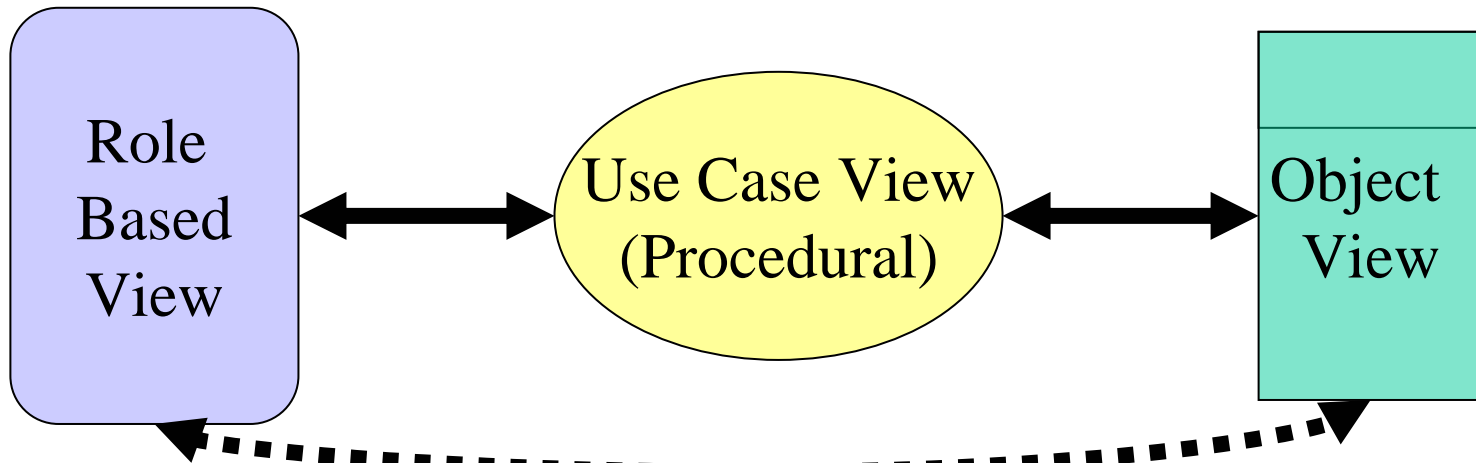


# Processes, Use Cases and Objects

- Three fundamentally different views, and notation used.
- Process: Role Based. Activities assigned to (and spread across) roles.
- Use Case: Procedural. Actors assigned to (and spread across) major activities.
- Object Oriented. Activities assigned to (cohesive) objects.

# Mappings



- Difficult to preserve mapping when notations are orthogonal.
  - Sometimes utilise further (structuring and overview) notations, such as POSD.

# Initially: Processes to Use Cases

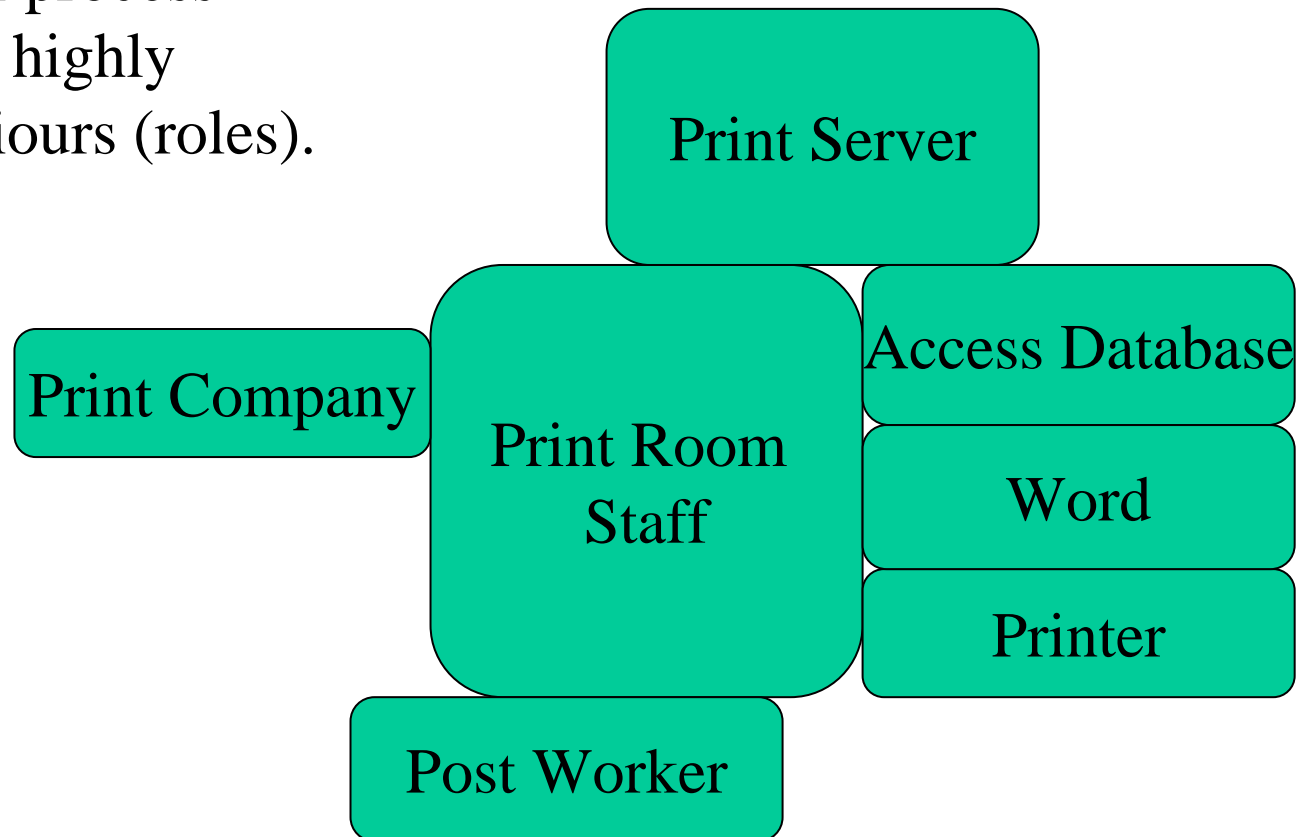
- Role Based. Roles (activities spread).
- Use Case: Abstract activities (actors spread)
- Hence, to map roles to use cases we will need to consider large-scale activities.
  - Typically chunks of interactions, (or actions).
- Hence, Use Cases may be found:
  - In the connections between roles.
  - In the chunks of activity within roles.

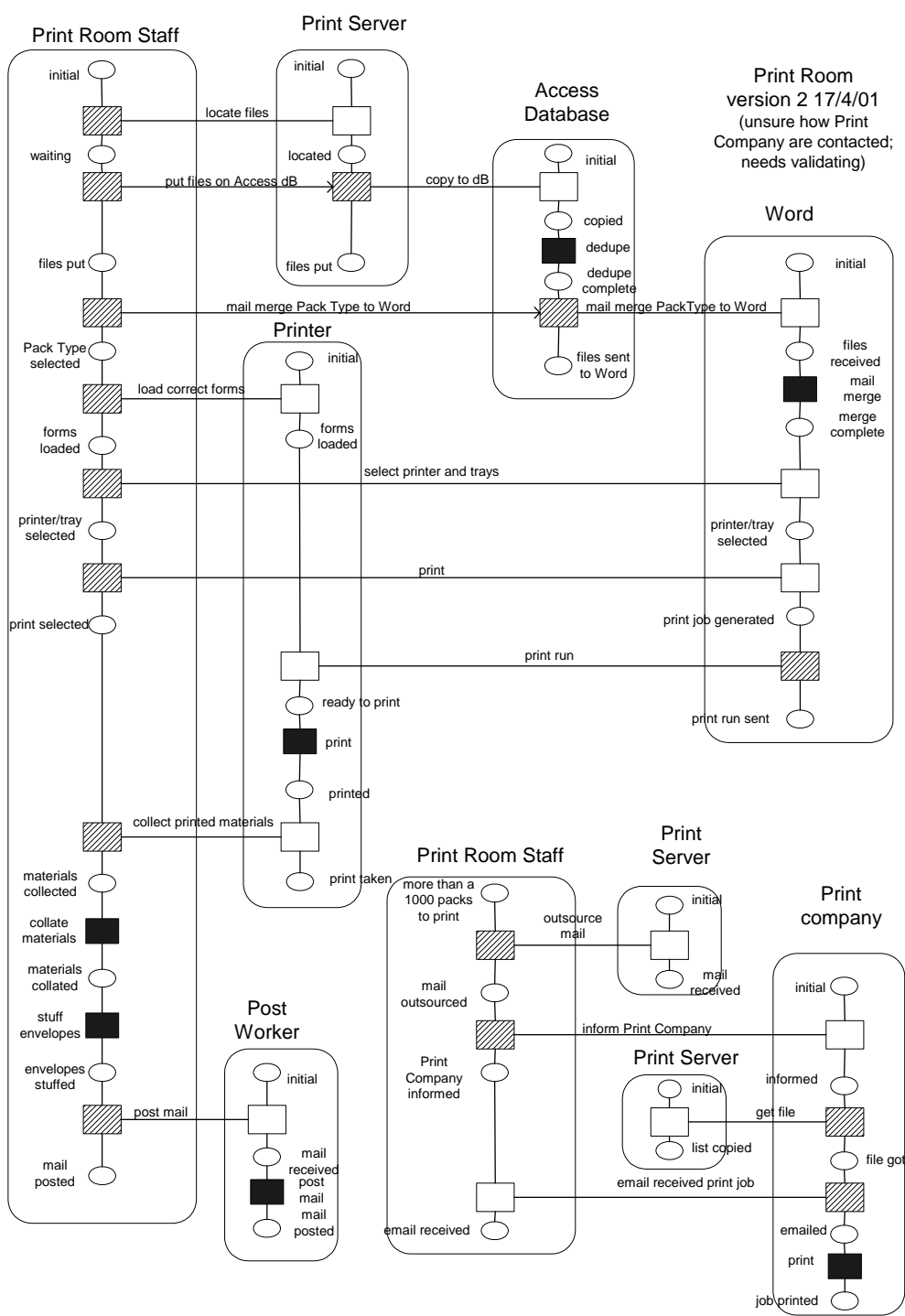
# Guidelines

- Use Case may be a chunk of connections (bundles of related interactions).
  - Hence, interactions become events within use cases (or whole use cases).
- Similarly, chunks of related actions, or combined activity (actions and interactions).
- Roles often actors (not quite the same).
- May use more abstract mechanisms to handle ‘grouping’ of activities.

# Overview Model: POSD For Printing

Simple view of process  
Suggests some highly  
coupled behaviours (roles).





# RAD (Process as was)

Detailed view of process  
 Again suggests some highly coupled roles.  
 Also suggest some complex (multiple-stage) interactions.

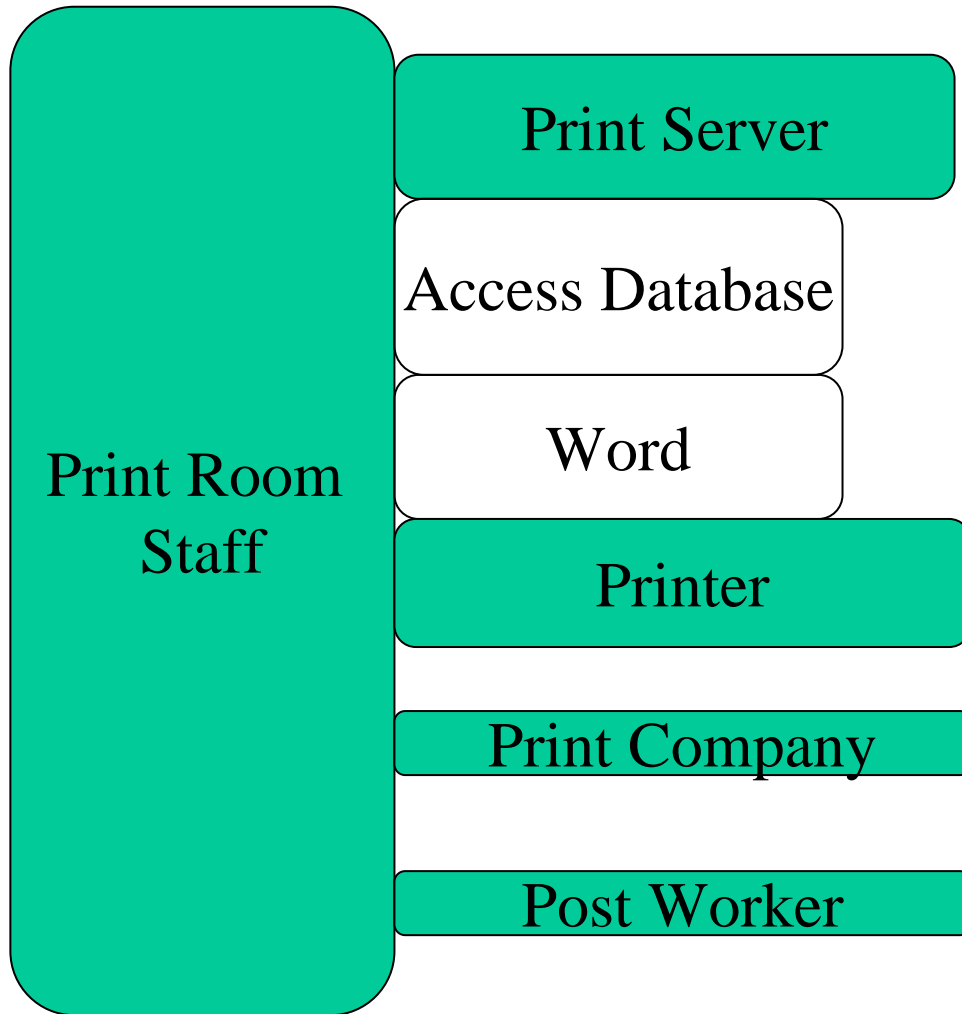
# Behaviours and Connections

- Behaviours (roles) share behaviours with other behaviours (roles) .
- Sometimes behaviour is simple, e.g., an interaction in a RAD.
- Sometimes the behaviour itself is complex and worthy of separate consideration.
  - Behaviours may span roles (or actors or Use Cases).
  - Can depend on perspective. (Email).

# Shared Behaviours

	Print Room Staff	Print Server	Access Database	Word	Printer	Post Worker	Print Company
Print Room Staff		I	I	I	I	I	I
Print Server	I		I				
Access Database	I	I		I			
Word	I		I		I		
Printer	I			I			
Post Worker	I						
Print Company	I						

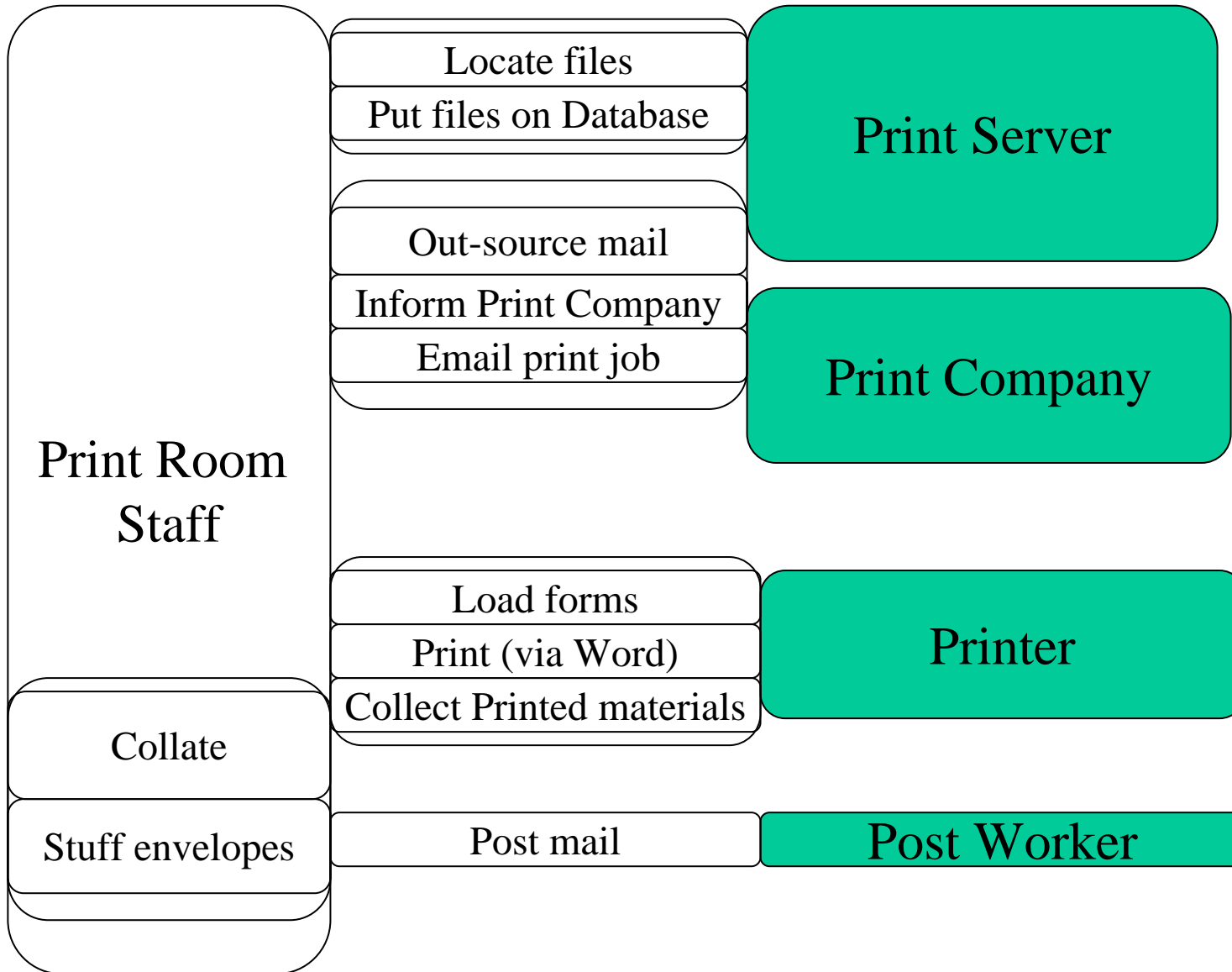
# Revised POSD For Printing



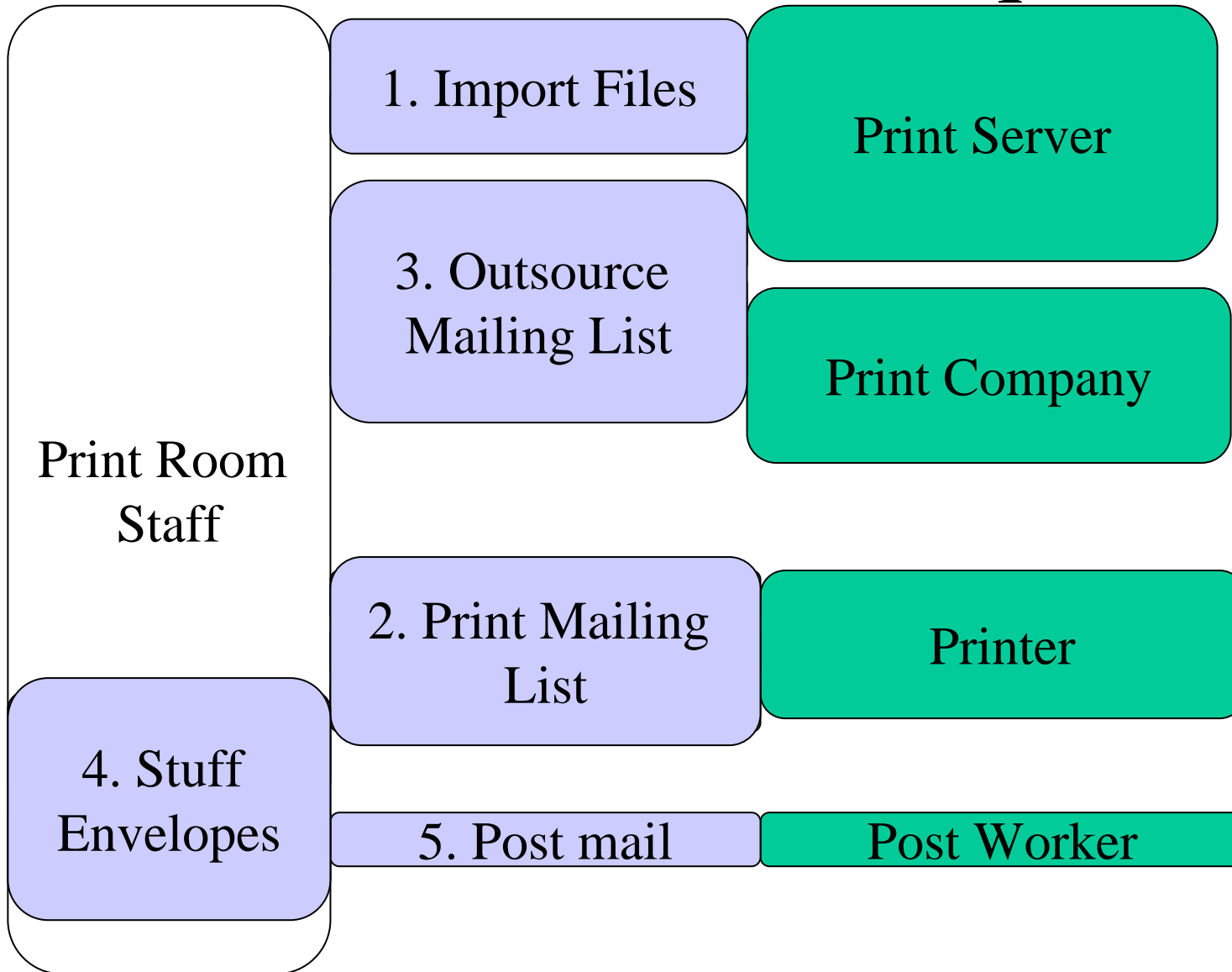
All three views (both POSD and RAD) are consistent in sharing of behaviours.

Note: Access and Word typically considered mechanism (omitted) in process model.

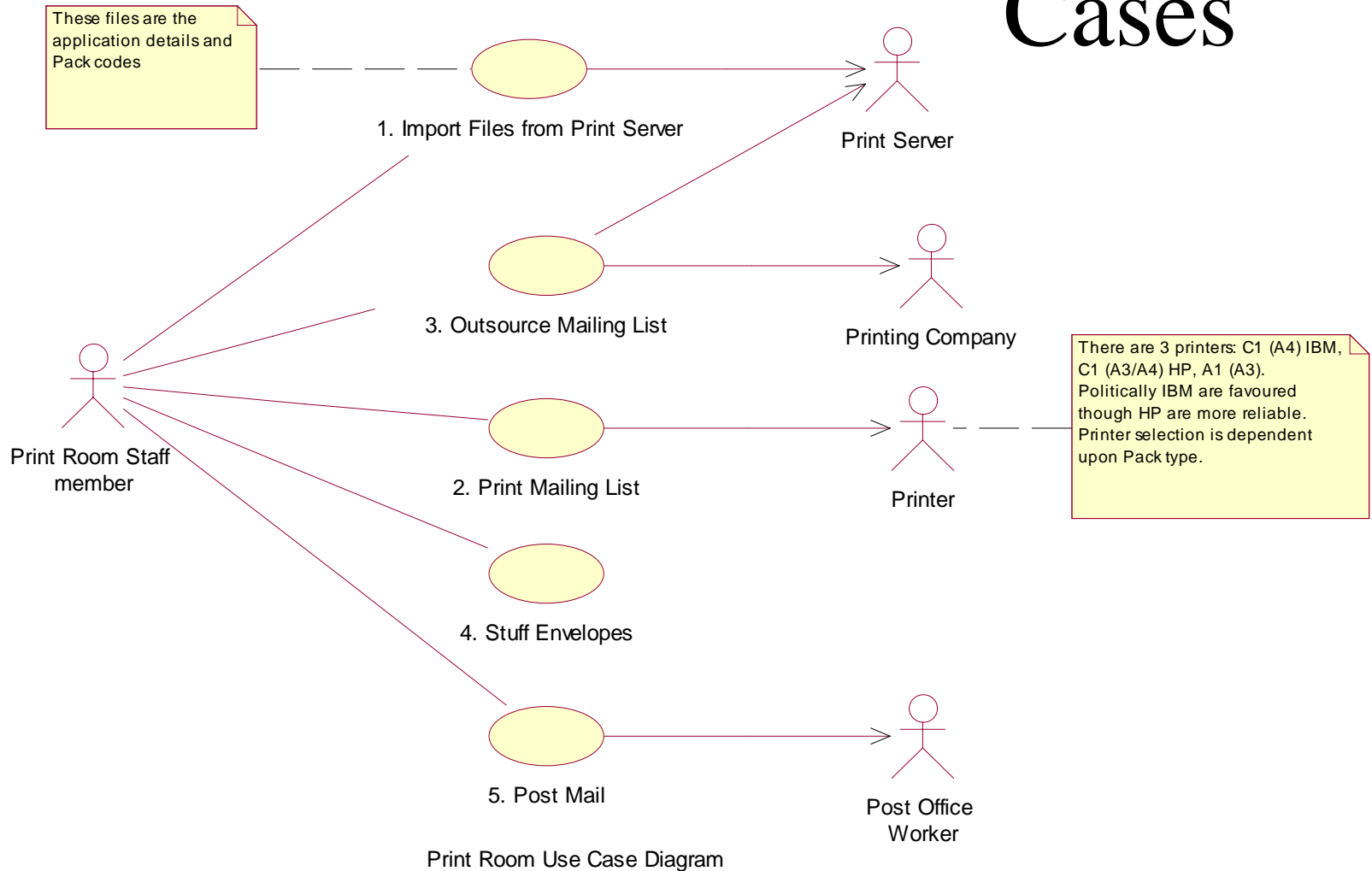
# Behaviour and connection



# Overview and Sequence



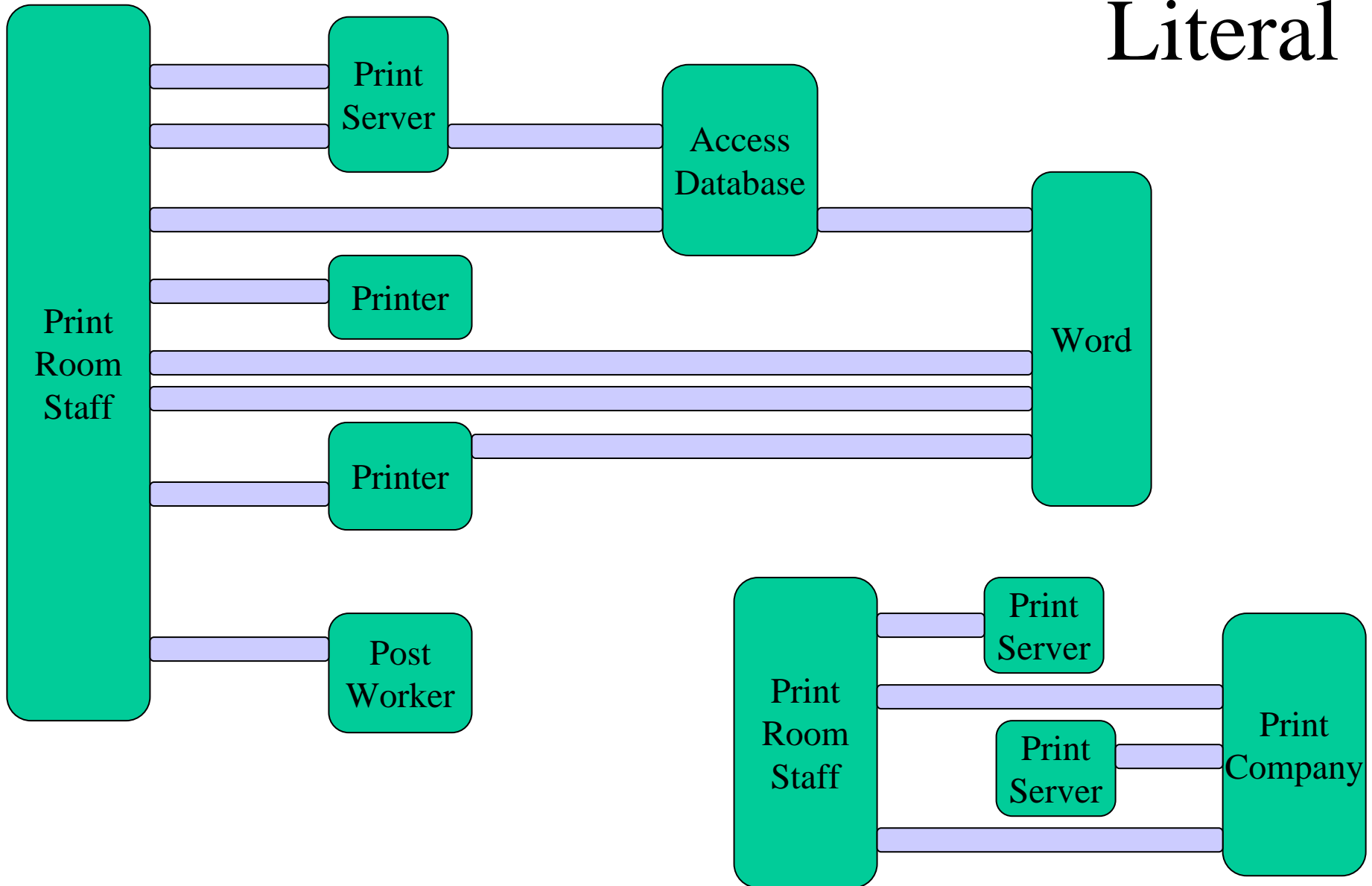
# Matching Use Cases



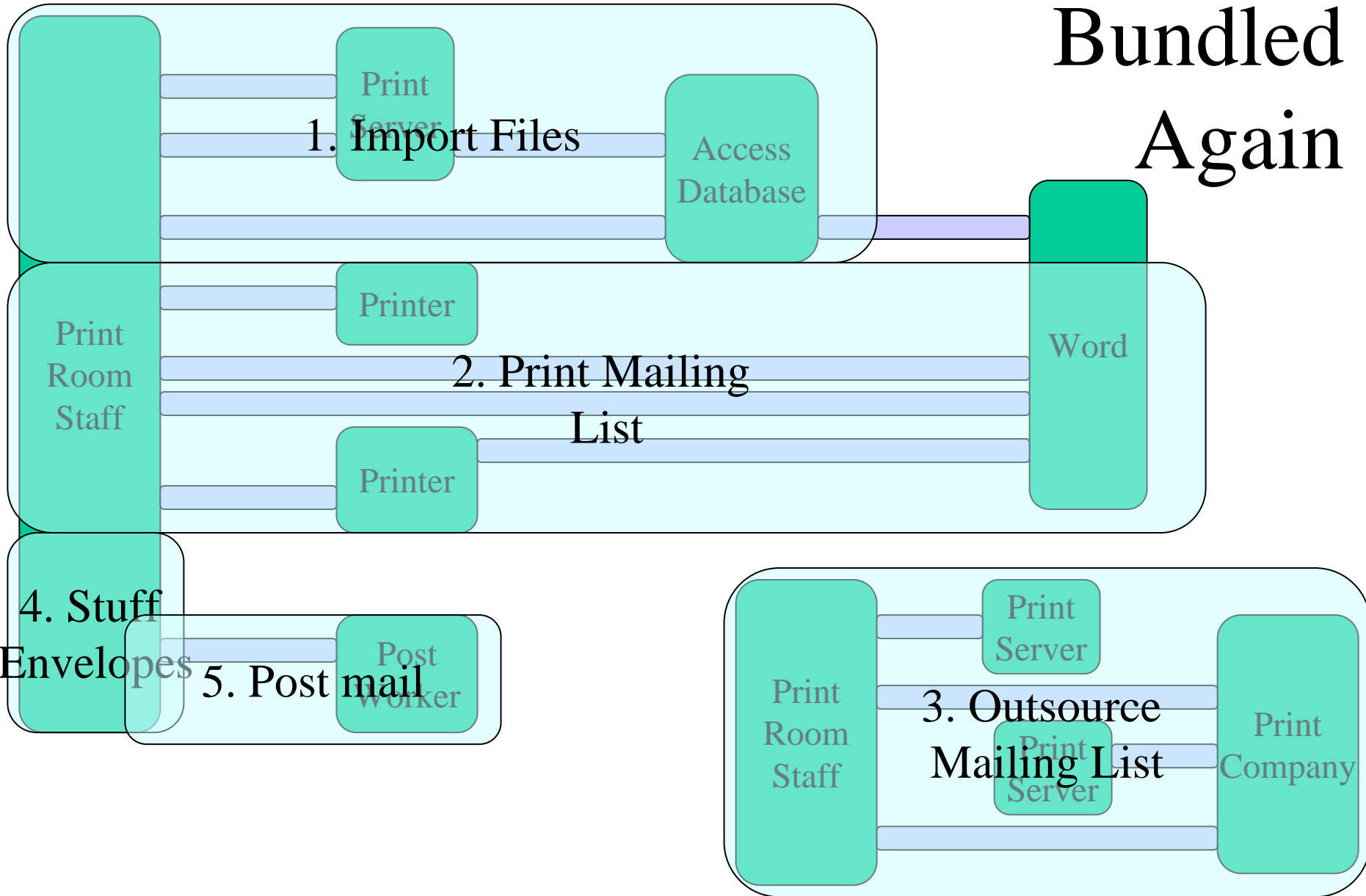
# Rules (what have we done?)

- Overview of process components (POSD).
- Consider process model (RAD).
- Move towards use cases (procedural view?).
  - Maintain connections.
  - Demonstrated preservation of mapping.
- What rules / guidelines can be used to move from RAD to Use case diagram?
  - To make this more mechanistic
  - To move (straight) from RAD to Use Case.

# RAD Connections: Literal



# Connections: Bundled Again



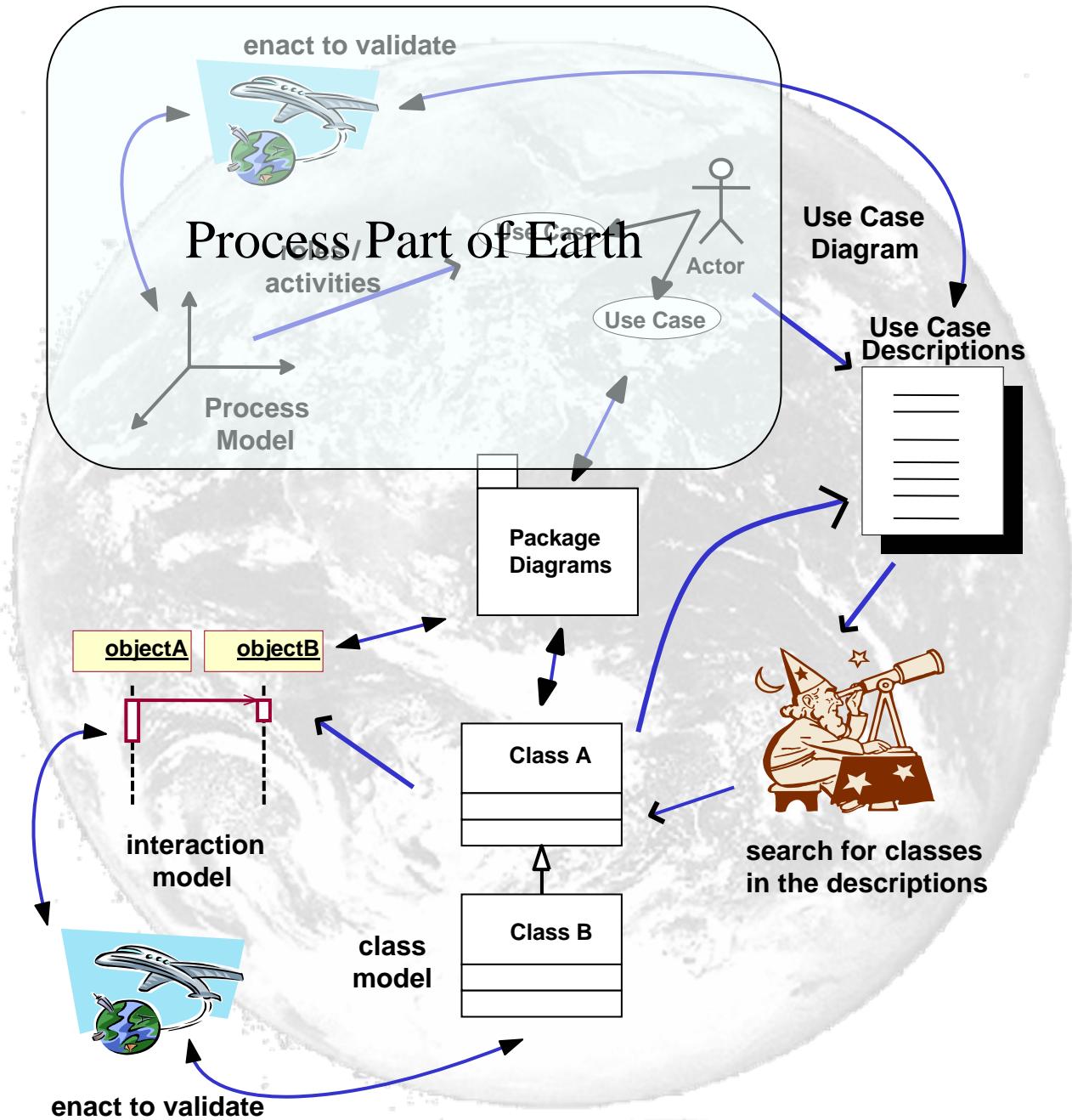
# Observations: Process so far

- RAD interaction detail not always helpful for moving towards Use Case.
  - Better to consider abstract connections (as POSD, or simplified RAD) and assign meaningful names.
  - POSD allows further scope for viewpoints.
  - Preservation of mapping helps to ensure that detail is not omitted.
- RAD phase helps ‘debug’ process.
  - Particularly where ‘enaction’ used.

# Next Steps: Use Cases onwards

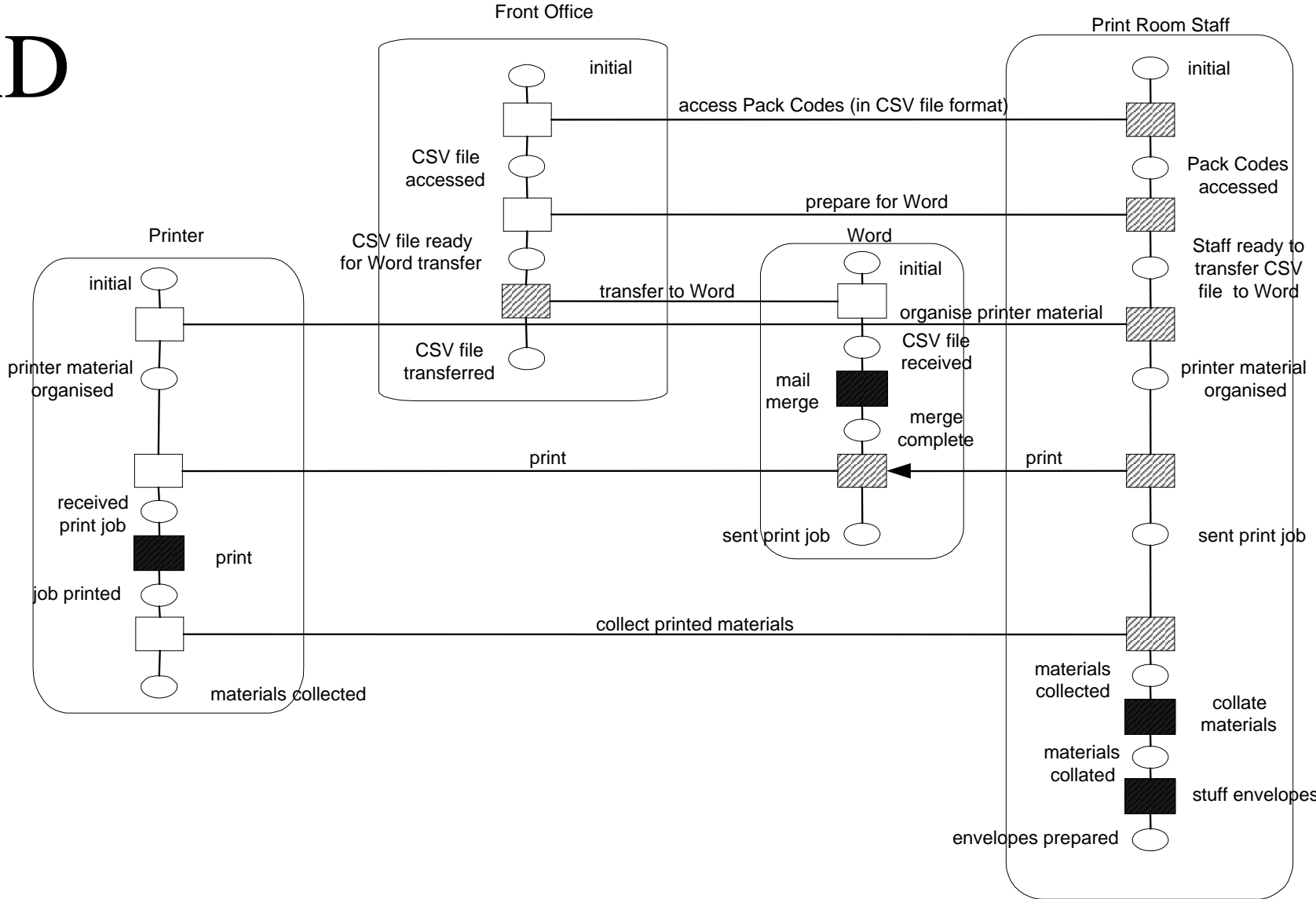
- Have moved from process description to use case diagram.
- Use cases help identify packages.
- Each Use cases has associated description.
- From descriptions we discover objects.
  - By asking sets of questions.
  - By refining descriptions.
  - By considering dependencies {further enactment}.

# EARTH method

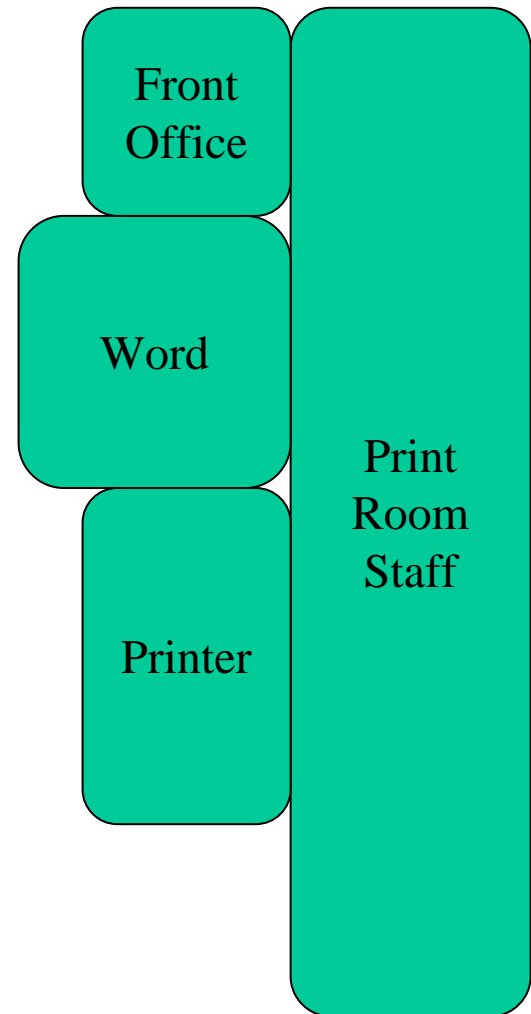
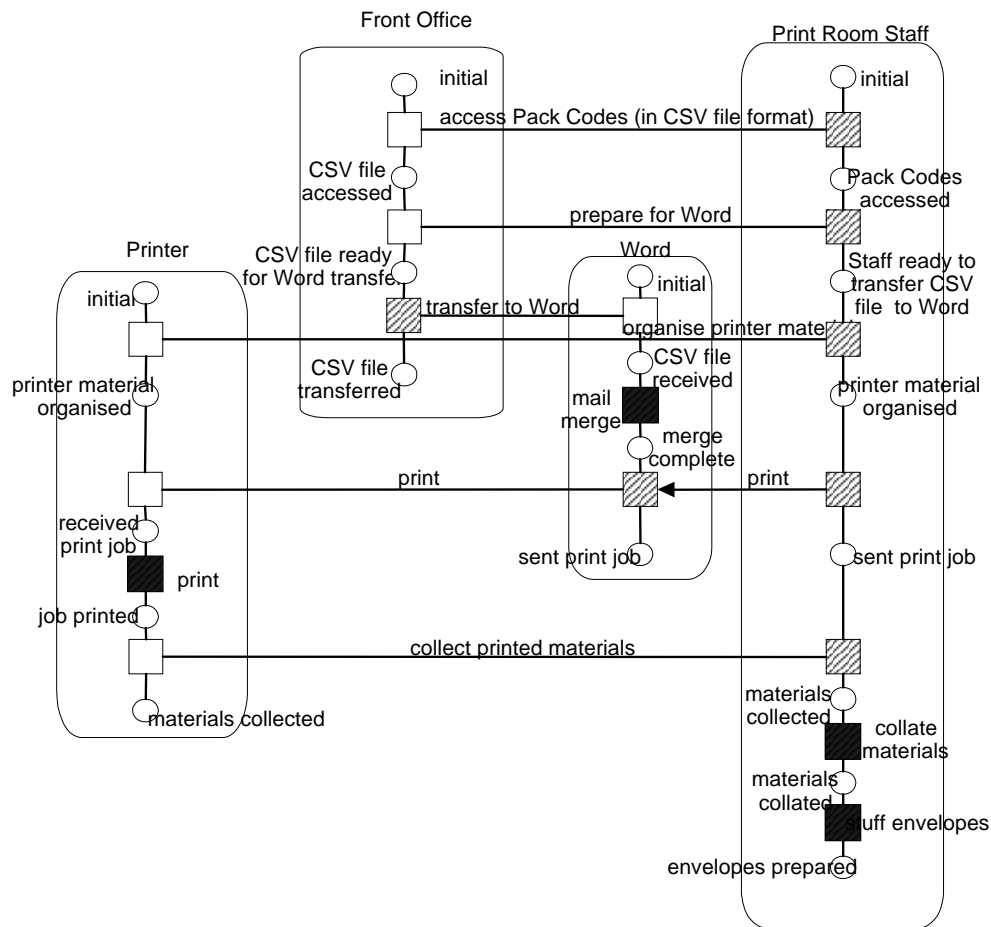


# Envisioned Process

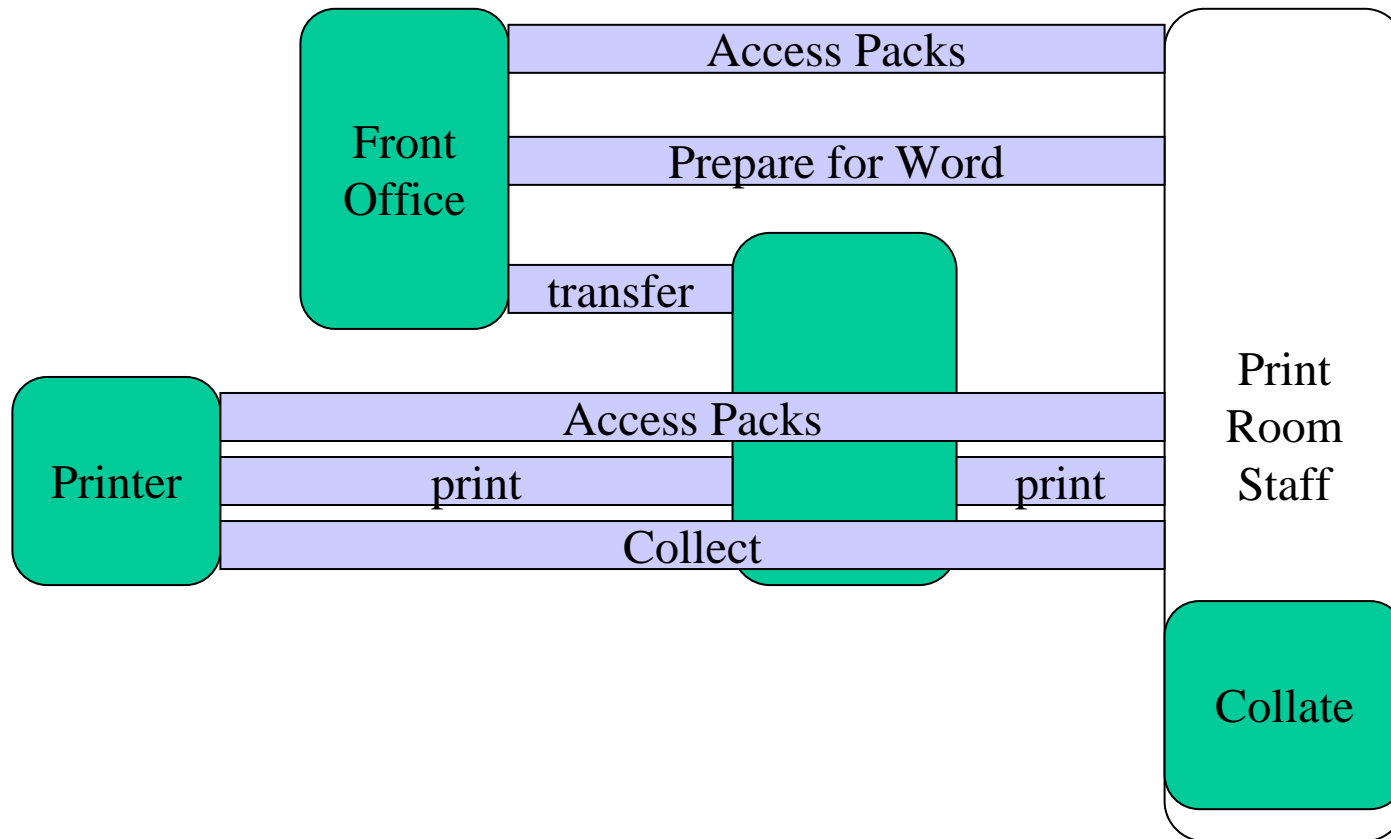
# RAD



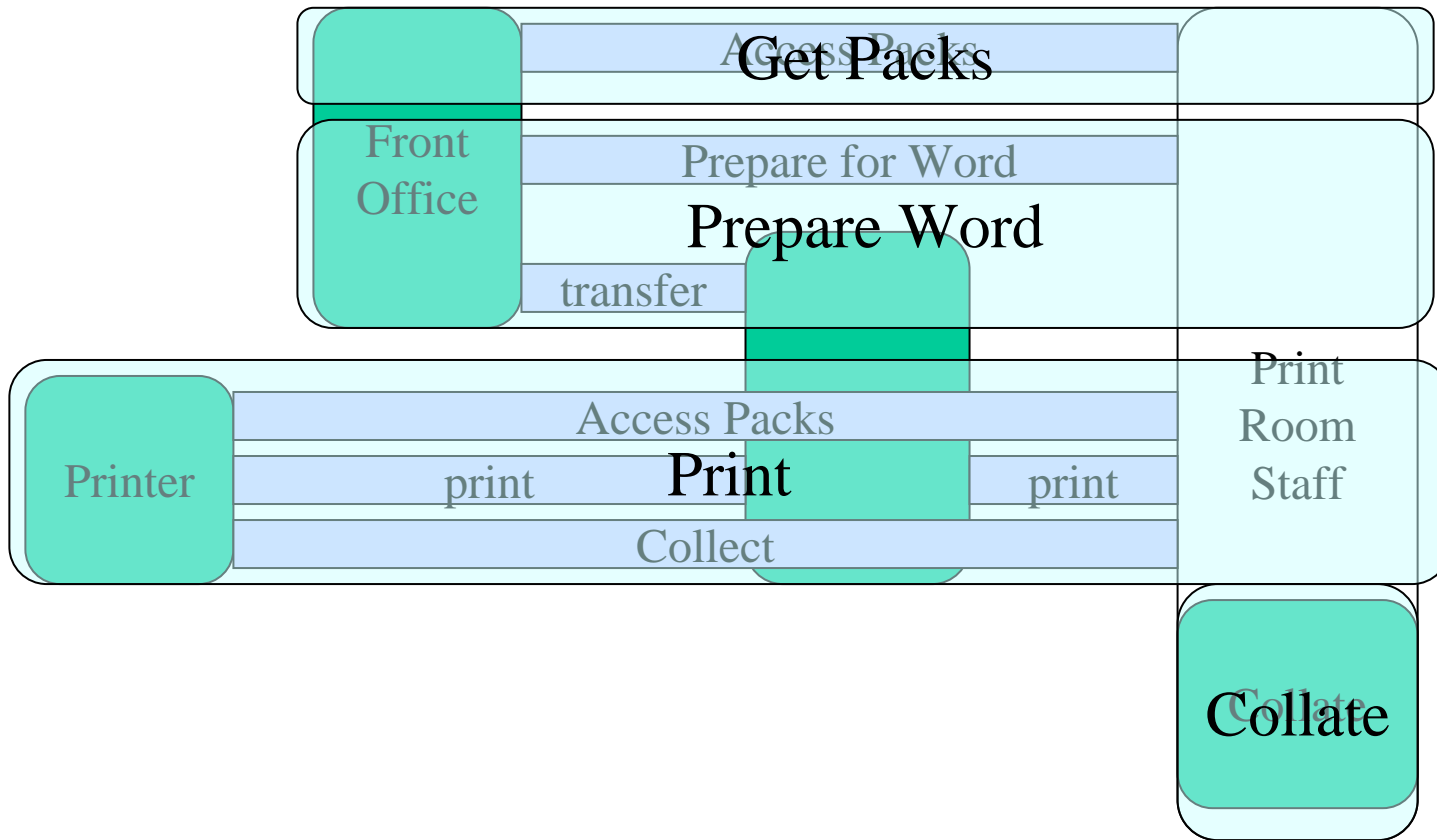
# POSD Overview



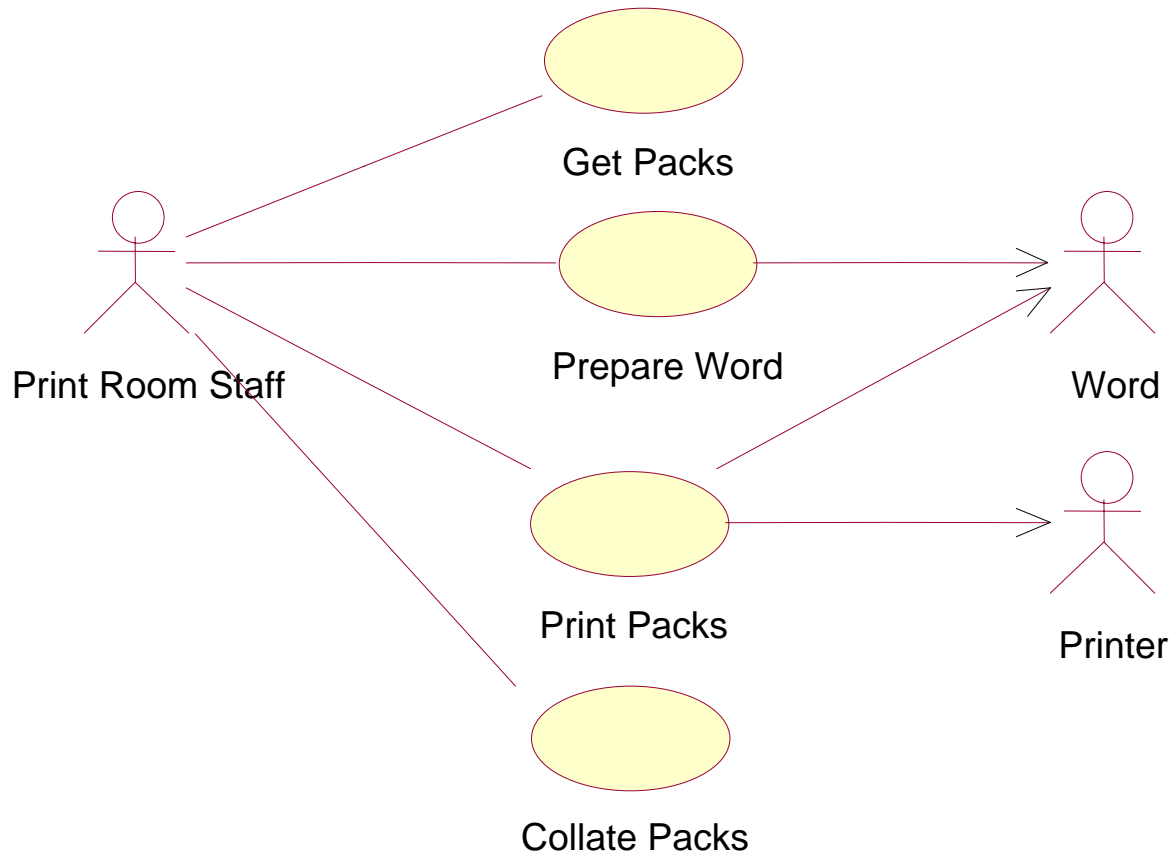
# Overview with Connections



# Connections Bundled



# Matching Use Case



# Lessons

- Much activity within connections.
  - Some bundling of both interactions and actions into use cases.
- Move from Use Case of process to Use Case as specification.
  - Hence, some process elements not described
- Beware inconsistent levels of abstraction.
  - E.g., single interactions as Use Case
  - Multiple actions and interactions as a Use Case.