

# Extending Gadgets with APIs

Brandon Dougherty  
Front-End Developer  
@bdougherty24

**SPOTLIGHTING INNOVATION**  
THE 2016 OMNIUPDATE USER TRAINING CONFERENCE

# Agenda



- API Calls / Endpoints
- Gadget Installation / Configuration
- Gadget Events and API Calls / Notifications
- Metadata API

# The Gadget Logic

- Page Parameters Undo Gadget
  - 1) Get the parameters of the current file
  - 2) Wait until the user saves the page parameters
  - 3) save the snapshot of the old parameters
  - 4) Utilize saved data from step 3 to revert.

# API Calls/Endpoints

- API calls represent specific operations that your application can execute to perform a task
  - » Read / Query
  - » Create, Update, Delete
  - » Perform system tasks

# API Calls/Endpoints

- API calls are made via the HTTP methods GET, POST
- The response body returns JSON

# API Calls/Endpoints

- An API call must contain an authorization token
- gadgetlib.js retrieves this token for you :)

1

/users/list?authorization\_token=xf1 g3t2re...

Gadget



OU Campus™

(Response body)  
[{username : john, id : 1},  
{username : alice, id : 2}...]

2

# The Gadget

- Page Parameters Revert Gadget
  - 2 API calls
    - GET, POST
  - Listen for 2 OU Campus Events
    - params-save, view\_changed
  - 3 Metadata API Calls
    - create, link, delete





# Let's Code!

<https://github.com/omniupdate/gadget-starter>

# Gadget Events

- JavaScript events that are triggered when various actions are performed in OU Campus
  - » File operations (move, copy, rename)
  - » Publish
  - » Message
  - » Workflow

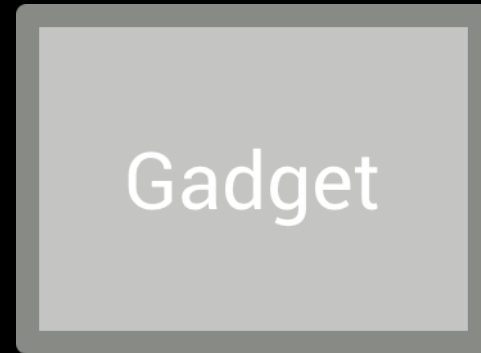
1

User clicks save  
in WYSIWYG editor



\*

Im listening for  
a save event!



2

OU Campus broadcasts the  
save event to all registered  
gadgets.

3

The gadget recieves the  
event.

# Metadata API

- Interface for Gadgets to store their own persistent data, or even share it!



# Metadata API

- Store data that is just visible to:
  - » Current User
  - » Publicly (within OU Campus account)
  - » Group

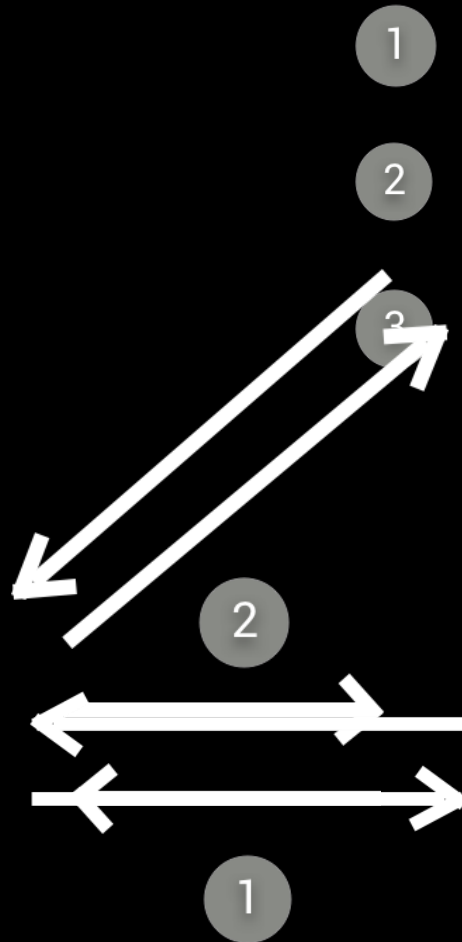
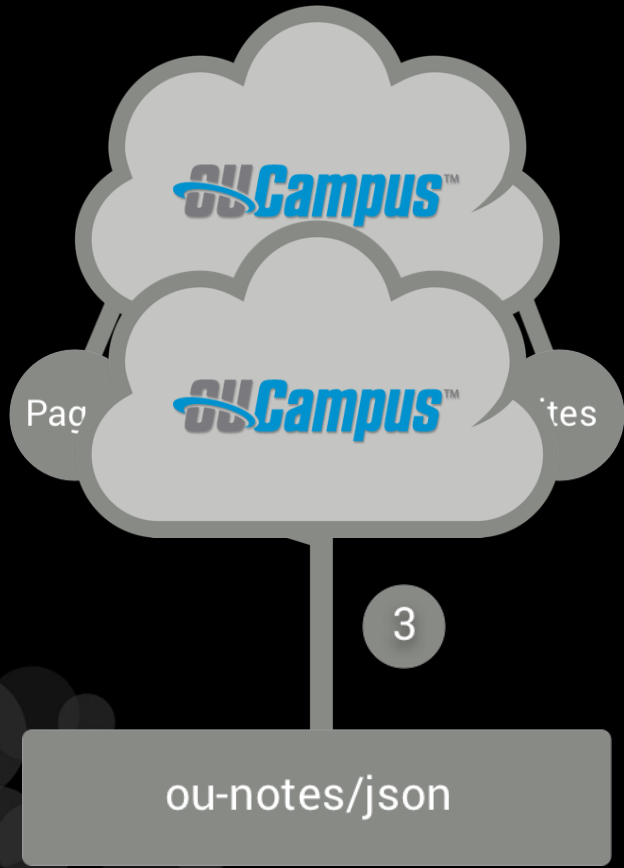
# Metadata API

- Create relationships with OU Campus data:
  - » Files
  - » Directories
  - » Assets
  - » Sites

# Metadata API

- Data is stored and accessed via a Mime Type
  - » Ex. ou-notes/json





1 Gadget calls /metadata/create, creating a new metadatum.

2  
Gadget

this  
off Mime Type.

linked to Pages,  
directories

2  
OU Notes  
Gadget

1



# Metadata API

- Uses Cases:
  - Store simple notes
  - Store Images/Screenshots (base64 string)
  - Share stored data between users



**Back to work!**



**Enjoy your gadget!**

<https://github.com/omniupdate/gadget-starter>



**Enjoy your gadget!**

<https://github.com/omniupdate/gadget-starter>

# Thank you!

*Don't forget to take our survey*

[outc16.com/surveys](https://outc16.com/surveys)