



# Real life experiences with the SharePoint Workflow

Paolo Pialorsi  
Senior Consultant  
PiaSys.com  
@paolopia

#SPSUK

This session mainly targets  
Developers and Workflow Designers  
that use SPD2013 and VS2013

# Thank You to Our Sponsors!



# About Me

- Project Manager, Consultant, Trainer
- More than 40 Microsoft certification exams passed, including MC(S)M
- Focused on SharePoint since 2002
- Author of 10 books about XML, SOAP, .NET, LINQ, and SharePoint
- Speaker at main IT conferences worldwide
- <http://www.piasys.com/>



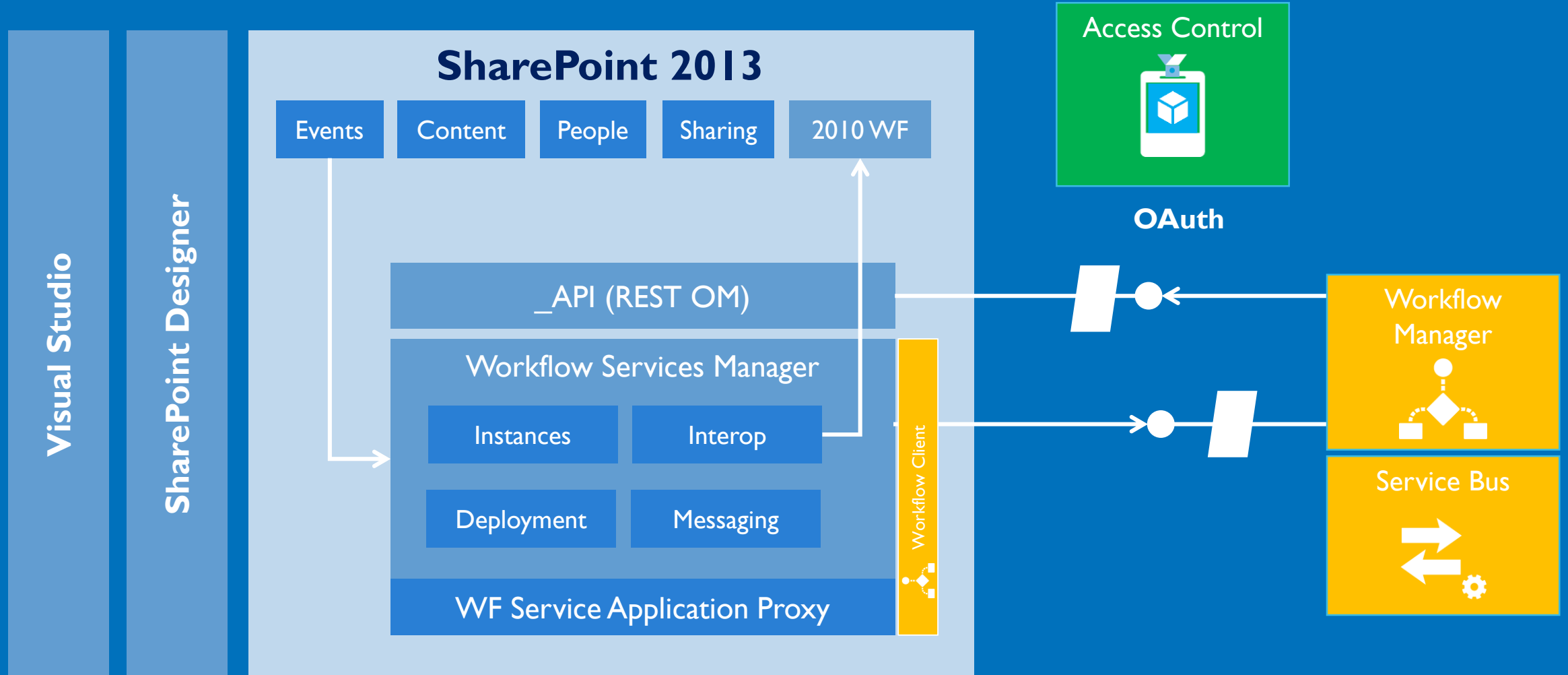
# Agenda

- Workflow Architecture (recall)
- Good to Know and Limitations
- Common Business Requirements
- Workflow Security
- Notes from the Fields

# Workflow Architecture

SharePoint 2013/SharePoint Online

# Workflow Architecture



# Good to Know and Limitations

With SharePoint Designer 2013 and Visual Studio 2013

# Known Facts

- The Workflow Manager uses the REST API to communicate with SharePoint 2013/Online
- The REST-based requests should leverage JSON serialization
- Security over HTTP(S) is based on OAuth/ACS
- The Workflow Manager is extensible
  - Using custom declarative Actions/Activities
  - Using custom code-based Activities
- In order to support Office 365 and SharePoint Online
  - You can't create code-based activities
  - Only declarative (markup-based) extensions are supported



# SPD2013 Native Actions/Conditions

- Are not enough for real business scenarios

Coordination Actions
Start a List Workflow
Start a Site Workflow
Core Actions
Add a Comment
Add Time to Date
Build Dictionary
Call HTTP Web Service
Count Items in a Dictionary
Do Calculation
Get an Item from a Dictionary
Log to History List
Pause for Duration
Pause until Date
Send an Email
Set Time Portion of Date/Time Field
Set Workflow Status
Set Workflow Variable

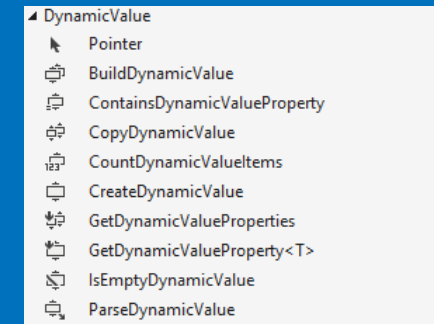
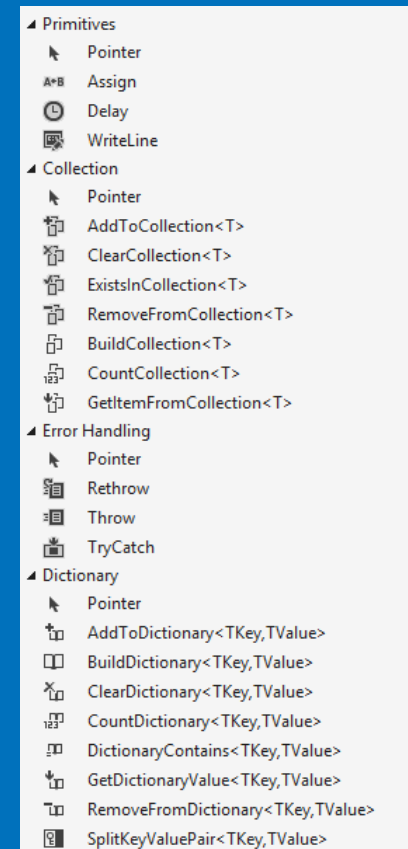
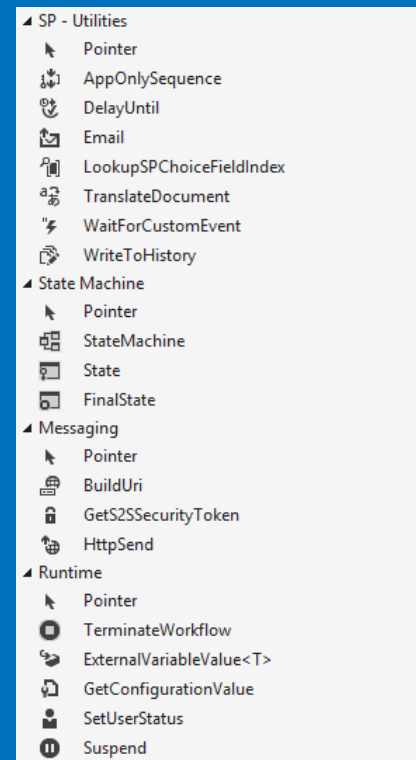
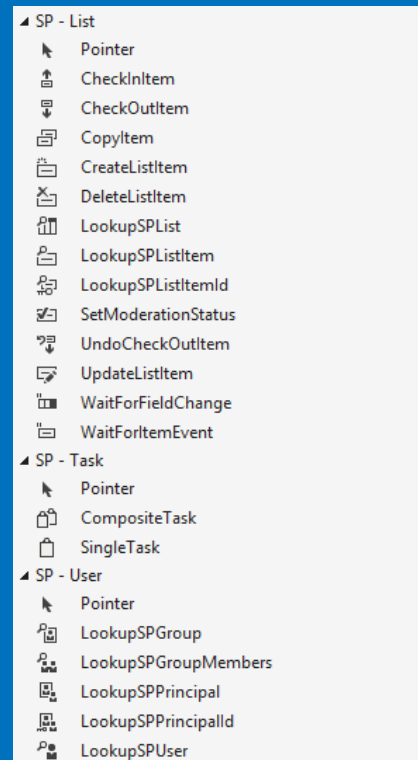
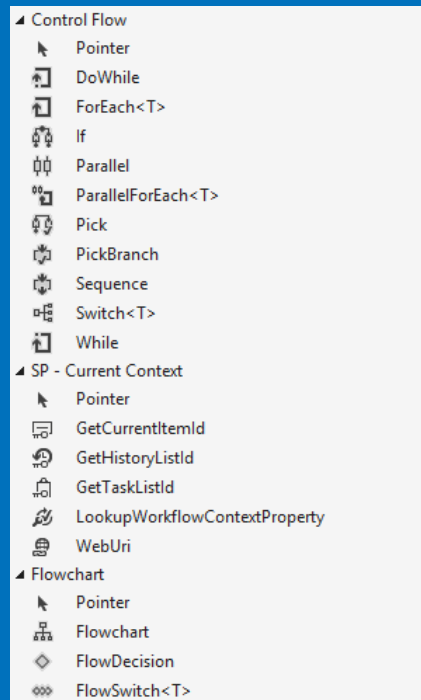
List Actions
Check In Item
Check Out Item
Copy Document
Create List Item
Delete Item
Discard Check Out Item
Set Field in Current Item
Translate Document
Update List Item
Wait for Event in List Item
Wait for Field Change in Current Item
Task Actions
Assign a task
Start a task process

Utility Actions
Extract Substring from End of String
Extract Substring from Index of String
Extract Substring from Start of String
Extract Substring of String from Index with Length
Find Interval Between Dates
Find Substring in String
Replace Substring in String
Trim String

Common Conditions
If any value equals value
Other Conditions
Created by a specific person
Created in a specific date span
Modified by a specific person
Modified in a specific date span
Person is a valid SharePoint user
Title field contains keywords

# VS2013 Native Actions/Activities

- Are still not enough, and somehow less than those of SPD2013



# You can add custom activities to VS2013

- In
- Rig
- Bro
- Ch
- Sel

The screenshot displays the 'Choose Toolbox Items' dialog box in Visual Studio 2013. The dialog is titled 'Open' and 'Choose Toolbox Items'. It is currently showing the 'System.Activities Components' tab. The list of activities is as follows:

Name	Namespace	Assembly Name
<input type="checkbox"/> RegisterForSharePoi...	Microsoft.SharePoint.WorkflowServices.Activi...	Microsoft.SharePoint.WorkflowServi
<input type="checkbox"/> ReplaceEmailTokens	Microsoft.SharePoint.WorkflowServices.Activi...	Microsoft.SharePoint.WorkflowServi
<input type="checkbox"/> SetField	Microsoft.SharePoint.WorkflowServices.Activi...	Microsoft.SharePoint.WorkflowServi
<input type="checkbox"/> SetRelatedItem	Microsoft.SharePoint.WorkflowServices.Activi...	Microsoft.SharePoint.WorkflowServi
<input type="checkbox"/> SetTimeField	Microsoft.SharePoint.WorkflowServices.Activi...	Microsoft.SharePoint.WorkflowServi
<input checked="" type="checkbox"/> SetWorkflowStatus	Microsoft.SharePoint.WorkflowServices.Activi...	Microsoft.SharePoint.WorkflowServi
<input type="checkbox"/> TranslateDocument	Microsoft.SharePoint.WorkflowServices.Activi...	Microsoft.SharePoint.WorkflowServi
<input type="checkbox"/> UndoCheckOutItem	Microsoft.SharePoint.WorkflowServices.Activi...	Microsoft.SharePoint.WorkflowServi
<input type="checkbox"/> UpdateListItem	Microsoft.SharePoint.WorkflowServices.Activi...	Microsoft.SharePoint.WorkflowServi
<input type="checkbox"/> WaitForCustomEvent	Microsoft.SharePoint.WorkflowServices.Activi...	Microsoft.SharePoint.WorkflowServi

Below the list, there is a 'Filter:' field and a 'Clear' button. The 'SetWorkflowStatus' activity is expanded, showing its properties:

- Language: Invariant Language (Invariant Country)
- Version: 15.0.0.0

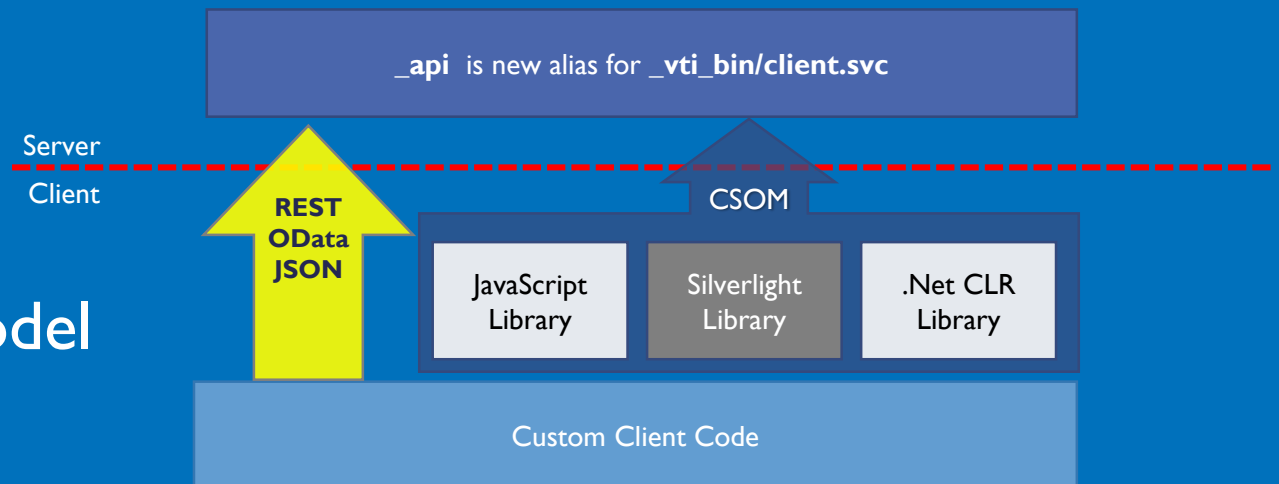
At the bottom of the dialog, there are 'OK', 'Cancel', and 'Reset' buttons. A 'Browse...' button is also visible next to the 'SetWorkflowStatus' activity. In the background, a file explorer window is visible on the left, and a keyboard shortcuts panel is on the right, showing shortcuts like Ctrl+X, Ctrl+C, Ctrl+V, and Del.

# But something is still missing ...

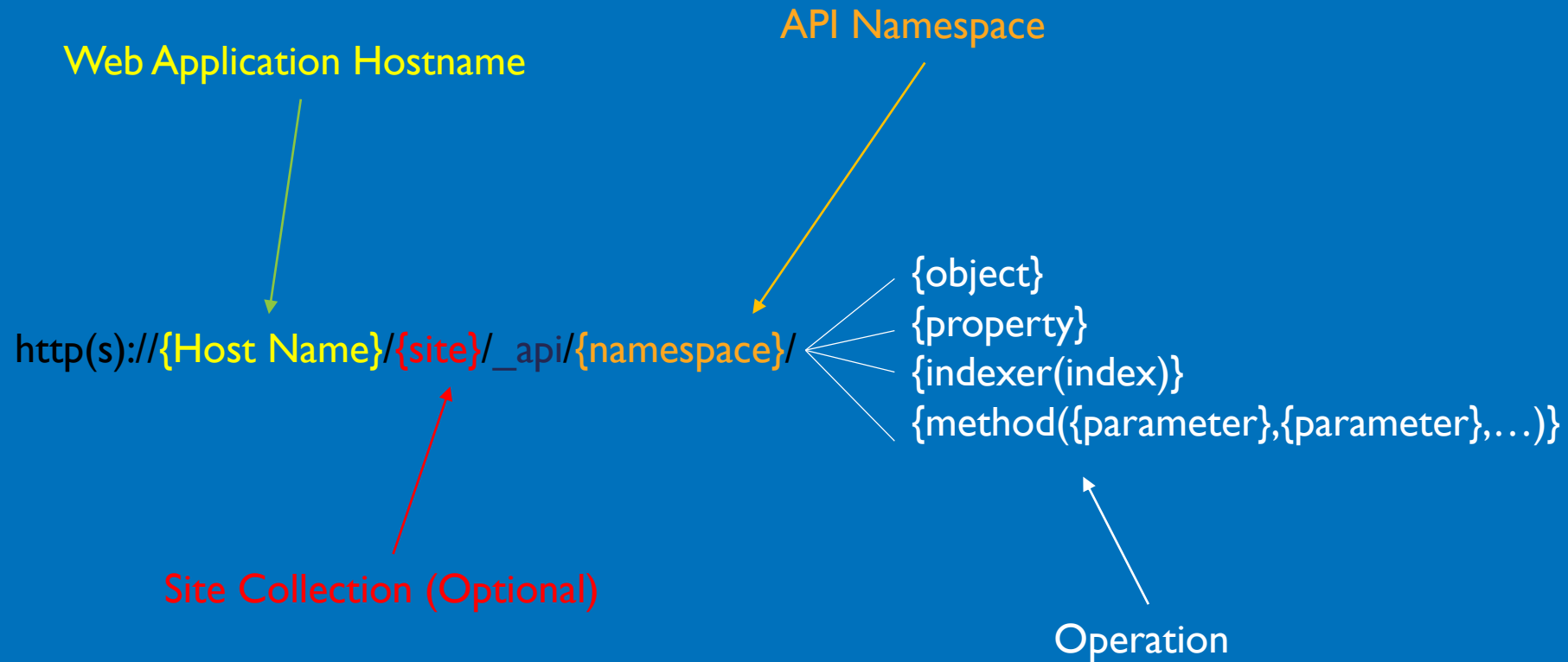
- Move File
- Upload File
- Configure Permissions on Items
- Manage Workflow Tasks
- Search for Items
- Publish/Approve Document
- Save data on a back-end database
- Etc.

# Luckily we have the REST API

- Support XML/OData/JSON format
- Can be used to query (OData) contents
- Can be used to manage (add, update, delete) content
- Make the Client Side Object Model (CSOM) remotely available via REST
- Useful for JavaScript/jQuery consumers
- But also for workflows in SPD2013 or Visual Studio 201x



# \_api Reference



# Sample URLs

- `_api/web/lists`
- `_api/web/lists/getById(guid'4E8A2867-31DB-4BA6-BFE2-4B89A040F14A')`
- `_api/web/lists/getByTitle('Documents')`
- `_api/web/lists/getByTitle('Documents')/Items`
- `_api/web/lists/getByTitle('Documents')/Items(4)`
- `_api/web/CurrentUser`
- `_api/web/SiteUsers`
- `_api/web/RoleAssignments`

# Demo

How to use the REST API



# Common Business Requirements

And how to face them

# Moving Documents

- Often document approval workflows need to move files around libraries
  - To support different authorization rules
  - To fire cascading approval workflows/processes
- There isn't an out of the box activity to move documents
- You simply have an SPD action to copy them
- Using REST API you can leverage the MoveTo method of File type

# Demo

A custom activity to Move a document ... more about security later ...

# Custom Workflow Forms and Tasks

- When you create real business workflows
- You typically need to feed workflow instances with arguments
- And you need to provide “usable” tasks to target users
- However, the out of the box UI of SharePoint is really “rude” and “minimalist”
- You should create Custom Workflow Forms and Tasks

# Understanding Workflow Forms

- Every workflow definition can have
  - Association Form: to associate a workflow definition to a target (0 or 1)
  - Initiation Form: to start a workflow instance on a target (0 or 1)
  - Task Forms: to provide custom UI for tasks (0 or more)
- Are common ASPX pages
  - But without .NET code, with client-side JSOM code only
  - You will need to interact with the Workflow Services Manager client library
- Visual Studio 2013 provides item templates for
  - Association and Initiation forms
- You need to manually define Tasks Forms
  - Leveraging custom Workflow Task content types and custom task lists

# Workflow Services Manager

Deployment  
Service

Instance  
Service

Subscription  
Service

Messaging  
Service

Interop  
Service

# Demo

Creating custom Workflow Forms and Tasks with custom UI

# Handling External Events

- Real business workflows need to interact with
  - External users
  - External software solutions (ERP, mobile apps, etc.)
  - Etc.
- You cannot rely only on tasks and users' interaction!
- The Workflow Manager engine supports asynchronous custom events
- The workflow instance can be suspended while waiting for events
  - No CPU or RAM consumption
  - Workflow instance persistence and bookmarking



# Demo

Handling External Events

# Handling Exceptions

- The out of the box engine provides
  - TryCatch (Try ... Catches ... Finally)
  - Throw
  - Rethrow
- Kind of Exceptions
  - Internal
  - External

# Internal Exceptions

- Occur in your own business logic
- You should manage them properly
- Providing some sort of logging
  - Mind the History List limits (more on this later ...)
- Where it is possible define a way out
  - Or a retry policy
- Native activities (like HttpSend) have out of the box retry policies

# External Exceptions

- Those that don't directly depend on you
  - Security Exceptions
  - Infrastructure issues
  - Etc.
- You should log them
  - And rethrow them
- Never try to solve or hide issues that don't depend on you!

# Other Common Needs

- Setting item-level permissions
- Approval tasks with predecessors
- Setting Workflow User Status and Workflow Status Field

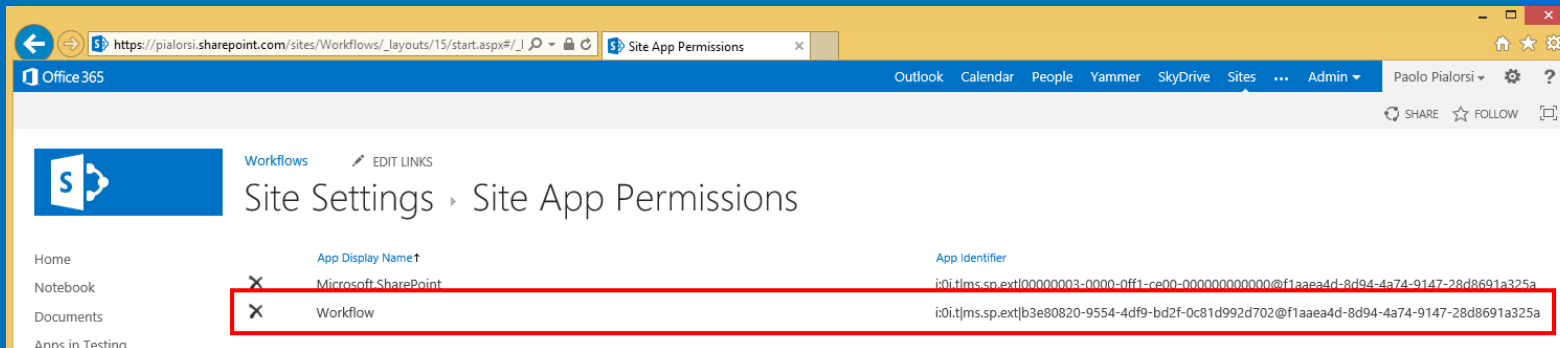
# Demo

Handling Exceptions and some other custom activities

# Workflow Security

# A Workflow is like an App

- The workflow engine has an App Principal
- You can manage app principals:
  - [https://{hostname}/sites/{site}/\\_layouts/15/appprincipals.aspx?Scope=Web](https://{hostname}/sites/{site}/_layouts/15/appprincipals.aspx?Scope=Web)
  - [https://{hostname}/sites/{site}/\\_layouts/15/appinv.aspx](https://{hostname}/sites/{site}/_layouts/15/appinv.aspx)
  - [https://{hostname}/sites/{site}/\\_layouts/15/appregnew.aspx](https://{hostname}/sites/{site}/_layouts/15/appregnew.aspx)



Version: 2.0

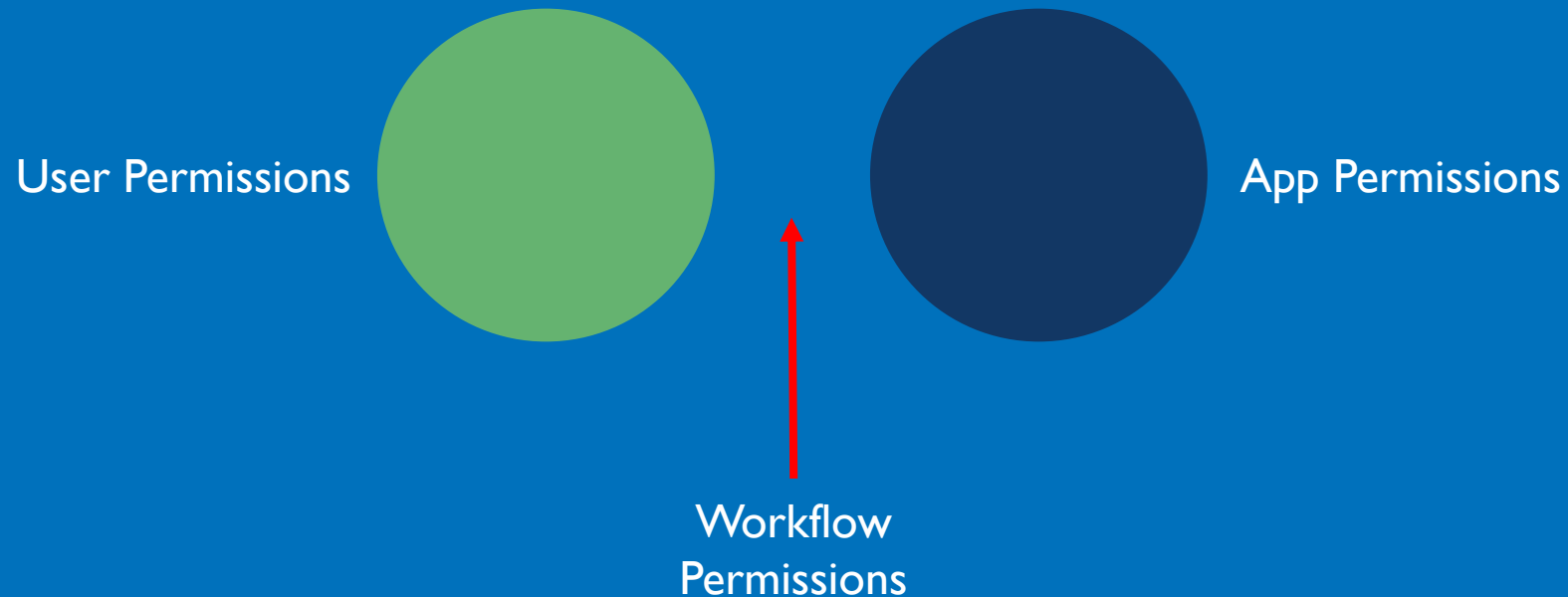
Created at 11/19/2013 11:06 PM by  Paolo Pialorsi

Last modified at 11/20/2013 12:11 AM by Workflow on behalf of  Paolo Pialorsi



# Default permission model

- Typically a workflow app has write permission level
- And while working leverages both
  - User permissions
  - Workflow App permissions

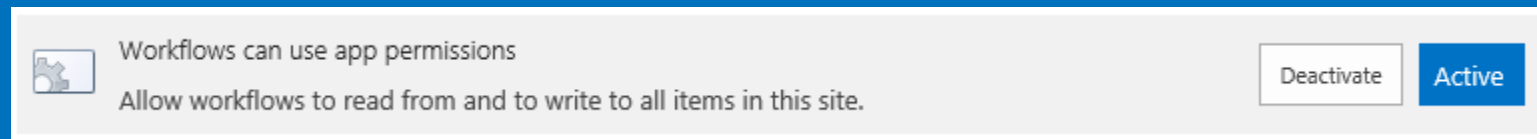


# Demo

Configuring Workflow App Security

# App Step and AppOnlySequence (recap)

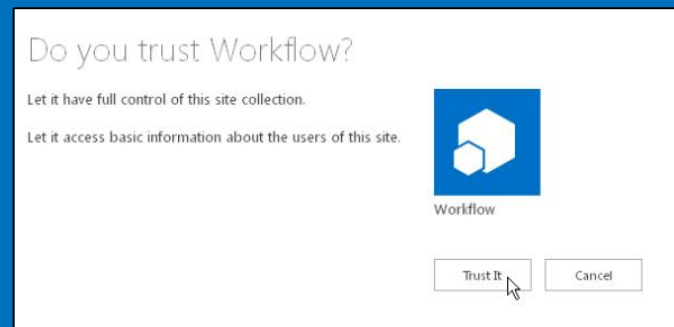
- You can configure a web site to «Allow workflows to use app permissions»



- Assign «Full Control» to the workflow App Principal (appinv.aspx)

```
<AppPermissionRequests>  
  <AppPermissionRequest Scope="http://sharepoint/content/sitecollection/web" Right="FullControl" />  
</AppPermissionRequests>
```

- And Trust it!



# Notes from the Fields

# Good Gotchas

- Remind that app web are deleted when apps are removed
  - Don't store fundamental data in the app site
  - Rely on external repositories (DB, host web, etc.)
- Carefully consider target lists size
  - Mind the list view threshold
  - Mind the number of tasks and history items
- Use dedicated task lists and history lists for each workflow association
  - Avoid sharing the same lists for all workflow associations
- Avoid leveraging on Workflow History
  - In particular in Office 365/SharePoint Online
- Be really careful with item-level permissions
  - You could end up losing control

# Thanks!

Feedback: [paolo@pialorsi.com](mailto:paolo@pialorsi.com) – [www.piasys.com](http://www.piasys.com) - [@PaoloPia](https://twitter.com/PaoloPia)

Demos shortly available on: <http://github.com/OfficeDev/PnP/>