# AGO Storage Bloat: Identification and Management with Python

TEXAS PARKS & WILDLIFE

Michael Potts

Eric Blazewicz

# Storage bloat disambiguation

Overconsumption of space

- Controllable:
  - Pork: test data, duplication of efforts
  - Space optimization:
- Uncontrollable:
  - Budget constraints
  - Habits

# Causes of AGO storage bloat at TPWD

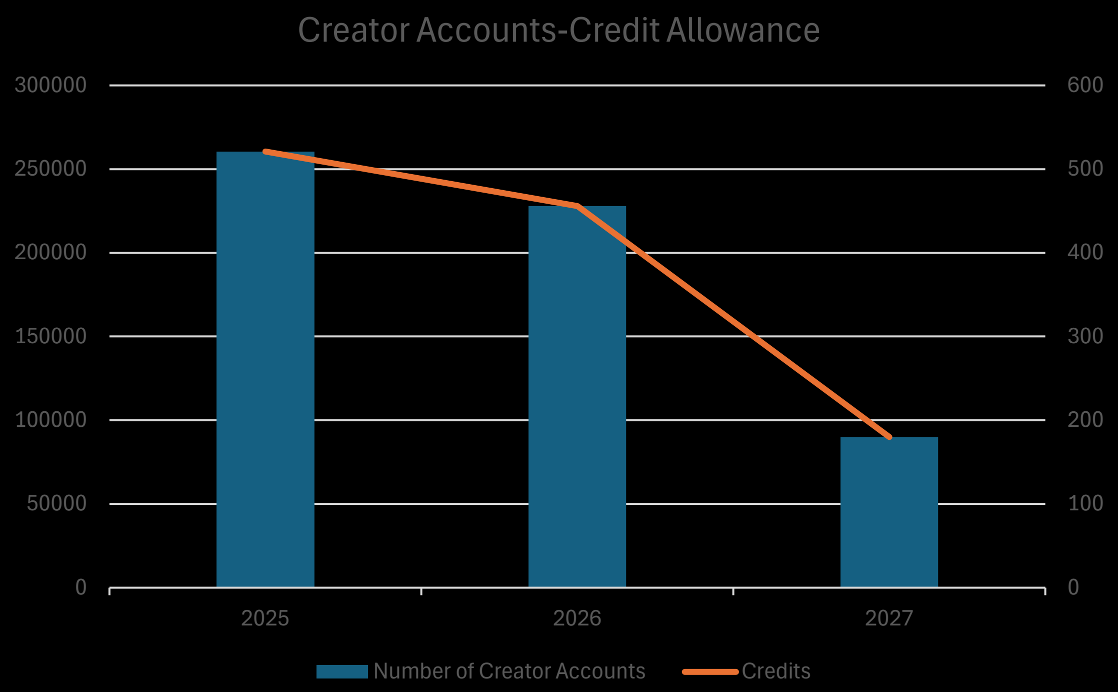
- Attachments
- New records and projects
- Change tracking (adds/deletes/updates)
- Stale content and old projects
- Content duplication and tests
- Non-geographic files and data
- Lack of content governance/unrestricted publishing rights

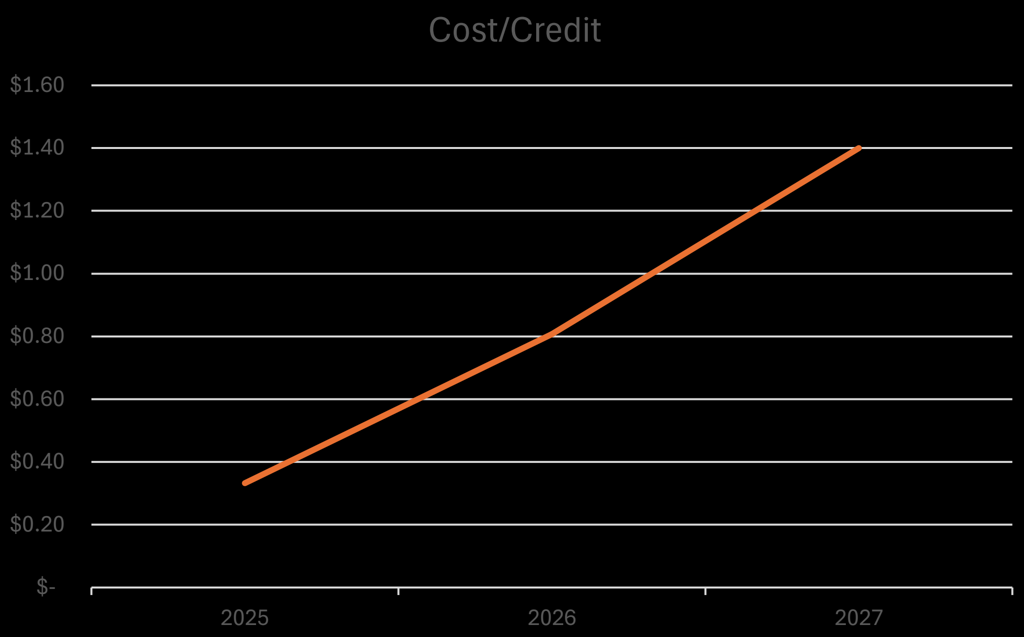
# Why is storage bloat unsustainable?

- Price increases per AGO user
- 500 credits per user per year: Creator, Professional, and Professional Plus
- 250 credits per user per year: Mobile Worker, Contributor
- $120 for 1,000 credits, which expire after 2 years
- Number of users for 2025 FY
  - 521 Creators: 260,500 credits
- Number of users for 2026 FY
  - 456 Creators: 228,000 credits
- Number of users for 2027 FY
  - ~180 Creators: ~90,000 credits

(ESRI Canada, 2025)

# Why is storage bloat unaffordable?





# Running out of credits

- If you run out of credits, your account enters a restricted state
  - Members cannot create new content.
  - Members cannot publish new layers of any type.
  - Members do not have access to credit-consuming tools.
- A negative credit balance will be pulled when credits are delivered to your organization's account upon subscription renewal.
- To minimize credits, must know which items to focus on.

(ArcGIS Online Help, 2025b)

# Methods of determining storage credit usage: No easy way to see per-item storage credit usage

- Credit report:
  - Total storage credit usage for org
- Item report:
  - Must calculate credits per item
- ArcGIS API scripting:
  - No storage credit attribute; must calculate credits per item

# Storage credit usage per item type

- 2.4 credits per 10 MB stored per month, calculated hourly
  - Feature storage: hosted feature layers
- 1.2 credits per 1 GB stored per month, calculated hourly
  - Imagery storage: hosted imagery services
  - Service definition files
  - Tile packages
  - Scene layer packages
  - Files: PDFs, images, etc.
  - Web maps
- 12 credits per 1 GB stored per month, per user
  - Any content in ArcGIS Notebooks workspaces

(ArcGIS Online Help, 2025b)

# Formulae for daily credit usage

- The item.size property is in bytes:
  - 2.4 credits per 10 MB
    - Feature storage: convert item.size to MB

```
if item.type == "Feature Service" and gis.properties.id in item.url:
    creds = item.size/1024/1024/10*2.4/30
```

  - 1.2 credits per 1 GB
    - File storage: convert item.size to GB

```
creds = item.size/1024/1024/1024*1.2/30
```

  - 12 credits per 1 GB
    - ArcGIS Notebooks

```
creds = item.size/1024/1024/1024*12/30
```

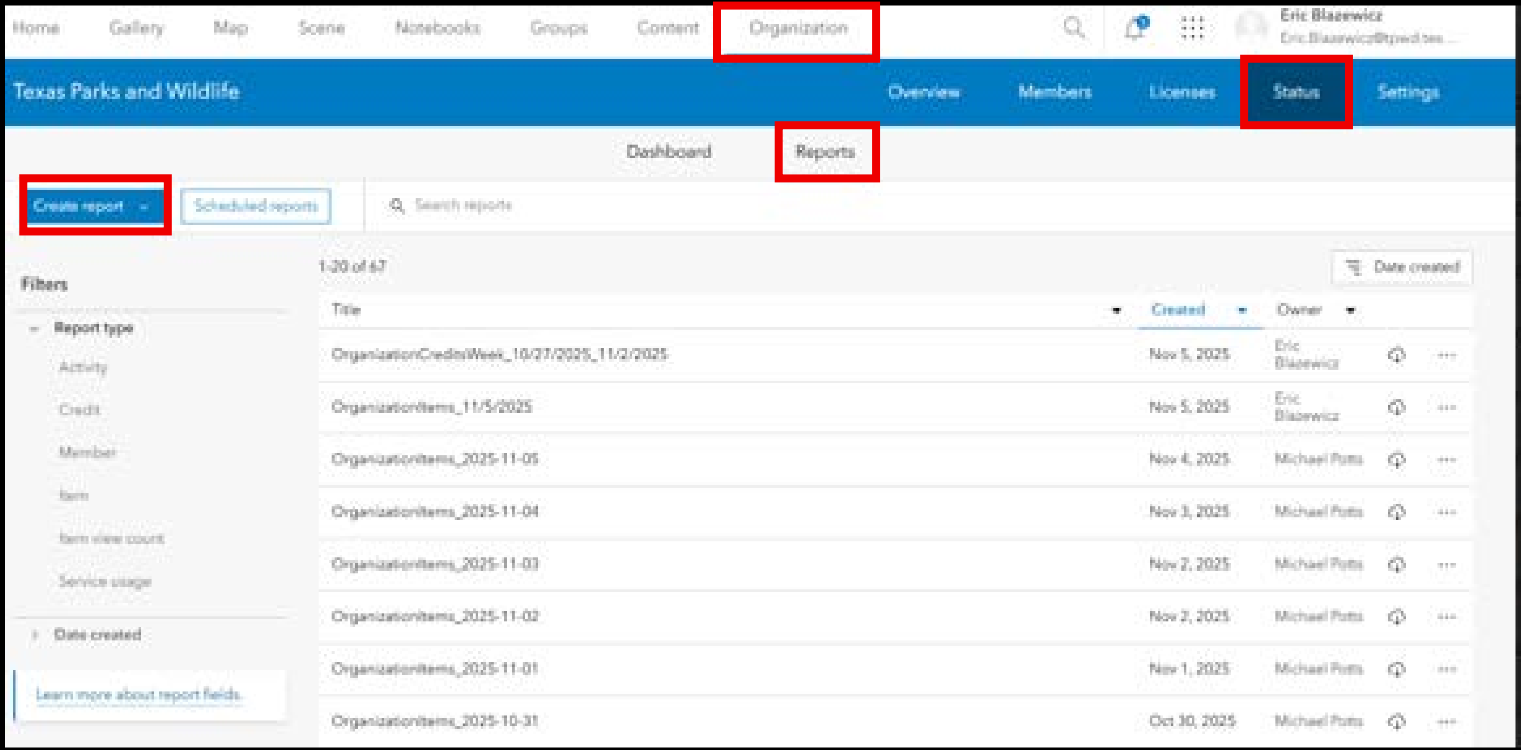
(ESRI Tech Support, 2025)

# Using AGO reports to find credit usage

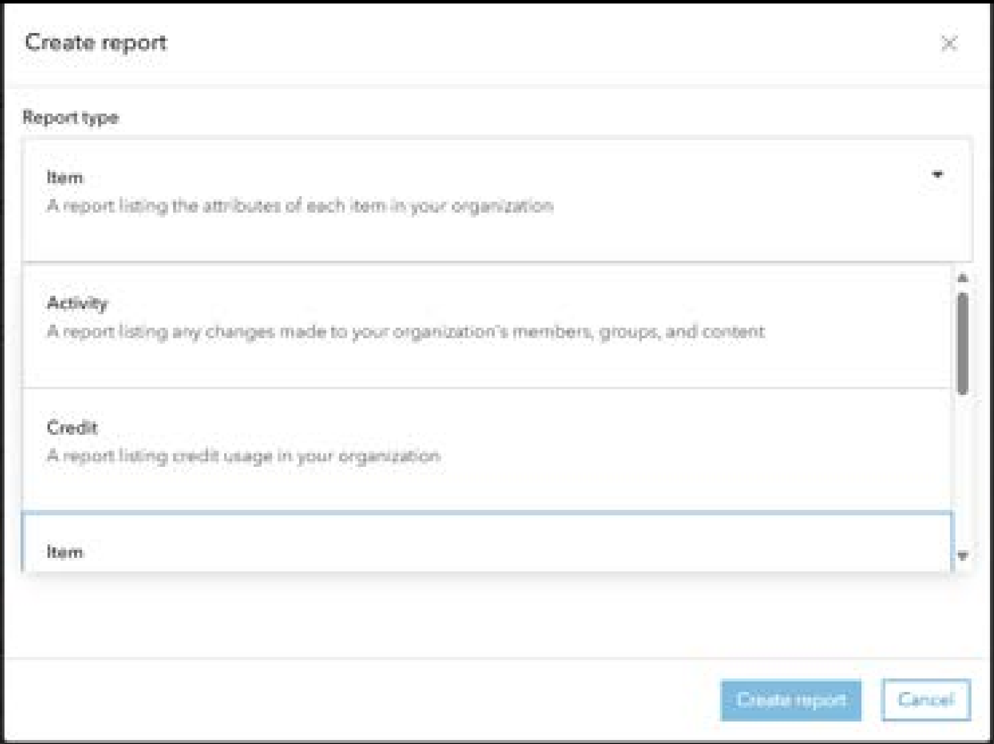
- Export a .csv from Organization > Status > Reports
- Credit report gives breakdown of usage in the org in a .csv:
  - Each row is a user, plus one row for the whole org. Only the org row has storage credits populated.
  - “Credit reports allow you to download a detailed breakdown of credit use in your organization in a .csv file. The rows display credit use by service for each organization member as well as storage credit consumption for the organization itself.” (ArcGIS Online Help, 2025a)
- Item report gives stats on each item, but not credits.
  - Each row is an item
  - File storage size column in MB
  - Feature storage size column in MB

(ESRI Australia, 2025; ArcGIS Online Help, 2025a)

# Using AGO reports to find credit usage



| Title | Created | Owner |
|---|---|---|
| OrganizationCreditsWeek_10/27/2025_11/2/2025 | Nov 5, 2025 | Eric Blazewicz |
| OrganizationItems_11/5/2025 | Nov 5, 2025 | Eric Blazewicz |
| OrganizationItems_2025-11-05 | Nov 4, 2025 | Michael Potts |
| OrganizationItems_2025-11-04 | Nov 3, 2025 | Michael Potts |
| OrganizationItems_2025-11-03 | Nov 2, 2025 | Michael Potts |
| OrganizationItems_2025-11-02 | Nov 2, 2025 | Michael Potts |
| OrganizationItems_2025-11-01 | Nov 1, 2025 | Michael Potts |
| OrganizationItems_2025-10-31 | Oct 30, 2025 | Michael Potts |



Item
A report listing the attributes of each item in your organization

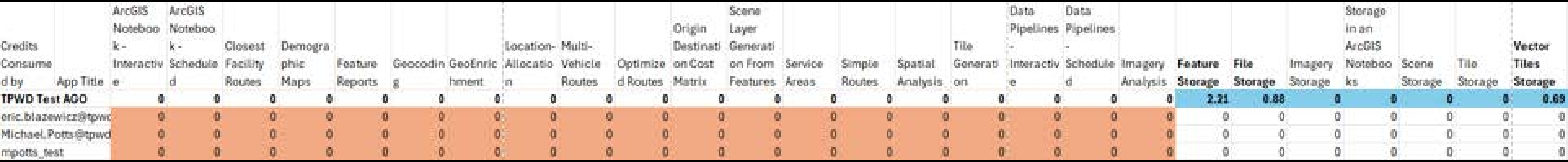
Activity
A report listing any changes made to your organization's members, groups, and content

Credit
A report listing credit usage in your organization

Item

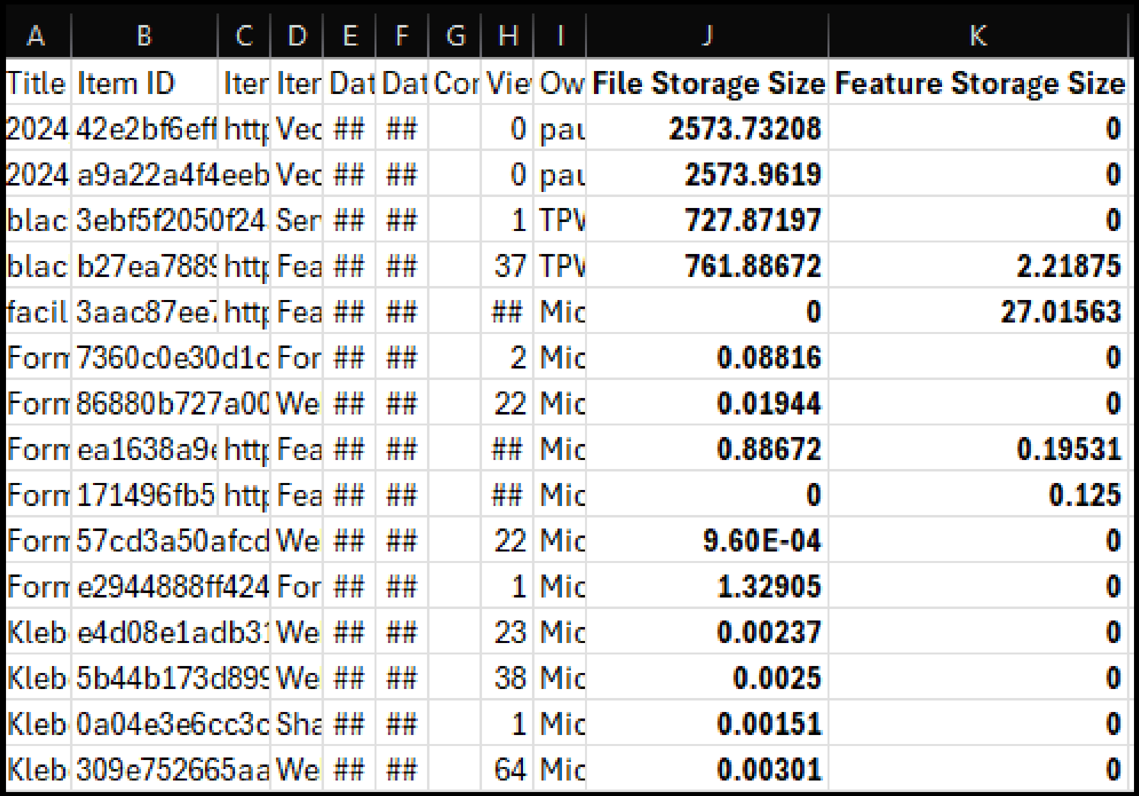
# Credit report is mishmash of data

- Analysis fields (orange)
  - Only apply to user rows
- Storage fields (blue)
  - Only apply to whole organization row



| Credits Consumed by | App Title | ArcGIS Notebook - Interactive | ArcGIS Notebook - Scheduled | Closest Facility Routes | Demographic Maps | Feature Reports | Geocoding | GeoEnrichment | Location-Allocation | Multi-Vehicle Routes | Optimized Routes | Origin Destination Cost Matrix | Scene Layer Generation From Features | Service Areas | Simple Routes | Spatial Analysis | Tile Generation | Data Pipelines - Interactive | Data Pipelines - Scheduled | Imagery Analysis | **Feature Storage** | **File Storage** | Imagery Storage | Storage in an ArcGIS Notebooks | Scene Storage | Tile Storage | **Vector Tiles Storage** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **TPWD Test AGO** | | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **2.21** | **0.88** | **0** | **0** | **0** | **0** | **0.69** |
| eric.blazewicz@tpwd | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Michael.Potts@tpwd | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| mpotts_test | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# Processing an item report



| A | B | C | D | E | F | G | H | I | J | K |
|---|---|---|---|---|---|---|---|---|---|---|
| Title | Item ID | Iter | Iter | Dat | Dat | Cor | Vie | Ow | **File Storage Size** | **Feature Storage Size** |
| 2024 | 42e2bf6eff | http | Vec | ## | ## | | 0 | pau | **2573.73208** | **0** |
| 2024 | a9a22a4f4eeb | | Vec | ## | ## | | 0 | pau | **2573.9619** | **0** |
| blac | 3ebf5f2050f24 | | Ser | ## | ## | | 1 | TPV | **727.87197** | **0** |
| blac | b27ea7889 | http | Fea | ## | ## | | 37 | TPV | **761.88672** | **2.21875** |
| facil | 3aac87ee7 | http | Fea | ## | ## | | ## | Mic | **0** | **27.01563** |
| Forn | 7360c0e30d1c | | For | ## | ## | | 2 | Mic | **0.08816** | **0** |
| Forn | 86880b727a00 | | We | ## | ## | | 22 | Mic | **0.01944** | **0** |
| Forn | ea1638a9e | http | Fea | ## | ## | | ## | Mic | **0.88672** | **0.19531** |
| Forn | 171496fb5 | http | Fea | ## | ## | | ## | Mic | **0** | **0.125** |
| Forn | 57cd3a50afcd | | We | ## | ## | | 22 | Mic | **9.60E-04** | **0** |
| Forn | e2944888ff424 | | For | ## | ## | | 1 | Mic | **1.32905** | **0** |
| Kleb | e4d08e1adb31 | | We | ## | ## | | 23 | Mic | **0.00237** | **0** |
| Kleb | 5b44b173d899 | | We | ## | ## | | 38 | Mic | **0.0025** | **0** |
| Kleb | 0a04e3e6cc3c | | Sha | ## | ## | | 1 | Mic | **0.00151** | **0** |
| Kleb | 309e752665aa | | We | ## | ## | | 64 | Mic | **0.00301** | **0** |

```python
csvDict, columns = {}, []
fileStorage, featStorage = 0, 0
with open(itemReportcsv, mode='r') as file:
    csvFile = csv.reader(file)
    for n, line in enumerate(csvFile):
        if n == 0:
            columns += line
        else:
            fileStorage += float(line[9])
            featStorage += float(line[10])

fileSize = fileStorage * 1024 * 1024 # Convert to bytes from MB
featSize = featStorage * 1024 * 1024 # Convert to bytes from MB

credsFile = fileSize/1024/1024/1024*1.2/30
credsFeat = featSize/1024/1024/10*2.4/30
totalCreds = credsFile + credsFeat

print(f“Total credits: {totalCreds}")
```

# ArcGIS API credit calculations per item

- Attachments bug: BUG-000156518
  - The size of item attachments [are] not consistently displayed on the item details page in ArcGIS Online.
  - Also affects item.size property in Python scripts and attachment size property of the Attachment Manager.
- Must take this bug into account for item credit calculations

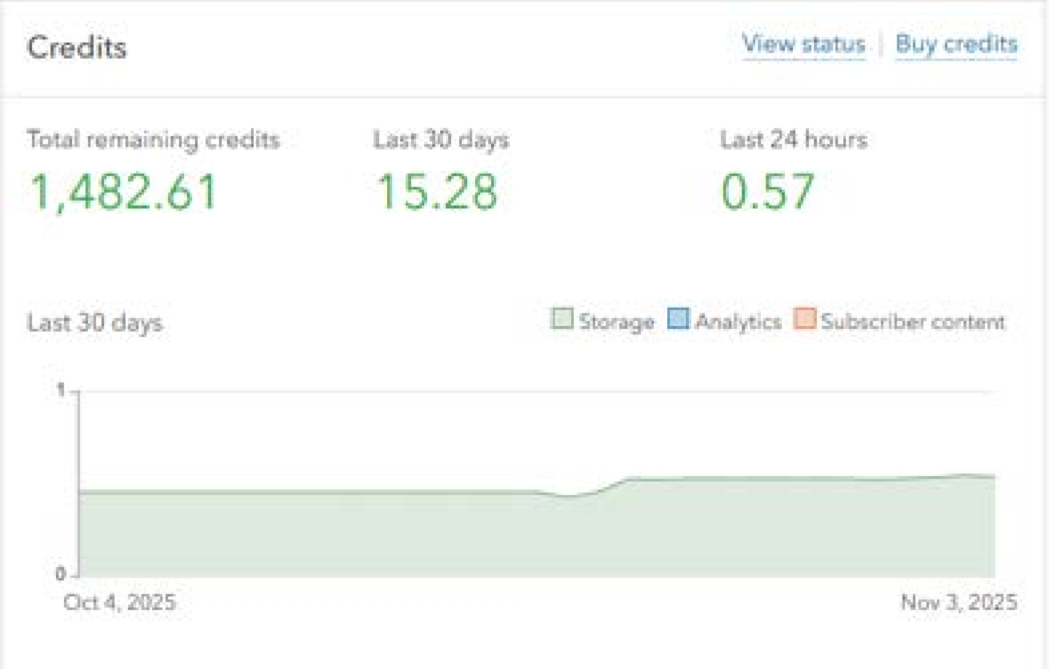
(ESRI Technical Support, 2023)

# Custom function for item size and credits

```
# Determine size and credits of an item, considering attachments.
def sizeAttachCreditTotaler(item): # With input item object, calculate size, attachments, and credit usage. N
    isize, asize, acnt, creds = item.size, 0, 0, 0 # Set item size and the empty vars
<...>
    try:
         # Have hosted feature service within org.
        if item.type == "Feature Service" and gis.properties.id in item.url:
            creds = isize/1024/1024/10*2.4/30
            if len(item.layers) > 0:
                for ix in range(len(item.layers)): # Handle attachments
                    if item.layers[ix].properties.hasAttachments: # Multiple attachment layers
                        atts = item.layers[ix].attachments.search()
                        acnt += len(atts)
                        for a in atts:
                            asize += a["SIZE"]  # Add up our attachments to calculate a credit subtractor.
                if isize > asize: # If the item size is less than its attachments, looking at BUG-000156518
                    creds -= asize/1024/1024/10*2.4/30 # Subtract the credit rate of attachments at max rate
                    creds += asize/1024/1024/1024*1.2/30 # Add attachments credits at basic rate
        elif item.type == "Notebook":
            creds = isize/1024/1024/1024*12/30
        else: # Any item not a hosted feature service or notebook
            creds = isize/1024/1024/1024*1.2/30 #  # Convert to GB and then credits For tiled imagery, storag
    except Exception as e:
        logWritr(f"7. !!!Error!!!: {e}: {item.id}")
    return isize, asize, acnt, creds
```

# Comparing credit calculation methods

- Displayed on AGO in Organization > Overview:
  - 0.57 in the last 24 hours
- Credit report:
  - 3.78 credits in previous week (minimum reporting period)
  - 3.78 / 7 = 0.54
- Items report calculations:
  - 0.554005064296875
- ArcGIS API credit calculations:
  - 0.553994041591881

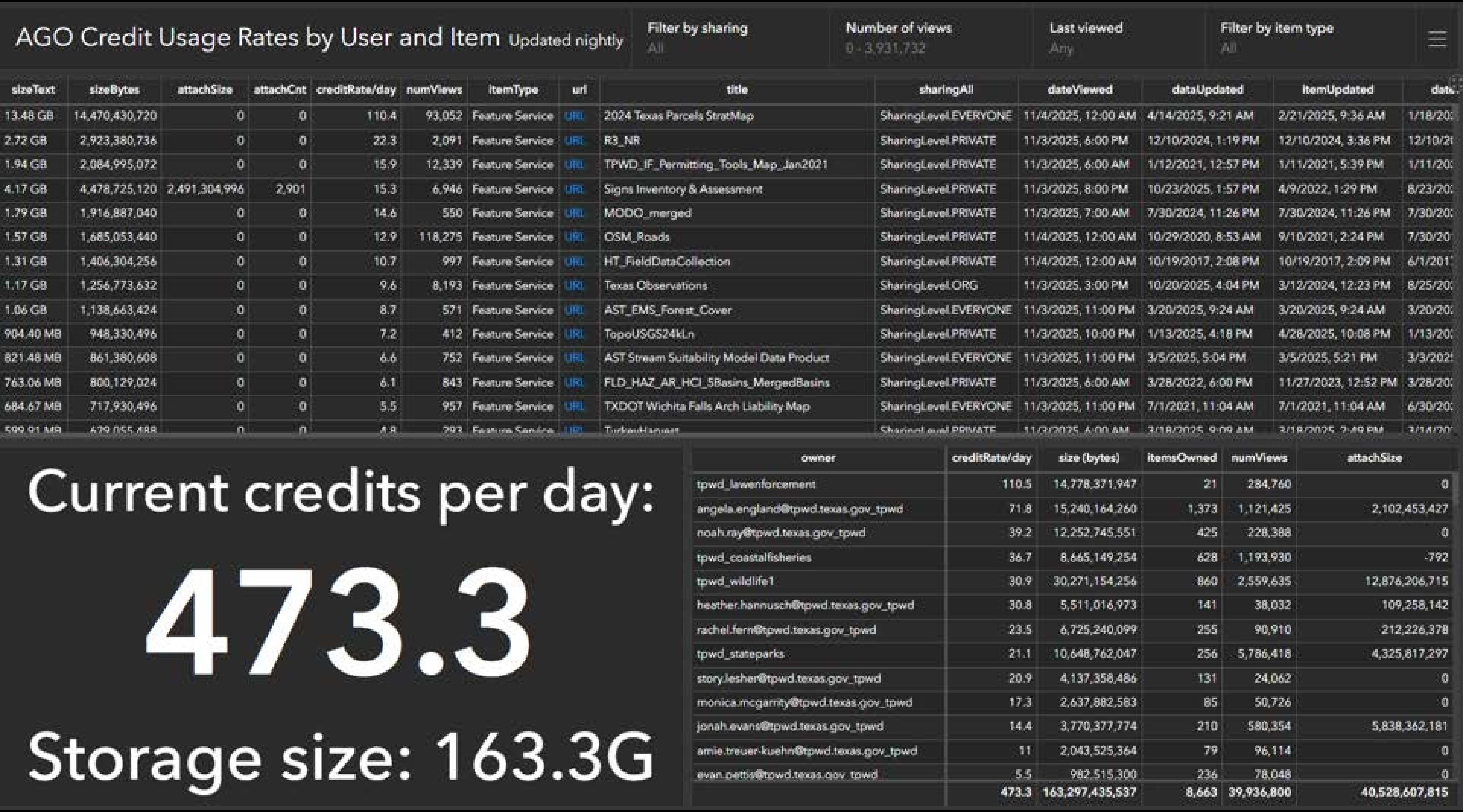


Credits

Total remaining credits 1,482.61

Last 30 days 15.28

Last 24 hours 0.57



# AGO Credit Usage Rates by User and Item Updated nightly

Filter by sharing: All

Number of views: 0 - 3,931,732

Last viewed: Any

Filter by item type: All

| sizeText | sizeBytes | attachSize | attachCnt | creditRate/day | numViews | itemType | url | title | sharingAll | dateViewed | dataUpdated | itemUpdated | data |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 13.48 GB | 14,470,430,720 | 0 | 0 | 110.4 | 93,052 | Feature Service | URL | 2024 Texas Parcels StratMap | SharingLevel.EVERYONE | 11/4/2025, 12:00 AM | 4/14/2025, 9:21 AM | 2/21/2025, 9:36 AM | 1/18/20 |
| 2.72 GB | 2,923,380,736 | 0 | 0 | 22.3 | 2,091 | Feature Service | URL | R3_NR | SharingLevel.PRIVATE | 11/3/2025, 6:00 PM | 12/10/2024, 1:19 PM | 12/10/2024, 3:36 PM | 12/10/2 |
| 1.94 GB | 2,084,995,072 | 0 | 0 | 15.9 | 12,339 | Feature Service | URL | TPWD_IF_Permitting_Tools_Map_Jan2021 | SharingLevel.PRIVATE | 11/3/2025, 6:00 AM | 1/12/2021, 12:57 PM | 1/11/2021, 5:39 PM | 1/11/20 |
| 4.17 GB | 4,478,725,120 | 2,491,304,996 | 2,901 | 15.3 | 6,946 | Feature Service | URL | Signs Inventory & Assessment | SharingLevel.PRIVATE | 11/3/2025, 8:00 PM | 10/23/2025, 1:57 PM | 4/9/2022, 1:29 PM | 8/23/20 |
| 1.79 GB | 1,916,887,040 | 0 | 0 | 14.6 | 550 | Feature Service | URL | MODO_merged | SharingLevel.PRIVATE | 11/3/2025, 7:00 AM | 7/30/2024, 11:26 PM | 7/30/2024, 11:26 PM | 7/30/20 |
| 1.57 GB | 1,685,053,440 | 0 | 0 | 12.9 | 118,275 | Feature Service | URL | OSM_Roads | SharingLevel.PRIVATE | 11/4/2025, 12:00 AM | 10/29/2020, 8:53 AM | 9/10/2021, 2:24 PM | 7/30/20 |
| 1.31 GB | 1,406,304,256 | 0 | 0 | 10.7 | 997 | Feature Service | URL | HT_FieldDataCollection | SharingLevel.PRIVATE | 11/4/2025, 12:00 AM | 10/19/2017, 2:08 PM | 10/19/2017, 2:09 PM | 6/1/201 |
| 1.17 GB | 1,256,773,632 | 0 | 0 | 9.6 | 8,193 | Feature Service | URL | Texas Observations | SharingLevel.ORG | 11/3/2025, 3:00 PM | 10/20/2025, 4:04 PM | 3/12/2024, 12:23 PM | 8/25/20 |
| 1.06 GB | 1,138,663,424 | 0 | 0 | 8.7 | 571 | Feature Service | URL | AST_EMS_Forest_Cover | SharingLevel.EVERYONE | 11/3/2025, 11:00 PM | 3/20/2025, 9:24 AM | 3/20/2025, 9:24 AM | 3/20/20 |
| 904.40 MB | 948,330,496 | 0 | 0 | 7.2 | 412 | Feature Service | URL | TopoUSGS24kLn | SharingLevel.PRIVATE | 11/3/2025, 10:00 PM | 1/13/2025, 4:18 PM | 4/28/2025, 10:08 PM | 1/13/20 |
| 821.48 MB | 861,380,608 | 0 | 0 | 6.6 | 752 | Feature Service | URL | AST Stream Suitability Model Data Product | SharingLevel.EVERYONE | 11/3/2025, 11:00 PM | 3/5/2025, 5:04 PM | 3/5/2025, 5:21 PM | 3/3/202 |
| 763.06 MB | 800,129,024 | 0 | 0 | 6.1 | 843 | Feature Service | URL | FLD_HAZ_AR_HCI_5Basins_MergedBasins | SharingLevel.PRIVATE | 11/3/2025, 6:00 AM | 3/28/2022, 6:00 PM | 11/27/2023, 12:52 PM | 3/28/20 |
| 684.67 MB | 717,930,496 | 0 | 0 | 5.5 | 957 | Feature Service | URL | TXDOT Wichita Falls Arch Liability Map | SharingLevel.EVERYONE | 11/3/2025, 11:00 PM | 7/1/2021, 11:04 AM | 7/1/2021, 11:04 AM | 6/30/20 |
| 599.91 MB | 629,055,488 | 0 | 0 | 4.8 | 293 | Feature Service | URL | TurkeyHarvest | SharingLevel.PRIVATE | 11/3/2025, 6:00 AM | 3/18/2025, 9:09 AM | 3/18/2025, 2:49 PM | 3/14/20 |

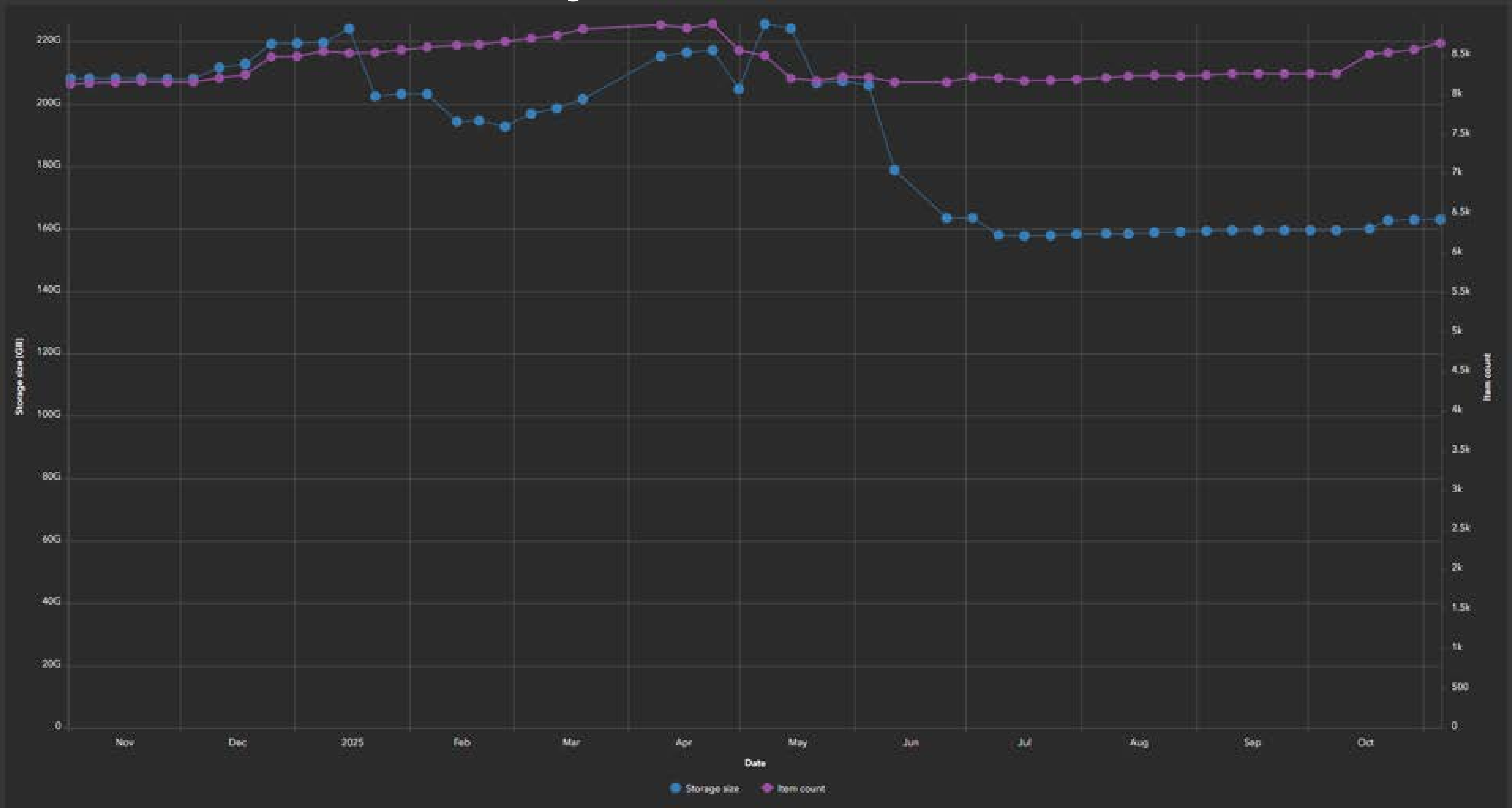
Current credits per day:

473.3

Storage size: 163.3G

| owner | creditRate/day | size (bytes) | itemsOwned | numViews | attachSize |
|---|---|---|---|---|---|
| tpwd_lawenforcement | 110.5 | 14,778,371,947 | 21 | 284,760 | 0 |
| angela.england@tpwd.texas.gov_tpwd | 71.8 | 15,240,164,260 | 1,373 | 1,121,425 | 2,102,453,427 |
| noah.ray@tpwd.texas.gov_tpwd | 39.2 | 12,252,745,551 | 425 | 228,388 | 0 |
| tpwd_coastalfisheries | 36.7 | 8,665,149,254 | 628 | 1,193,930 | -792 |
| tpwd_wildlife1 | 30.9 | 30,271,154,256 | 860 | 2,559,635 | 12,876,206,715 |
| heather.hannusch@tpwd.texas.gov_tpwd | 30.8 | 5,511,016,973 | 141 | 38,032 | 109,258,142 |
| rachel.fern@tpwd.texas.gov_tpwd | 23.5 | 6,725,240,099 | 255 | 90,910 | 212,226,378 |
| tpwd_stateparks | 21.1 | 10,648,762,047 | 256 | 5,786,418 | 4,325,817,297 |
| story.lesher@tpwd.texas.gov_tpwd | 20.9 | 4,137,358,486 | 131 | 24,062 | 0 |
| monica.mcgarrity@tpwd.texas.gov_tpwd | 17.3 | 2,637,882,583 | 85 | 50,726 | 0 |
| jonah.evans@tpwd.texas.gov_tpwd | 14.4 | 3,770,377,774 | 210 | 580,354 | 5,838,362,181 |
| amie.treuer-kuehn@tpwd.texas.gov_tpwd | 11 | 2,043,525,364 | 79 | 96,114 | 0 |
| evan.pettis@tpwd.texas.gov_tpwd | 5.5 | 982,515,300 | 236 | 78,048 | 0 |
|  | 473.3 | 163,297,435,537 | 8,663 | 39,936,800 | 40,528,607,815 |

# Storage size vs item count over time



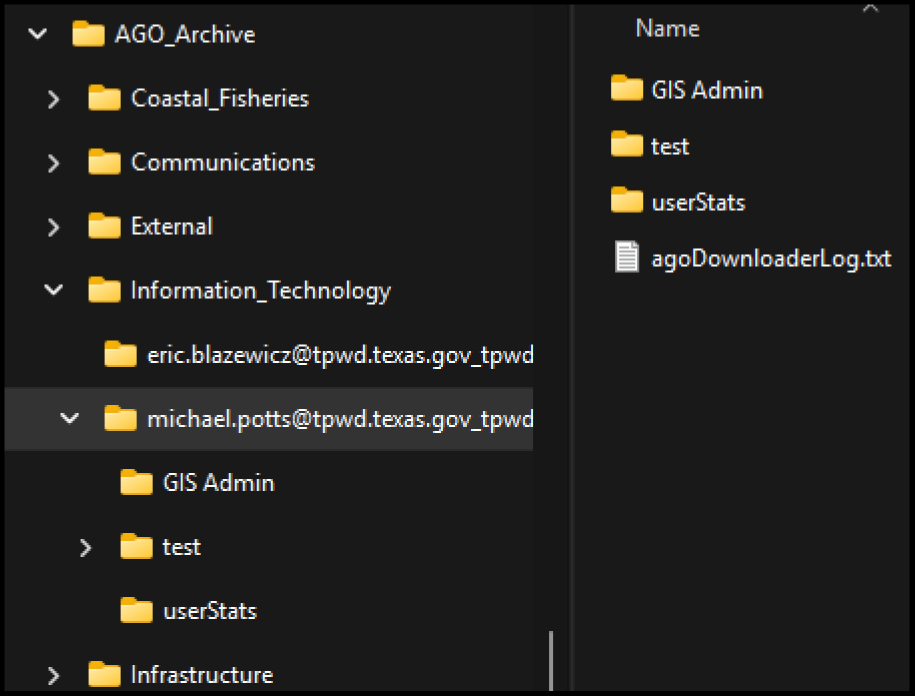
# Offloading static content options

- Export compressed (.zip) file geodatabases to read-only repository
- Load data to file geodatabases on a shared drive for desktop GIS access
- Load data to file geodatabases on a server and publish as map services
    - Maximal service performance by this method
- Host data in Portal

# Archiving AGO content to disk

```
outDir = os.path.join(r"\\GIS-DATASRVR\GISProjects\AGO_Archive",divison,username)
```

- Create a directory in a shared drive
    - Folder for each TPWD division
        - Read-only folder for each user
            - Folder corresponding to user’s AGO folders



```
# Create dir_path if it doesn't already exist.
def dirExists(dir_path):
    if not os.path.exists(dir_path):
        os.makedirs(dir_path)
        logWritr(f"Directory '{dir_path}' created.")
    else:
        print(f"Directory '{dir_path}' already exists.")
    return dir_path
```

# Create log file

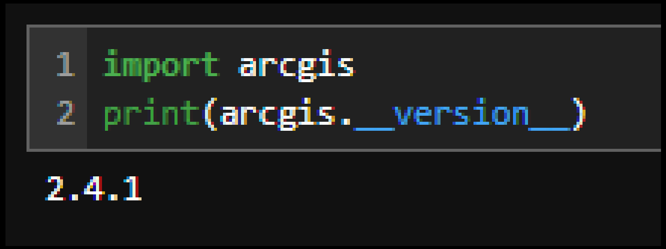
- The log file tracks downloads and serves as table of contents
- Download items to read-only directory on shared drive with a log file

```python
def logWritr(logMsg):
    fname = "agoDownloaderLog"
    activityLog = open(outDir + "\\" + fname + ".txt", "a")
    msgTime = datetime.datetime.now()
    activityLog.write(str(msgTime) + " | " + logMsg +  "\n")
    print(str(msgTime) + " | " + logMsg)
    activityLog.close()
```

```
2025-10-30 13:04:35.593800 | --> Folder: 'test' (27edb677c1b442c29af52f4f983e7061); 3 items
2025-10-30 13:05:30.705533 | Downloaded 0001, boundaries test, Feature Service, fc23ec2a2b504c0f8d03313fcc79ebbd, 1.51 MB
2025-10-30 13:05:31.186031 | Downloaded 0002, boundaries test, Web Map, 1a47881684ca4b27a72068d04e474fc5, 1.12 KB
2025-10-30 13:05:31.673037 | Downloaded 0003, boundaries test, Service Definition, 4546a23a9eee4c85ba75a07b28aee446, 571.27 KB
```

# Build dictionary of folders with input folder IDs

- Due to changes between ArcGIS Python API 2.3 and 2.4, must use generator object to build dictionary of folder names

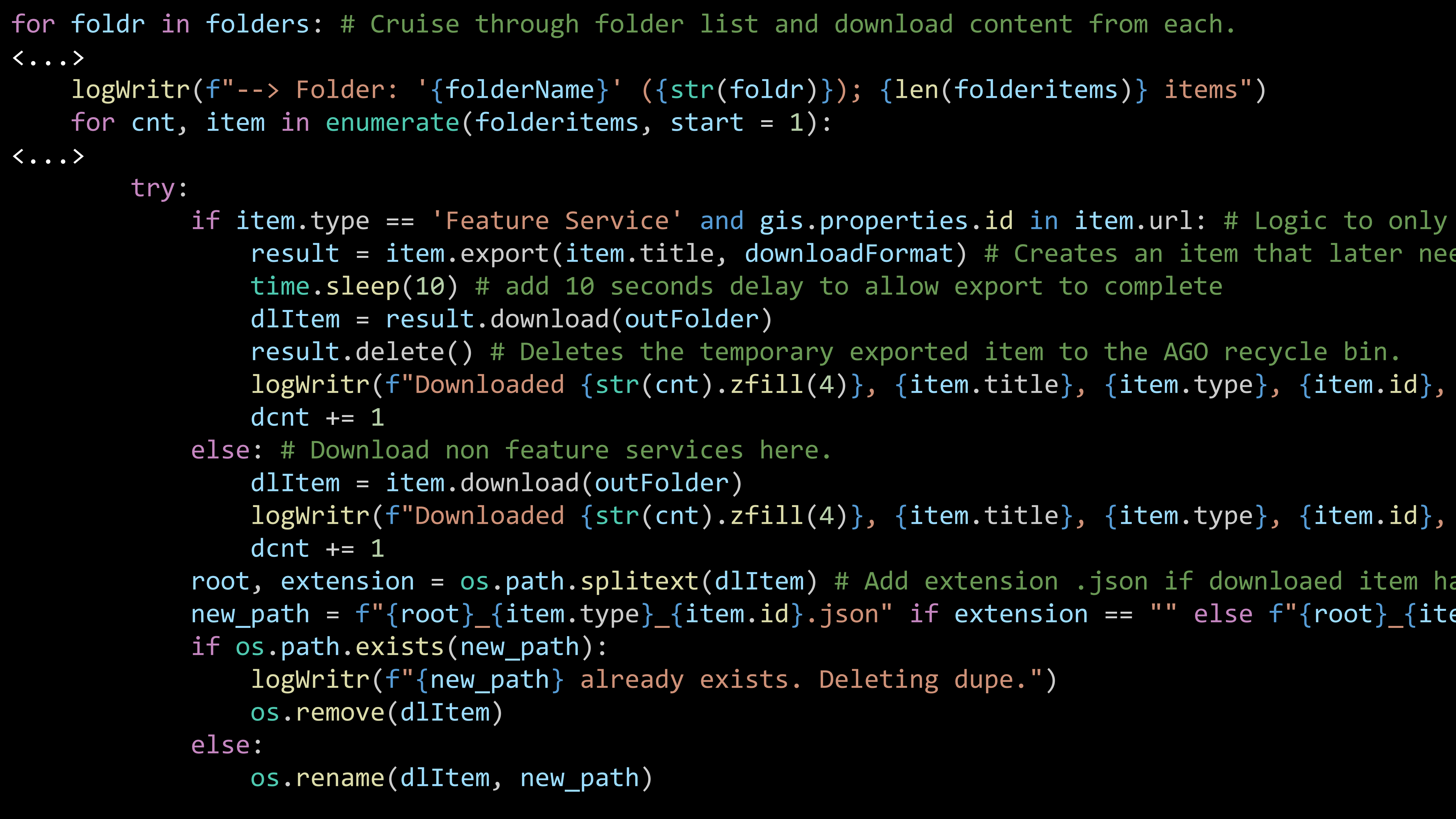


```python
import arcgis
print(arcgis.__version__)
```

```
2.4.1
```

```python
##### ##### Inputs Start ##### #####
gis = GIS(profile = "python_playground")
downloadFormat = "File Geodatabase" # Options: Shapefile, CSV, File Geodatabase, Feature Collec
username = "michael.potts@tpwd.texas.gov_tpwd" #
usr = gis.users.get(username)
##### ##### Inputs End ##### #####

folders = ["27edb677c1b442c29af52f4f983e7061", "631c58742ee740478a0a95f75c4d1d25", "63d22ed9e79
folders_mgr = gis.content.folders
folderGenerator = folders_mgr.list(owner=usr) # Folder generator object.
foldersDictList = [f.properties for f in list(folderGenerator)]
for f in foldersDictList:
    folderNameDict[f.get("id")] = f.get("title")
```



```
for foldr in folders: # Cruise through folder list and download content from each.
<...>
    logWritr(f"--> Folder: '{folderName}' ({str(foldr)}); {len(folderitems)} items")
    for cnt, item in enumerate(folderitems, start = 1):
<...>
        try:
            if item.type == 'Feature Service' and gis.properties.id in item.url: # Logic to only
                result = item.export(item.title, downloadFormat) # Creates an item that later nee
                time.sleep(10) # add 10 seconds delay to allow export to complete
                dlItem = result.download(outFolder)
                result.delete() # Deletes the temporary exported item to the AGO recycle bin.
                logWritr(f"Downloaded {str(cnt).zfill(4)}, {item.title}, {item.type}, {item.id},
                dcnt += 1
            else: # Download non feature services here.
                dlItem = item.download(outFolder)
                logWritr(f"Downloaded {str(cnt).zfill(4)}, {item.title}, {item.type}, {item.id},
                dcnt += 1
            root, extension = os.path.splitext(dlItem) # Add extension .json if download item ha
            new_path = f"{root}_{item.type}_{item.id}.json" if extension == "" else f"{root}_{ite
            if os.path.exists(new_path):
                logWritr(f"{new_path} already exists. Deleting dupe.")
                os.remove(dlItem)
            else:
                os.rename(dlItem, new_path)
```

# Error handling

- Sometimes items don’t export or download successfully

```
for foldr in folders: # Cruise through folder list and download content from each.
<...>
    logWritr(f"--> Folder: '{folderName}' ({str(foldr)}); {len(folderitems)} items")
    for cnt, item in enumerate(folderitems, start = 1):
<...>
        try:
<...>
        except Exception as e:
            errList.append((item.title,item.type,item.id)) # If an error occurs, build list of fa
            ecnt += 1
            logWritr(f"--!!! An error occurred1: {e}; {item.title}, {item.id}")
endTime = datetime.datetime.now()
logWritr(f"Downloaded {str(dcnt)} and skipped {str(scnt)} items with {str(ecnt)} errors in {str(e
logWritr("Errors:" + str(errList))
```

# Deleting items in a recycle bin

- When an item is exported, an exported item is created in the user’s content
- Must clear recycle bin to prevent extra credit usage

```
# Deletes all items in the current user's recycle bin.
my_user = gis.users.me
content = my_user.recyclebin.content
for item in content:
    item.delete()
    print(f"{item.properties['title']} deleted")
```

# Offloading dynamic content for data collection options

- Load items into SDE geodatabases
    - Publish feature classes to Server as feature services
    - Consume these feature services in AGO, Portal, and other web apps
- Load items directly into Portal
    - Publish hosted feature layers to consume in apps

# Downloading AGO content directly to a file geodatabase

```
def directlyExportFeatures(inid, outDir):
    item = gis.content.get(inid)
    outgdb = f"{item.title}_{item.type}_{item.id}.gdb"
    gdbPath = os.path.join(outDir,outgdb)
    if not arcpy.Exists(os.path.join(outDir,outgdb)):
        arcpy.management.CreateFileGDB(outDir, outgdb)
        logWritr(f"Created new gdb: {gdbPath}")
    else:
        logWritr(f"{gdbPath} already exists")
    for l in item.layers: # Export features in loop
        out_features = os.path.join(gdbPath,l.properties.name)
        arcpy.conversion.ExportFeatures(l.url, out_features)
        logWritr(f"Created new fc: {out_features}")
```

# Reliably downloading content to Enterprise GDB

- Download item
- Unzip item
- Copy item into Enterprise geodatabase

# Download content and unzip

```
item = gis.content.get(itemId) # cwd all points
outFolder = os.path.join(downloads, item.title)
os.makedirs(outFolder)
downloadFormat = "File Geodatabase" # Options: Shapefile, CSV, File Geodatabase, Feature Coll
if item.type == 'Feature Service' and gis.properties.id in item.url: # Logic to only grab hos
    result = item.export(f"{item.title}_gdb", downloadFormat) # Creates an item that later ne
    time.sleep(10) # add 10 seconds delay to allow export to complete
    dlItem = result.download(outFolder)
    result.delete() # Deletes the temporary exported item to the AGO recycle bin.
logWritr(f"Downloaded {item.title} to {outFolder}")
with zipfile.ZipFile(dlItem, 'r') as zip_ref: # Unzip downloaded content.
    unzip = zip_ref.extractall(outFolder)
logWritr(f"Unzipped download successfully.")
for root, dirs, files in os.walk(outFolder):
    for dir_name in dirs:
        if dir_name.endswith(".gdb"):
            arcpy.env.workspace = os.path.join(outFolder,dir_name)
            fcs = arcpy.ListFeatureClasses()
```

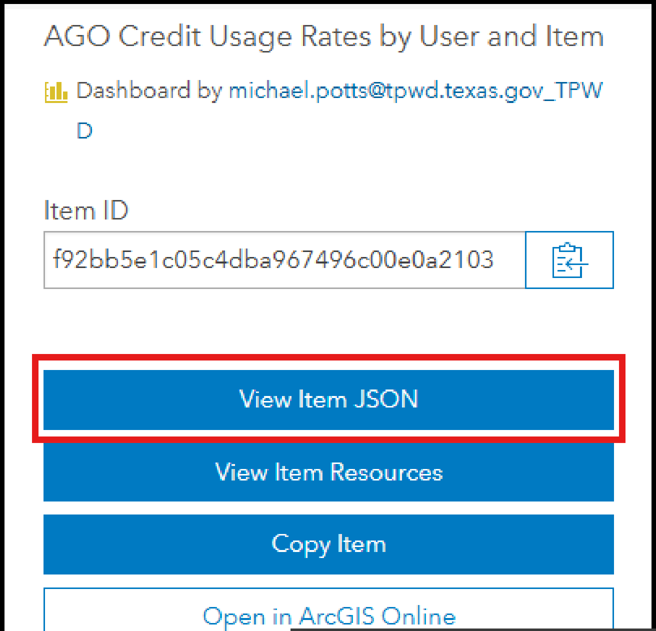
# Copy .gdb features to Enterprise GDB

```
def copyDataGDB2SDE(srcGDB, dstSDE):
    arcpy.env.workspace = str(srcGDB)
    indaList = arcpy.ListTables("*","ALL") + arcpy.ListFeatureClasses("*","ALL") + arcpy.ListRasters("*","ALL
    rc_list = [c.name for c in arcpy.Describe(str(srcGDB)).children if c.datatype == "RelationshipClass"] # N
    # To prevent duplication of fcs, handle members of relationship classes by removing the destination.
    for rc in rc_list:
        rc_path = str(srcGDB) + "\\" + rc
        des_rc = arcpy.Describe(rc_path)
        origin = des_rc.originClassNames
        destination = des_rc.destinationClassNames
        indaList.remove(destination[0])
    indaList.sort()
    for dat in indaList:
        FCFullName = str(dat).strip()
        fcInName = str(srcGDB) + "\\" + str(FCFullName)
        fcOutName = str(dstSDE) + "\\" + str(FCFullName).split(".")[-1] # take last com
        in_data = fcInName
        out_data = fcOutName
        print(f"Copying: {in_data} to: {out_data}")
        arcpy.management.Copy(in_data, out_data)
```

# Updating apps with ArcGIS Online Assistant (Beta)

- https://assistant.esri-ps.com/
- Works for both ArcGIS Online and ArcGIS Portal
- Apps are coded in .JSON
- App portability between platforms (AGO ←→ Portal) is not great, must test on your own depending on versions
- Example of offloading data to Enterprise and using ArcGIS Online Assistant to update data sources

- Find model app in ArcGIS Online Assistant
- Click on item > View item JSON
- In Data tab > Edit JSON
- Copy JSON to a code editor, like Visual Studio Code

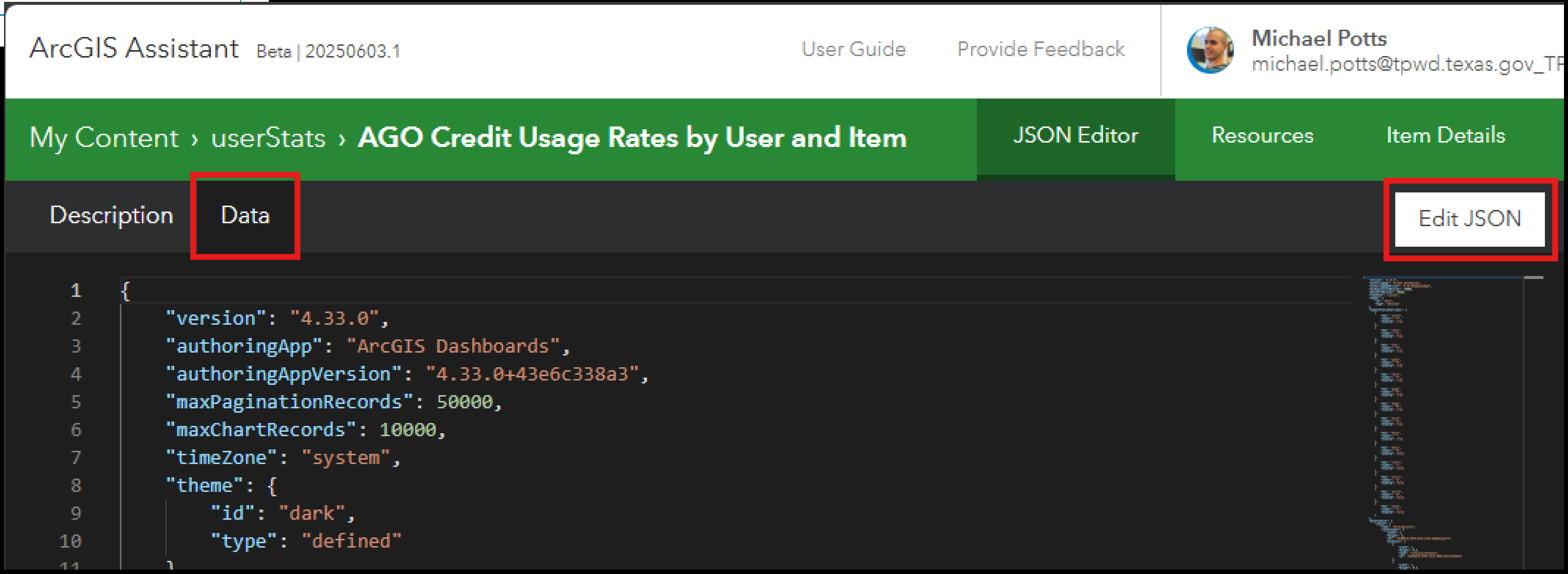


AGO Credit Usage Rates by User and Item

Dashboard by michael.potts@tpwd.texas.gov_TPWD

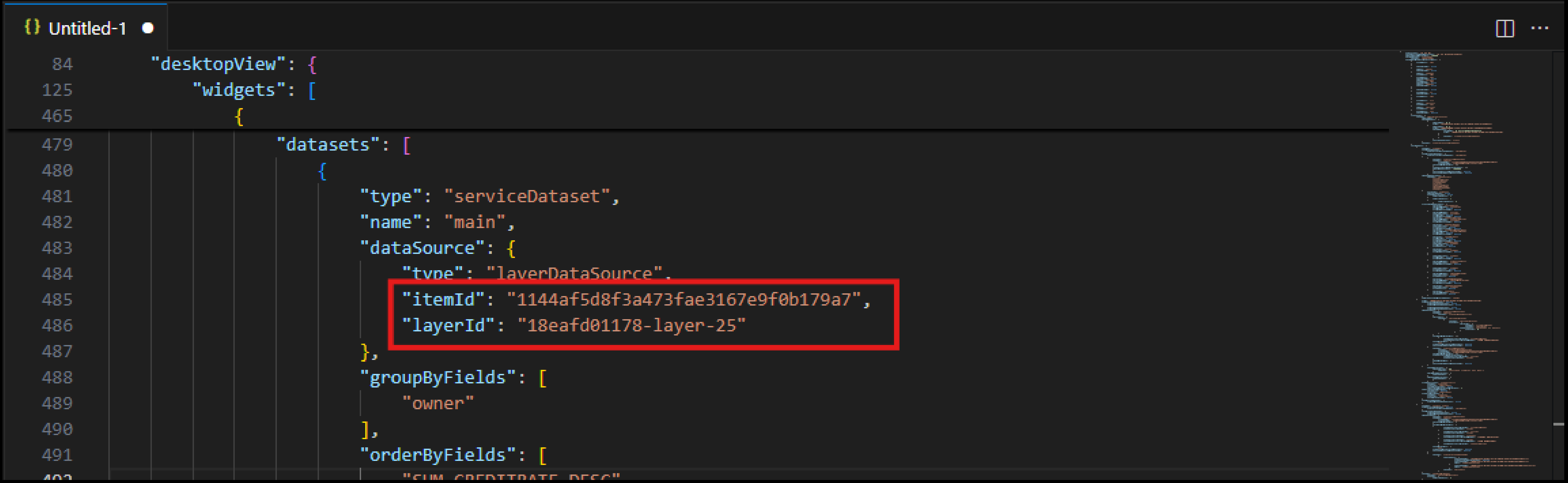
Item ID

f92bb5e1c05c4dba967496c00e0a2103

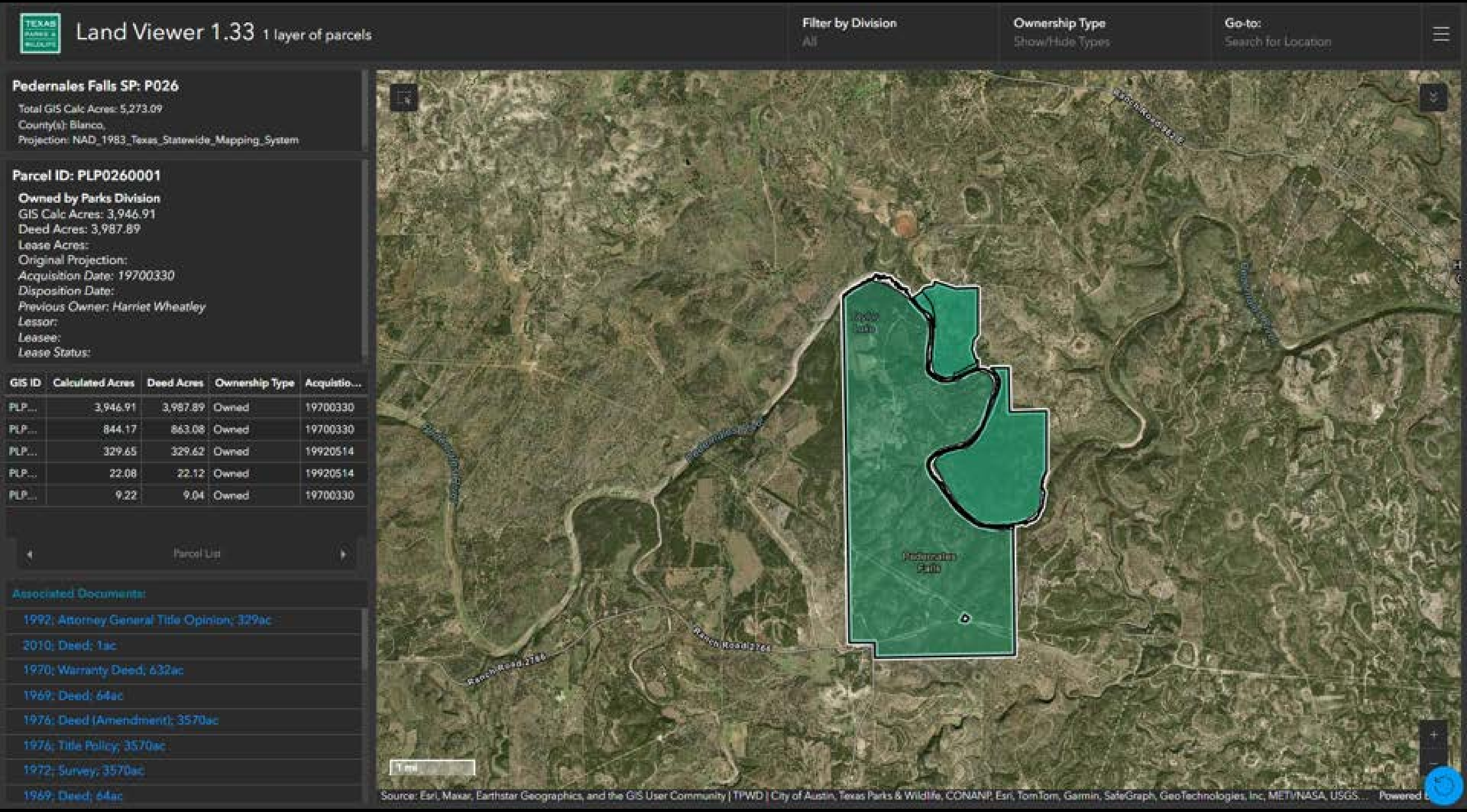


```
{
    "version": "4.33.0",
    "authoringApp": "ArcGIS Dashboards",
    "authoringAppVersion": "4.33.0+43e6c338a3",
    "maxPaginationRecords": 50000,
    "maxChartRecords": 10000,
    "timeZone": "system",
    "theme": {
        "id": "dark",
        "type": "defined"
    }
```

- May have to create a donor app to generate if using a webmap instead of layer reference
- Find and replace all occurrences
  - itemId: webmap
  - layerId: retrieve uniquely value generated from a donor app
- Paste updated JSON code into ArcGIS Online Assistant in the donor app



```
    "desktopView": {
        "widgets": [
            {
                "datasets": [
                    {
                        "type": "serviceDataset",
                        "name": "main",
                        "dataSource": {
                            "type": "layerDataSource",
                            "itemId": "1144af5d8f3a473fae3167e9f0b179a7",
                            "layerId": "18eafd01178-layer-25"
                        },
                        "groupByFields": [
                            "owner"
                        ],
                        "orderByFields": [
                            "SUM_CREDITRATE_DESC"
```



Land Viewer 1.33 1 layer of parcels

Filter by Division
All

Ownership Type
Show/Hide Types

Go-to:
Search for Location

**Pedernales Falls SP: P026**

Total GIS Calc Acres: 5,273.09
County(s): Blanco,
Projection: NAD_1983_Texas_Statewide_Mapping_System

**Parcel ID: PLP0260001**

**Owned by Parks Division**
GIS Calc Acres: 3,946.91
Deed Acres: 3,987.89
Lease Acres:
Original Projection:
*Acquisition Date: 19700330*
*Disposition Date:*
*Previous Owner: Harriet Wheatley*
*Lessor:*
*Leasee:*
*Lease Status:*

| GIS ID | Calculated Acres | Deed Acres | Ownership Type | Acquisitio... |
|---|---|---|---|---|
| PLP... | 3,946.91 | 3,987.89 | Owned | 19700330 |
| PLP... | 844.17 | 863.08 | Owned | 19700330 |
| PLP... | 329.65 | 329.62 | Owned | 19920514 |
| PLP... | 22.08 | 22.12 | Owned | 19920514 |
| PLP... | 9.22 | 9.04 | Owned | 19700330 |

◂ Parcel List ▸

Associated Documents:

- 1992; Attorney General Title Opinion; 329ac
- 2010; Deed; 1ac
- 1970; Warranty Deed; 632ac
- 1969; Deed; 64ac
- 1976; Deed (Amendment); 3570ac
- 1976; Title Policy; 3570ac
- 1972; Survey; 3570ac
- 1969; Deed; 64ac

Source: Esri, Maxar, Earthstar Geographics, and the GIS User Community | TPWD | City of Austin, Texas Parks & Wildlife, CONANP, Esri, TomTom, Garmin, SafeGraph, GeoTechnologies, Inc, METI/NASA, USGS... Powered b

# Conclusions

- Storage bloat happens to all of us.
- There are multiple ways to prevent storage bloat.
- There are multiple challenges we must face because of storage bloat.
- There are multiple ways to identify bloat.
- There are multiple ways to offload content.

# References

“ArcGIS Online Credits Pricing.” ESRI Canada. 19 October 2025. <https://www.esri.ca/en-ca/store/products/buy/credits>.

“Manage Your ArcGIS Online Credits in Five Minutes.” ESRI Australia. 10 January 2025. <https://esriaustraliatechblog.wordpress.com/2025/01/10/manage-your-arcgis-online-credits-in-five-minutes/>.

“Report fields—ArcGIS Online Help | Documentation.” ArcGIS Online Help. 15 October 2025a. <https://doc.arcgis.com/en/arcgis-online/reference/report-fields.htm>.

“Understand credits.” ArcGIS Online Help. 19 October 2025b. <https://doc.arcgis.com/en/arcgis-online/administer/credits.htm>.

“The size of item attachments is not consistently displayed on the item details page in ArcGIS Online.” ESRI Technical Support. 22 March 2023. <https://support.esri.com/en-us/bug/the-size-of-item-attachments-is-not-consistently-displa-bug-000156518>.

“Query regarding credit calculation, No 04028613.” ESRI Technical Support Cases. 21 October 2025.