
Assembly Modeling

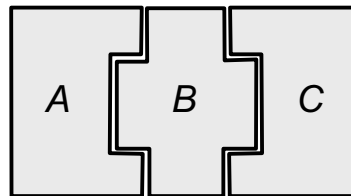
MEEM5408 Design Automation: Theory and Implementation

MichiganTech

Assemblies

- An assembly is a collection of components (parts and sub-assemblies) arranged in a specific way.

E.g., Widget assembly has components: A, B, C.



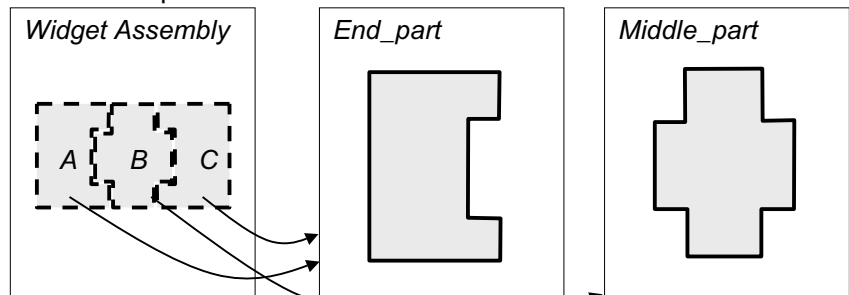
- Assemblies are useful for:

MEEM5408 Design Automation: Theory and Implementation

MichiganTech

Assembly Part Info.

- In early Assembly Modelers, all the information for each component was put in the file. This was very inefficient when the same part was used multiple times. E.g., A and C are the same. Information for A and C was included twice.
- Now CAD systems keep **pointers** to the part or sub-assembly information. I.e., an assembly only stores virtual copies (instances) of sub-components.

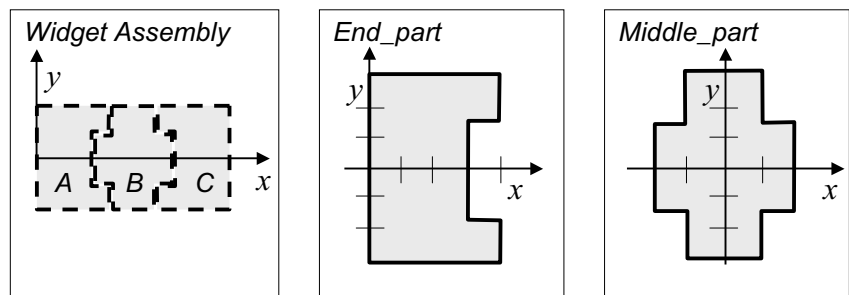


MEEM5408 Design Automation: Theory and Implementation

MichiganTech

Assembly Positioning

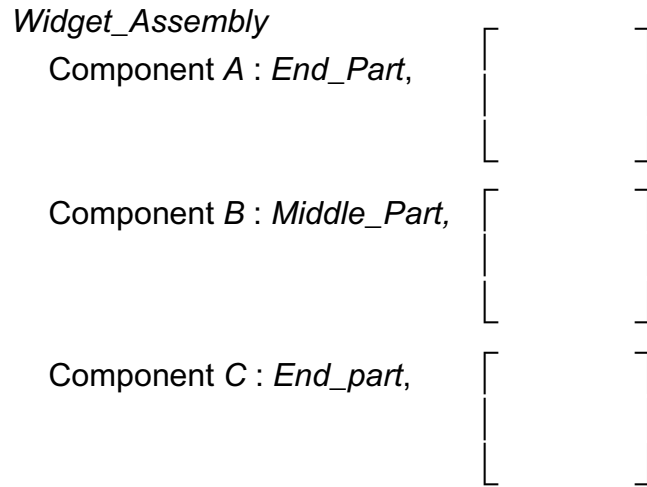
- Since copies of parts are virtual, it is also necessary to keep track of the position of each component.
- The position of each component is given as a transformation matrix.



MEEM5408 Design Automation: Theory and Implementation

MichiganTech

Assembly Positioning



MEE5408 Design Automation: Theory and Implementation

MichiganTech

Assembly Constraints

- Components can be positioned with respect to each other using **positioning commands**.
- However, modern CAD systems use **constraints** (mating conditions) to automatically position components.
- Assembly constraints allow components to be automatically re-positioned when the size of a **part changes** (e.g., if *Middle_part* becomes wider, component C automatically moves right.)
- UG assembly constraints are: **mate, align, angle, parallel, perpendicular, center, distance, tangent**

MEE5408 Design Automation: Theory and Implementation

MichiganTech

Assembly Data Structure

- In UG, a part file can contain:
 -
 -
- For each component, an assembly keeps track of:
 -
 -
 -
 -
 -
 -

MEEM5408 Design Automation: Theory and Implementation

MichiganTech

Bottom-Up vs. Top-Down Design

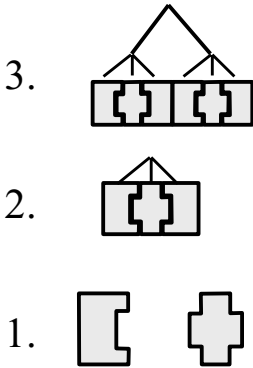
- **Bottom-up:** component parts are designed and edited apart from their usage in a higher assembly.
 1. Create part solid models.
 2. Combine parts into sub-assemblies.
 3. Combine sub-assemblies into assemblies.
- **Top-down:** the hierarchy of assemblies and sub-assemblies is designed first, then part solid models are designed in place.
 1. Create highest level assembly.
 2. Add empty sub-assemblies and parts to assemblies.
 3. Create solid models in empty part files.

MEEM5408 Design Automation: Theory and Implementation

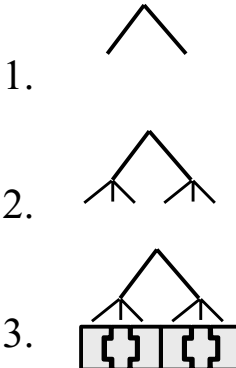
MichiganTech

Bottom-Up vs. Top-Down Design

Bottom Up



Top Down

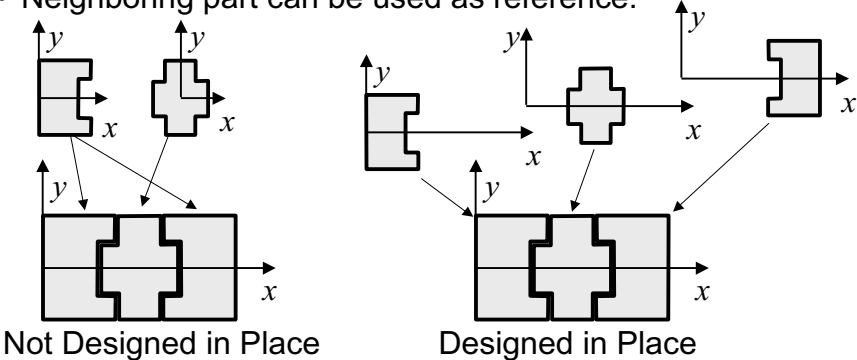


MEEM5408 Design Automation: Theory and Implementation

MichiganTech

Design in Place

- Also known as “Design in Context.”
- Solid Models are created such that the part and assembly coordinate systems line up in the assembly.
- Neighboring part can be used as reference.



MEEM5408 Design Automation: Theory and Implementation

MichiganTech