



Visual Studio **ALM** Days 2012



TFS Pluggins

Thomas Trotzki, artiso, ALM Consultant

Motivation

Das Vorbild – TFS Build Automation

Buids werden seit TFS 2010 mittels Windows Workflow Foundation automatisiert

Build Process Templates als XAML, grafischer Editor, Debugging, ...

Erweiterbarkeit – eigene Build Process Templates und eigene Workflow Activities

Das Problem – Kontext “Build” passt nicht immer

Automatisierung bei Planung – Zuweisung einer User Story zur Umsetzung (workitem state)

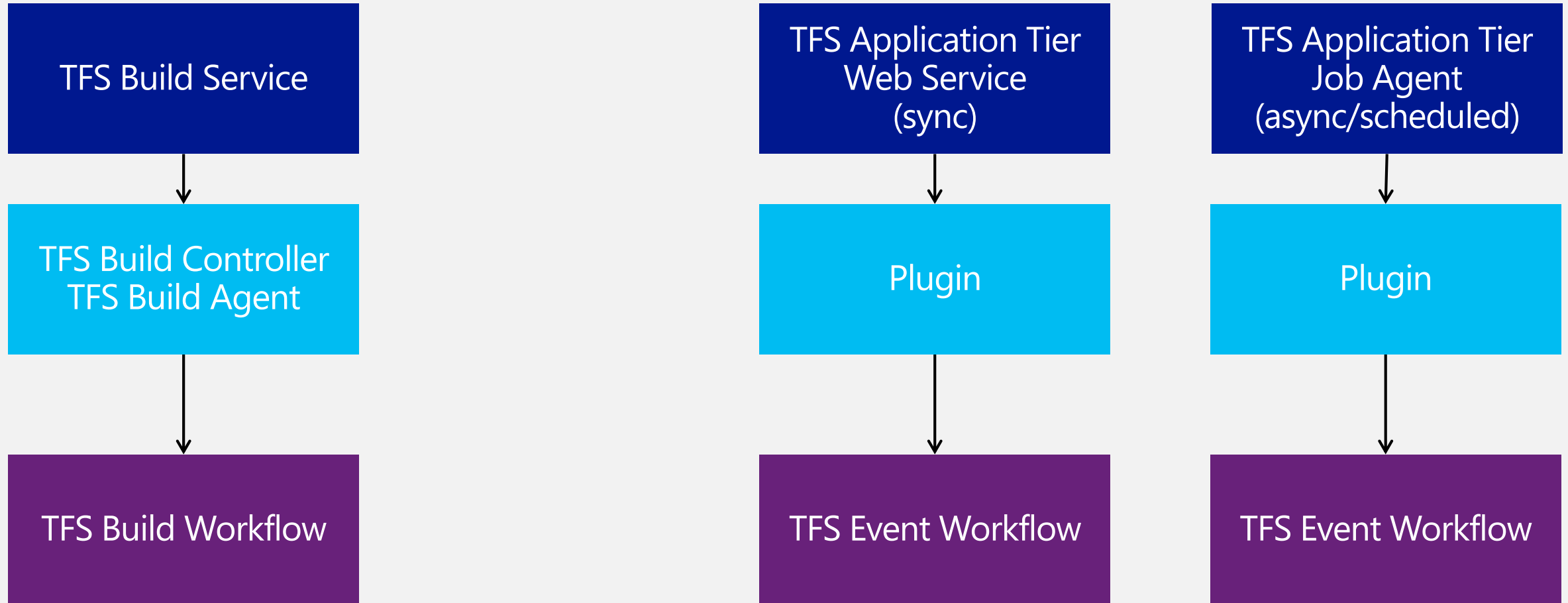
Automatisierung bei Freigabe – Freigabe eines Builds durch QM (build quality)

Die Lösung – TFS Event Workflows

Server Sided TFS Plugins – Trigger durch Events auf dem TFS Application Tier

Kombination mit Windows Workflow Foundation – teilweise Reuse der Build Workflows / Activities

Architektur



TFS Web Services - Plugins

TFS Build Service

TFS Build Controller
TFS Build Agent

TFS Application Tier
Web Service
(sync)

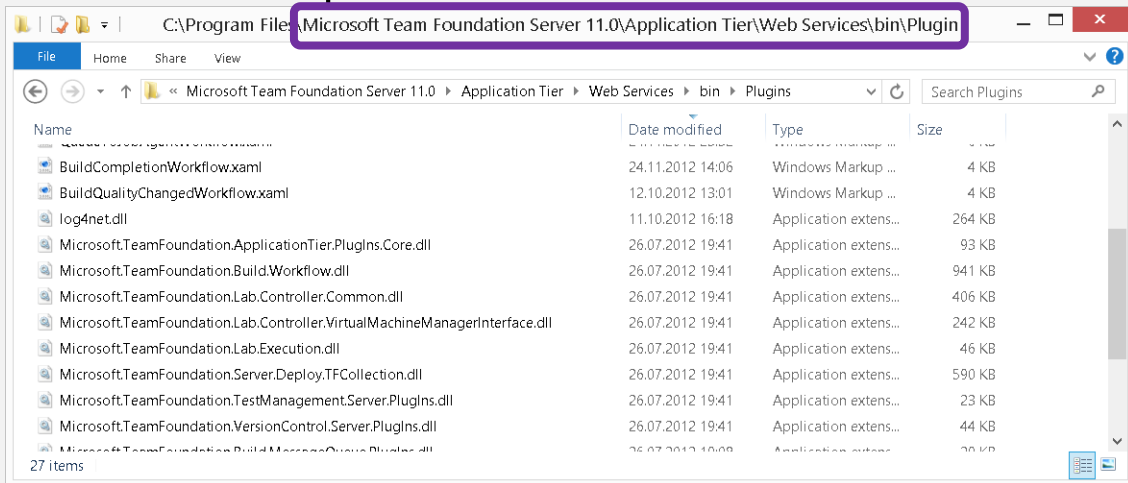
Plugin

TFS Event Workflow

TFS Application Tier
Job Agent
(async/scheduled)

Plugin

TFS Event Workflow



TFS Web Services – Plugins

```
public interface ISubscriber
```

```
{
```

```
    string Name { get; }
```

```
    SubscriberPriority Priority { get; }
```

```
    Type[] SubscribedTypes();
```

```
    EventNotificationStatus ProcessEvent(
```

```
        TeamFoundationRequestContext requestContext,
```

```
        NotificationType notificationType,
```

```
        object notificationEventArgs,
```

```
        out int statusCode,
```

```
        out string statusMessage,
```

```
        out ExceptionPropertyCollection properties);
```

```
}
```

Welche Events
interessieren?

Event Handler

Request Context

Notification
DecisionPoint

Event spezifische
Daten

DecisionPoint
Failure Info

TFS Web Services – Plugins

```
public Type[] SubscribedTypes()
{
    return new Type[]
    {
        typeof(WorkItemChangedEvent),
        typeof(BuildCompletionNotificationEvent),
        typeof(BuildQualityChangedNotificationEvent),
        typeof(BuildDefinitionChangedEvent),
        typeof(CheckinNotification),
        typeof(TestSuiteChangedNotification),
        // ...
    };
}
```

Workitem
Tracking

Build

Version Control

Testing

... and a lot more

TFS Web Services – Plugins

```
public EventNotificationStatus ProcessEvent(TeamFoundationRequestContext requestContext,
NotificationType notificationType, object notificationEventArgs, ... )
{
    // we only handle notifications
    if (notificationType != NotificationType.Notification)
        return EventNotificationStatus.ActionPermitted;

    WorkItemChangedEvent workitemChangedEvent = notificationEventArgs as WorkItemChangedEvent;
    if (null != workitemChangedEvent)
    { // ... handle workitem changed event }

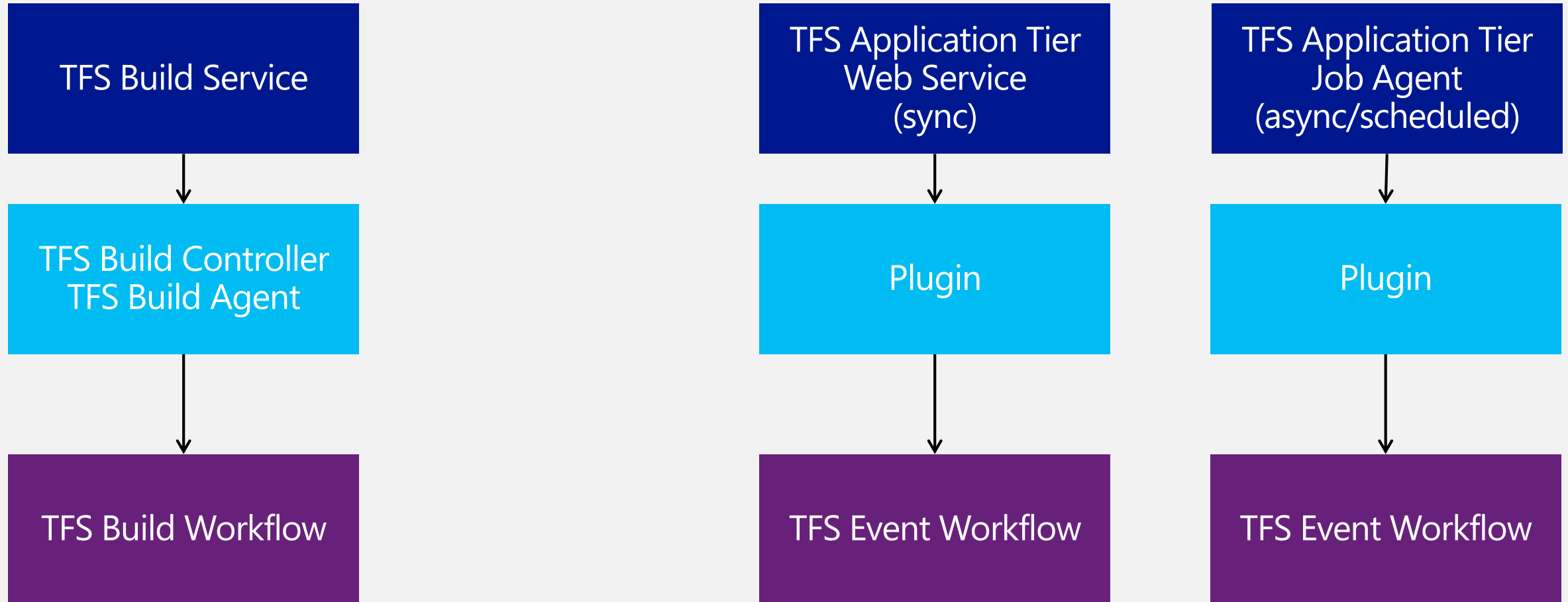
    return EventNotificationStatus.ActionPermitted;
}
```

Alle DecisionPoint erlauben

Event Typ abfragen und casten

Event bearbeiten

Workflows



Workflows – Laden und Ausführen

```
public EventNotificationStatus ProcessEvent(TeamFoundationRequestContext requestContext,
NotificationType notificationType, object notificationEventArgs, ... )
{
    // ...
    try
    {
        var workflow = this.GetWorkflow("workitemChangedWorkflow.xaml");
        var workflowParameters = new Dictionary<string, object> {
            { "TFSEvent", notificationEventArgs },
            { "TeamFoundationRequestContext", requestContext }
        };
        WorkflowInvoker.Invoke(workflow, workflowParameters);
    }
    catch (Exception e) { // ... }
    // ...
}
```

Workflow laden
Xaml/dll

Parameter festlegen

Ausführen

Workflows – Editor

The screenshot shows the Visual Studio Workflow Editor for a file named 'TFSEventWorkflows - DumpWorkitem.xaml'. The workflow diagram, titled 'DumpWorkitem', contains a 'MainWorkflow' container with two activities: 'GetWorkItemChangedEven' and 'WriteLogMessage'. Below the diagram is a table defining the workflow's arguments.

Name	Direction	Argument type	Default value
TFSEvent	In	WorkItemChangedI	<input type="text" value="Enter a VB expression"/>
TeamFoundationRequestContext	In	TeamFoundationRe	<input type="text" value="Enter a VB expression"/>

At the bottom of the editor, there are tabs for 'Variables', 'Arguments' (which is selected), and 'Imports'. The status bar at the very bottom shows a zoom level of 100%.

Workflow Activities

Parameter festlegen

Demo

Debug TFS Web Service Plugin

Workflows – Eigene Code Activities

```
public sealed class GetTeamProjectCollection : CodeActivity
```

CodeActivity

```
{
```

```
    [RequiredArgument]
```

```
    public InArgument<TeamFoundationRequestContext> requestContext { get; set; }
```

Parameter festlegen

```
    [RequiredArgument]
```

```
    public OutArgument<TfsTeamProjectCollection> teamProjectCollection { get; set; }
```

```
    protected override void Execute(CodeActivityContext context)
```

Ausführen

```
{
```

```
    TeamFoundationRequestContext requestContext = context.GetValue(this.requestContext);
```

```
    var tfLocationService = requestContext.GetService<TeamFoundationLocationService>();
```

```
    var accessMapping = tfLocationService.GetServerAccessMapping(requestContext);
```

```
    string teamProjectCollectionUrl = tfLocationService.GetHostLocation(
        requestContext, accessMapping);
```

InParameter
abfragen

```
    var teamProjectCollection = new TfsTeamProjectCollection(
        new Uri(teamProjectCollectionUrl));
```

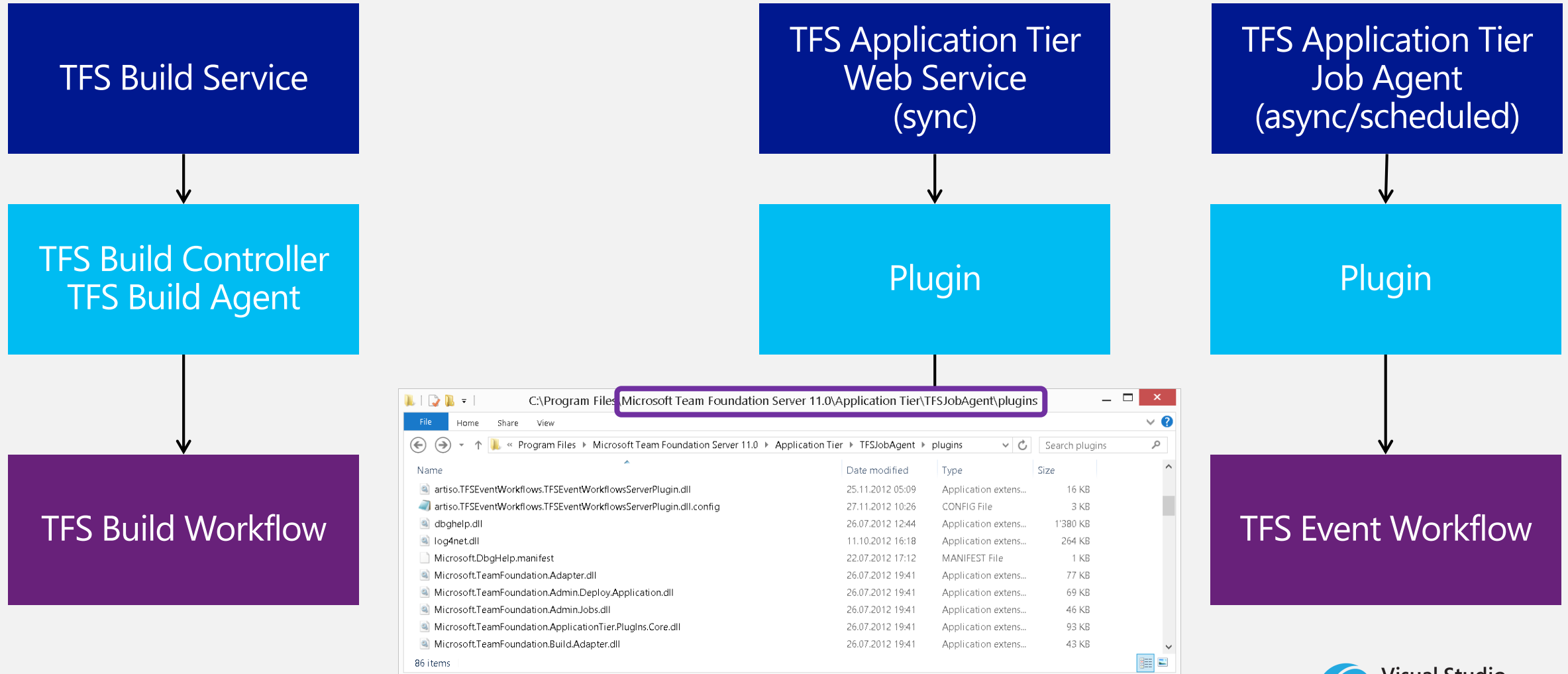
```
    context.SetValue(this.teamProjectCollection, teamProjectCollection);
```

OutParameter
festlegen

```
}
```

```
}
```

TFS Job Agent



Job Agent - Queueing

```
public sealed class QueueToJobAgent : CodeActivity
{
    // ...
    protected override void Execute(CodeActivityContext context)
    {
        object notificationEventArgs = context.GetValue(this.notificationEventArgs);
        TeamFoundationRequestContext requestContext = context.GetValue(this.requestContext);
        XmlNode xmlData = SerializeXml(notificationEventArgs);
        var jobService = requestContext.GetService<TeamFoundationJobService>();
        var jobGuid = jobService.QueueOneTimeJob(
            requestContext, "TFSEventWorkflow Job",
            "artiso.TFSEventWorkflows.TFSEventWorkflowsServerPlugin.WorkflowRunner",
            xmlData,
            false);
        int result = jobService.QueueJobsNow(requestContext, new List<Guid>() { jobGuid }, true);
    }
    // ...
}
```

RequestContext und Event als Parameter

Event in Xml serialisieren

Job Service ermitteln

Job Queueen

Optional: Priorität erhöhen

Job Agent - Ausführen

```
public class WorkflowRunnerJob : ITeamFoundationJobExtension
{
    public TeamFoundationJobExecutionResult Run(
        TeamFoundationRequestContext requestContext,
        TeamFoundationJobDefinition jobDefinition,
        DateTime queueTime,
        out string resultMessage)
    {
        XmlNode xmlData = jobDefinition.Data;
        object notificationEventArgs = DeserializeXml(xmlData);
        this.workflowRunner.ProcessEvent(requestContext, notificationEventArgs);

        return TeamFoundationJobExecutionResult.Succeeded;
    }
}
```

Job Extension

RequestContext
JobDefinition

Xml Daten
abrufen

Event
deserialisieren

Event
bearbeiten

Demo

Debug TFS Job Agent Plugin

Beispiele – Workitem Tracking

Neue “User Story”

Neue Tasks erstellen

Neue TestCases erstellen

➔ `Microsoft.TeamFoundation.WorkItemTracking.Server.WorkItemChangedEvent`

Aggregation

Aufwand auf Parent aggregieren

Status-Änderungen aggregieren

➔ `Microsoft.TeamFoundation.WorkItemTracking.Server.WorkItemChangedEvent`

Achtung: lange Laufzeit, Rekursion, JobAgent einbinden

Beispiele – Version Control

Checkin von Quellcode

Build triggern

➔ CI Build einrichten

Checkin von Quellcode

Change History erstellen

➔ `Microsoft.TeamFoundation.VersionControl.Server.CheckinNotification`

Checkin einer Spezifikation

“Task” für Sichtung der Spezifikation erstellen

Word Dokumenten-Gliederung in Workitem-Tree umsetzen

➔ `Microsoft.TeamFoundation.VersionControl.Server.CheckinNotification`

Beispiele – Build

Anstoßen eines Deployment Builds

Erfolgreicher „Nightly“ triggert automatisch Lab Build

➔ `Microsoft.TeamFoundation.Build.Server.BuildCompletionNotificationEvent`

Änderung der Build Quality

Anlegen einer „Task“

Deployment

➔ `Microsoft.TeamFoundation.Build.Server.BuildQualityChangedNotificationEvent`

Änderung einer BuildDefinition

BuildDefinition in Version Control einchecken

➔ `Microsoft.TeamFoundation.Build.Server.BuildDefinitionChangedEvent`

Beispiele – Testing

Anstoßen eines Deployment Builds

Änderung an TestSuite löst Lab Build aus

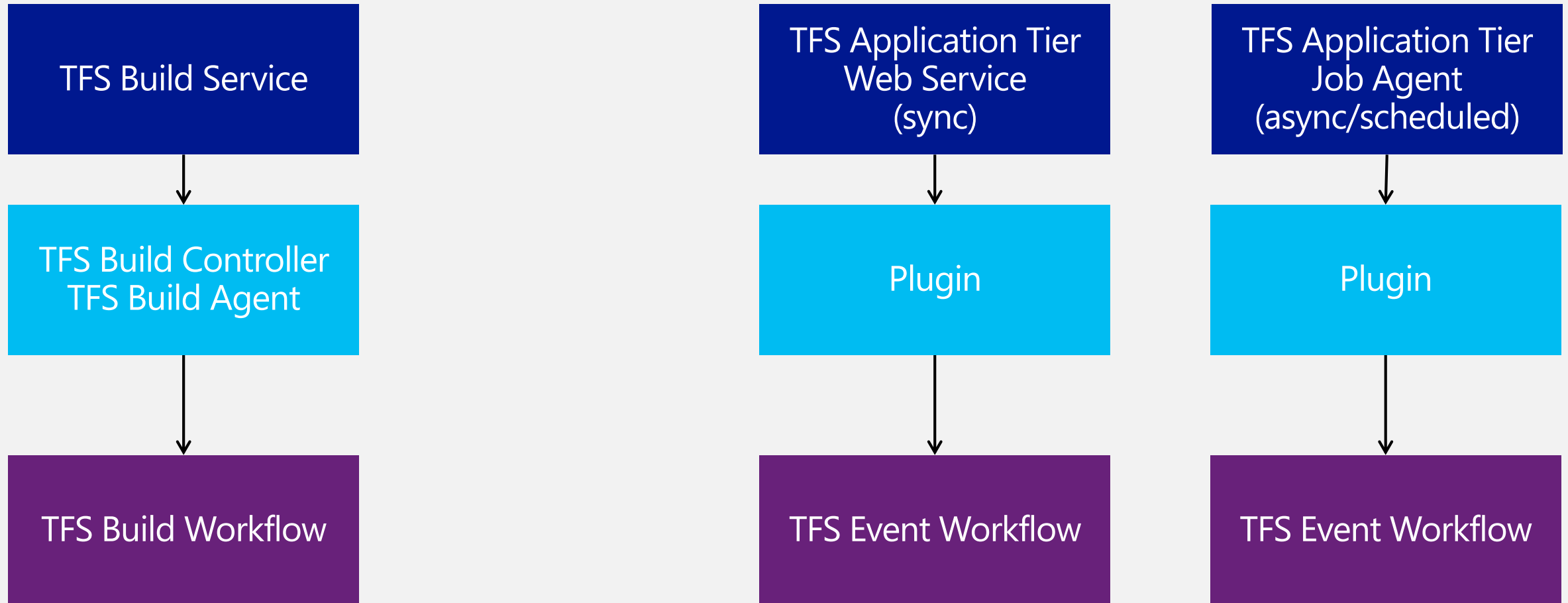
➔ `Microsoft.TeamFoundation.TestManagement.Server.TestSuiteChangedNotification`

Änderung einer TestSuite

TestSuite in Version Control einchecken

➔ `Microsoft.TeamFoundation.TestManagement.Server.TestSuiteChangedNotification`

Bring It Together



Bring It Together

Vorhandene Build Workflows

Sie haben schon in Build Automatisierung investiert

Aufsplitten

Alles was keinen BuildController Context bzw. BuildAgent Context erfordert kann problemlos in einem Event Workflow genutzt werden

Alles Andere: Event Workflow triggert Build!



Visual Studio **ALM** Days 2012



TFS Pluggins

Visit <http://tfseventworkflows.codeplex.com>

Fragen?

Herzlichen Dank