Global Project Linking and External Resources

PRESENTED BY
Matthew Ellison and Thomas Bro-Rasmussen
WHAT WE’LL COVER IN THIS SESSION

• What Global Project Linking (GPL) and External Resources (ER) have in common
• Questions to consider when choosing between them
• The differences between GPL and ER
• Setting up and linking a global project
• Suggestions on what to include in a global project
• Setting up and synchronizing external resources
THE CONCEPT OF A FLARE PROJECT

- SELF-CONTAINED

**Content**
- Topics
- Images
- Snippets
- Style sheet

**Project**
- TOC
- Skin
- Glossary
- Target
- etc.
WHAT WE ARE AIMING FOR

FLARE PROJECT

CENTRAL REPOSITORY OF FILES

Files copied and synchronized

FLARE PROJECT

FLARE PROJECT

FLARE PROJECT
EXTERNAL RESOURCES

FLARE PROJECT

- style.css*
- warning.flsnp
- logo.png*

Local Folder
Or
SharePoint Server

FLARE PROJECT

- style.css*
- warning.flsnp
- logo.png*
QUESTIONS TO CONSIDER

- Who will be responsible for maintaining shared resources?
- How do you plan to edit shared resources?
- Do you want to allow team members to edit shared resources within their own projects?
- Do you want to allow team members to remove links to shared resources?
- Do you want shared resources to be project- or machine-specific?
<table>
<thead>
<tr>
<th></th>
<th>Global Project Linking</th>
<th>External Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow of updates</td>
<td>only from global project to “child” project</td>
<td>both directions</td>
</tr>
<tr>
<td>Shared resources stored in…</td>
<td>a Flare project</td>
<td>a folder or SharePoint Server</td>
</tr>
<tr>
<td>Special project file required in “child” project</td>
<td>Yes (Flare Project Import File)</td>
<td>No</td>
</tr>
<tr>
<td>Option to auto-include linked files?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Option to auto-reimport before generate output?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Warning when edit shared resources in “child” project?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Changes can be merged?</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Global Project Linking
PROJECT LINKING

• An important single-sourcing technique
• …for sharing content between two different Flare projects
• [What’s wrong with just copying files?]
• The magic (.flimpl) is in the receiving project
POSSIBLE LINKING STRATEGIES

- Related projects
- Adhoc/opportunistic
- Top-down (global)
SHARING BETWEEN RELATED PROJECTS

Project A

Project B

Project C

Project D

Import File

ProjectA.flprj

*.flsnp

Flare Project
Import File
ADHOC/OPPORTUNISTIC SHARING

Project A

Project B

Project C

Project D

Flare Project Import File

ProjectD.flprj

Example1.flsnp
GLOBAL PROJECT

Project B

Project C

Project D

Global Project A

Global.

Global.flprj

*.flsnp

Flare Project Import File
MERGED PROJECT

- Project A
- Project C
- Project D

Merged

Flare Project Import File for each sub project
OVERVIEW OF THE GPL PROCESS

1. Create and populate a global project

2. Add a Flare Project Import File to a “child” project, and enter the required settings

3. Copy the Flare Project Import File to other “child” projects

I recommend adding the Flare Project Import File to your project template
Demonstration of the GPL process
SETTING UP OPTIONS FOR SOURCE PROJECT

Global project

Import everything...

...except this topic and any target
EDITING A LINKED ITEM

- Best practice: edit only in global project
- Visual indicator of linked item
- Warning if you edit in “child” project
REMOVED LINKS ITEMS

Flare Project Import File

Click to restore link

Restore Link  Open
TWO OPTIONS FOR RE-IMPORTING

- Manual
- Automatic
Newer version in global project selected automatically
WHAT TO INCLUDE IN A GLOBAL PROJECT

- Images (icons, logos, …)
- Snippets (standard warnings, copyright & legal, …)
- Style sheets
- Page layouts
- Skins
- Master pages
- Glossary
WHAT TO INCLUDE IN A GLOBAL PROJECT (CONTINUED)

- Condition tag set
- Variable set
- Synonym file
- Sample topic (demonstrating every style)
- and more…
NAMING CONVENTIONS

• Need to agree and plan consistent file-naming conventions across all projects

• Filenames must:
  – be unique
  – clearly indicate content and type of topic

• Filenames may:
  – indicate source project/author
  – include codes or version numbers
BRINGING SOURCE CONTROL INTO THE PICTURE

FLARE PROJECT

Central Source Control Repository

AUTHOR A

AUTHOR B

AUTHOR C

GLOBAL PROJECT

Source Control
SUGGESTED BEST PRACTICE FOR SOURCE CONTROL

• Always pull (get latest files) from source control before:
  – importing from Global Project
  – building output

• Only one author should have responsibility for working on the Global Project
External Resources
OVERVIEW OF THE ER PROCESS

1. Add folder(s) for external resources

2. Populate folder(s) with files

3. Copy files from external resources to “child” projects (and create mapping)
Demonstration of the ER process
Click to add new External Resources folder

Use both if you want to select several files
COPYING TO PROJECT

Select to create mapping (not selected by default)
EDITING A MAPPED ITEM

- Can be done either in external resources folder or in “child” project

- Visual indicator of mapped item

- No warning if you edit in “child” project
SYNCHRONIZING FILES

File changed in project

File changed in "child" project

Files synchronized successfully.
MERGE CHANGES WINDOW

Click here to accept change

OK appears when all conflicts have been resolved
THINGS TO CONSIDER WITH ER

• Since Flare “only” synchronizes on date, it is important that all users computers are “date-aligned”

• Using a Source Control across time zones has to be accounted for.
An alternative approach
CONTROLLING “ER” AT BUILD TIME

• The Auto-Reimport uses in GPL can indirectly be applied to ER
• In all Flare targets you have the ability to start a script prior to the build (and after)
• Basically it adds “auto” to ER
THIS IS WHAT FLARE DOES (IN THE FLPRJ)

<Mapping ProjectPath="Content/copyright.flsnp"

• These files also appear in the FileSync folder.

In the Flare project file files synchronized are shown.
THIS IS WHAT DOS BATCH WOULD DO

FROM

c:\Data\Dropbox\Kunder\madcap\madworld 2018\US\GPL\External resources\text\language\manufacturer\copyright.flsnp

TO

c:\Data\Dropbox\Kunder\madcap\madworld 2018\US\GPL\External resource folder\Content\n
FINAL DOS COMMAND

XCOPY “c:\Data\Dropbox\Kunder\madcap\madworld 2018\US\GPL\External resources\text\language\manufacturer\copyright.flsnp”
“c:\Data\Dropbox\Kunder\madcap\madworld 2018\US\GPL\External resource folder\Content\” /e /i /y /q /r >NUL
REFINED DOS BATCH

off
MODE CON: COLS=84 LINES=20
cls
set source=c:\Data\Dropbox\Kunder\madcap\madworld 2018\US\GPL\External resources\text\language\manufacturer
set target=:\Data\Dropbox\Kunder\madcap\madworld 2018\US\GPL\External resource folder\Content
cls
echo .
echo ... Copying files ...
XCOPY "%source%\copyright.flsnp" "%target%" /e /i /y /q /r >NUL
SAVING A DOS BATCH FILE

• Save the code from before into a DOS BATCH file: e.g.: External resources.bat
RUN BATCH FILE FROM TARGET

Insert path+batch filename

When target builds this will run first and show in build log
SUMMARY

• GPL and ER both offer a way to copy shared resources to multiple projects

• GPL provides a more centrally-controlled workflow

• ER offers more flexibility and is arguably easier to set up

• ER+GPL (BATCH) can be controlled from e.g. Excel/Database
Matthew Ellison and Thomas Bro-Rasmussen