ESPC
[LEARNING HUB]

LIVE WEBINAR

Mastering Cloud Storage with SharePoint Embedded: A Modern Approach

with **Jamel Abed,** Microsoft 365 MVP | Senior Solutions Architect | International Speaker | SharePoint Embedded | Microsoft 365 | Azure | Power Platform | SharePoint | Microsoft Certified

THURSDAY
20th MAR | 3PM CET

REGISTER →

Jamel Abed

Freelance Microsoft 365 Consultant

Working with SharePoint since 2007

Founder of SPNEXT in Paris (FRANCE)

Founder of MCEXPERT in Brussels (BELGIUM)

sessionize MOST ACTIVE SPEAKER 2024

MVP Microsoft® Most Valuable Professional

M365 Development & SharePoint

X SP_JAMEL

in ProfessionalSharePoint

Agenda

Agenda

Company Challenge Managing Applications and Data

SharePoint Embedded Overview

Architecture

Concepts

Costs of SharePoint Embedded

SharePoint Embedded Administration

Developer Experience and Tools

PowerShell and Graph API

Demo

Company Challenge Managing Applications and Data

Company Challenge

How is storage is allocated for your tenant ?

Base Storage per tenant

1 TB

Extra storage for each licenced User

10 GB per User

Example: If a company has 200 users licensed under E3, it would accumulate a total of 3 TB (1 TB base plus an additional 2 TB from users).

Additional charge: $0.20 monthly for every GB.

ESPC

Company Challenge – Customer Scenario

A tenant with 300 000 Users

Developing and designing projects for multiple business unit across the country

Applications depend on SharePoint sites for file storage

Challenges

- Some business units claim they are only using their free allocations.

- Others suggest that costs should be shared among different business units. But how ?

- Others don't want to pay the extra and justify that they don't have sufficient data.

Concerns About AI and File Privacy

- Several business units are worried about AI.

- They want to ensure their files remain private from Copilot.

Company Challenge

Solution Needed

To manage costs effectivelly

We need a storage solution that:

- Functions similarly to SharePoint document libraries.

- Ensures clear cost allocation and oversight.

- Also addresses concerns regarding AI and data privacy.

- Keeps files secure within the customer's tenant.

SharePoint Embedded Overview

SharePoint Embedded Overview

Novembre 2023 : Public Preview
Mai 2024 : General Avalibility (GA)

Content stored in containers

SPE allows you to create a SharePoint partition dedicated to your application

Containers reside in the customer's Microsoft 365 tenant

Document centric solution

Access only via Graph API

SharePoint Embedded Overview

Containers are linked to the Container Type

The container type is related to the application

Content available within Microsoft 365 tenant

Security, versionning, sharing, collaboration, Purview, Sensitive label..

Containers and content are not part of Microsoft 365 Storage quotas

Pay as you go via Azure

0.78 TB available of 1.98 TB

Containers **770 GB**

Active containers

Use this page to manage the file storage containers for your applications.
Learn more about containers
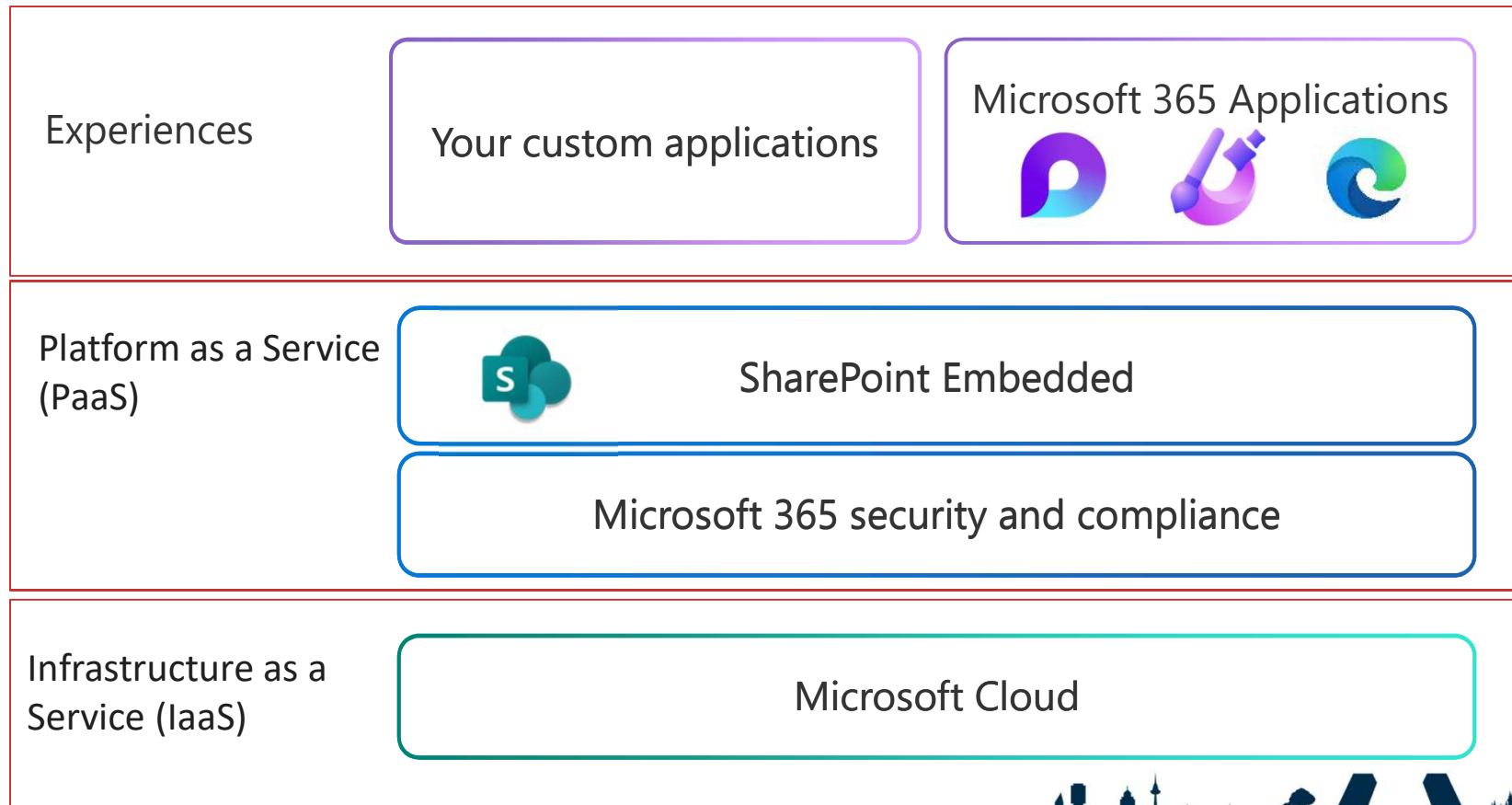
1.05 TB available of 1.05 TB ⓘ

↓ Export   🗑 Delete

🔍 Search containers

Container name	Application name ⌄	Publisher ⌄	Storage (GB) ⌄	Owners ⓘ	Sensitivity label	Created on ↓ ⌄
Knoxville Microsoft 365 Community Day	AMPDUBAI2	Other	0.01	JA	None	2/27/25, 12:35 PM
Collabdays BREMEN 2	AMPDUBAI2	Other	0.00	JA	None	2/22/25, 12:42 AM
AMP DUBAI Container 1	AMPDUBAI2	Other	0.14	JA	None	2/13/25, 3:51 AM
Pages	Loop	Microsoft	0.00	JA	None	11/6/24, 6:47 AM
Ideas	Loop	Microsoft	0.00	JA	None	10/28/24, 11:25 PM

ESPC

SharePoint Embedded

Experiences	Your custom applications	Microsoft 365 Applications
Platform as a Service (PaaS)	SharePoint Embedded	
	Microsoft 365 security and compliance	
Infrastructure as a Service (IaaS)	Microsoft Cloud	

Edge Workspaces
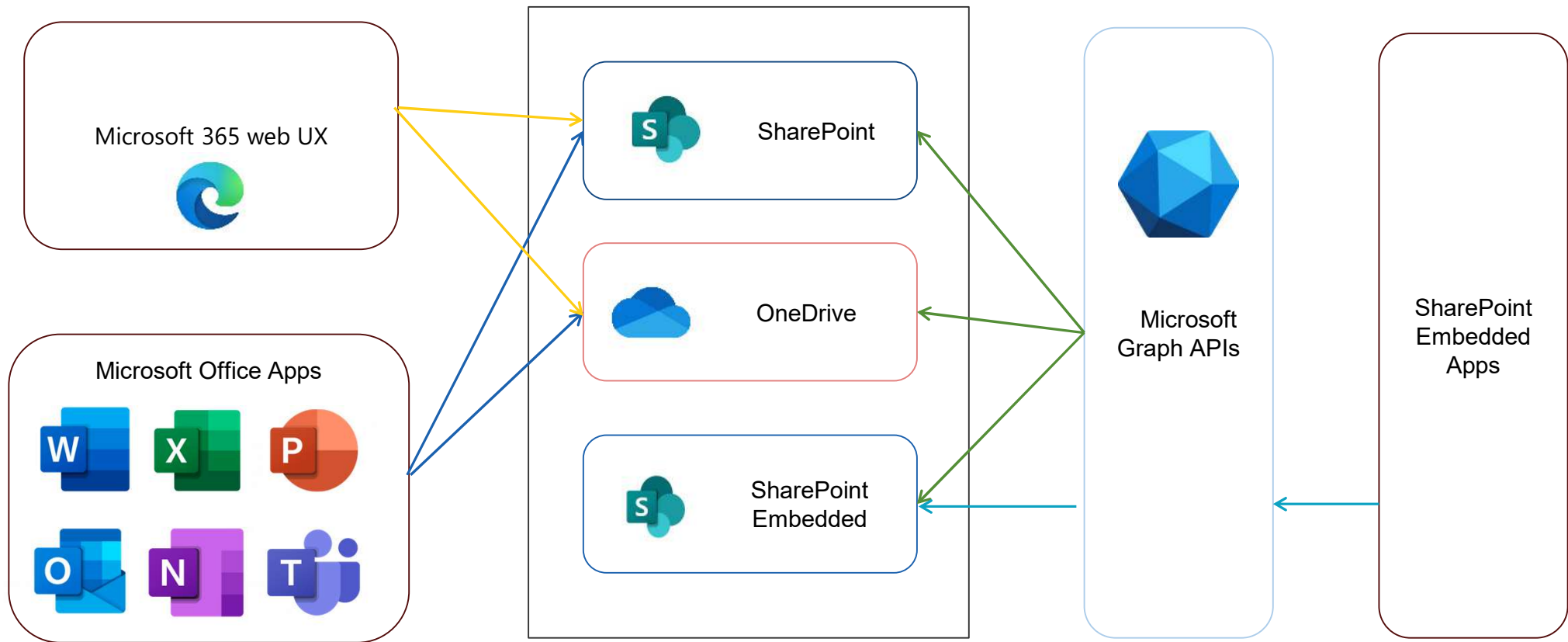
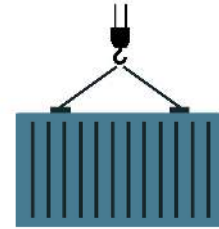# Microsoft 365 Storage Partitions

M365 STORAGE

Architecture

Architecture

SharePoint Embedded
applications

Container Type

File storage containers

Files

ESPC

SharePoint Embedded Applications

- Microsoft Entra Applications

- By default, these applications exclusively access your SharePoint Embedded content.

- They are utilized for managing and accessing Containers.

- A new permission has been introduced in Microsoft Graph: **FileStorageContainer.Selected**.

- An application is permitted to have only one Container Type at any given moment.

FileStorageContainer.Selected

File Storage Containers

- Dedicated SharePoint partition for your application

- Sample container ID: "**b!7COoNcPsJ0yqk1l9ydQy_ujQ1RWSTpdIi-XEc2o4BWq4IjN520MBQr3JOj-IXySE**"

- Applications have the capability to create multiple containers

- Each application can have up to 100,000 containers per tenant

- All containers are associated with the application through the container type

- Access is permitted only through the API

- Seamlessly integrated with Microsoft 365 experiences

- Each container has a data limit of 25TB.

Container Type

- Serves as the universal identifier for your application.

- An example of a container type ID is: "**4b6d4bb6-c390-4591-bc2b-5ccbaeacb091**"

- It is created within the provider's Tenant.

- It is registered with the consumer Tenant(s).

- Each file storage container is linked to exactly one container type.

- Connect billing to the relevant containers.

- **In summary, a container type signifies a group of containers.**

Files

- SharePoint Embedded supports all file formats.

- Each file is assigned a unique identifier for ease of use.

- Files can be arranged into folders to enhance management efficiency.

- Users can take advantage of versioning capabilities, which enable them to monitor alterations made to each file.

- The integration with other Microsoft applications facilitates real-time sharing and collaboration.

SharePoint Embedded / Delegated / Files / **Upload file**

PUT https://graph.microsoft.com/v1.0/drives/ {{ContainerId}} /root:/ {{file_name_with_extension}} :/content

Params Authorization Headers (9) Body Scripts Settings

Body Cookies Headers (11) Test Results

201 Created · 858 ms · 2.1 KB · Save Response

Pretty Raw Preview Visualize JSON

```
2    "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#drives
          ('b%21YWItAr-teUuuuk_trLP71IkT2I8N5txHvfeZH8Flu9DpoLsOGR0YSYa5AYh7Tyz2')/root/$entity",
3    "@microsoft.graph.downloadUrl": "https://spnextmvp.sharepoint.com/contentstorage/
          CSP_022d6261-adbf-4b79-aeba-4fedacb3fbd4/_layouts/15/download.aspx?
          UniqueId=9650e9eb-8de0-4d6f-bbdb-bc90cb60bf85&Translate=false&tempauth=v1.
          eyJzaXRlaWQiOiIwMjJkNjI2MS1hZGJmLTRiNzktYWViYS00ZmVkYWNiM2ZiZDQiLCJhcHBfZGlzcGxheW5hbWUiOiJTaGFyZVBvaW50IEVtYmVk
          ZGVkIEFwcCIsImFwcGlkIjoiYmM3ZDEwZDYtYjU0Yy00NDU0LTk5Y2ItZTExNWQxNTU0ZjkwIiwiYXVkIjoiMDAwMDAwMDMtMDAwMC0wZmYxLWNl
          MDAtMDAwMDAwMDAwMDAwL3NwbmV4dG12cC5zaGFyZXBvaW50LmNvbUAwNGM2NTJmNC03Y2JlLTQxMTYtOThhNi0wM2M4NjAxZDQ4YTAiLCJleHAi
          OiIxNzMyMyMjYzODk2In0.
          CgoKBHNuaWQSAjY0EgsIzsurwpm0xT0QBRoNMjAuMTkwLjE3Ny4yNSoscVp3eG1JWVNNajZWYmlYMlBuK0JIMCtBWG1iTVFVck85ZDMybzNPQitW
          VT0wsAE4AUIQoWZO9MRQAKB8lpojj0iXPUoQaGFzaGVkcHJvb2Z0b2tlbllIWyJrbXNpIl1yKTBoLmZ8bWVtYmVyc2hpcHdxMDAzMjAwM2UzZmQz
          ODhlQGxpdmUuY29tegEyggESCfRSxgS-fBZBEZimA8hgHUigkgEFamFtZWyaAQRhYmVkogEYamFtZWwuYWJlZEBzcG5leHRtdnAuY29tqgEQMTAw
          MzIwMDNFM0ZEMzg4RbIBIGNvbnRhaW5lci5zZWxlY3RlZCBhbGxmaWxlcy5yZWFkyAEB.
          t5YIT19f-cdyuRDsY8Rn2dTLjEppb_oK7H1v1-znKXs&ApiVersion=2.0",
4    "createdDateTime": "2024-11-22T07:24:56Z",
5    "eTag": "\"{9650E9EB-8DE0-4D6F-BBDB-BC90CB60BF85},1\"",
6    "id": "01HNJ36UXL5FIJNYENN5G3XW54SDFWBP4F",
7    "lastModifiedDateTime": "2024-11-22T07:24:56Z",
8    "name": "AMPMONACOFILE1.docx",
9    "webUrl": "https://spnextmvp.sharepoint.com/contentstorage/CSP_022d6261-adbf-4b79-aeba-4fedacb3fbd4/_layouts/15/Doc.
          aspx?sourcedoc=%7B9650E9EB-8DE0-4D6F-BBDB-BC90CB60BF85%7D&file=AMPMONACOFILE1.docx&action=default&
          mobileredirect=true",
10   "cTag": "\"c:{9650E9EB-8DE0-4D6F-BBDB-BC90CB60BF85},1\"",
11   "size": 0,
12   "createdBy": {
13       "application": {
```

Consepts

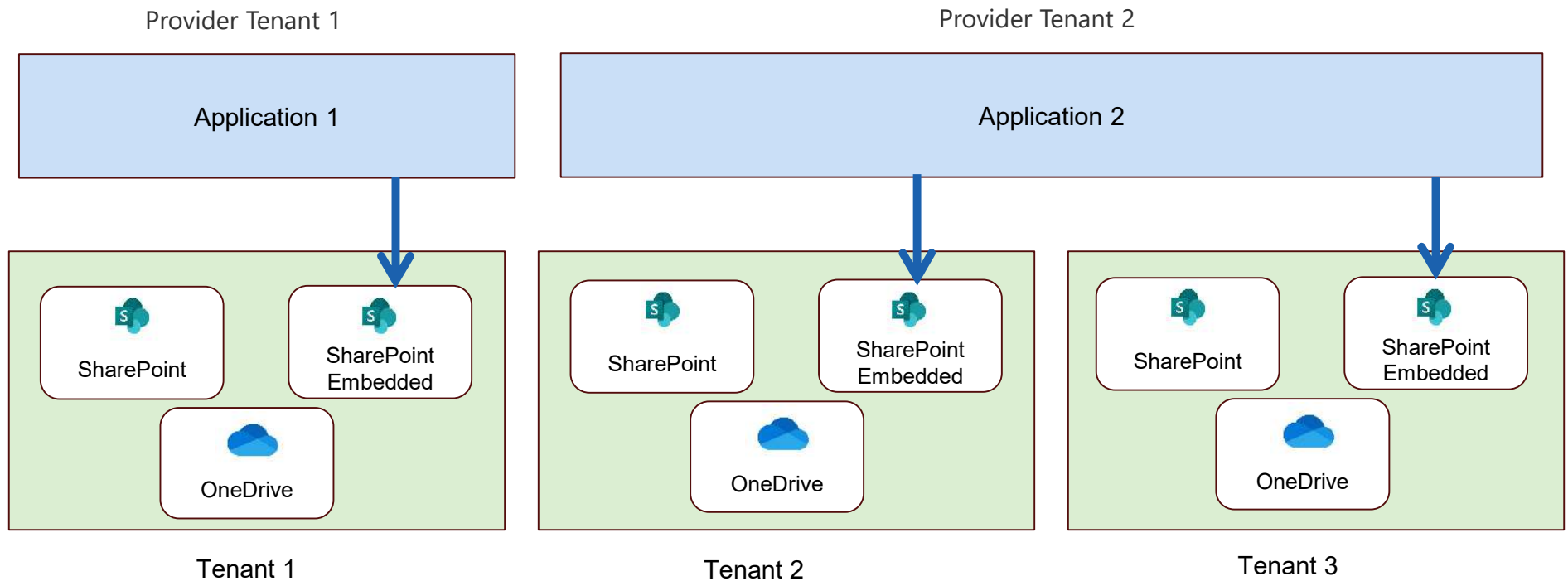# Concepts – Tenants SPE

Developer Tenant
(Provider)

Consumer Tenant(s)

Single-tenant applications

Multi-tenant applications

Concepts - Data Storage Model

Data is always controlled by the end client

Container Permissions

Container Permissions

Owner

- Full control over the Container

Manager

- Add, update, delete, and manage permissions for the Container and content within the Container.

Editor

- Add, update, and delete content within the Container

Reader

- Only view content within the Container.

Container Permissions

Unlike SharePoint, permission inheritance cannot be broken

Permissions can be added to container, files and Folders.

Custom Properties

Usueful as metadata for your applications

Custom properties can be indexed for search

Search disabled/enabled is defined with the boolean property **isSearchable**.

Can be used as a filter when querying Containers

PATCH ∨ https://graph.microsoft.com/beta/storage/fileStorage/containers/{{ContainerId}}/customProperties

Params   Authorization   Headers (9)   Body •   Scripts   Settings

○ none   ○ form-data   ○ x-www-form-urlencoded   ● raw   ○ binary   ○ GraphQL   JSON ∨

```json
1   {
2       /*
3       Format
4       "{custom_name}": {
5           "value": "{property_value}",
6           "isSearchable": true | false //Whether or not the property can be used with search queries
7       }
8       */
9       "urlPath": {
10          "value": "northwind",
11          "isSearchable": false
12      },
13      "tags": {
14          "value": "northwind traders nwt nw",
15          "isSearchable": true
16      },
17      "department": {
18          "value": "finance",
19          "isSearchable": true
20      }
21  }
```

Containers Fields

- Boolean

- DateTime

- Currency

- Choice

- HyperlinkOrPicture

- Number

- PersonOrGroup

- Text

SharePoint Embedded Administration

Administration Tools
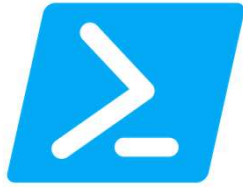
PowerShell

SharePoint Admin Center

Microsoft Purview

Microsoft Graph API

SharePoint Embedded Administrator

SPE can be managed by Global Administrators or SharePoint Embedded Administrators (new admin role)

SharePoint Admin Center - Global Administrators

SharePoint Admin Center - SharePoint Embedded Administrators

Costs of SharePoint Embedded

Costs of SharePoint Embedded

Billable standard Container Types

Pay via Azure subscription.

Trial Container Types

• Each container type is valid for 30 days.

• Maximum of 5 Containers.

• Limited to 1GB.

ESPC

Costs of SharePoint Embedded

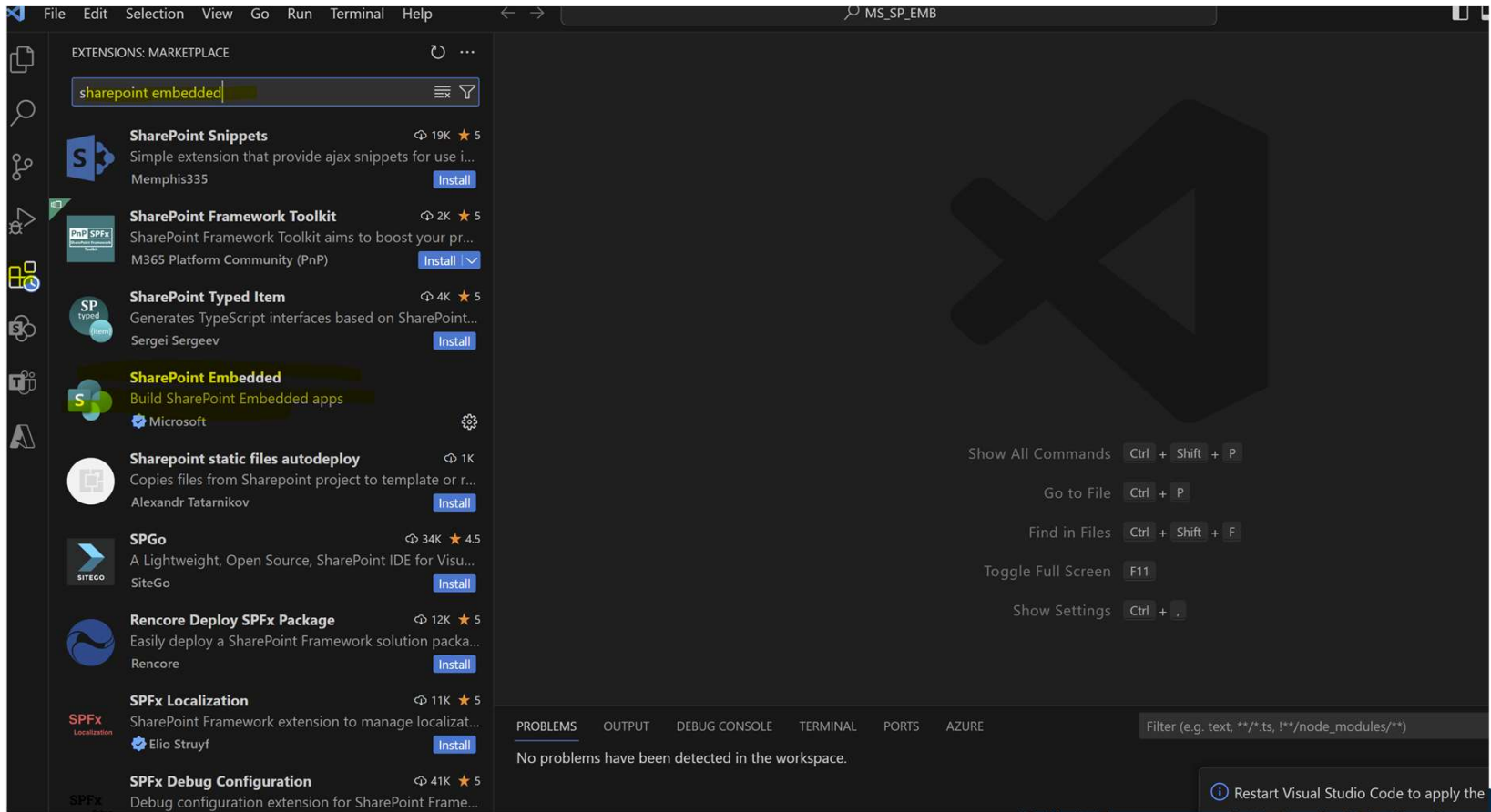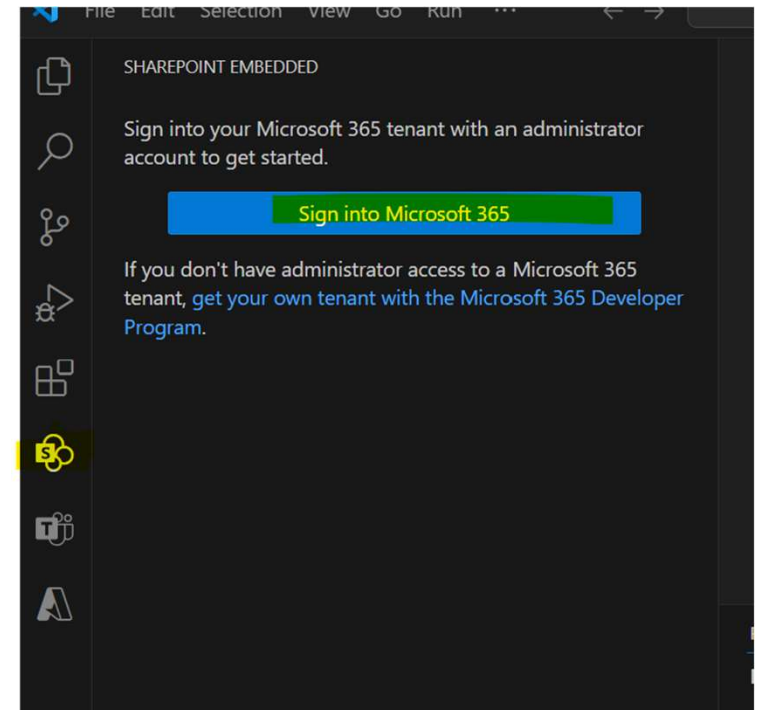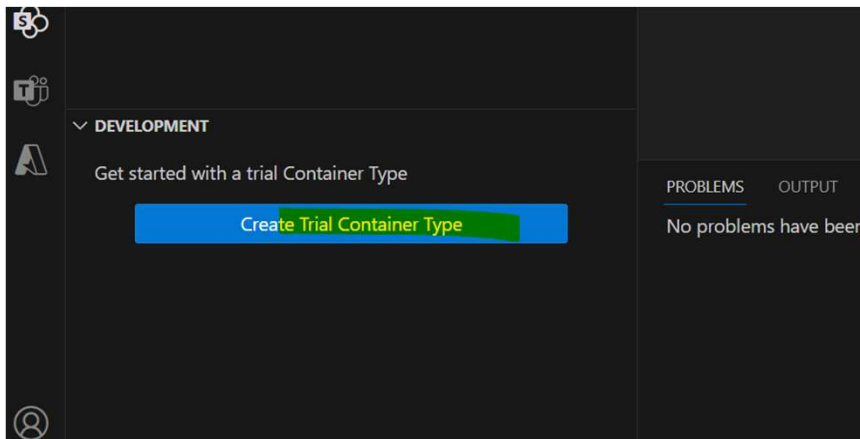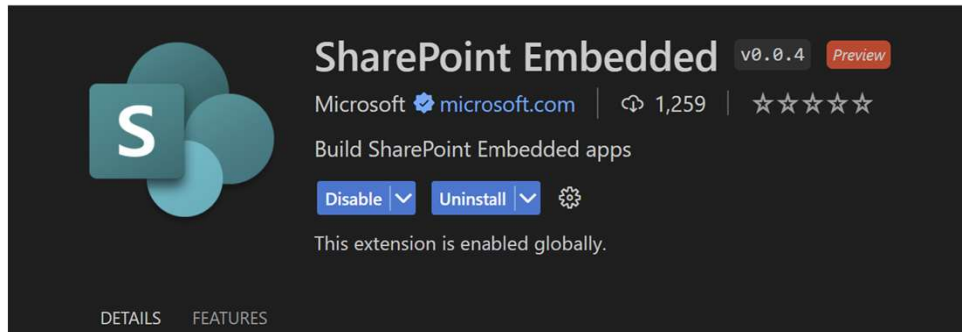| SharePoint Embedded Service Meters	Meter Unit	Price
Storage	$/GB/Day	$0.00667
Graph API Transactions	$/1 API call	$0.00050

0.00667 × 30 =

0.2001

Developer Experience and Tools

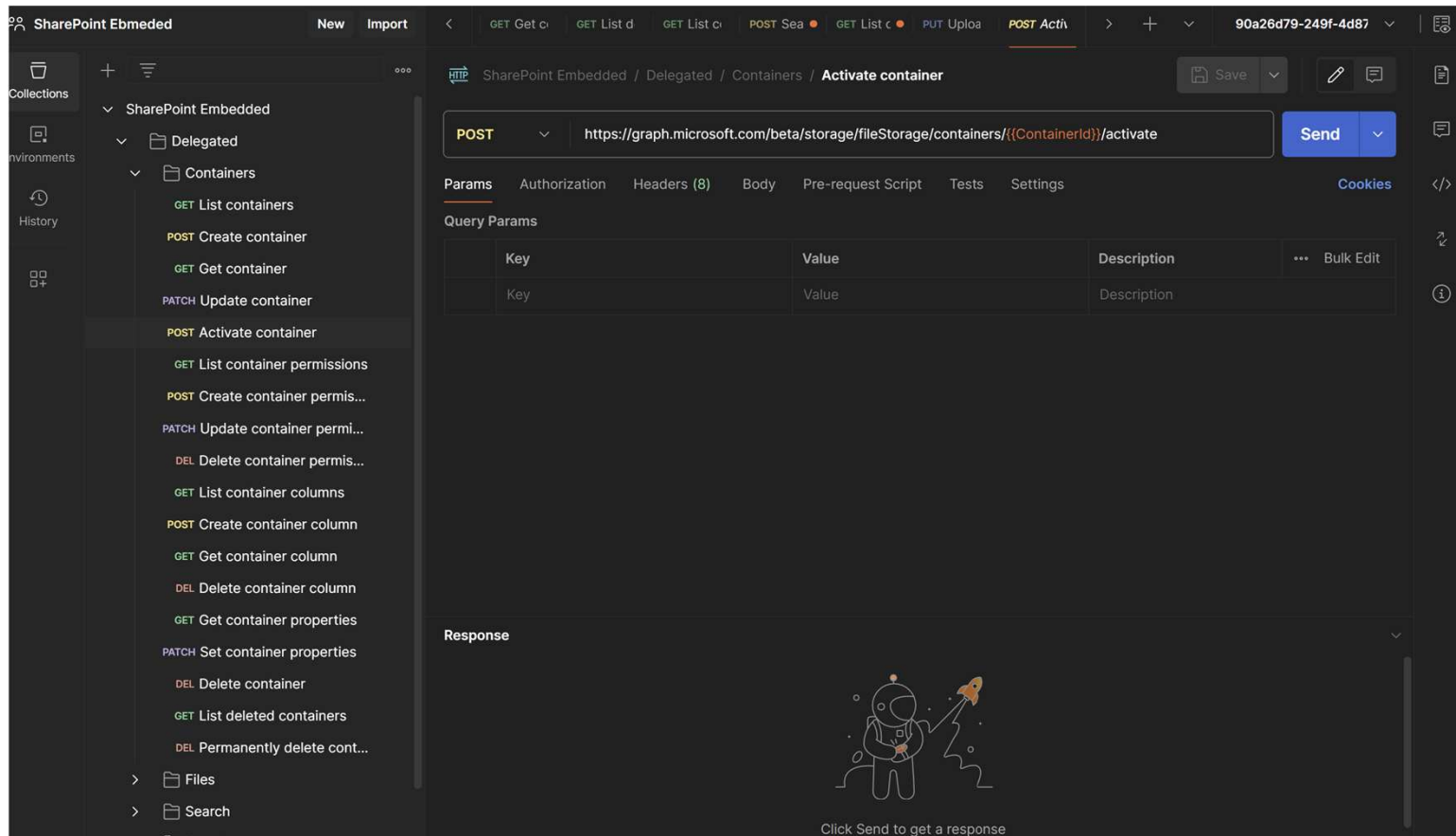# Configure the SharePoint Embedded extension of Visual Studio Code

Configure the SharePoint Embedded extension of Visual Studio Code

Configure Postman for SharePoint Embedded

https://aka.ms/spe-postman

PowerShell Commands
and Graph API

Creation Container Type

- There are two types of containers that a Tenant developer can create:

  - Trial container types.

  - Standard container types.

Creation trial container Type

Import-Module "Microsoft.Online.SharePoint.PowerShell"

Connect-SPOService -Url <SPOAdminUrl>

New-SPOContainerType
[–TrialContainerType]
[-ContainerTypeName <ContainerTypeName>]
[-OwningApplicationId <OwningApplicationId>]

Standard container type

Import-Module "Microsoft.Online.SharePoint.PowerShell"
Connect-SPOService -Url <SPOAdminUrl>

New-SPOContainerType
[–ContainerTypeName <ContainerTypeName>]
[-OwningApplicationId <OwningApplicationId>]
[-AzureSubscriptionId <AzureSubscriptionId>]
[-ResourceGroup <ResourceGroup>]
[-Region <Region>]

DEMO

Retrieve container Type

Get-SPOContainerType

Get-SPOContainerType -ContainerTypeId <ContainerTypeId>

Link the container type with the consumer Tenant

CALL API

PUT {RootSiteUrl}/_api/v2.1/storageContainerTypes/{containerTypeId}/applicationPermissions

Content-Type: application/json

```json
{
  "value": [
    {
      "appId": "71392b2f-1765-406e-86af-5907d9bdb2ab",
      "delegated": ["full"],
      "appOnly": ["full"]
    }
  ]
}
```

Link the container type with the consumer Tenant

Result

HTTP/1.1 200 OK

Content-type: application/json

```
{
  "value": [

   {

     "appId": "71392b2f-1765-406e-86af-5907d9bdb2ab",

     "delegated": ["full"],

     "appOnly": ["full"]

   }

  ]

}
```

Creation of a container

POST https://graph.microsoft.com/beta/storage/fileStorage/containers

Content-Type: application/json

```json
{
  "displayName": "My Application Storage Container",
  "description": "Description of My Application Storage Container",
  "containerTypeId": "91710488-5756-407f-9046-fbe5f0b4de73"
}
```

Creation of a container

Response

HTTP/1.1 201 Created

Content-type: application/json

```json
{
  "@odata.type": "#microsoft.graph.fileStorageContainer",
  "id": "b!ISJs1WRro0y0EWgkUYcktDa0mE8zSlFEqFzqRn70Zwp1CEtDEBZgQlCPkRbil_5Z",
  "displayName": "My Application Storage Container",
  "description": "Description of My Application Storage Container",
  "containerTypeId": "91710488-5756-407f-9046-fbe5f0b4de73",
  "status": "inactive",
  "createdDateTime": "2021-11-24T15:41:52.347Z"
}
```

Update a container

GET

https://graph.microsoft.com/beta/storage/fileStorage/containers/{**container Id**}

Content-Type: application/json

```
{

  "displayName": "Updated Name",

  "description": "Updated Description"

}
```

Update a container

Response

HTTP/1.1 200 OK

Content-type: application/json

```json
{
 "@odata.type": "#microsoft.graph.fileStorageContainer",
 "id": "b!ISJs1WRro0y0EWgkUYcktDa0mE8zSlFEqFzqRn70Zwp1CEtDEBZgQICPkRbil_5Z",
 "displayName": "Updated Name",
 "description": "Updated Description",
 "containerTypeId": "91710488-5756-407f-9046-fbe5f0b4de73",
 "status": "active",
 "createdDateTime": "2021-11-24T15:41:52.347Z"
}
```

Retrieve containers

PowerShell

Get-SPOContainer -OwningApplicationId {{**appId**}}

Get-SPOContainer -OwningApplicationId {{**appId**}} -Identity {{**containerId**}}

Graph API

GET ../beta/storage/fileStorage/containers?$filter=containerTypeId eq {{**containerTypeId**}}

GET ../beta/storage/fileStorage/containers/{{**containerId**}}

Delete containers

#delete container
Remove-SPOContainer -Identity {{**containerId**}}

#get deleted containers
Get-SPODeletedContainer

#restore deleted container
Restore-SPODeletedContainer -Identity {{**containerId**}}

#permanently delete container
Remove-SPODeletedContainer -Identity {{**containerId**}}

Delete containers Graph API

#delete container
DELETE ../beta/storage/fileStorage/containers/{{**containerId**}}

#get deleted containers
GET ../beta/storage/fileStorage/deletedContainers

#permanently delete container
DELETE ../beta/storage/fileStorage/deletedContainers/{{**containerId**}}

Manage Container permissions

#get permission(s)

GET ../beta/storage/fileStorage/containers/{{**containerId**}}/permissions

GET ../beta/storage/fileStorage/containers/{{**containerId**}}/permissions/{{permissionId}}

#create permission

POST ../beta/storage/fileStorage/containers/{{**containerId**}}/permissions

{ "roles": ["reader"], "grantedToV2": { "user": { "userPrincipalName": ".." } } }

#update permission

PATCH ../beta/storage/fileStorage/containers/{{**containerId**}}/permissions/{{permissionId}}

{ "roles": ["writer"], "grantedToV2": { "user": { "userPrincipalName": ".." } } }

#delete permission

DELETE ../beta/storage/fileStorage/containers/{{**containerId**}}/permissions/{{permissionId}}

Manage container custom properties

#get custom properties|property

GET ../beta/storage/fileStorage/containers/{{**containerId**}}/customProperties

GET ../beta/storage/fileStorage/containers/{{**containerId**}}/customProperties/{{propertyId}}

#create & update custom properties

PATCH ../beta/storage/fileStorage/containers/{{**containerId**}}/customProperties

```
{
  "property1": { "value": "value1", "isSearchable": false },
  "tags": { "value": "northwind nw", "isSearchable": true }
}
```

#delete custom property (set value to null)

PATCH ../beta/storage/fileStorage/containers/{{**containerId**}}/customProperties

```
{ "property1": { "value": [null|""], "isSearchable": false } }
```

Work with container

#get file(s)

GET ../v1.0/drives/{{containerId}}/items/root/children

GET ../v1.0/drives/{{**containerId**}}/items/{{**driveItemId**}} // driveItem = file | folder

#upload file

PUT .. /v1.0/drives/{{**containerId**}}/root:/{{**fileNameWithExtension**}}:/content

<binary stream>

#delete file / folder

DELETE ../v1.0/drives/{{**containerId**}}/items/{{fileId|folderId}}

#create folder

POST ../v1.0/drives/{{**containerId**}}/root/children

{ "name": "New Folder", "folder": {} }

Conclusion

Conclusion

- SharePoint storage is not free; you pay extra beyond the quotas.

- Since Microsoft has developed applications with SharePoint embedded storage, you can do it too.

- Study the costs of your project's storage before developing or using cloud storage.

Ressources

https://aka.ms/spe-postman

https://learn.microsoft.com/fr-fr/sharepoint/dev/embedded/overview

Set up and configure SharePoint Embedded - Training | Microsoft Learn

Thank you