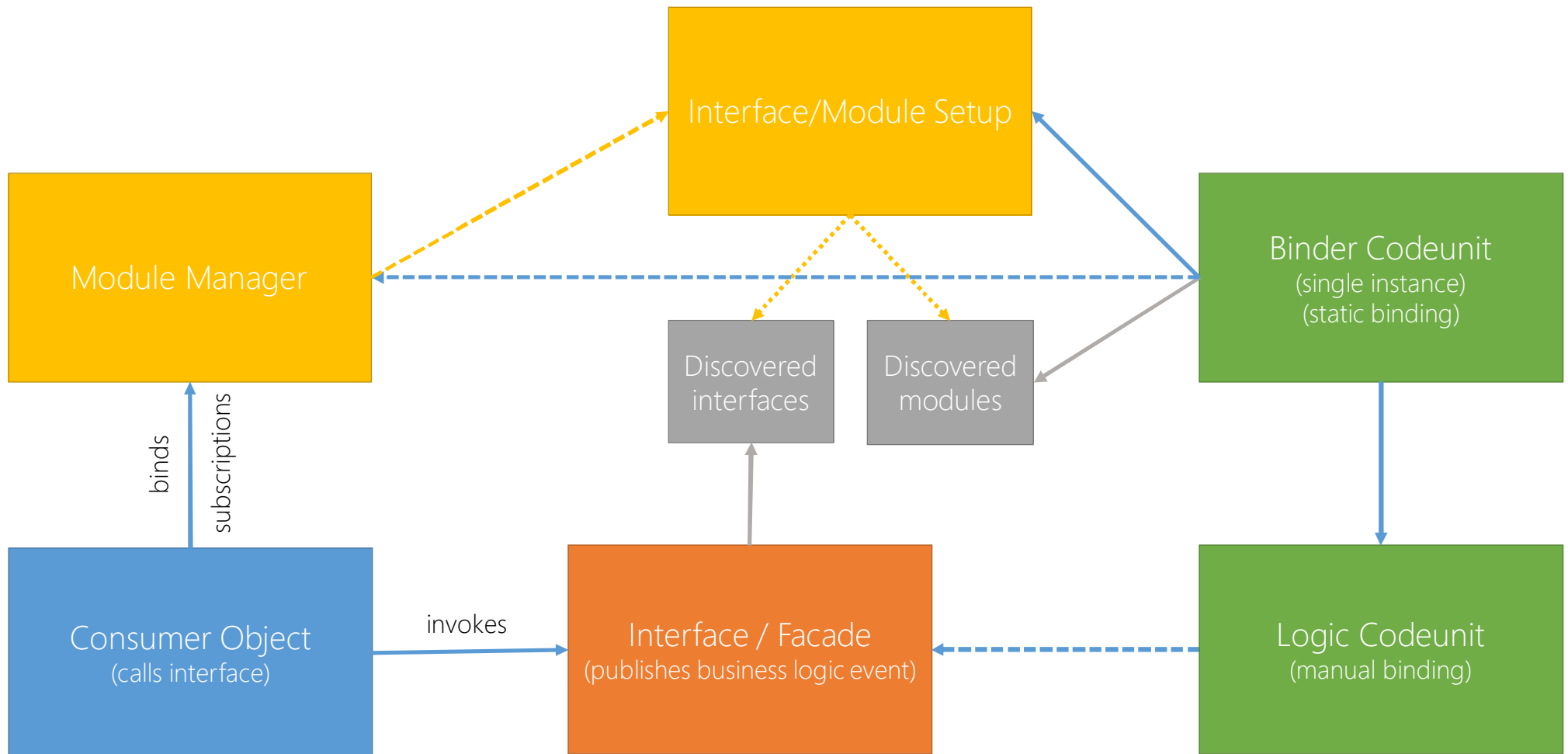
Combines:

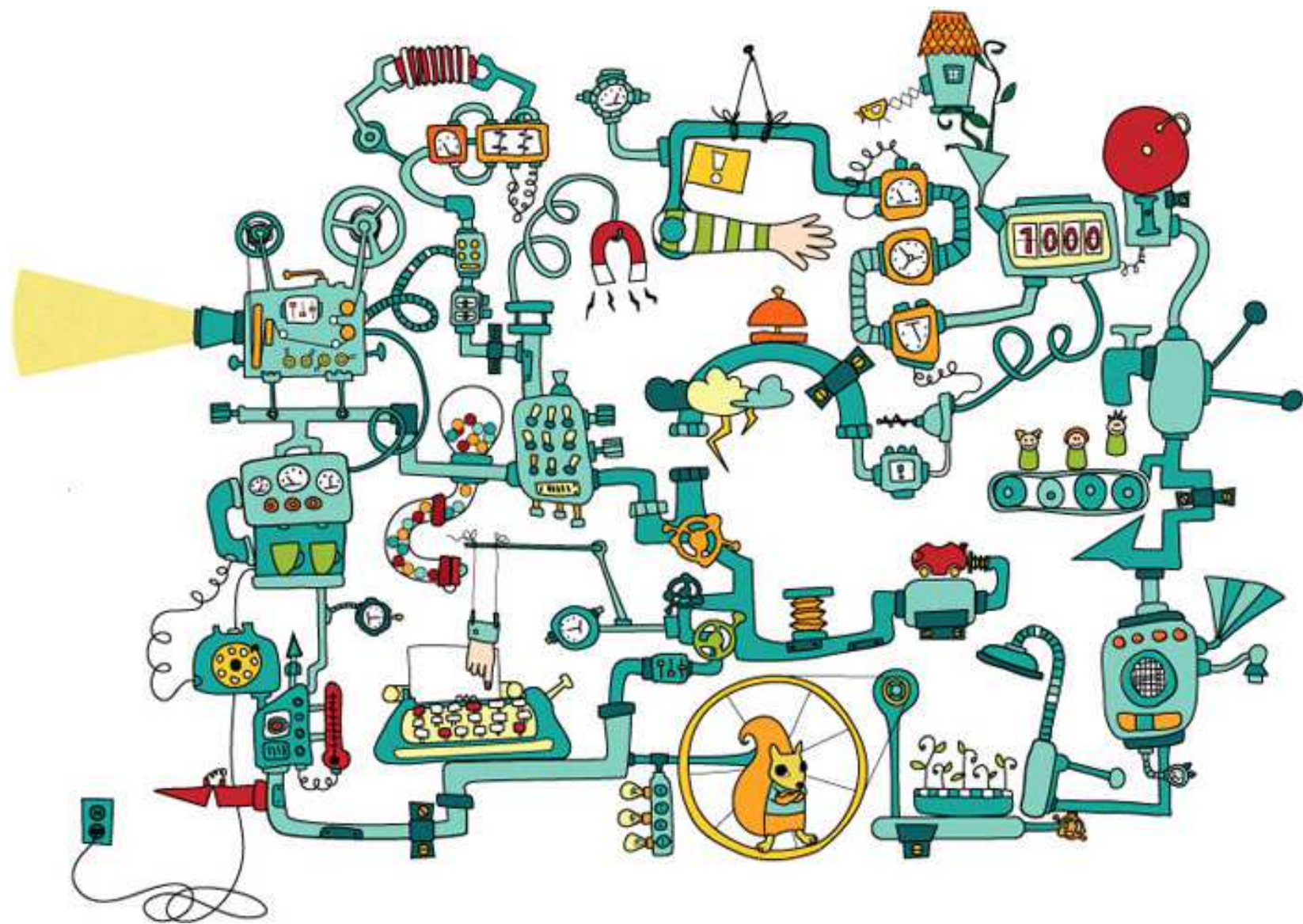
- Discovery Event Pattern
- "Handled" Pattern

Achieves:

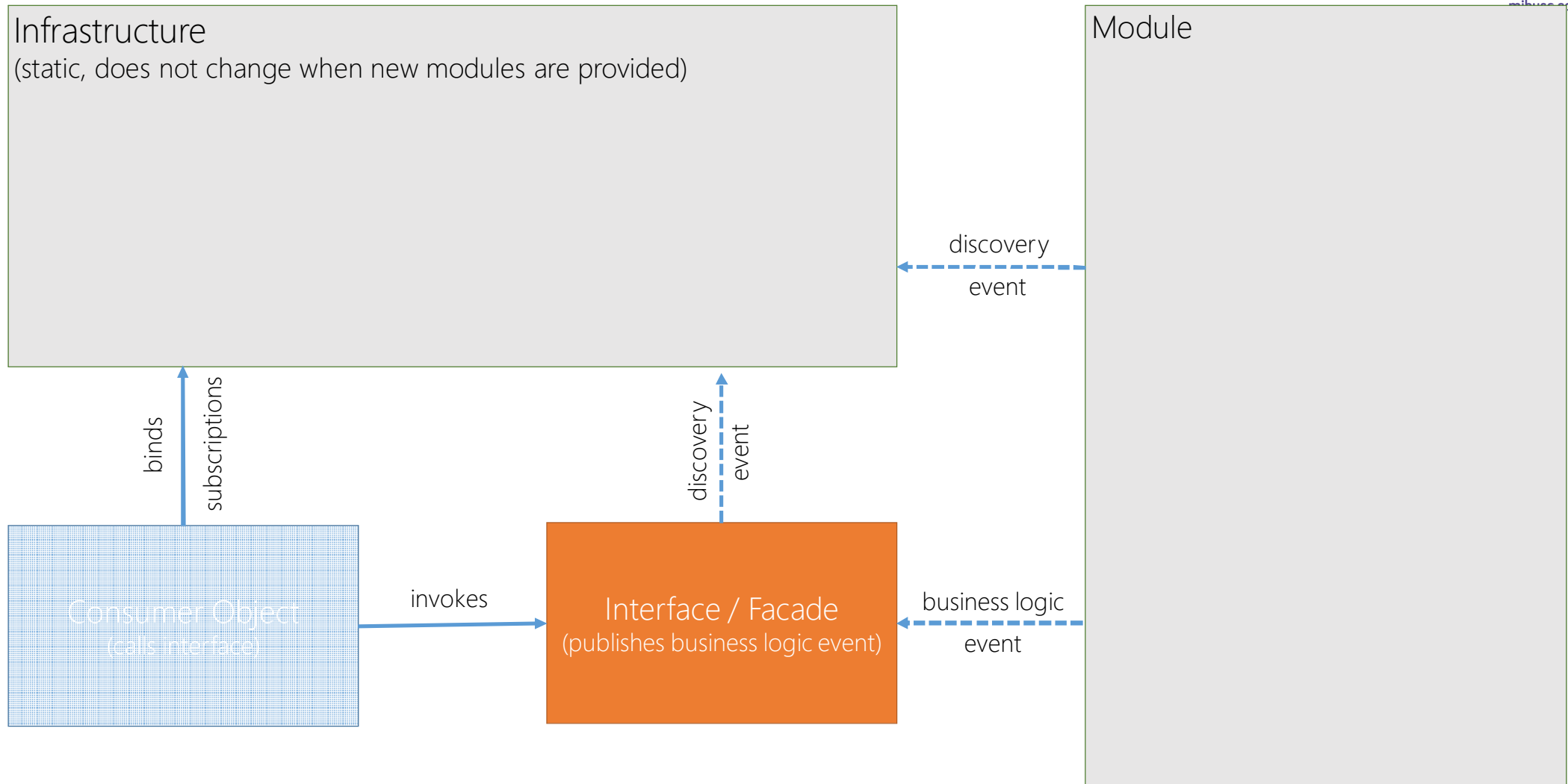
- Logic and infrastructure separation
- Infrastructure stability

A Closer Look: Module Binder Pattern





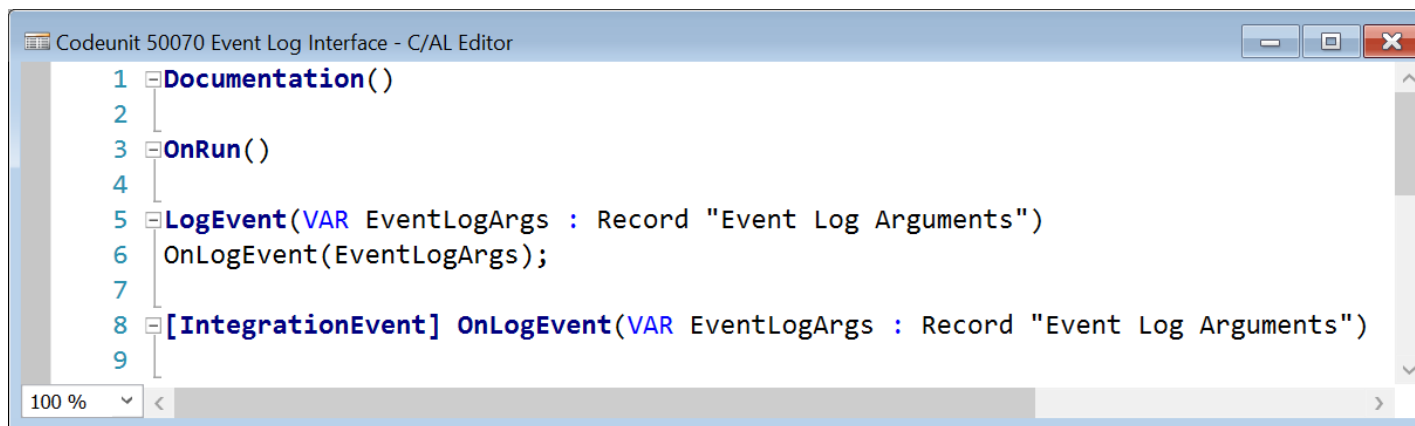
A Closer Look: Module Binder Pattern



Interface

Facade for the business logic
Publishes business logic event

Interface / Facade
(publishes business logic event)



```
Codeunit 50070 Event Log Interface - C/AL Editor
1 Documentation()
2
3 OnRun()
4
5 LogEvent(VAR EventLogArgs : Record "Event Log Arguments")
6   OnLogEvent(EventLogArgs);
7
8 [IntegrationEvent] OnLogEvent(VAR EventLogArgs : Record "Event Log Arguments")
9
```

Module

A two-codeunit block

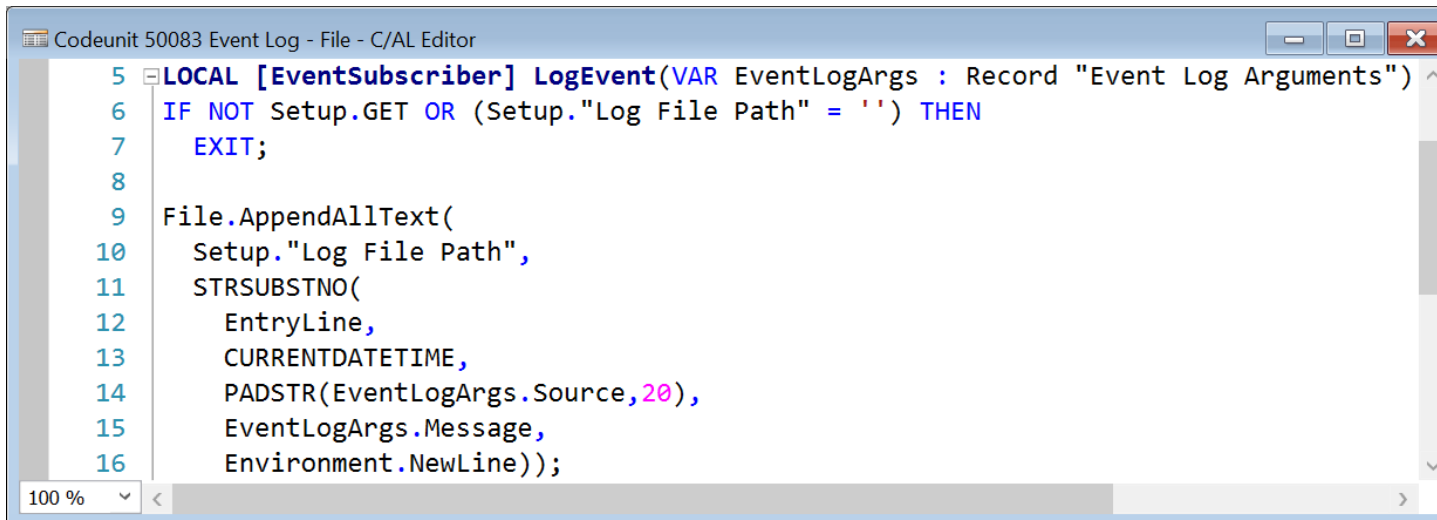
Separates business logic from infrastructure

Retains state

Module: Business Logic Codeunit

Subscribes to interface event
Executes specific business logic

Logic Codeunit
(manual binding)



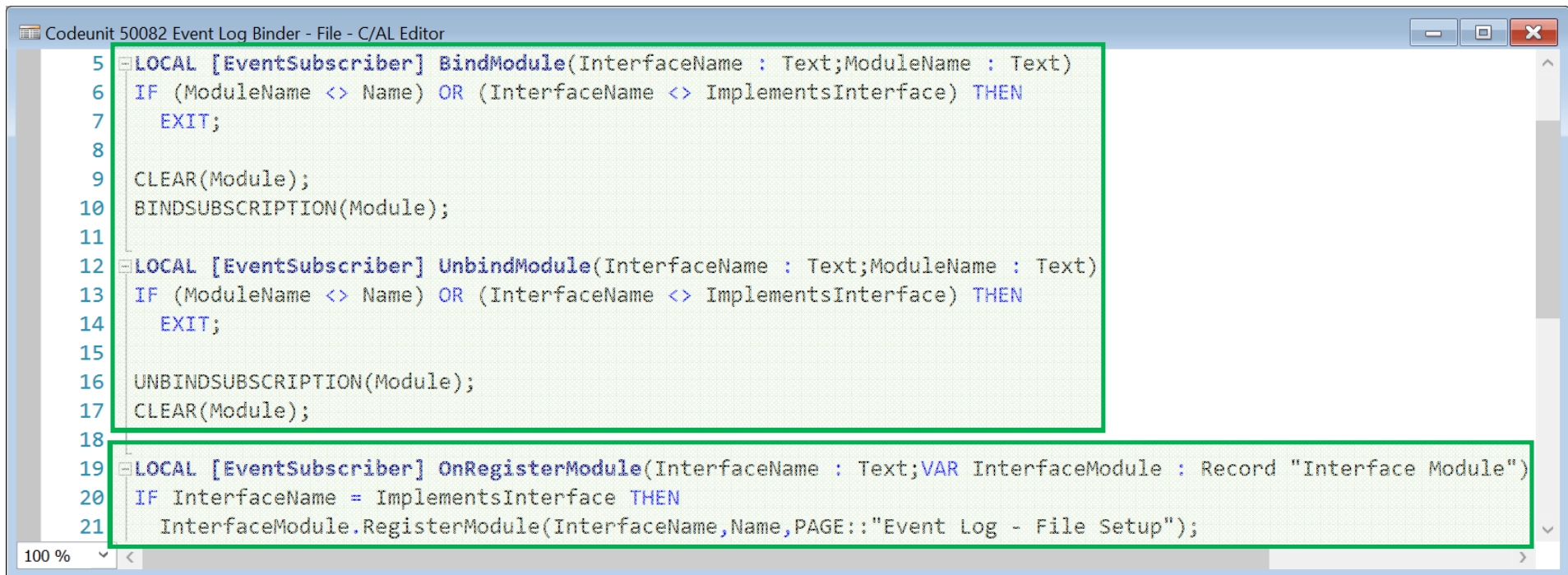
```
Codeunit 50083 Event Log - File - C/AL Editor

5 LOCAL [EventSubscriber] LogEvent(VAR EventLogArgs : Record "Event Log Arguments")
6 IF NOT Setup.GET OR (Setup."Log File Path" = '') THEN
7     EXIT;
8
9 File.AppendAllText(
10     Setup."Log File Path",
11     STRSUBSTNO(
12         EntryLine,
13         CURRENTDATETIME,
14         PADSTR(EventLogArgs.Source,20),
15         EventLogArgs.Message,
16         Environment.NewLine));
```

Module: Binder

Binds/unbinds business logic codeunit
Discovers module for a specific interface

Binder Codeunit
(single instance)
(static binding)



```
Codeunit 50082 Event Log Binder - File - C/AL Editor

5 LOCAL [EventSubscriber] BindModule(InterfaceName : Text;ModuleName : Text)
6 IF (ModuleName <> Name) OR (InterfaceName <> ImplementsInterface) THEN
7     EXIT;
8
9 CLEAR(Module);
10 BINDSUBSCRIPTION(Module);
11
12 LOCAL [EventSubscriber] UnbindModule(InterfaceName : Text;ModuleName : Text)
13 IF (ModuleName <> Name) OR (InterfaceName <> ImplementsInterface) THEN
14     EXIT;
15
16 UNBINDSUBSCRIPTION(Module);
17 CLEAR(Module);
18
19 LOCAL [EventSubscriber] OnRegisterModule(InterfaceName : Text;VAR InterfaceModule : Record "Interface Module")
20 IF InterfaceName = ImplementsInterface THEN
21     InterfaceModule.RegisterModule(InterfaceName,Name,PAGE::"Event Log - File Setup");
```

Module Manager

Handles discovery of interfaces and modules

Handles binding/unbinding of module binders

Module Manager

```
Codeunit 50060 Module Management - C/AL Editor
5 BindInterface(InterfaceName : Text;WithError : Boolean)
6 IF NOT Setup.GET(InterfaceName) THEN BEGIN
7     IF WithError THEN
8         ERROR(ErrorNoSetup,InterfaceName,SetupPage.CAPTION);
9     MESSAGE(ErrorNoSetup,InterfaceName,SetupPage.CAPTION);
10    EXIT;
11 END;
12
13 BindModule(InterfaceName,Setup."Module Name");
14
15 UnbindInterface(InterfaceName : Text)
16 IF NOT Setup.GET(InterfaceName) THEN
17     EXIT;
18
19 UnbindModule(InterfaceName,Setup."Module Name");
20
21 LOCAL BindModule(InterfaceName : Text;ModuleName : Text)
22 OnBindModule(InterfaceName,ModuleName);
23
24 LOCAL UnbindModule(InterfaceName : Text;ModuleName : Text)
25 OnUnbindModule(InterfaceName,ModuleName);
```

Challenges

Call
multiplication

Explicit
subscriptions

Rogue
subscribers

Single
subscribers

Solution

Use Event Subscription virtual table to:

- Verify that there are only manual subscribers

- Verify that there are active subscribers before calling the method

```
[TryFunction] MakeSureSubscriptionsAreManual(CodeunitID : Integer)
WITH Subscription DO BEGIN
    SETRANGE("Publisher Object Type","Publisher Object Type"::Codeunit);
    SETRANGE("Publisher Object ID",CodeunitID);
    SETRANGE("Subscriber Instance",'Static-Automatic');
    IF FINDFIRST THEN BEGIN
        AllObj.GET(AllObj."Object Type"::Codeunit,"Subscriber Codeunit ID");
        ERROR(ErrorManualSubscription,CodeunitID,"Subscriber Codeunit ID",AllObj."Object Caption");
    END;
END;
```

```
[TryFunction] MakeSureSubscriberExists(CodeunitID : Integer;FunctionName : Text)
WITH Subscription DO BEGIN
    SETRANGE("Publisher Object Type","Publisher Object Type"::Codeunit);
    SETRANGE("Publisher Object ID",CodeunitID);
    SETRANGE("Published Function",FunctionName);
    SETRANGE("Subscriber Instance",'Manual');
    SETRANGE(Active,TRUE);
    SETFILTER("Active Manual Instances",'>=%1',1);
    IF ISEMPY THEN
        ERROR(ErrorNoSubscriber,FunctionName,CodeunitID);
    END;
```

Demo 8

Module Binder Pattern

Combining Module Binder with Control Add-ins

Specific Control Add-in functionality becomes a specific module:

- Control Add-in interface defines general functionality of (all) control add-ins
- Individual module binder handles infrastructure:
 - Binds an individual module to the interface
 - Responds to custom JavaScript events that are too generic for the interface itself
 - Invokes module
- Individual module:
 - Responds to specific Control Add-in events (e.g. requesting UI customizations)
 - Performs specific Control Add-in behavior

Examples:

- Chess
- Twitter

Problem statement (and a crazy solution for the end)

Reference to JavaScript Control Add-in may proliferate across the application.

Reason:

- Each object that requires JavaScript will have to be made aware of it explicitly

Issues:

- A lot of code duplication
- Bad maintainability

Goal:

- Enable event-based JavaScript discovery



Demo 9

Control Add-in Module Binder

**NAV
TECH
DAYS
2016**

mibuso.com

THANK YOU!

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

**NAV
TECH
DAYS
2016**

mibuso.com