



Table of Contents

SSI's Multi-Project Management Tools	3
Overview.....	3
Multi-Project Management Basics.....	4
What is a Subproject?	4
What is a Master Project?	5
What is a Handoff?	5
Reasons to Breakdown a Project into Subprojects.....	6
Understanding the Handoff Method.....	7
Types of Handoff Relationships	9
Class Exercise – Code and Flag Handoffs.....	11
Setting up a Project Unique ID (PUID)	15
Class Exercise – Setting up a Project Unique ID Field	15
Using “Shell” Files	17
Setting up the Subproject Shell	17
Setting up the Master Project Shell file.....	17
Class Exercise – Creating Shell files	18
Class Exercise – Setting up Project B.....	24
Integrating Subprojects into the Master Project.....	26
“Linked” Subprojects and Master Project	27
“Unlinked” Subprojects and Master Project.....	30
Class Exercise – Insert Subprojects into the Master Project Shell	31
Using the Handoff Linker Tool	33
Class Exercise – Using the Handoff Linker Tool.....	34
Synchronizing Subprojects.....	37
Class Exercise – Using the Sync Remote Projects Tool to Post Data to Subprojects	38



Multi-Project Environment Roles Explained	43
Subproject Owner	43
Master Project Integrator	43
SSI's Recommended Multi-Project Management Process	44



SSI's Multi-Project Management Tools

Overview

SSI's Multi-Project Tools are built to help users who currently or intend to work in a Multi-Project Environment. These tools give Microsoft Project users the ability to decompose a large, complex project into smaller, separate Subproject files without losing dependencies. These tools work using the Handoff or Giver/Receiver method and are designed to be a substitute to Microsoft Project's built in Linked Subproject feature, which is known to cause many issues, the biggest of which being far increased probability of file corruption.

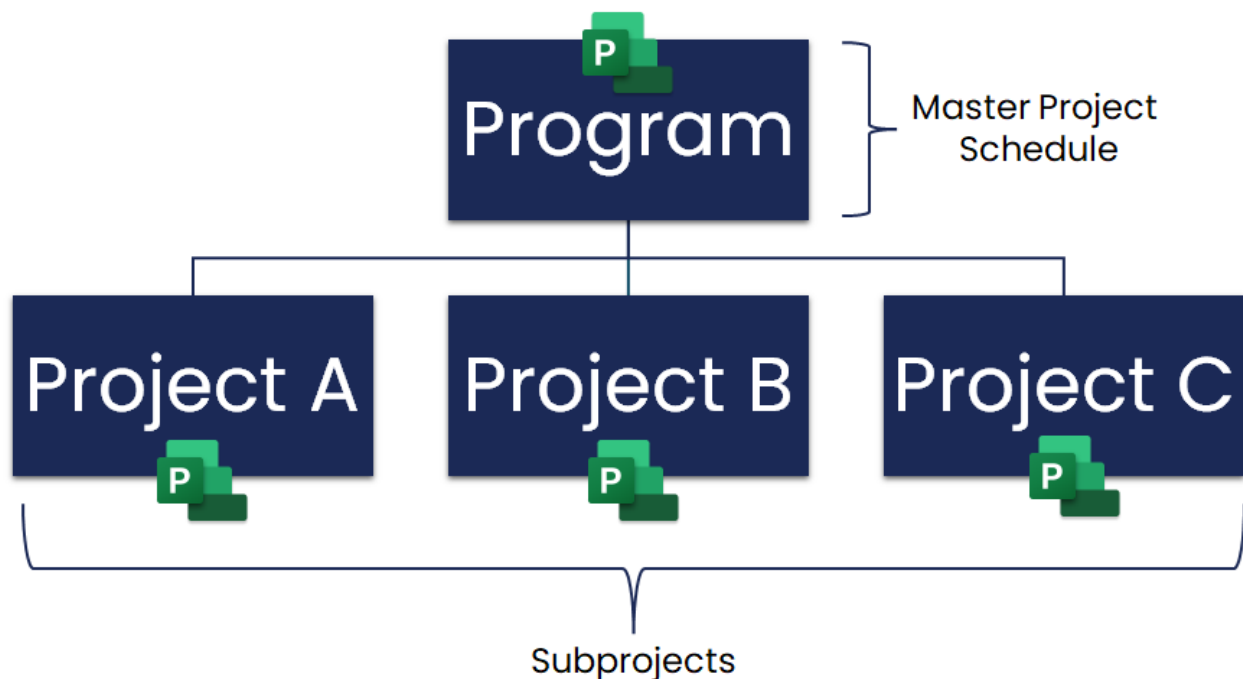
SSI Multi-Project Tools are useful for:

1. Decomposing a very large project into separate, stand-alone Subproject files.
2. Combining two or more project files into a larger Master Project Schedule.
3. Explicitly identifying dependencies between Subprojects using unique codes and flags.
4. Managing shared resources and Handoffs between Subprojects.
5. Summarizing groups of tasks at a high level in another project.



Multi-Project Management Basics

A **Multi-Project Environment** is composed of two or more **Subprojects** that are periodically or permanently integrated together to create a **Master Project**. Subprojects will often contain **Handoffs** between each other that are connected together in the Master Project. This structure allows **Microsoft Project** users to break down a large project and delegate its parts to the necessary people, giving authority and responsibility of those parts to different schedulers/teams.



What is a Subproject?

A **Subproject** is a part or section of the Program schedule that has been separated into its own file and can be combined with the other Subproject files to produce the **Master Project**. Breaking down a large program into smaller Subprojects can be a useful technique to manage the schedule. It allows **Microsoft Project** users to have ownership and work simultaneously on different parts of the larger Program Schedule and helps to prevent the project file from becoming bloated, slow, and hard to work with.

**What is a Master Project?**

Think of the **Master Project** as a collection of consolidated projects. These projects will often contain interdependencies (predecessor and successor relationships) or **Handoffs** between each other. Projects that are used to make up a Master Project Schedule are called **Subprojects**. The Master Project file is used to perform program level analysis, like time phased resource usage or critical/driving path analyses, and usually serves as the schedule deliverable if the program schedule must be submitted.

What is a Handoff?

For the most part Projects will have **Internal Dependencies**. These are predecessor and successor relationships between their own tasks and milestones. However, in a Multi-Project Environment, Subprojects may contain **External Dependencies**. These are predecessor and successor relationships that cross from tasks in one subproject to another. For brevity, we refer to these **External Dependencies** as **Handoffs** in this document. As an example, imagine two project schedules – Project A and Project B. Both projects are separate .mpp files. Project B has some tasks that are dependent on tasks from Project A and vice versa. These dependency points between the projects are **Handoffs**.

Here are some other common names for **Handoffs**:

- Cross-Project Dependencies
- External Dependencies
- Givers/Receivers
- Inputs/Outputs
- Makers/Takers

SSI refers to a predecessor in the handoff relationship will as a **Giver** and a successor in the handoff relationship as a **Receiver**.



Reasons to Breakdown a Project into Subprojects

To determine whether you should break up a large project into a master project and Subprojects, consider the following:

- **Several people can own and manage different parts of the program schedule at the same time**
 - Most of the time, a project file is owned, managed, and modified by a single project scheduler. But often a project schedule will become or has become so large that many schedulers are needed to manage it. Microsoft Project only allows a single user to access a file at a time. By breaking down a large program into Subprojects, different schedulers can own and operate different sections of the schedule simultaneously and independently.
- **Your project is very large and detailed** – If your project contains more than a few hundred tasks, breaking it into Subprojects can make it easier to manage. If some parts of the project contain work that is broken down into more detail than others, it may make sense to make those parts into Subprojects so that most users see only a rolled-up description of the Subproject, but interested parties can view it in more detail.
- **Your organization is not centralized** – In a decentralized or distributed organization, a Master Project and Subprojects give workers more control over their own work than one centralized project file does.
- **You work on multiple projects** – Project managers may have a set of projects they work on all the time, whether the projects are interrelated or not. Instead of opening them one by one, you can open all the Subprojects at the same time when you open the Master Project. This approach also makes it easier to generate reports on multiple projects. If the projects are interrelated, the project manager can create Handoffs, or task dependencies, between tasks in different projects. Creating Handoffs between projects makes it easier for different project schedulers to see how work by other project schedulers affects their schedules.
- **Your project has multiple stakeholders who care about different parts of the project** – When people want to look at different details, project managers can put all the project files on a server and customize different views for various stakeholders. The same project file can be used as a Subproject in different master projects to customize the information displayed.



Understanding the Handoff Method

SSI's Multi-Project Tools require teams to use the **Handoff Method** or **Giver/Receiver Method**. This is just a way of identifying and marking dependencies that exist between Subprojects so they can be easily connected and synchronized with each other.

The Handoff Method requires the use of a single text field and two flag fields in order to mark Handoffs in Microsoft Project. The text field is used to input **The Handoff Code**. **The Handoff Code** is a unique code or number used by both the **Giver(s)** and the **Receiver(s)** of the Handoff relationship. The unique code is usually determined and assigned by the receiving party. The flag fields are used to input which item(s) will be the giver(s) or predecessor(s), and which item(s) will be the receiver(s) or successor(s). Examine the following images:

Project UID ▾	Task Name ▾	Handoff Code ▾	Giver ▾	Receiver ▾
ProjA_1	Project A Requirements		No	No
ProjA_2	Write the Requirements Document		No	No
ProjA_3	Deliver Requirements Document	ProjA-001	Yes	No
ProjA_4	Write the Requirements Verification Document		No	No
ProjA_5	Deliver Verification Document	ProjA-002	Yes	No
ProjA_6	Receive Test Report from Project B	ProjB-001	No	Yes
ProjA_7	Review Test Report		No	No

Figure 1 – Project 'A' Example Handoffs

Project UID ▾	Task Name ▾	Handoff Code ▾	Giver ▾	Receiver ▾
ProjB_1	Project B (Product)		No	No
ProjB_3	Receive Requirements Document Published by Project A	ProjA-001	No	Yes
ProjB_4	Design Product Per Requirements		No	No
ProjB_5	Receive Requirements Verification Document Published by Project A	ProjA-002	No	Yes
ProjB_7	Test and Verify Product Performance		No	No
ProjB_8	Generate Test Report		No	No
ProjB_6	Deliver Test Report	ProjB-001	Yes	No

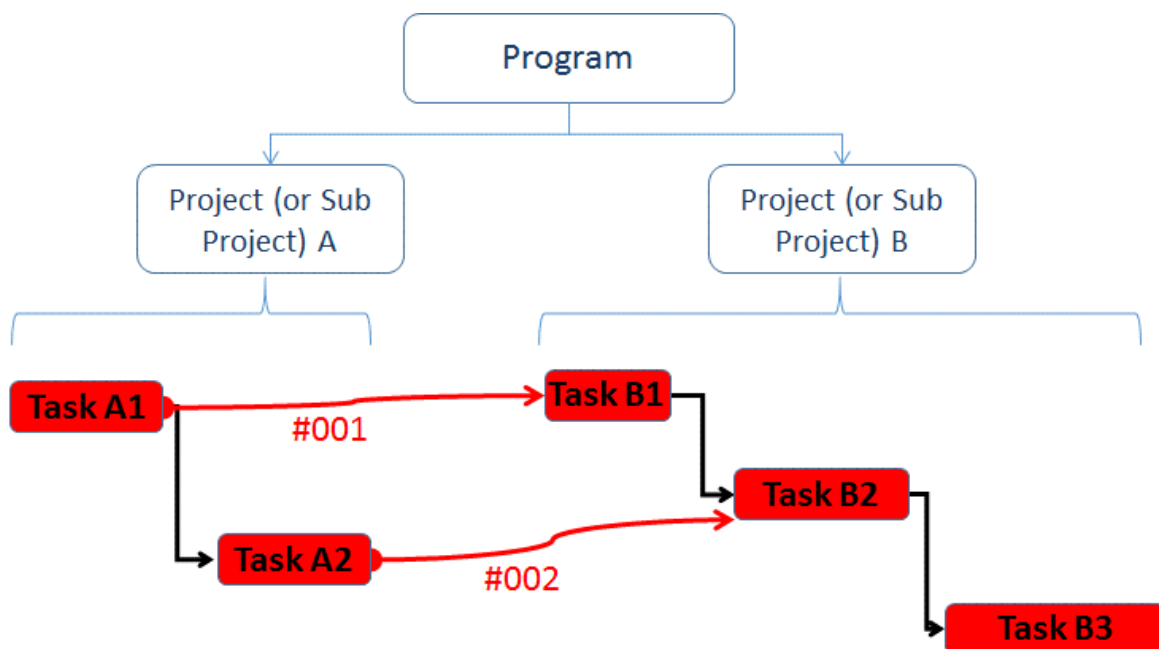
Figure 2 – Project 'B' Example Handoffs



In these two example projects the following **Handoffs** can be found:

- **ProjA-001**, which connects Unique ID "ProjA_2" as a predecessor to "ProjB_3"
- **ProjA-002**, which connects Unique ID "ProjA_5" as a predecessor to "ProjB_5"
- **ProjB-001**, which connects Unique ID "ProjB_6" as a predecessor to "ProjA_6"

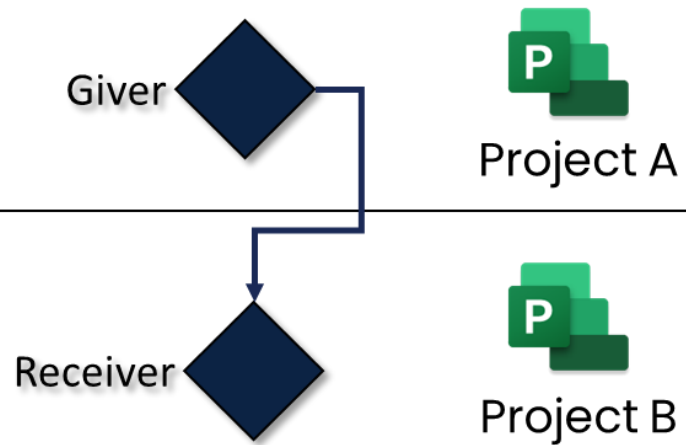
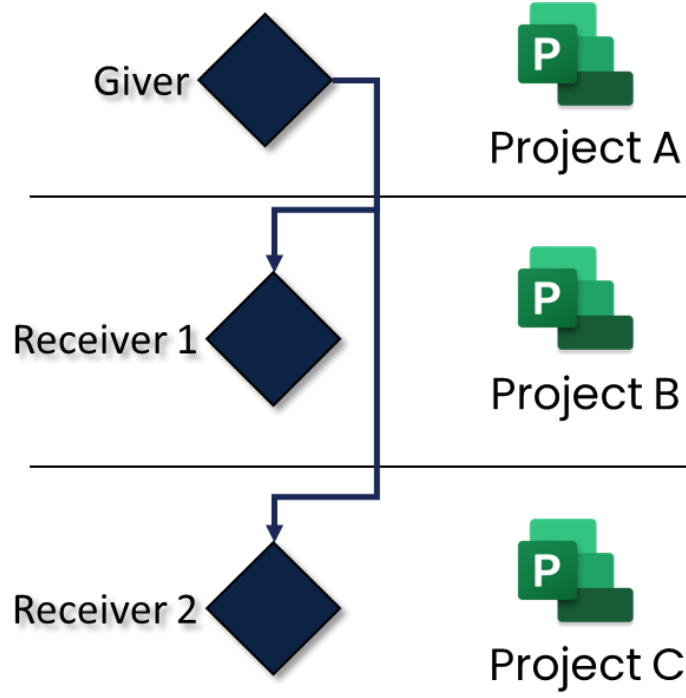
In these examples, the syntax for the Handoff Code is the abbreviated name of the project that is the predecessor in the Handoff relationship, followed a dash, and then by the number of the giver that is coming from that project. This convention is just our personal preference, but you may choose any convention you like, as long as your Handoff codes can connect givers and receivers together.





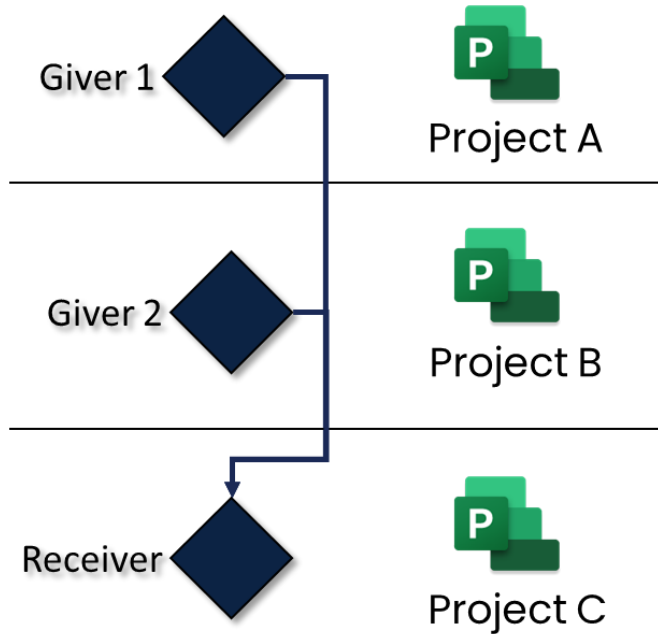
Types of Handoff Relationships

It is possible to have the following types of relationships between givers and receivers:

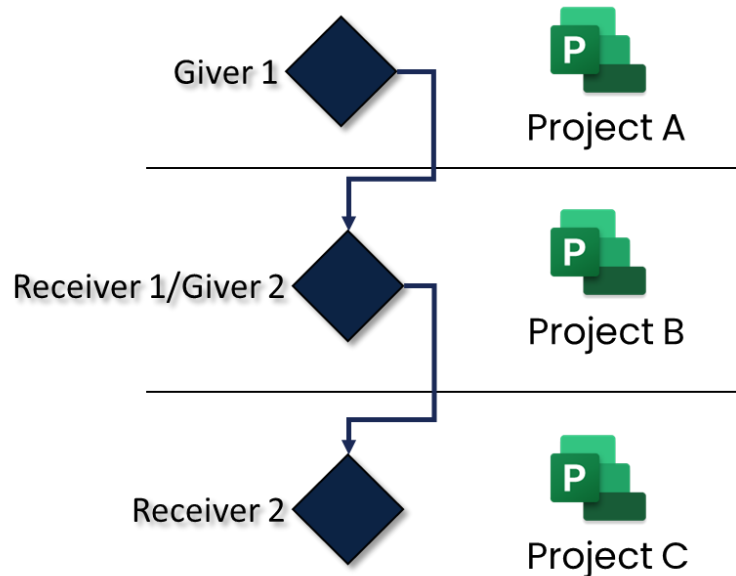
One-to-One – in this type of handoff relationship, there is a single giver that connects to a single receiver.	 The diagram illustrates a one-to-one handoff relationship. A single 'Giver' (represented by a dark blue diamond) is connected by a line to a single 'Receiver' (also a dark blue diamond). The Giver is associated with 'Project A' (indicated by a green 'P' icon), and the Receiver is associated with 'Project B' (indicated by a green 'P' icon'). A horizontal line separates the Giver/Project A section from the Receiver/Project B section.
One-to-Many – in this type of Handoff relationship, there is a single Giver that connects to many Receivers. The Receivers can all be within a single subproject or across multiple separate subprojects. In the example to the right, the Giver from <i>Project A</i> feeds the Receivers from <i>Project B</i> and <i>Project C</i>, but it could feed additional Receivers in either of those projects.	 The diagram illustrates a one-to-many handoff relationship. A single 'Giver' (dark blue diamond) is connected by a line that splits into two separate lines, each leading to a different 'Receiver' (dark blue diamond). The Giver is associated with 'Project A' (green 'P' icon). The first Receiver is labeled 'Receiver 1' and is associated with 'Project B' (green 'P' icon'). The second Receiver is labeled 'Receiver 2' and is associated with 'Project C' (green 'P' icon'). Horizontal lines separate the Giver/Project A section from the Receiver 1/Project B section, and the Receiver 2/Project C section.



Many-to-One - in this type of Handoff relationship, there are many Givers that connect to a single Receiver. There can be many Givers within a single subproject or across multiple separate subprojects. In the example to the right, the Givers from *Project A* and *Project B* feed the Receiver from *Project C*, but there could be more Givers in Project a or Project B that feed the same Receiver.



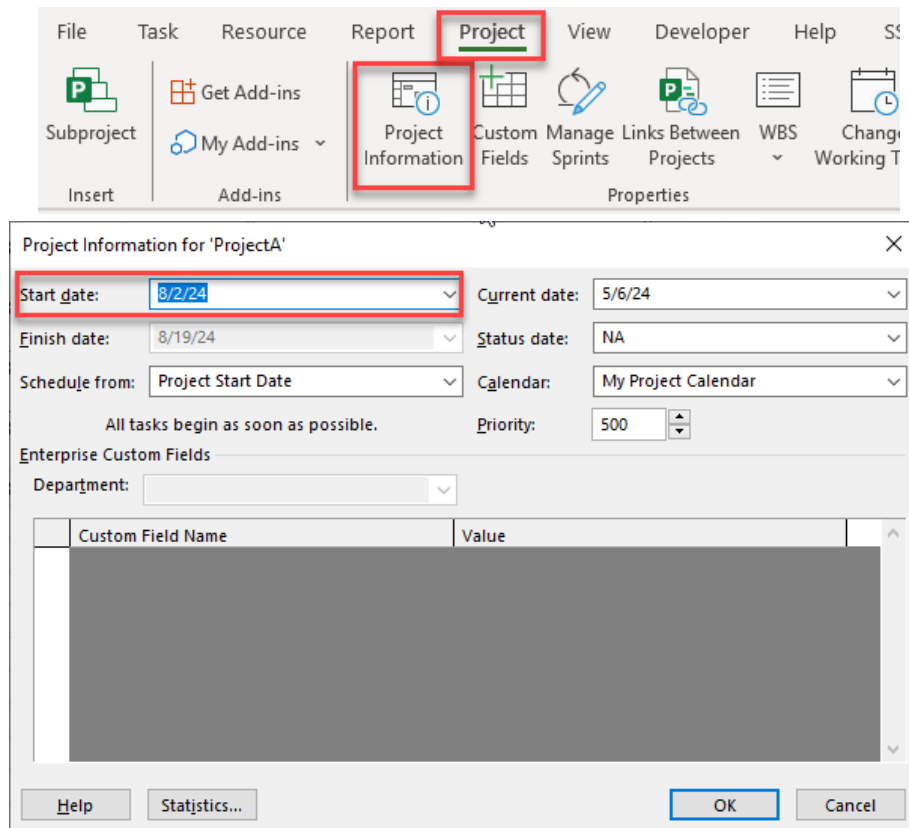
Giver/Receiver Combo - an item can be used as both a Giver and a Receiver at the same time, essentially creating a "pass through" item. In the example to the right, the milestone in *Project B* is both a Giver and a Receiver. It Receives Giver 1 from *Project A* and feeds *Receivers 2* in *Project C*.



Class Exercise – Code and Flag Handoffs

The solution to this exercise is located here: [Project A Solution](#) (ProjectA.mpp)

1. Open a blank project in Microsoft Project
2. Name Row 1 “Project A Requirements”. This will become the Top Level Summary task of the Project
3. Select **Project Information** from the **Project** ribbon and set the start date of the project to 08/02/2024.

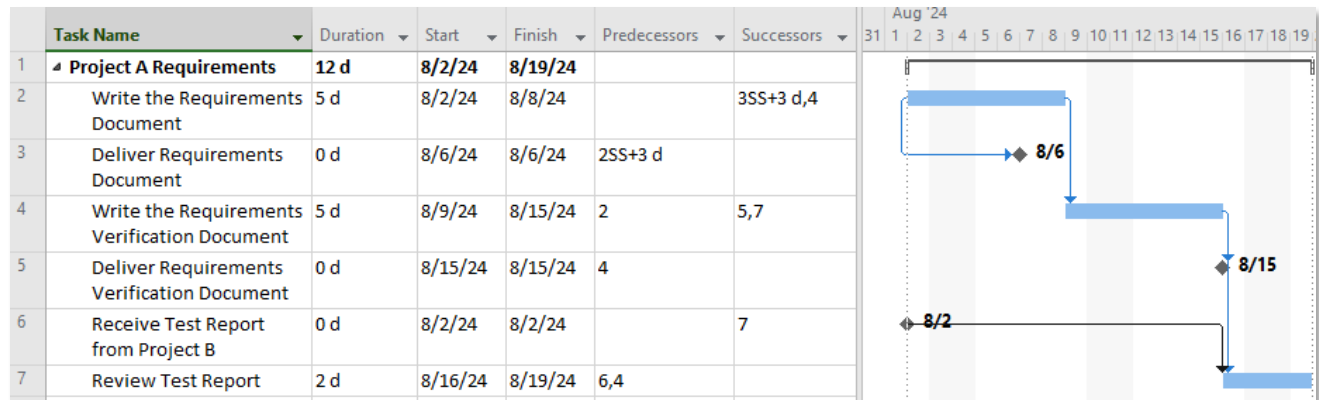


4. Add the following tasks:
 - Write the Requirements Document (5-day duration)
 - Deliver Requirements Document (0-day milestone)
 - Write the Requirements Verification Document (5-day duration)

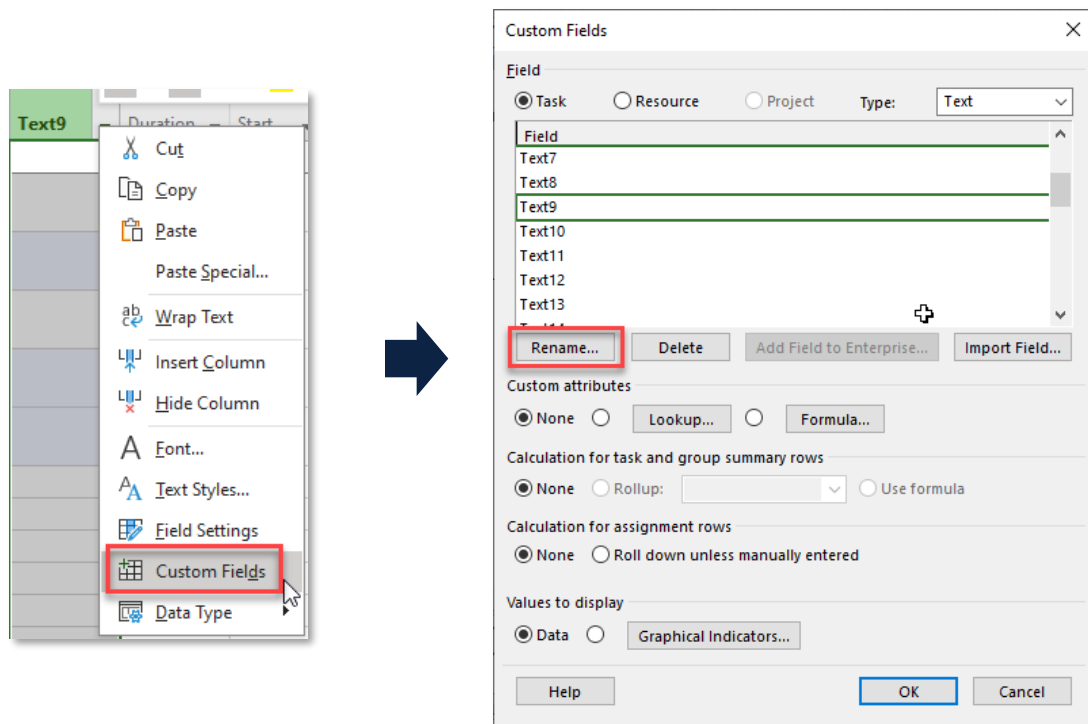


- Deliver Requirements Verification Document (0-day milestone)
- Receive Test Report from Project B (0-day milestone)
- Review Test Report (2-day duration)

Connect them like shown in the following image:



5. Add the columns named Text9, Flag8, Flag9 into the Gantt Table.
6. Rename these columns by right-clicking on each Column header and selecting **Custom Fields**. Using the **Custom Fields Dialog**, rename Text9 to "Handoff Code", rename Flag8 to "Receiver" and rename Flag9 to "Giver". Refer to these custom names in the steps below.



7. Enter "ProjA-001" in **Handoff Code** of the row named "**Deliver Requirements Document**" and mark it as a Giver by setting the **Giver Flag = Yes**
8. Enter "ProjA-002" in **Handoff Code** of row named "**Deliver Requirements Verification Document**" and mark it as a Giver by setting the **Giver Flag = Yes**.
9. Enter "ProjB-001" in **Handoff Code** of the row named "**Receive Test Report from Project B**" and mark it as a Receiver by setting the **Receiver Flag = Yes**



Task Name ▼	Handoff Code ▼	Giver ▼	Receiver ▼
▸ Project A Requirements		No	No
Write the Requirements Document		No	No
Deliver Requirements Document	ProjA-001	Yes	No
Write the Requirements Verification Document		No	No
Deliver Requirements Verification Document	ProjA-002	Yes	No
Receive Test Report from Project B	ProjB-001	No	Yes
Review Test Report		No	No

10. Save this project as **ProjectA.mpp**

End of Exercise

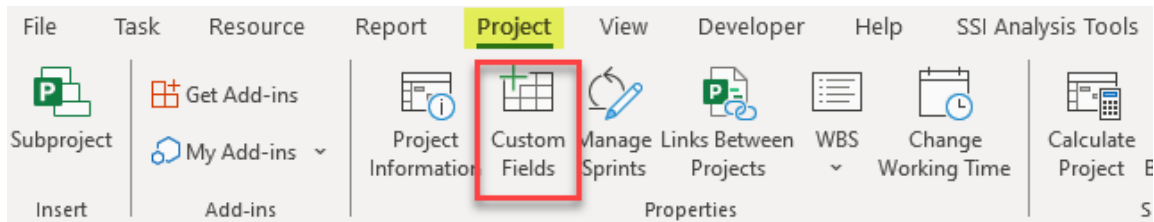


Setting up a Project Unique ID (PUID)

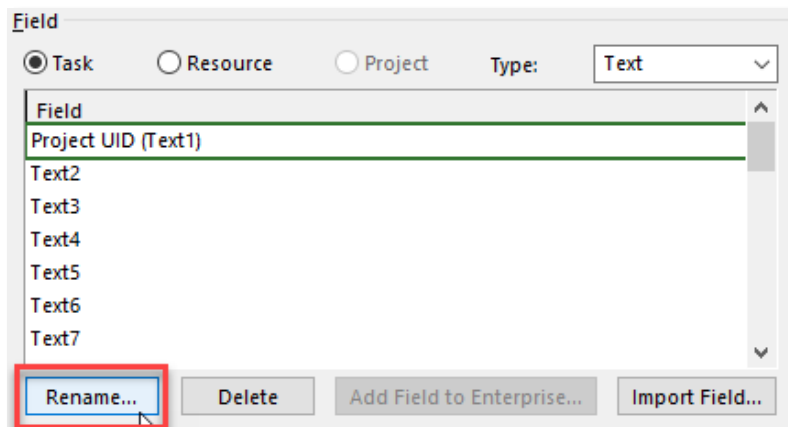
A **Project Unique ID** is an alternative Unique ID field that should be setup in Subproject files. The goal of this field is to always have a way to reference the "true" unique ID of a task within a Subproject, regardless if it has been integrated into a Master Project Schedule.

Class Exercise – Setting up a Project Unique ID Field

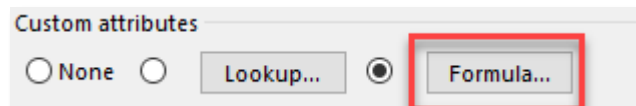
1. Open the Project A file you just created in the previous exercise.
2. Open the **Custom Fields Dialog** by clicking the **Project** Ribbon, then clicking the **Custom Fields** button.



3. In the **Custom Fields Dialog**, rename Text1 to "Project UID".



4. Set the Custom Attributes of the Field to utilize a formula





5. Set the formula to “ProjA_” & [Unique ID]

6. Set the Calculation for task and group summary rows to “Use Formula”

7. Click OK to close the Custom Fields Dialog.

End of Exercise



Using “Shell” Files

When working in a **Multi-Project Environment**, it can be helpful to create **Shell files** for the Subprojects and the Master Project. **Shell files** are blank MS Project files that only contain the common fields, calendars, and resources that will be utilized across all subproject files. The Shell files serve as the starting point for the creation of any new Subproject and any new Master Project integration.

Setting up the Subproject Shell

In an ideal scenario, the Subproject shell is the file that will be the starting point for all Subprojects before content is entered into them. Prior to setting this up, the schedule team should decide what custom fields are important to have common across all schedule files and which calendars are needed. In addition, the Subproject shell should always contain a Project Unique ID (PUID) field to ensure a unique identifier of any given task can be maintained when the Subproject is inserted into the Master Project. To create a Subproject Shell file, follow these steps:

1. Identify the common fields that will be necessary to have in each Subproject file. At a minimum, this will include two flag fields to identify givers and receivers, one text field for Giver/Receiver ID, and one Text field for PUID.
2. Open a blank Microsoft Project file.
3. Customize the task and resource fields to contain the common fields and formulas.
4. Save the shell file with a name indicating it is the shell file (ex: “Program XYZ Subprj Shell”)

Setting up the Master Project Shell file

The **Master Project Shell file** serves as the starting point anytime a schedule integration is performed. Depending on the program, the Master Project Shell file may or may not contain any of its own tasks. Either way, the Master Project Shell file must contain all the common elements between the Subproject files such as calendars, fields, and resources. To create a Master Project Shell file, follow these steps:

1. Open a blank Microsoft Project file and any of the Subproject files. (Important that the Subproject files must be setup first).
2. Use the organizer to move the common fields, calendars, views, groups, tables, etc. into the shell file.
3. Remove any formulas from fields that are specific to Subproject files. The values of these fields in the Subproject files will still be copied into the Master file even without the formula existing in the Master file.
4. Copy all resources being used in the Subproject files into the resource sheet of the shell file.
5. Set the start date of the shell file to be the earliest start date of the Subproject files.
6. Set all critical settings to be the desired values.

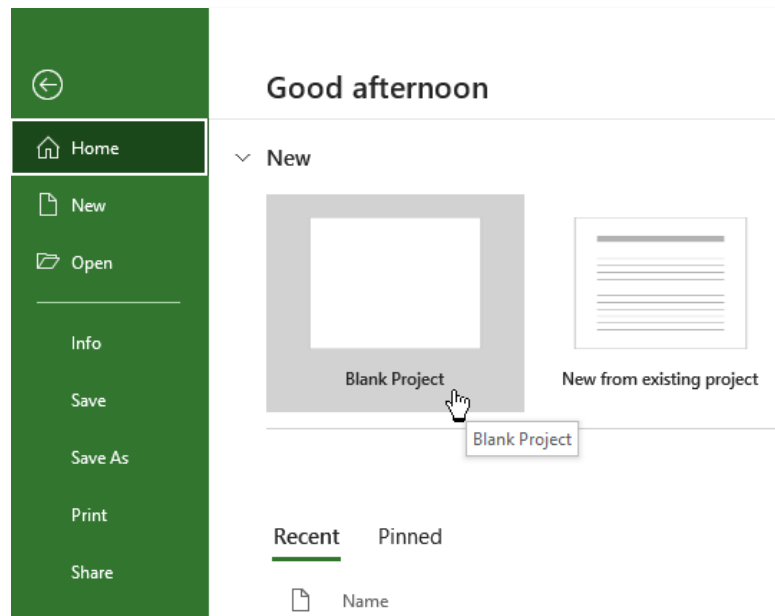


7. Save the shell file with a name indicating it is the shell file (example: "Program XYZ Shell")

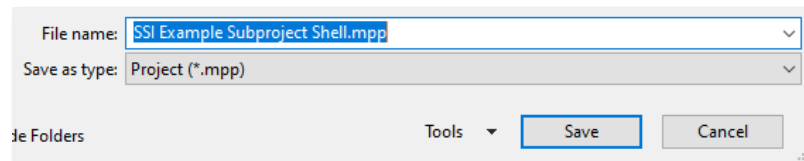
Class Exercise – Creating Shell files

The solution to this exercise is located [here](#).

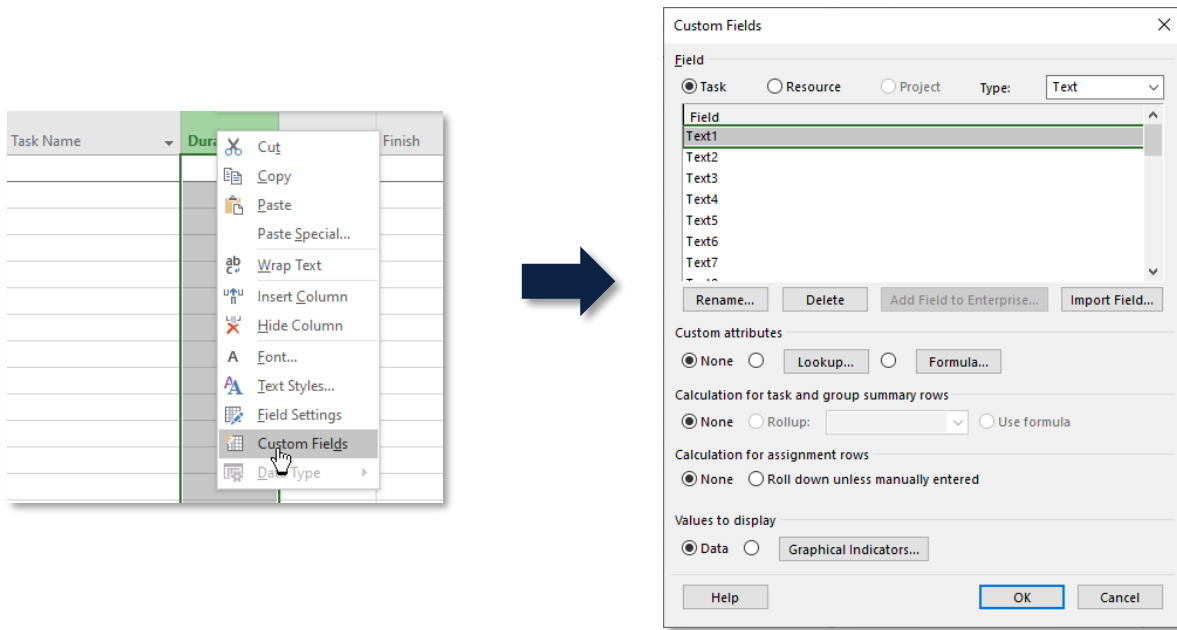
1. Open a blank *Microsoft Project* file



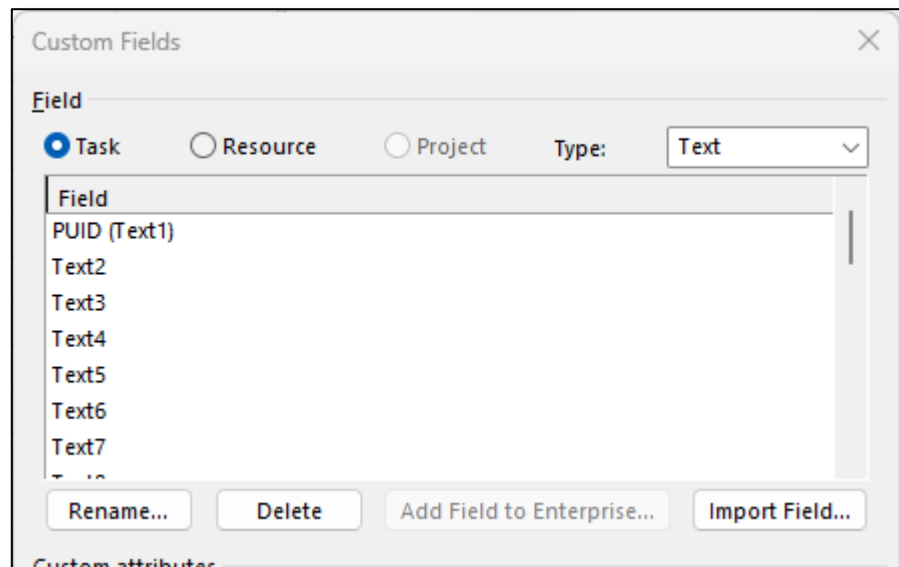
2. Save the file with the name "SSI Example Subproject Shell"



- Right click on any task field header and choose the **Custom Fields** option. This will open the Custom Fields window.



- Rename Flag8 to be called "Receiver"
- Rename Flag9 to be called "Giver"
- Rename Text9 to be called "Handoff Code" (it shows GR code in the image but just ignore that)
- Rename Text10 to be called "SSi Analysis"
- Rename Text1 to be called "PUID"





9. Add any custom calendars necessary into the project by using the *Organizer* or clicking the *Project* tab → *Change Working Time* and creating a custom calendar

The 'Change Working Time' dialog box is shown. It has a dropdown for 'For calendar:' set to 'My Class Calendar' and a 'Create New Calendar ...' button. Below is a legend with 'Working' (white), 'Nonworking' (grey), '31 Edited working hours', '31 Exception day', and '31 Nondefault work week'. A calendar grid for July 2020 is displayed, with a hand cursor over the 3rd. Text on the right says 'Click on a day to see its working times: July 03, 2020 is nonworking.' and 'Based on: Exception 'Holiday for Indep...' on calendar 'My Class Calendar'.' At the bottom, there are 'Exceptions' and 'Work Weeks' tabs. The 'Exceptions' table has columns 'Name', 'Start', and 'Finish', with one entry: '1 Holiday for Independence Day 2020' from '07/03/2020' to '07/03/2020'. Buttons for 'Details...' and 'Delete' are on the right.

10. Use the Project Information form to set a start date for the shell file.

The 'Project' tab in the ribbon is highlighted with a red box. Within it, the 'Project Information' button is also highlighted with a red box. Other buttons in the 'Project' tab include 'Custom Fields', 'Manage Sprints', 'Links Between Projects', 'WBS', and 'Change Working Time'.

The 'Project Information for 'ProjectA'' dialog box is shown. It has fields for 'Start date:' (8/2/24), 'Current date:' (5/6/24), 'Finish date:' (8/19/24), 'Status date:' (NA), 'Schedule from:' (Project Start Date), 'Calendar:' (My Project Calendar), and 'Priority:' (500). Below these is a section for 'Enterprise Custom Fields' with a 'Department:' dropdown and a table for 'Custom Field Name' and 'Value'. At the bottom are 'Help', 'Statistics...', 'OK', and 'Cancel' buttons.

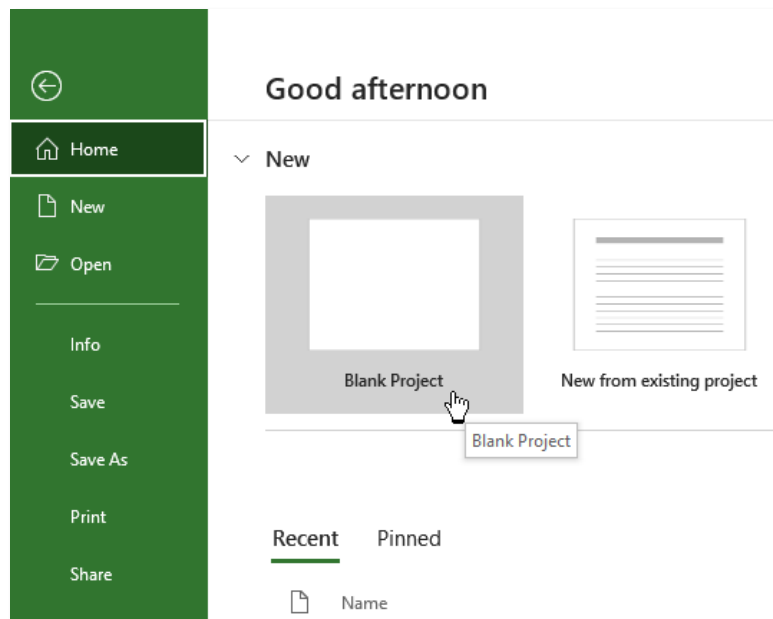


11. Add any resources that will be used by subproject files into the resource sheet.

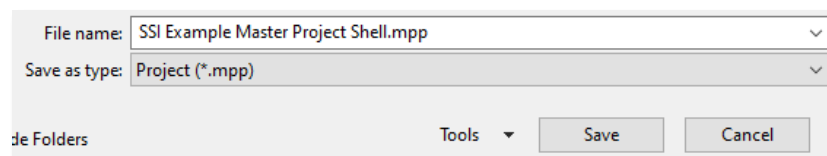
	Resource Name	Type	Initials	Group	Max.	Std. Rate	Accrue	Code
1	SW Engineer	Work	S	Eng	100%	\$75.00/hr	Prorated	SW
2	HW Engineer	Work	H	Eng	100%	\$75.00/hr	Prorated	HW
3	Systems Engineer	Work	S	Eng	100%	\$75.00/hr	Prorated	Sys
4	Program Manager	Work	P	PMO	100%	\$75.00/hr	Prorated	PM
5	PM Office	Work	P	PMO	100%	\$75.00/hr	Prorated	PM
6	Security Engineer	Work	S	Eng	100%	\$75.00/hr	Prorated	Sec
7	Production Lead	Work	P	Mfg	100%	\$75.00/hr	Prorated	Prod
8	Production Support	Work	P	Mfg	100%	\$75.00/hr	Prorated	Prod
9	Test Engineer	Work	T	Test	100%	\$75.00/hr	Prorated	Prod
10	Test Tech	Work	T	Test	100%	\$75.00/hr	Prorated	Prod
11	Material	Material	M	Cost		\$1.00	Prorated	Cost
12	Travel	Material	T	Cost		\$1.00	Prorated	Cost
13	ODC	Material	O	Cost		\$1.00	Prorated	Cost

12. Save the project file

13. Open another blank project file

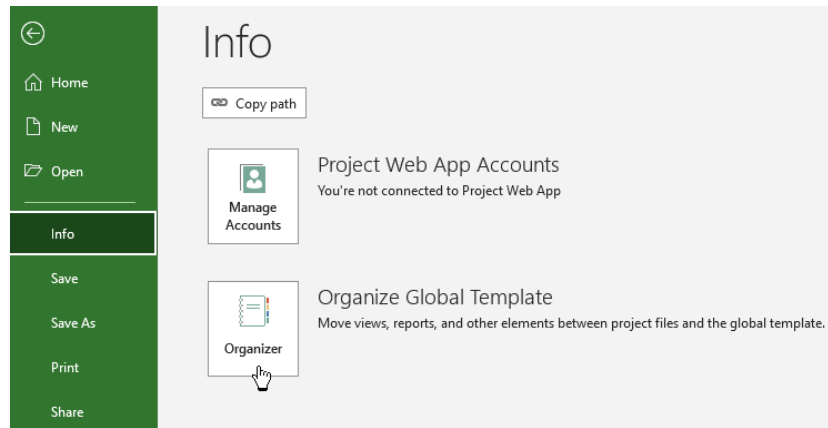


14. Save the file with the name "SSI Example Master Project Shell"

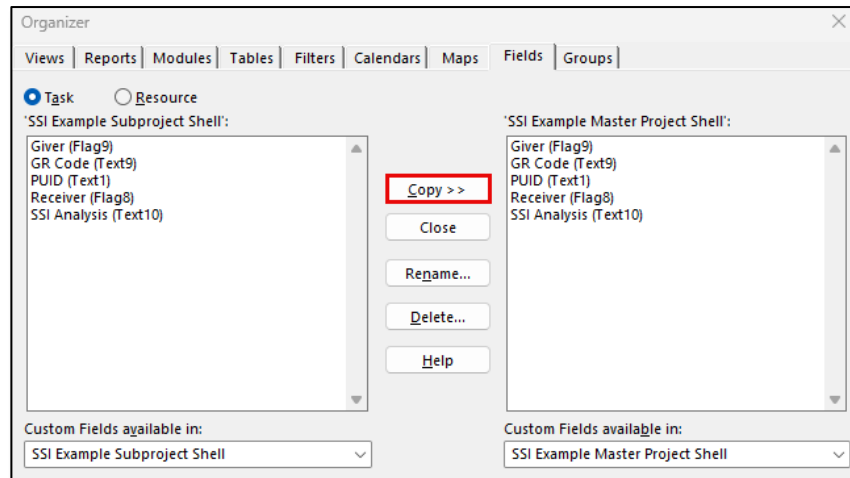




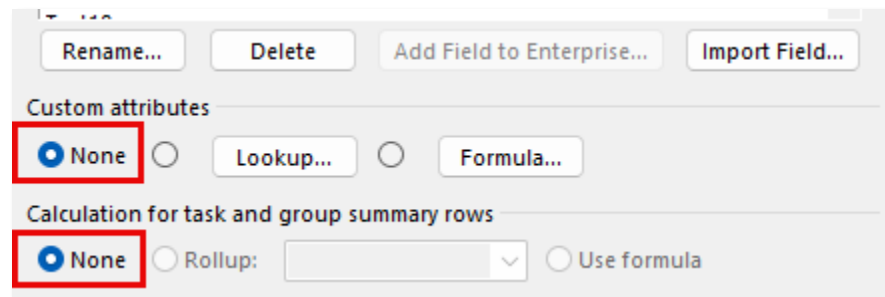
15. Open the Project Organizer by clicking *File* → *Info* → *Organizer*



16. Use the *Organizer* to copy over the fields and calendars from the Subproject Shell file to the Master Project shell file



17. Remove any formula from the PUID field and set the Custom Attributes to *None*



18. Use the Project Information form to set a start date for the project. This date should be whatever the earliest start date of any of the subproject files will be.



File Task Resource Report **Project** View Developer Help SSI

Subproject Get Add-ins My Add-ins Project Information Custom Fields Manage Links Between Projects WBS Change Working Time

Insert Add-ins Properties

Project Information for 'ProjectA'

Start date: 8/2/24 Current date: 5/6/24

Finish date: 8/19/24 Status date: NA

Schedule from: Project Start Date Calendar: My Project Calendar

All tasks begin as soon as possible. Priority: 500

Enterprise Custom Fields

Department:

Custom Field Name	Value
-------------------	-------

Help Statistics... OK Cancel

19. Add any resources that will be used by subproject files into the resource sheet.

	Resource Name	Type	Initials	Group	Max.	Std. Rate	Accrue	Code
1	SW Engineer	Work	S	Eng	100%	\$75.00/hr	Prorated	SW
2	HW Engineer	Work	H	Eng	100%	\$75.00/hr	Prorated	HW
3	Systems Engineer	Work	S	Eng	100%	\$75.00/hr	Prorated	Sys
4	Program Manager	Work	P	PMO	100%	\$75.00/hr	Prorated	PM
5	PM Office	Work	P	PMO	100%	\$75.00/hr	Prorated	PM
6	Security Engineer	Work	S	Eng	100%	\$75.00/hr	Prorated	Sec
7	Production Lead	Work	P	Mfg	100%	\$75.00/hr	Prorated	Prod
8	Production Support	Work	P	Mfg	100%	\$75.00/hr	Prorated	Prod
9	Test Engineer	Work	T	Test	100%	\$75.00/hr	Prorated	Prod
10	Test Tech	Work	T	Test	100%	\$75.00/hr	Prorated	Prod
11	Material	Material	M	Cost		\$1.00	Prorated	Cost
12	Travel	Material	T	Cost		\$1.00	Prorated	Cost
13	ODC	Material	O	Cost		\$1.00	Prorated	Cost

20. Save the project file

End of Exercise



Class Exercise – Setting up Project B

The solution to this exercise is located here: [Project B Solution](#) (ProjectB.mpp)

1. Open the Subproject Shell file you created in the previous exercise
2. Name Row 1 “Project B (Product)”. This will become the Top Level Summary task of the Project
3. Add the following tasks:
 - Receive Requirements Document From Project A (0-day milestone)
 - Design Product Per Requirements (4-day duration)
 - Receive Requirements Verification Document From Project A (0-day milestone)
 - Test and Verify Product Performance (5-day duration)
 - Generate Test Report (2-day duration)
 - Deliver Test Report (0-day milestone)

Connect them like shown in the following image:

	Task Name	Duration	Start	Finish	Predecessors	Successors
1	Project B (Product)	11 d	8/2/24	8/16/24		
2	Receive Requirements Document Published by Project A	0 d	8/2/24	8/2/24		3
3	Design Product Per Requirements	4 d	8/2/24	8/7/24	2	5
4	Receive Requirements Verification Document Published by Project A	0 d	8/2/24	8/2/24		5
5	Test and Verify Product Performance	5 d	8/8/24	8/14/24	4,3	6
6	Generate Test Report	2 d	8/15/24	8/16/24	5	7
7	Deliver Test Report	0 d	8/16/24	8/16/24	6	

4. Set the Custom Attributes of the PUID Field to utilize a formula.

Custom attributes

☐ None ☐ Lookup... ☒ Formula...

5. Set the formula to “ProjB_” & [Unique ID]



Formula for 'Project UID' ✕

Edit formula

Project UID =

ProjB_" & [Unique ID]

+ - * / & MOD \ ^ () = <> < > AND OR NOT

Insert: Field ▾ Function ▾ Import Formula...

Help OK Cancel

6. Add the columns named Text9, Flag8, Flag9 into the Gantt Table.
7. Enter "ProjA-001" in **Handoff Code** of the row named **"Receive Requirements Document from Project A"** and mark it as a Receiver by setting the **Receiver Flag = Yes**.
8. Enter "ProjA-002" in **Handoff Code** of row named **"Receive Requirements Verification Document from Project A"** and mark it as a Receiver by setting the **Receiver Flag = Yes**.
9. Enter "ProjB-001" in **Handoff Code** of the row named **"Deliver Test Report"** and mark it as an Giver by setting the **Giver Flag = Yes**

Task Name		Handoff Code	Giver	Receiver	Duration	Aug '24																
						1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	Project B (Product)		No	No	11 d																	
2	Receive Requirements Document From Project A	ProjA-001	No	Yes	0 d																	
3	Design Product Per Requirements		No	No	4 d																	
4	Receive Requirements Verification Document From Project A	ProjA-002	No	Yes	0 d																	
5	Test and Verify Product Performance		No	No	5 d																	
6	Generate Test Report		No	No	2 d																	
7	Deliver Test Report	ProjB-001	Yes	No	0 d																	

10. Save this project as **ProjectB.mpp**

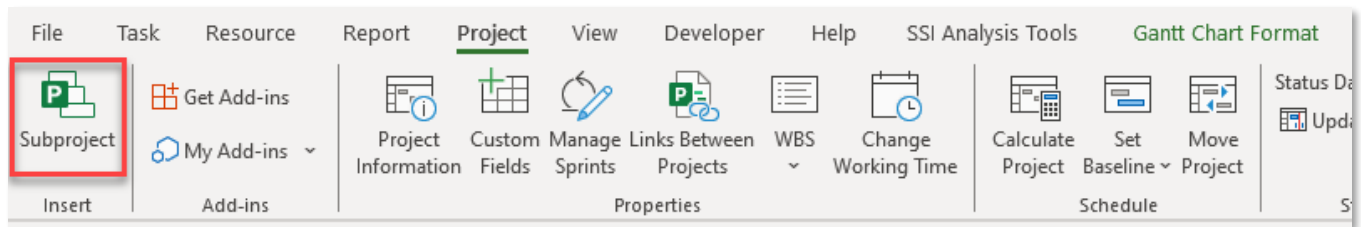
End of Exercise



Integrating Subprojects into the Master Project

At some point(s) all of the Subproject files will need to be integrated together, and their Handoffs will need to be connected in order to form the Master Project Schedule.

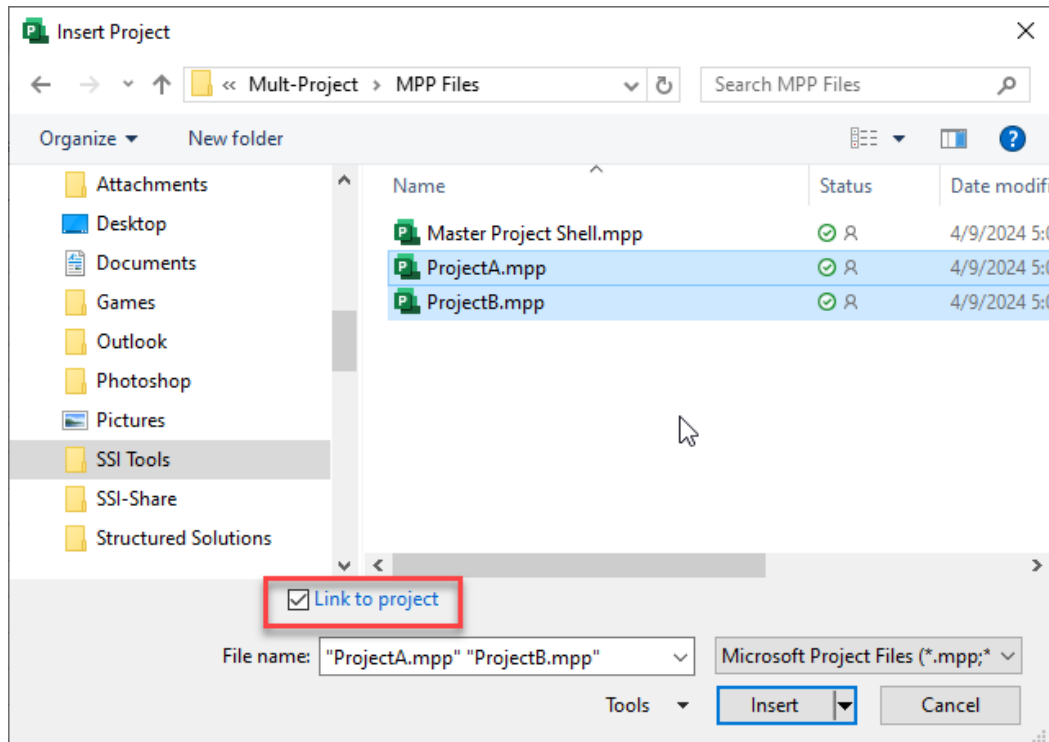
In Microsoft Project, The Subproject button in the Project ribbon is a built in tool to insert Subproject files into a consolidated, or Master Project Schedule.



The Subproject button gives users two methods of inserting one or more projects into another, "Linked" and "Unlinked". Before inserting Subprojects, it's necessary to understand the two different methods.

"Linked" Subprojects and Master Project

Subproject files can be inserted into a master project dynamically "linked" to its source file. To insert "linked" Subproject files into a Master Project file, leave the "Link to project" option checked when selecting files from the Subproject button:



When you insert a "linked" Subproject file into the Master Project, you are really just using the Master Project to view the contents of the Subproject. Any changes that are made to tasks from the Subproject in the Master Project will automatically be reflected in the Subproject file and vice versa.

While this may sound nice on paper, in practice this method is known to cause many issues. Here are a few:

- **The Master Project and Subprojects are very susceptible to file corruption.** If you ever rename, move, overwrite or save the files on a WAN (Wide Area Network) you roll the file corruption dice. It's a matter of when not if one or more files corrupt.
- "Linked" Subproject and master project files are much slower (take longer to calculate) and cumbersome to work in than unlinked files.



- When delivering or moving an Master Project, you must also deliver/move the collection of Subproject files too.
- The name and location of source and master files is significant. If a source Subproject file name is changed or its file is no longer accessible, its data in the Master Project will be incorrect or not available.
- You cannot create revisions of Subproject files.
- Linking a Predecessor from one Subproject to a Successor in another creates an External task in each Subproject. These links can make the project file unstable. If the data supporting either end of this Subproject-to-Subproject dependency becomes inaccessible, the form named Links Between Projects displays warnings concerning the source of the dependency.

	External Task	Project	Task Name	
1	No	Project 2	Project 2	
2	Yes	C:\Users\Arnold\Desktop\Demonst - Copy\Project 1.mpp	Deliver Document	
3	No	Project 2	Receive Project 1's Document	
4	No	Project 2	Review Project 1's Document	

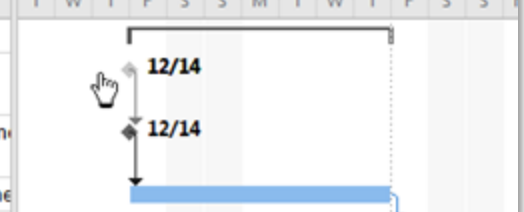


Figure 3 - External "Ghost" Task

Task Name	Duration	Start	Finish	Predecessors
Project B (Product)	17 days	8/4/21	8/27/21	
Deliver Requirements Document	0 days	8/4/21	8/4/21	
Deliver Requirements Document	0 days	8/4/21	8/4/21	
Deliver Requirements Document	0 days	8/4/21	8/4/21	
Receive Requirements Document Published by Project A	0 days	8/4/21	8/4/21	D:\OneDrive\SSI Tools\Training Materials\Example Files\Mult-Project\01-18-2022\linked\ProjectA.mpp\3,D:\OneDrive\SSI Tools\Training Materials\Example Files\Mult-Project\01-18-2022\linked\ProjectA r2.mpp\3,D:\OneDrive\SSI Tools\Trainings
Design Product Per Requirements	4 days	8/5/21	8/10/21	5
Deliver Verification Document	0 days	8/13/21	8/13/21	
Deliver Verification Document	0 days	8/13/21	8/13/21	
Deliver Verification Document	0 days	8/20/21	8/20/21	
Receive Requirements Verification Document Published by Project A	0 days	8/20/21	8/20/21	D:\OneDrive\SSI Tools\Training Materials\Example Files\Mult-Project\01-18-2022\linked\ProjectA.mpp\6,D:\OneDrive\SSI Tools\Training Materials\Example Files\Mult-Project\01-18-2022\linked\ProjectA r2.mpp\6,D:\OneDrive\SSI Tools\Trainings
Test and Verify Product Performance	5 days	8/23/21	8/27/21	10,6
Deliver Test Report	0 days	8/27/21	8/27/21	11
Receive Test Report from Project B	0 days	8/27/21	8/27/21	12
Receive Test Report from Project B	0 days	8/27/21	8/27/21	12
Receive Test Report from Project B	0 days	8/27/21	8/27/21	12

Figure 4 - External Dependencies

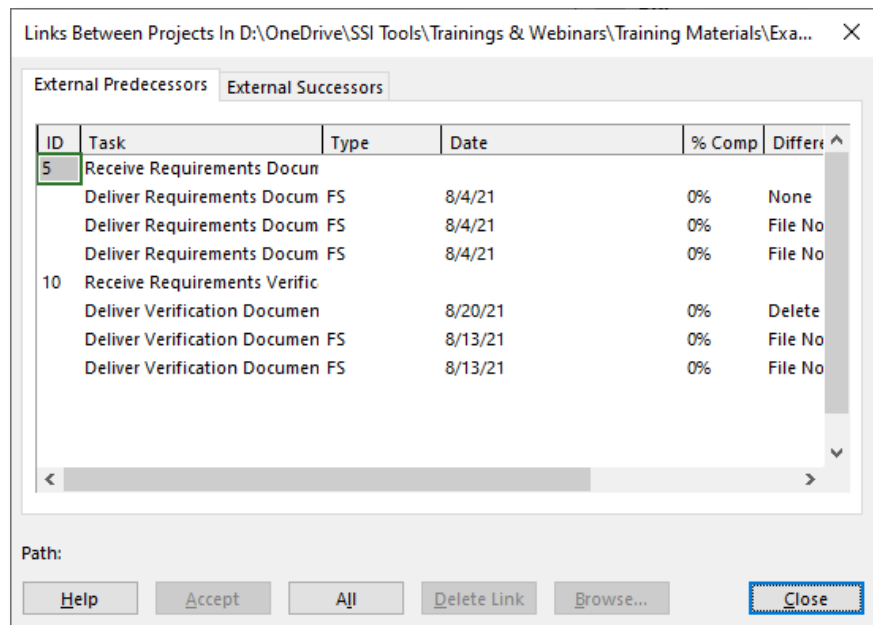
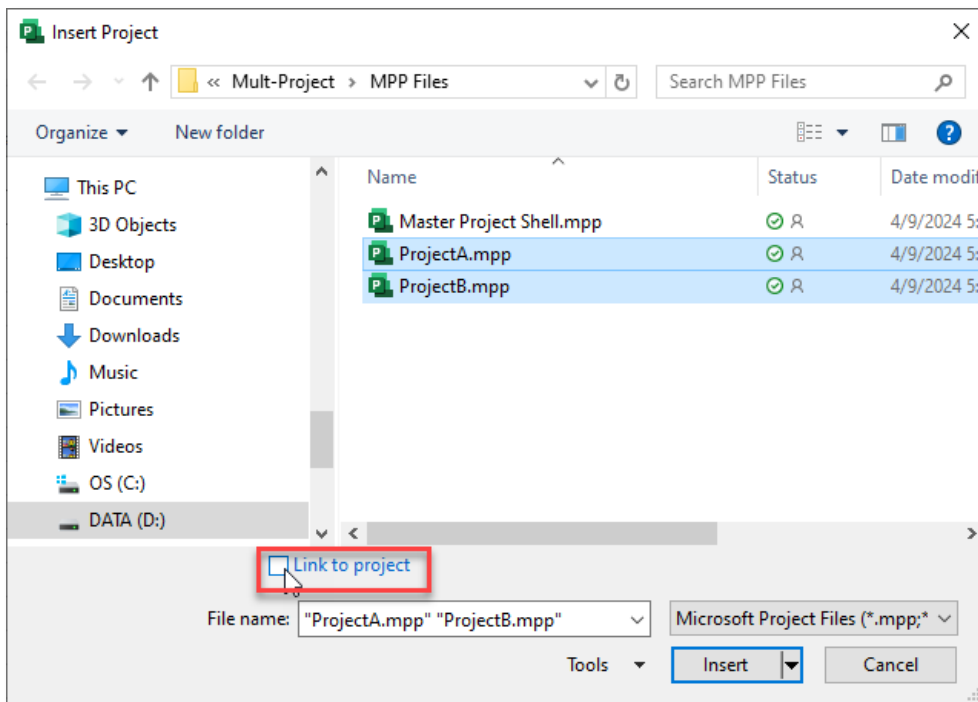


Figure 5 - External Dependencies Error Dialog

“Unlinked” Subprojects and Master Project

Subproject files can be inserted into a master project “unlinked” from its source file. To insert “unlinked” Subproject files into a master project file, turn the “Link to project” option off when selecting files from the Subproject button:



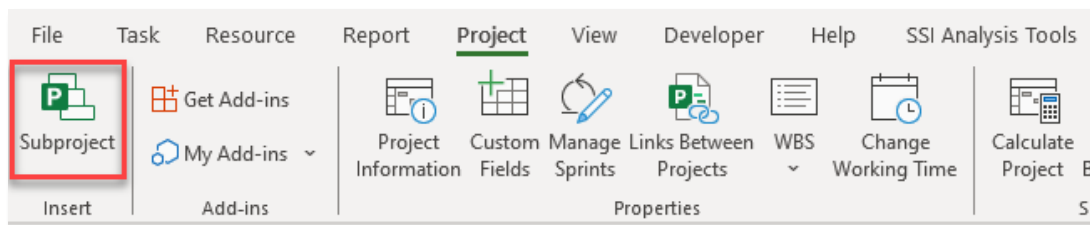
When you insert an “unlinked” Subproject file into the Master Project, you are really creating a copy of the contents of the Subproject within the master project. The downside to this method is that any changes that are made to tasks from the Subproject in the Master Project will not automatically be reflected in the Subproject file and vice versa. Instead, the master project and Subprojects act as their own singular Microsoft Project files with no connection to each other. The benefit of this method is that it avoids all the problematic issues from “linked” Subprojects and master projects.

SSI's Multi-Project Tools are designed make it easy to integrated unlinked Subprojects into a Master Project shell, and automatically synchronize dates that are changed within the Subprojects as a result of connecting the all the **Handoffs** in the Master Project.

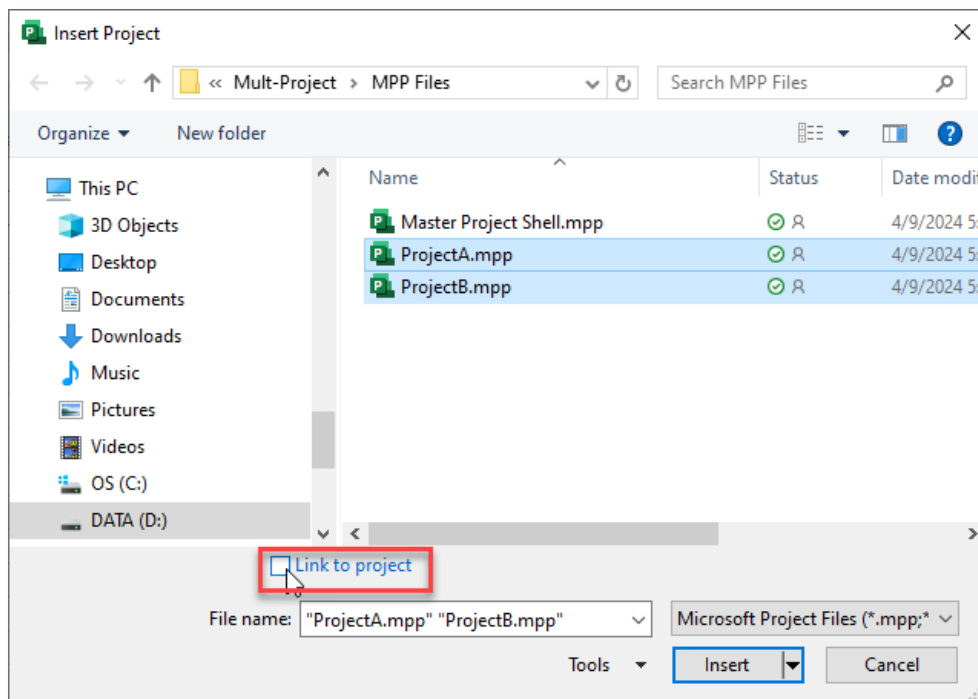
Class Exercise – Insert Subprojects into the Master Project Shell

The solution to this exercise is located [here](#).

1. Open the Master Project Shell file
2. Click on row 1 of the Master Project Shell
3. On the Project ribbon, click the **Subproject** button.

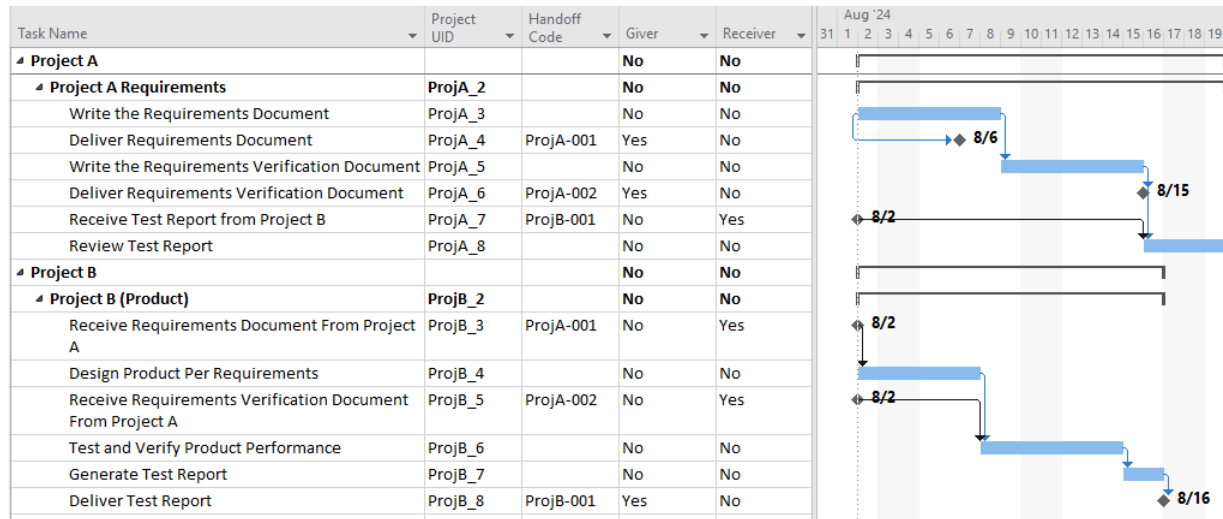


4. In the **Insert Project Dialog**, browse to the Project A and Project B files created in the previous exercises. Ensure you uncheck the **Link to project** option, then click the **Insert** button.





5. Expand both Project Summaries



Click **File** → **Save As** to save the project file. Give it the title **“Project AB Unlinked”**

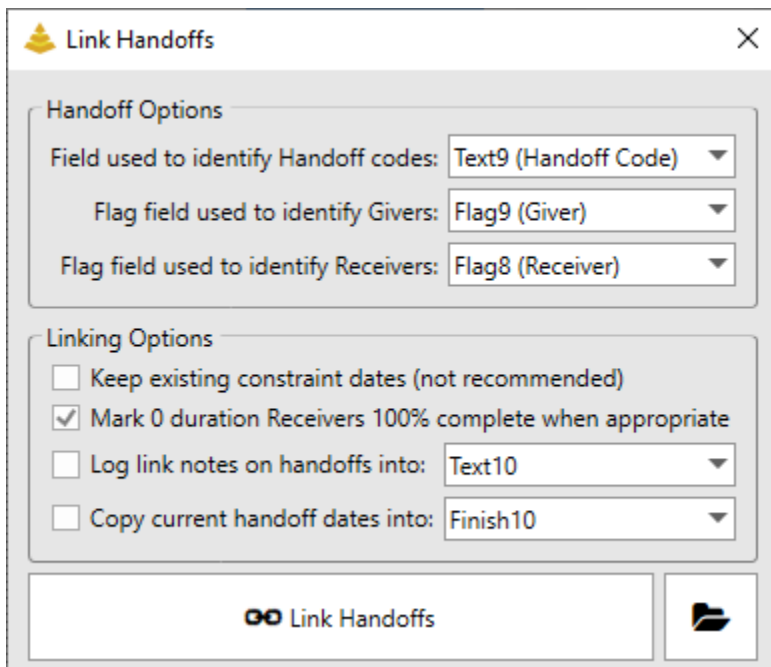
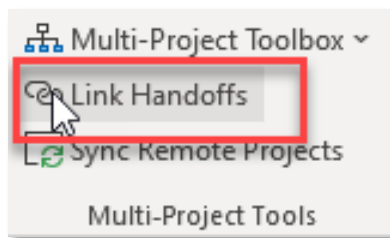
End of Exercise



Using the Handoff Linker Tool

The Handoff Linker Tool allows you to automatically link Handoffs from Subprojects within the a Master Project file. In other words, this tool can be used to create dependencies (predecessor and successor relationships) between Subproject files once they have all been integrated within a Master Project.

To open this tool, click the **Link Handoffs** button in the Multi-Project Tools group of the *SSI Analysis Tools* ribbon.

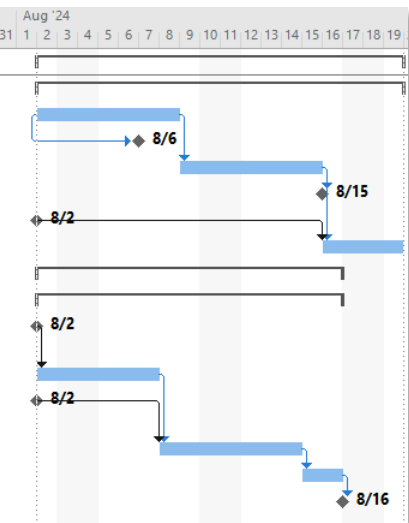


For detailed information on each of the options on the Handoff Linker dialog, see [this page](#).

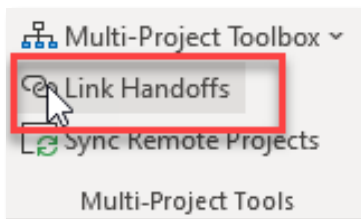
Class Exercise – Using the Handoff Linker Tool

1. Open the Project AB Unlinked file from the previous exercise

Task Name	Project UID	Handoff Code	Giver	Receiver	Aug '24
Project A			No	No	
Project A Requirements	ProjA_2		No	No	
Write the Requirements Document	ProjA_3		No	No	
Deliver Requirements Document	ProjA_4	ProjA-001	Yes	No	
Write the Requirements Verification Document	ProjA_5		No	No	
Deliver Requirements Verification Document	ProjA_6	ProjA-002	Yes	No	
Receive Test Report from Project B	ProjA_7	ProjB-001	No	Yes	
Review Test Report	ProjA_8		No	No	
Project B			No	No	
Project B (Product)	ProjB_2		No	No	
Receive Requirements Document From Project A	ProjB_3	ProjA-001	No	Yes	
Design Product Per Requirements	ProjB_4		No	No	
Receive Requirements Verification Document From Project A	ProjB_5	ProjA-002	No	Yes	
Test and Verify Product Performance	ProjB_6		No	No	
Generate Test Report	ProjB_7		No	No	
Deliver Test Report	ProjB_8	ProjB-001	Yes	No	



2. the **Handoff Linker Tool** by clicking the **Link Handoffs** button in the Multi-Project Tools group of the SSI Analysis Tools ribbon.



3. Ensure the **Field used to identify Handoff Codes** is set to Text9, the **Flag field used to identify Givers** is set to Flag9, and the **Flag field use to identify Receivers** is set to Flag8.

Handoff Options

Field used to identify Handoff codes: Text9 (Handoff Code)

Flag field used to identify Givers: Flag9 (Giver)

Flag field used to identify Receivers: Flag8 (Receiver)



4. Click the **Link Handoffs** button on the bottom of the Handoff Linker form

5. Click **Yes** to the message box that asks if you would like to export the **Link Log**



6. Review all Handoffs have been connected in the Master Project.

Task Name	Project UID	Handoff Code	Giver	Receiver	Aug '24
Project A			No	No	31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
Project A Requirements	ProjA_2		No	No	
Write the Requirements Document	ProjA_3		No	No	
Deliver Requirements Document	ProjA_4	ProjA-001	Yes	No	
Write the Requirements Verification Document	ProjA_5		No	No	
Deliver Requirements Verification Document	ProjA_6	ProjA-002	Yes	No	
Receive Test Report from Project B	ProjA_7	ProjB-001	No	Yes	
Review Test Report	ProjA_8		No	No	
Project B			No	No	
Project B (Product)	ProjB_2		No	No	
Receive Requirements Document From Project A	ProjB_3	ProjA-001	No	Yes	
Design Product Per Requirements	ProjB_4		No	No	
Receive Requirements Verification Document From Project A	ProjB_5	ProjA-002	No	Yes	
Test and Verify Product Performance	ProjB_6		No	No	
Generate Test Report	ProjB_7		No	No	
Deliver Test Report	ProjB_8	ProjB-001	Yes	No	

The Gantt chart displays the project schedule for August 2024. It shows the duration of various tasks and their dependencies. Key dates marked on the timeline are 8/6, 8/15, and 8/26. The chart uses blue bars to represent task durations and arrows to show the flow of dependencies between tasks.

7. Save the project.

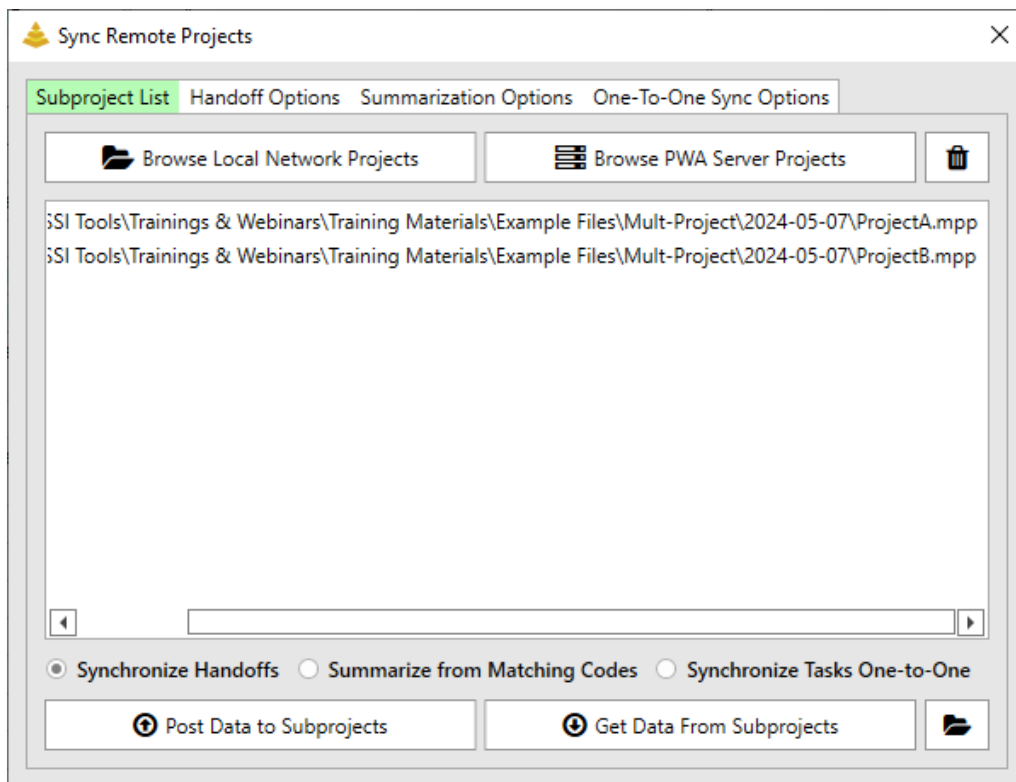
End of Exercise

Synchronizing Subprojects

Integrating subprojects together into the Master Project using the **“Unlinked” method** means when Handoffs are connected, Receivers in Subproject files are not automatically updated to reflect their true dates. Therefore, we need some way to synchronize the dates of Handoffs in our Subproject files in order for the **“Unlinked” method** to be a viable option for integrating Subprojects together.

The **Sync Remote Projects** tool allows users to automatically synchronize Handoff dates between Subprojects or the Master Project by either **Getting** data from those projects or **Posting** data to those projects.

The tool also supports users working in a **Project Server/Project Online environment**. connecting to your **Project Web App (PWA)**, the tool can **Get** and **Post** data from enterprise projects saved on PWA, as well as projects saved on a local network.

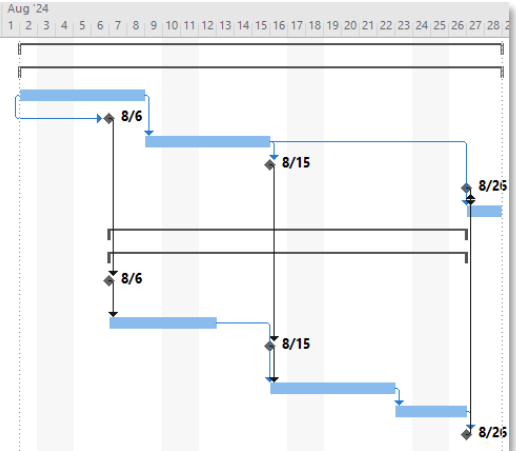


This tool even has the ability to **summarize** groups of similar tasks and synchronize tasks one-to-one through a **Project Unique ID**, but those features will not be covered in this document.

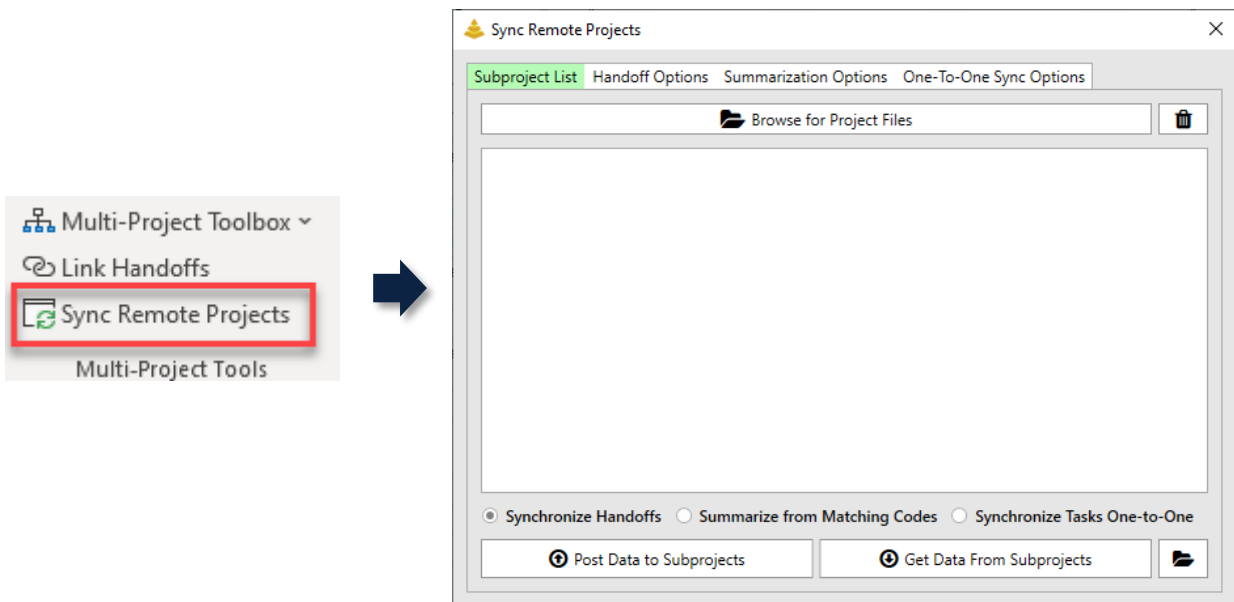
Class Exercise – Using the Sync Remote Projects Tool to Post Data to Subprojects

- Open the integrated **Project AB Unlinked** from the previous exercise

Task Name	Project UID	Handoff Code	Giver	Receiver	Aug '24
Project A			No	No	
Project A Requirements	ProjA_2		No	No	
Write the Requirements Document	ProjA_3		No	No	
Deliver Requirements Document	ProjA_4	ProjA-001	Yes	No	
Write the Requirements Verification Document	ProjA_5		No	No	
Deliver Requirements Verification Document	ProjA_6	ProjA-002	Yes	No	
Receive Test Report from Project B	ProjA_7	ProjB-001	No	Yes	
Review Test Report	ProjA_8		No	No	
Project B			No	No	
Project B (Product)	ProjB_2		No	No	
Receive Requirements Document From Project A	ProjB_3	ProjA-001	No	Yes	
Design Product Per Requirements	ProjB_4		No	No	
Receive Requirements Verification Document From Project A	ProjB_5	ProjA-002	No	Yes	
Test and Verify Product Performance	ProjB_6		No	No	
Generate Test Report	ProjB_7		No	No	
Deliver Test Report	ProjB_8	ProjB-001	Yes	No	



- Open the **Sync Remote Projects** tool from the *Multi-Project Tools* group in the *SSI Analysis Tools* ribbon





3. Navigate to the **Handoff Options** tab on the Sync Remote Projects form

The screenshot shows the 'Sync Remote Projects' window with the 'Handoff Options' tab highlighted in green. The 'Handoff Identification' section contains three dropdown menus: 'Handoff code field' set to 'Text9 (Handoff Code)', 'Giver flag field' set to 'Flag9 (Giver)', and 'Receiver flag field' set to 'Flag8 (Receiver)'. The 'Posting Data to Remote Subprojects' section has three checkboxes, with the first one checked and its dropdown set to 'Start'. The 'Getting Data From Remote Subprojects' section has three checkboxes, with the first one checked and its dropdown set to 'Start'.

4. Ensure the **Handoff code field** is set to Text9, the **Giver flag field** is set to Flag9, and the **Receiver flag field** is set to Flag8.

This close-up shows the 'Handoff Identification' section with the following settings: 'Handoff code field' is 'Text9 (Handoff Code)', 'Giver flag field' is 'Flag9 (Giver)', and 'Receiver flag field' is 'Flag8 (Receiver)'.

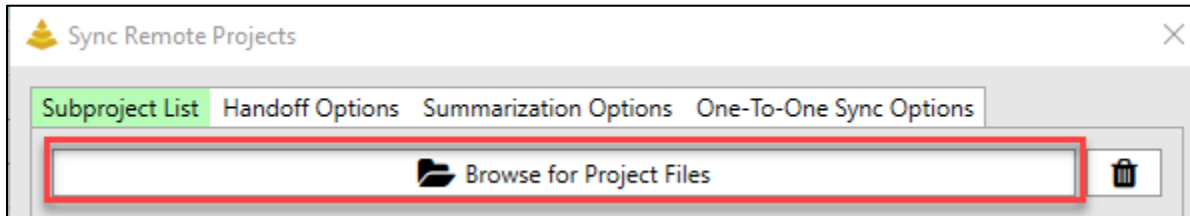
5. Ensure the option to **Post finish dates of Givers to matching remote Receivers** is checked and the field is set to **Start**

This close-up shows the 'Posting Data to Remote Subprojects' section with the checkbox 'Post finish dates of Givers to matching remote Receivers in:' checked and the dropdown menu set to 'Start'.

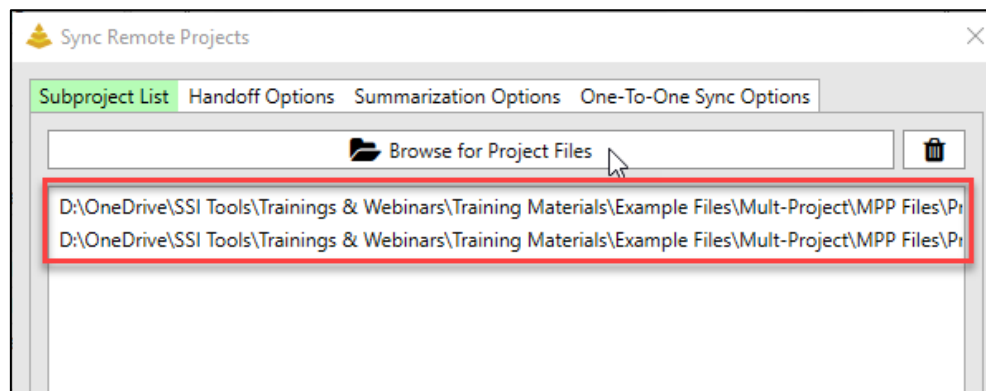
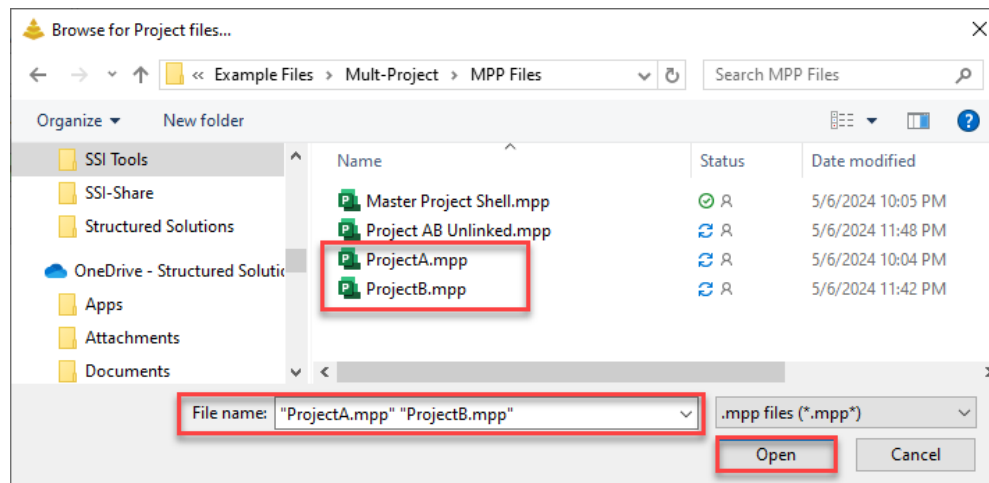
6. Navigate back to the **Subproject List** tab

The screenshot shows the tab bar at the top of the window with 'Subproject List' highlighted in green, and other tabs including 'Handoff Options', 'Summarization Options', and 'One-To-One Sync Options'.

7. Click the **Browse for Project Files** button



8. Using the File Explorer that opens, Browse for **Project A** and **Project B** that were created in the previous exercises, then click the **Open** button (Note: the file paths you see will be different from the images)





9. Ensure the **Synchronize Handoffs** option is selected.

The screenshot shows a control panel with three radio buttons: 'Synchronize Handoffs' (selected), 'Summarize from Matching Codes', and 'Synchronize Tasks One-to-One'. Below the buttons are two main action buttons: 'Post Data to Subprojects' and 'Get Data From Subprojects'. A red box highlights the 'Synchronize Handoffs' radio button.

10. Click the **Post Data to Subprojects** button.

This screenshot is similar to the previous one, but the 'Post Data to Subprojects' button is now highlighted with a red box, indicating it should be clicked.

11. Click **Yes** to the message box that appears asking if you would like to download the log

A message box titled 'SSI Tools' is displayed. It contains a question mark icon and the text: 'Data has been posted to remote subprojects. Would you like to download the log?'. At the bottom, there are two buttons: 'Yes' (highlighted with a red box) and 'No'.

12. Inspect Project A and Project B and ensure that the start dates of the Receivers match up with the finish dates of the Givers in the Master Project.

Task Name	Handoff Code	Giver	Receiver	Start	Finish	Aug '24																													
						31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
Project A Requirements		No	No	8/2/24	8/28/24																														
Write the Requirements Document		No	No	8/2/24	8/8/24																														
Deliver Requirements Document	ProjA-001	Yes	No	8/6/24	8/6/24																														
Write the Requirements Verification Document		No	No	8/9/24	8/15/24																														
Deliver Requirements Verification Document	ProjA-002	Yes	No	8/15/24	8/15/24																														
Receive Test Report from Project B	ProjB-001	No	Yes	8/26/24	8/26/24																														
Review Test Report		No	No	8/27/24	8/28/24																														

The Gantt chart visualizes the tasks from the table. It shows a sequence of tasks with arrows indicating dependencies. Key dates marked on the chart include 8/6, 8/15, and 8/26. The tasks are color-coded: blue for 'Project A' tasks and orange for 'Project B' tasks.



Task Name	Handoff Code	Giver	Receiver	Start	Finish	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
Project B (Product)		No	No	8/6/24	8/26/24																								
Receive Requirements Document From Project A	ProjA-001	No	Yes	8/6/24	8/6/24																								
Design Product Per Requirements		No	No	8/7/24	8/12/24																								
Receive Requirements Verification Document From Project A	ProjA-002	No	Yes	8/15/24	8/15/24																								
Test and Verify Product Performance		No	No	8/16/24	8/22/24																								
Generate Test Report		No	No	8/23/24	8/26/24																								
Deliver Test Report	ProjB-001	Yes	No	8/26/24	8/26/24																								

End of Exercise

Multi-Project Environment Roles Explained

When working in a **Multi-Project Environment**, **communication** and **teamwork** are vital to the success of the scheduling process. It's extremely important for teams to identify and clearly define who is doing what, so the entire team can be on the same page and execute their role effectively as the program progresses through time.

Subproject Owner

The role of a **Subproject Owner** is to develop a credible schedule that yields accurate dates – including the dates for their own Givers. This person or team should be the primary owner of their Subproject file and should have authority over changes to their Subproject. In addition, SSI recommends the Subproject Owner be responsible for the following:

- Identifying and communicating their needed Receivers with other Subproject Owners
- Making any baseline changes within their file
- Collecting and performing status updates in their file
- Conducting quality control checks/audits within their file
- Understanding the driving paths to Givers and major milestones within their file
- Understanding resource utilization within their file

Master Project Integrator

The role of a **Master Project Integrator** is to combine two or more Subprojects and connect any **Handoffs** to form the **Master Project**. Once a Master Project has been created and Handoffs have been linked, the Master Project has calculated the cumulative effect of Givers that drive Receivers. The Master Project Integrator should be responsible for communicating synchronized dates to Subprojects. In addition, SSI recommends the Master Project Integrator be responsible for the following:

- Setting up the Subproject and Master Project Shell files
- Communicating with the Subproject Owners when and where they need to provide their current Subproject files for integration.
- Communicating any Handoff issues, such as potential circular relationships, to the Subproject Owners
- Performing Master Project level analyses
- Conducting and driving Master Project level meetings (Program Critical/Driving Path Review meetings, Program Headcount Review meetings, etc.)

SSI's Recommended Multi-Project Management Process

Structured Solutions recommends the following when working in or considering switching to a Multi-Project Environment:

1. Identify and assign Multi-Project roles to team members
2. Identify which custom fields, calendars, tables, filters, groups, and resources are/will be commonly used across Subproject files and create a Shell file for your Subprojects and Master Project.
3. In each Subproject, make use of a Project Unique ID (PUID) field so there is always a reference back to a task's "native" Unique ID when Subprojects are integrated into a Master Project.
4. Use the Handoff Method to identify dependencies (predecessors and successors) that cross Subproject boundaries. This will make it possible to use many of SSI's Multi-Project tools. Only create a Handoff between Subprojects if there is a demand from the Receiver.
5. Use the "Unlinked" method to insert Subproject files into the Master Project shell. This method avoids all of the problematic issues teams will usually encounter when using the "Linked" method. It also gives total change control and responsibility of any given Subproject to a single individual or team, which is usually a good thing.
6. Treat the Master Project as a temporary read-only file to see where dates are, submit deliverables, and perform program level analyses. All the actual day-to-day work should occur within the Subprojects. Reintegrate the Master Project file as often as needed by the program team, using the most up to date Subprojects. If changes outside of connecting Handoffs cannot be avoided, use SSI's Change Tracking Tools to record changes so they can easily be replicated in the Subprojects that are affected.



The following diagram represents SSI's recommended battle rhythm for working on and integrating Subprojects into the Master Project. This process should be performed on a regular recurring basis. In the diagram, the yellow bubbles represent tasks that are the responsibility of the Subproject Owners, while the blue bubbles represent tasks that are the responsibility of the Master Project Integrator:

