Comparing the Decompositions Produced by Software Clustering Algorithms using Similarity Measurements

2001 IEEE International Conference on Software Maintenance (ICSM'01).

Brian S. Mitchell & Spiros Mancoridis

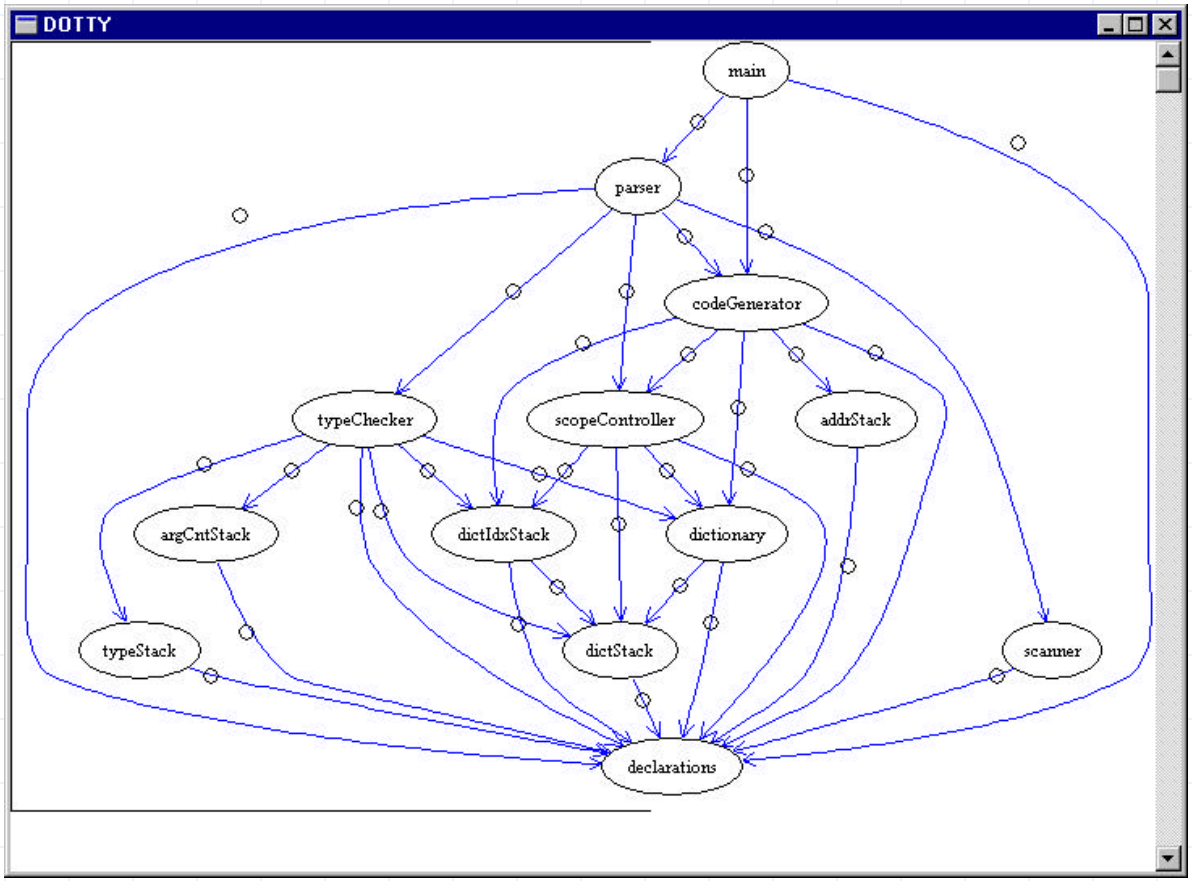
Math & Computer Science, Drexel University

Motivation

Using module dependencies when determining the similarity between two decompositions is a good idea...

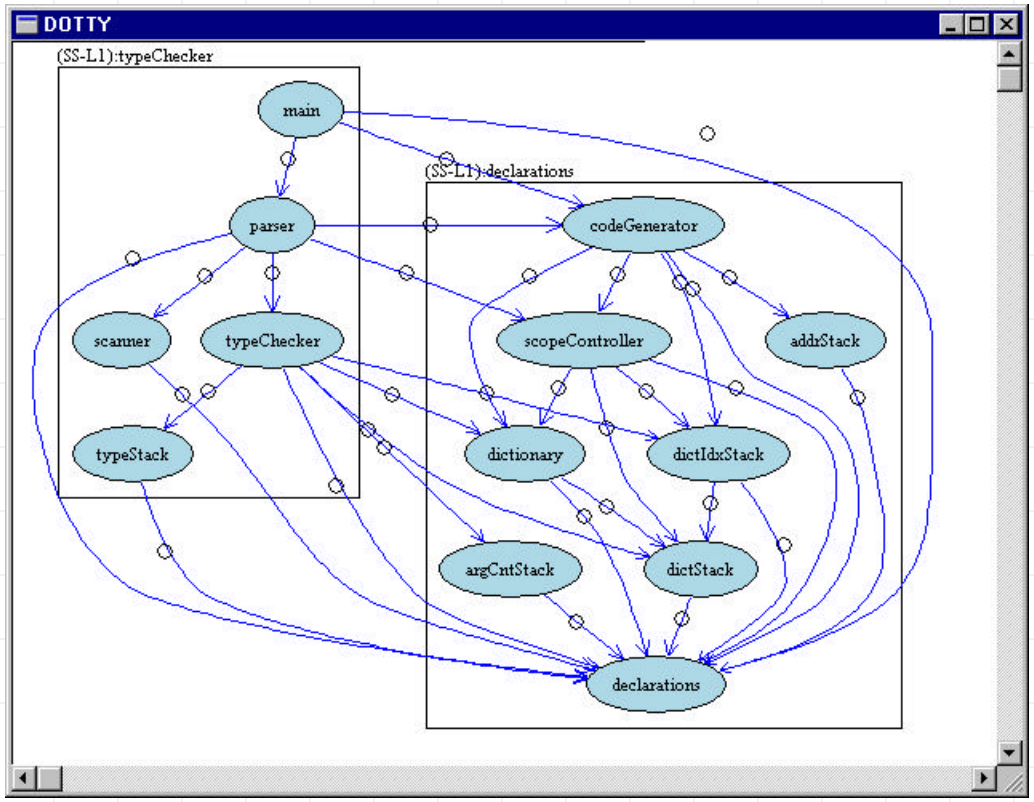
Clustering the Structure of a System (1)

Given the structure of a system…



Clustering the Structure of a System (2)

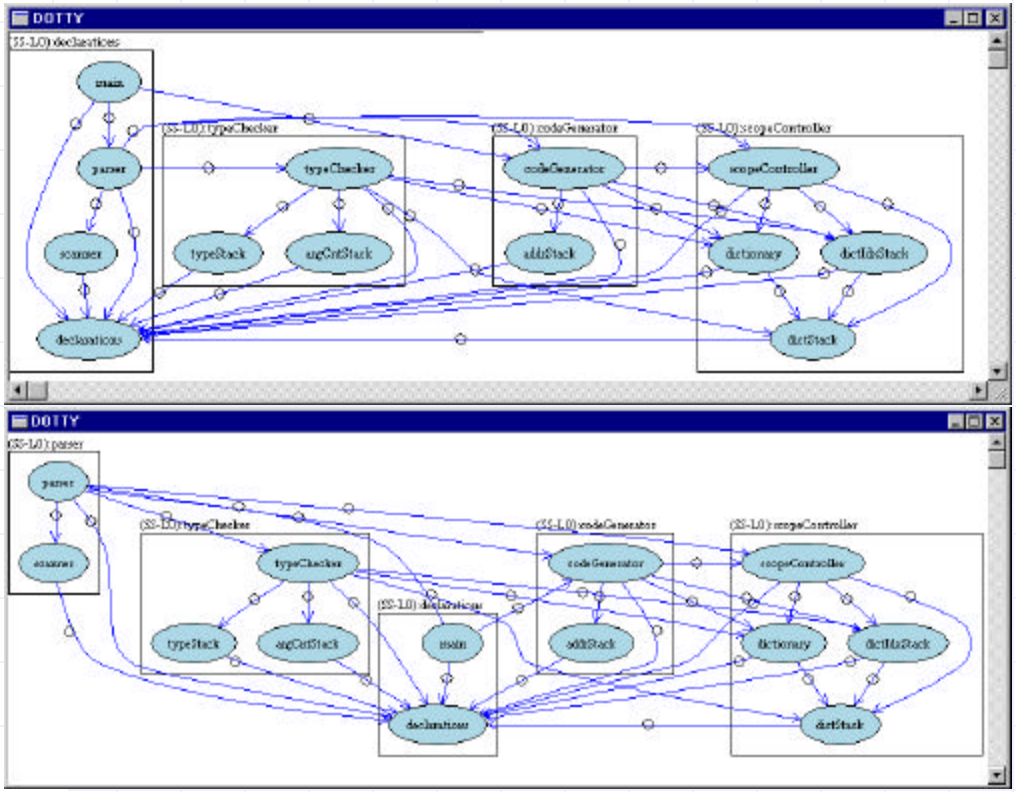
The goal is to partition the system structure graph into clusters…

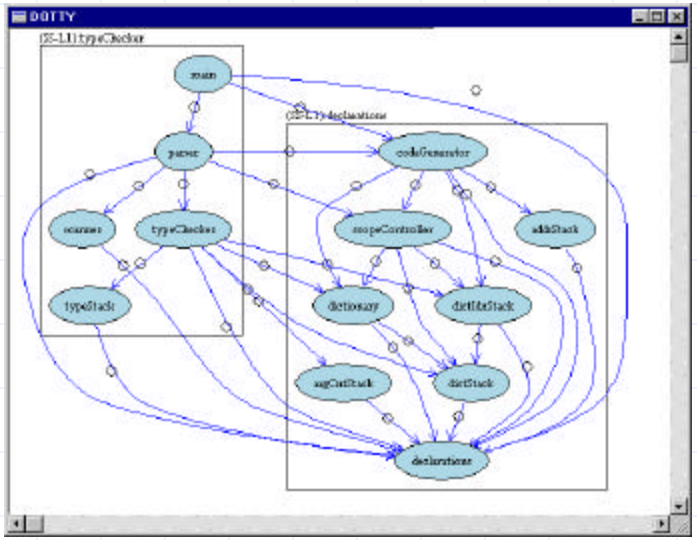


The clusters should represent the subsystems

Clustering the Structure of a System (3)

But how do we know that the clustering result is good?



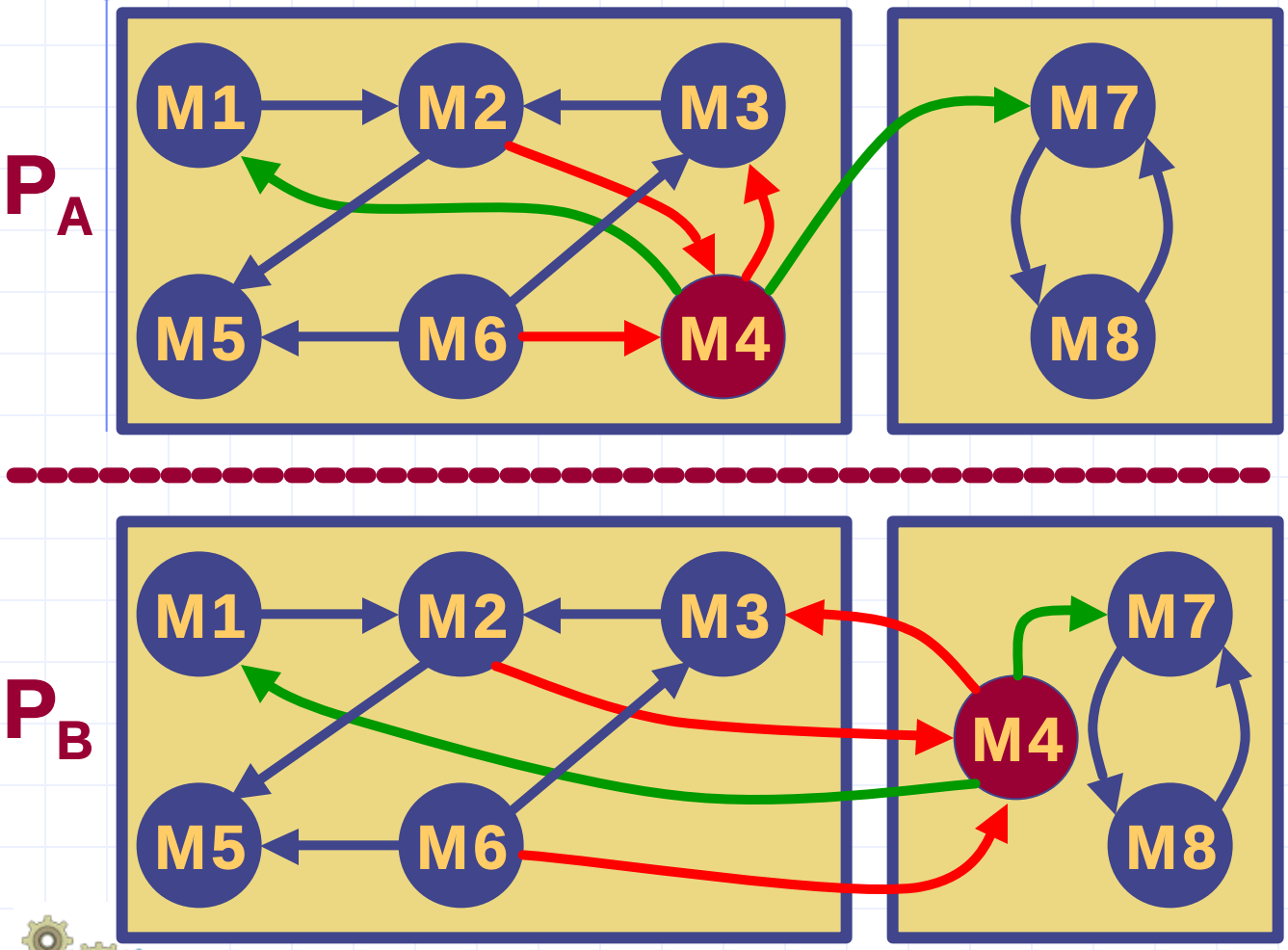


Ways to Evaluate Software Clustering Results...

Given a software clustering result, we can:

- Assess it against a mental model
- Assess it against a benchmark standard
- Techniques:
  - Subjective Opinions
  - Similarity Measurements

Example: How "Similar" are these Decompositions?



Blue Edges:
Similarity still the same...

Green Edges:
Similarity still the same...

Red Edges:
Not as similar...

Conclusions:
Once we add the **red** edges the similarity between P_A and P_B decreases

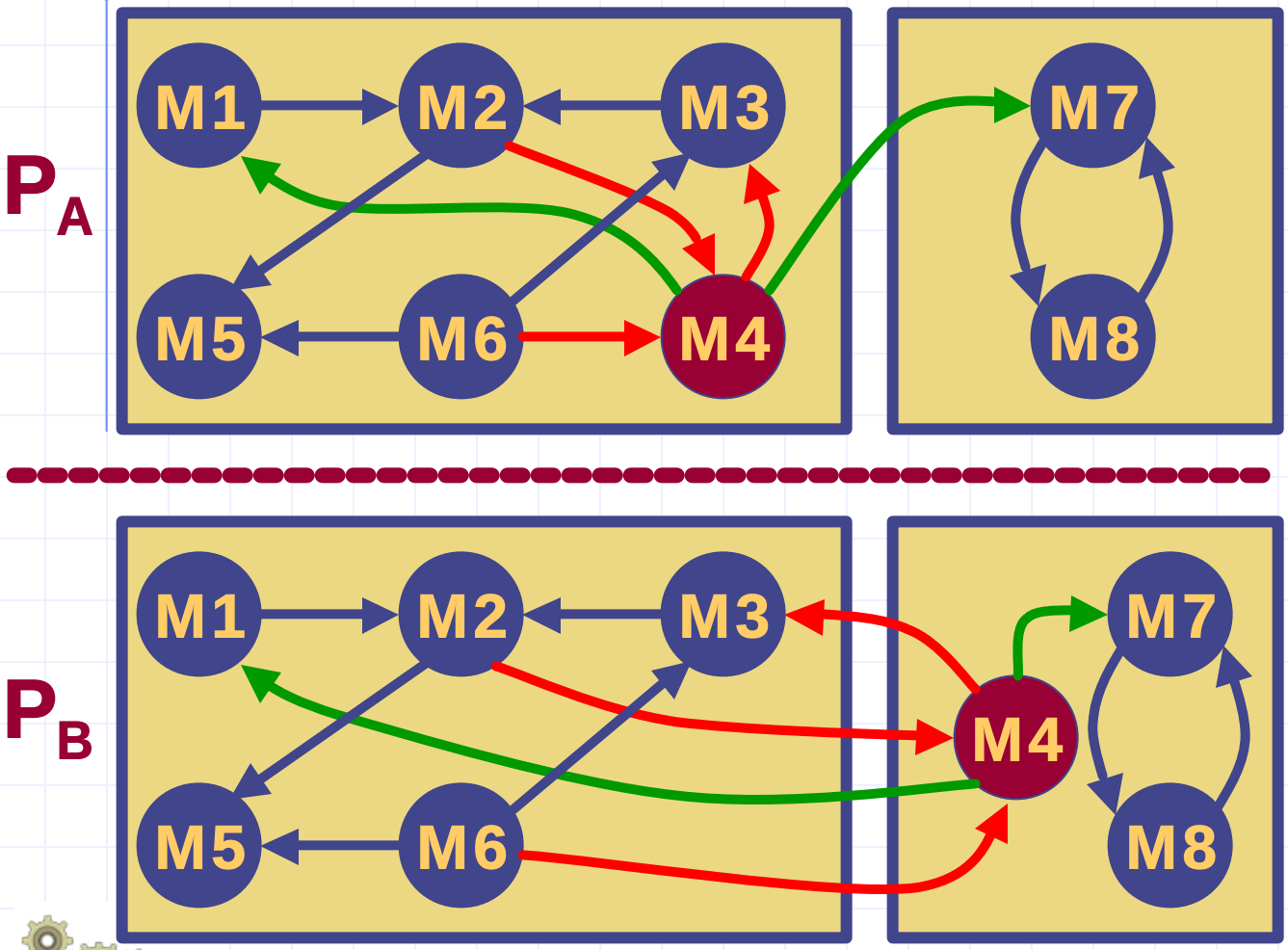
Observations

- Edges are important for determining the similarity between decompositions
- Existing measurements don't consider edges:
  - Precision / Recall *(similarity)*
  - MoJo *(distance)*
- **Our idea:** Use the edges to determine similarity

Research Objectives

- Create new similarity measurements that use dependencies (edges)
  - **EdgeSim** *(similarity)*
  - **MeCl** *(distance)*
- Evaluate the new similarity measurements against MoJo & Precision/Recall
- Use similarity measurements to support evaluation of software clustering results (see our WCRE'01 paper)

Example: How "Similar" are these Decompositions?

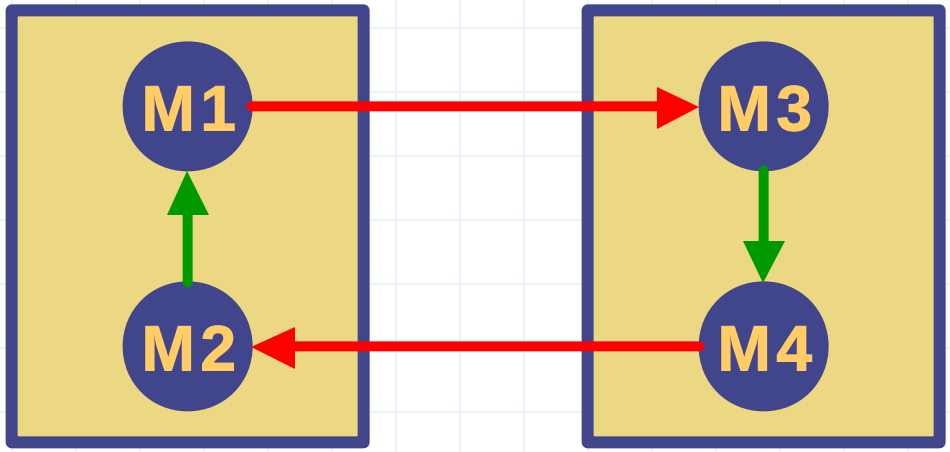


Add Blue Edges:
PR, MoJo, MeCl & EdgeSim **unchanged**.

Add Green Edges:
PR, MoJo, MeCl & EdgeSim **unchanged**.

Add Red Edges:
PR, MoJo **unchanged**.
EdgeSim, MeCl **reduced**.

Definitions

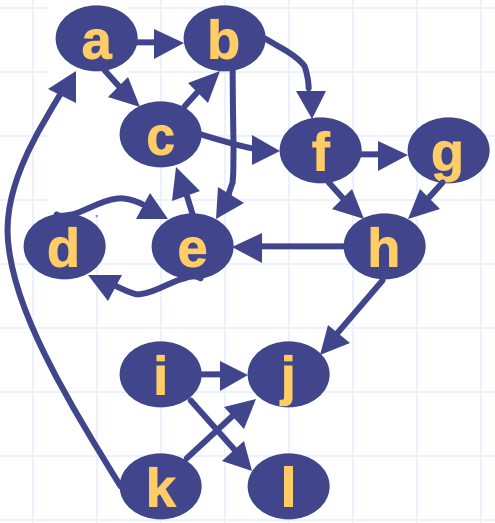


Internal / Intra-Edge: Edge within a cluster

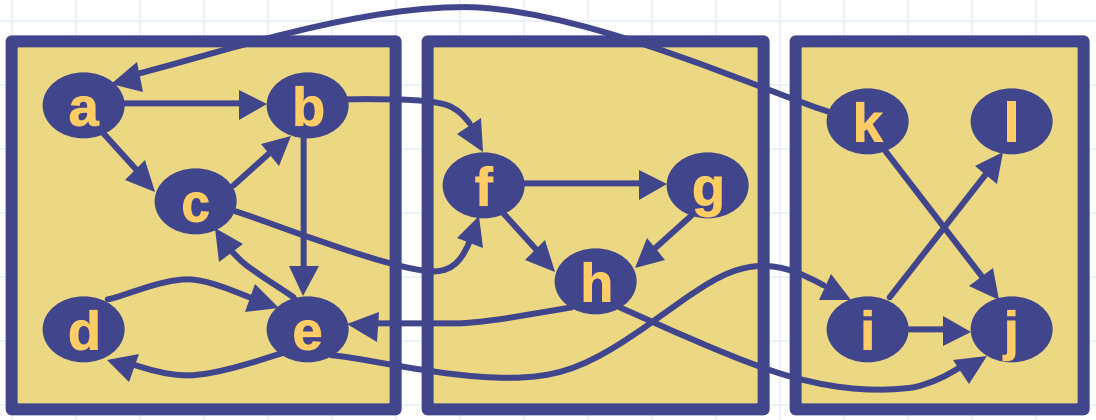
External / Inter-Edge: Edge between two clusters

EdgeSim Example

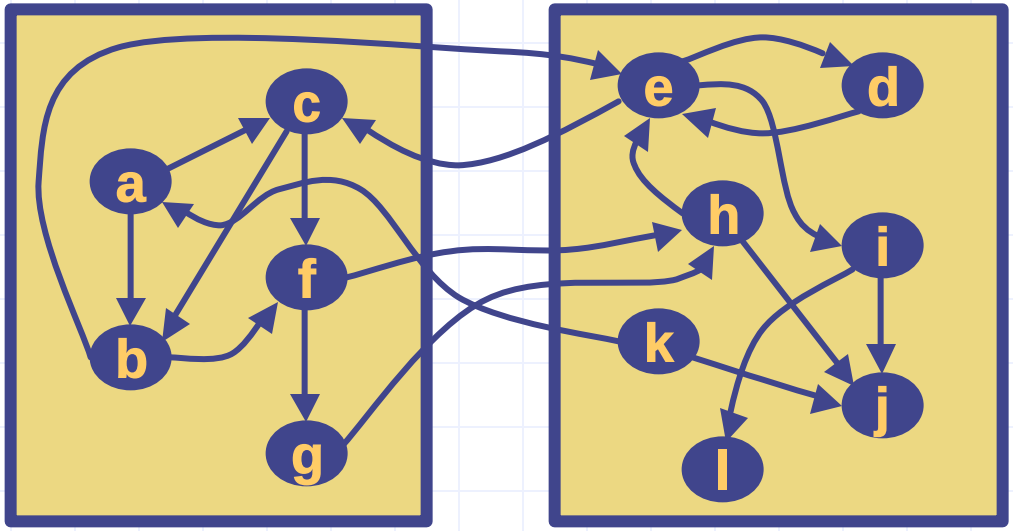
MDG



P_A

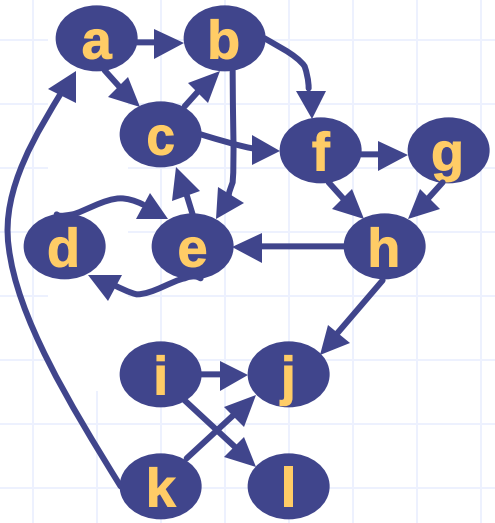


P_B

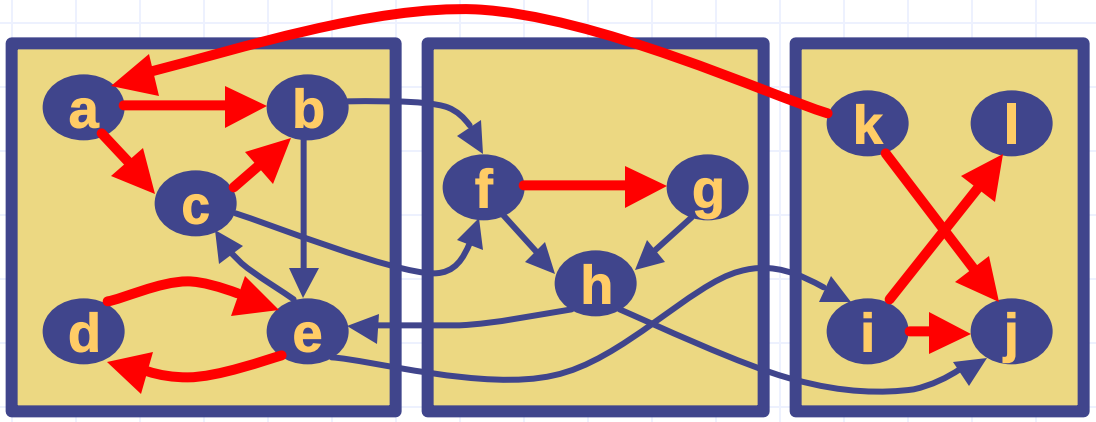


EdgeSim Example

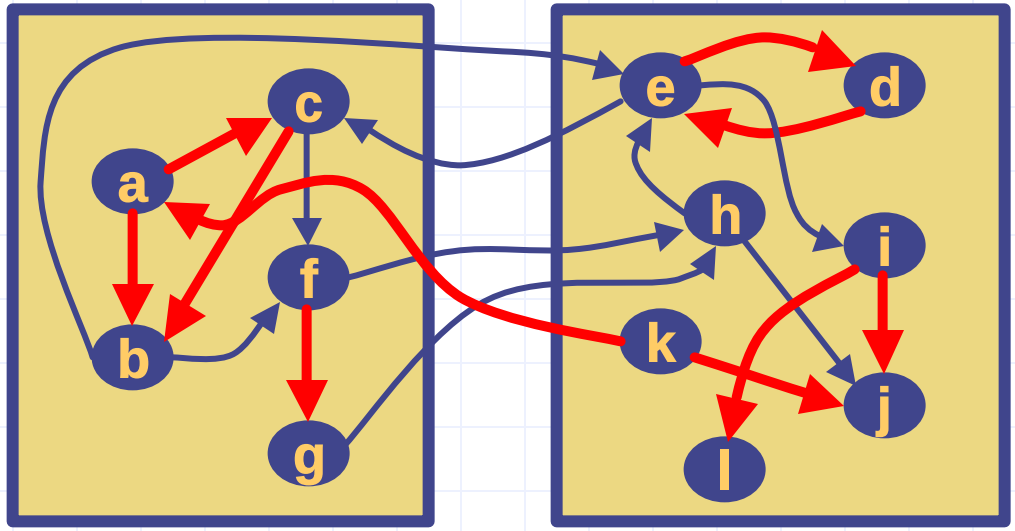
MDG



P_A



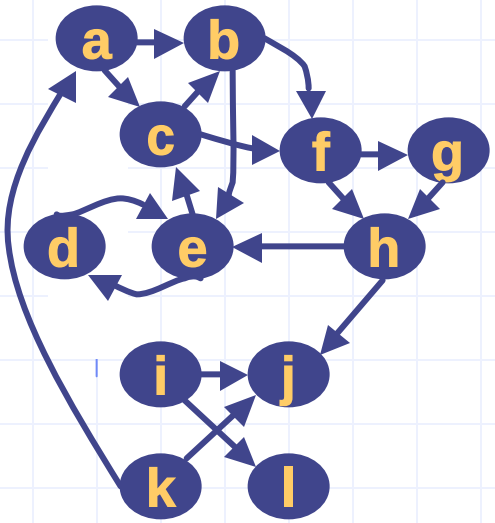
P_B



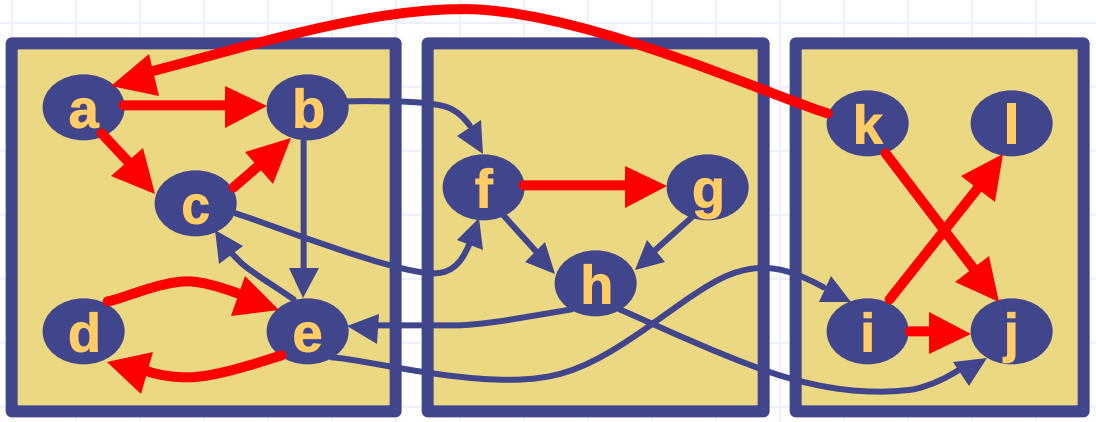
Step 1:
Find Common Inter- and Intra-Edges

EdgeSim Example

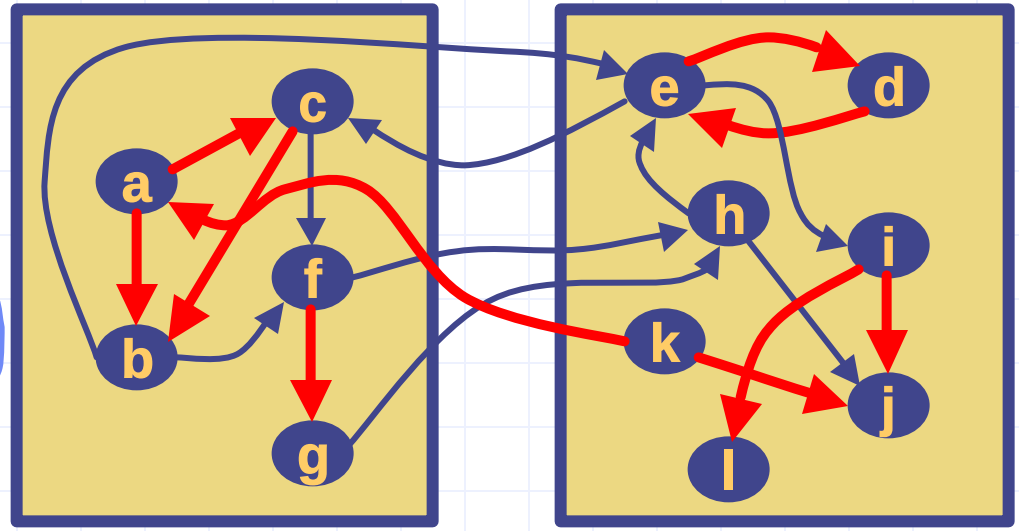
MDG



P_A



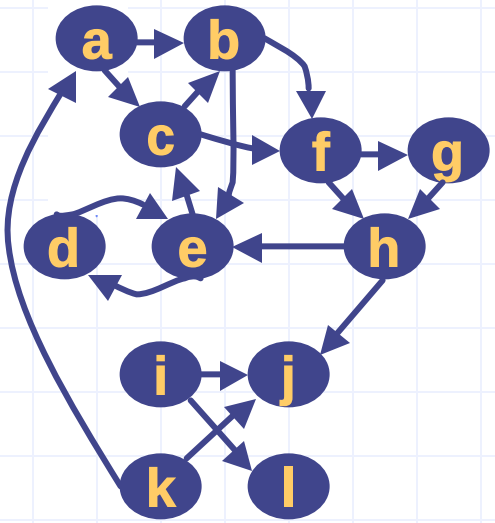
P_B



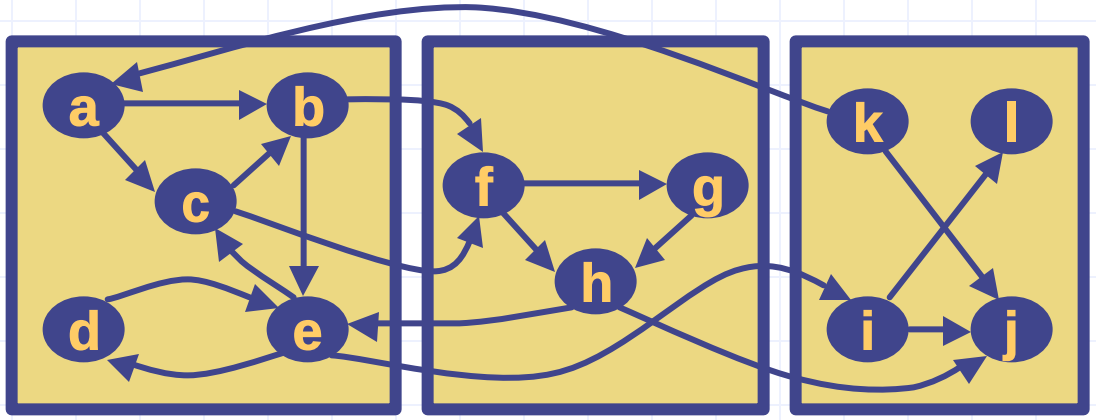
$$\frac{\text{Common Edge Weight}}{\text{Total Edge Weight}} = \frac{10}{19} = 53\%$$

MeCI Example

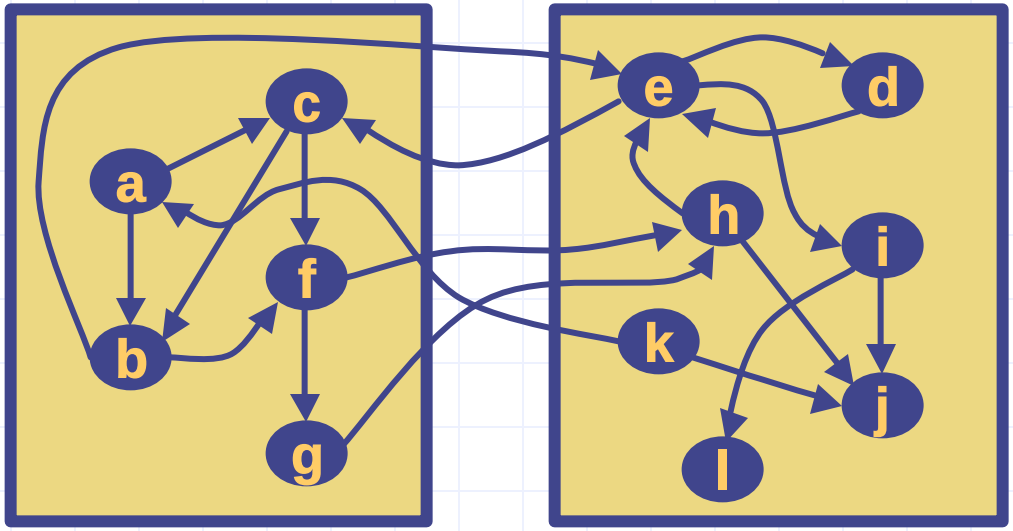
MDG



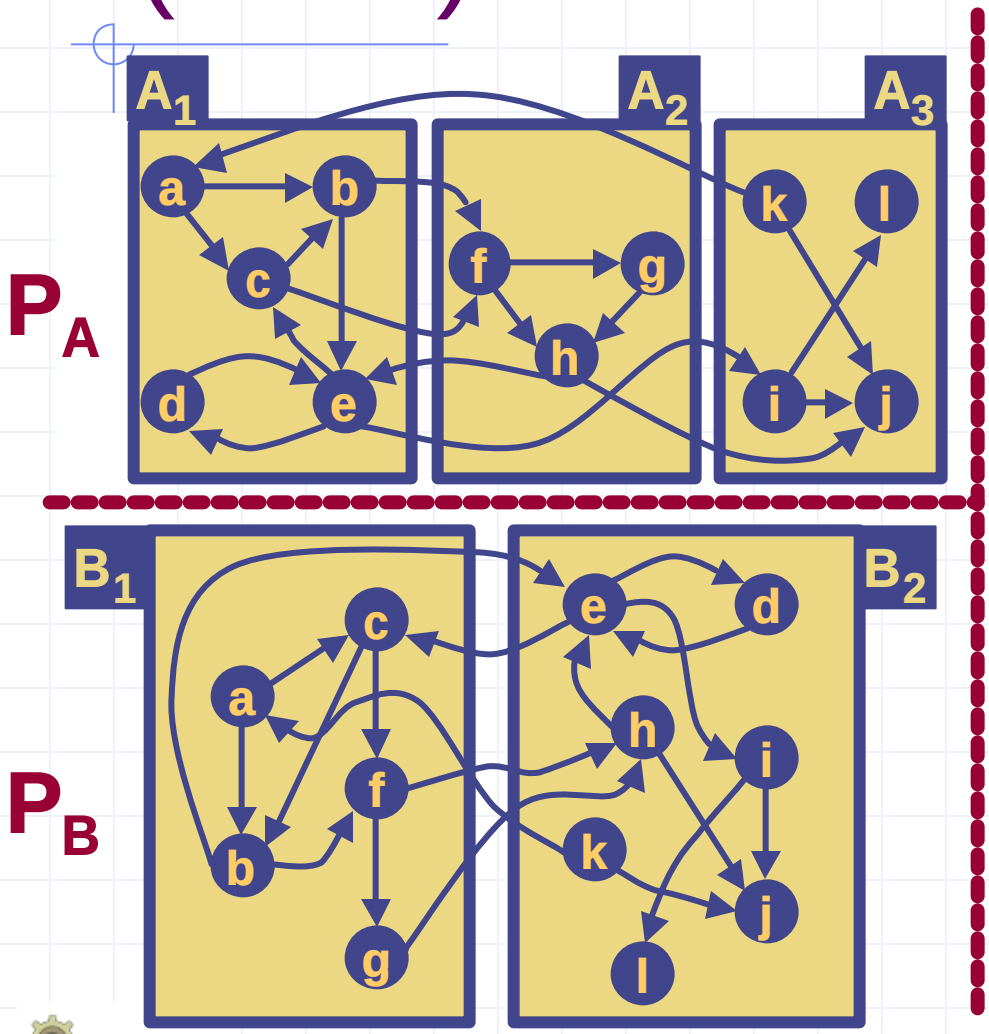
P_A



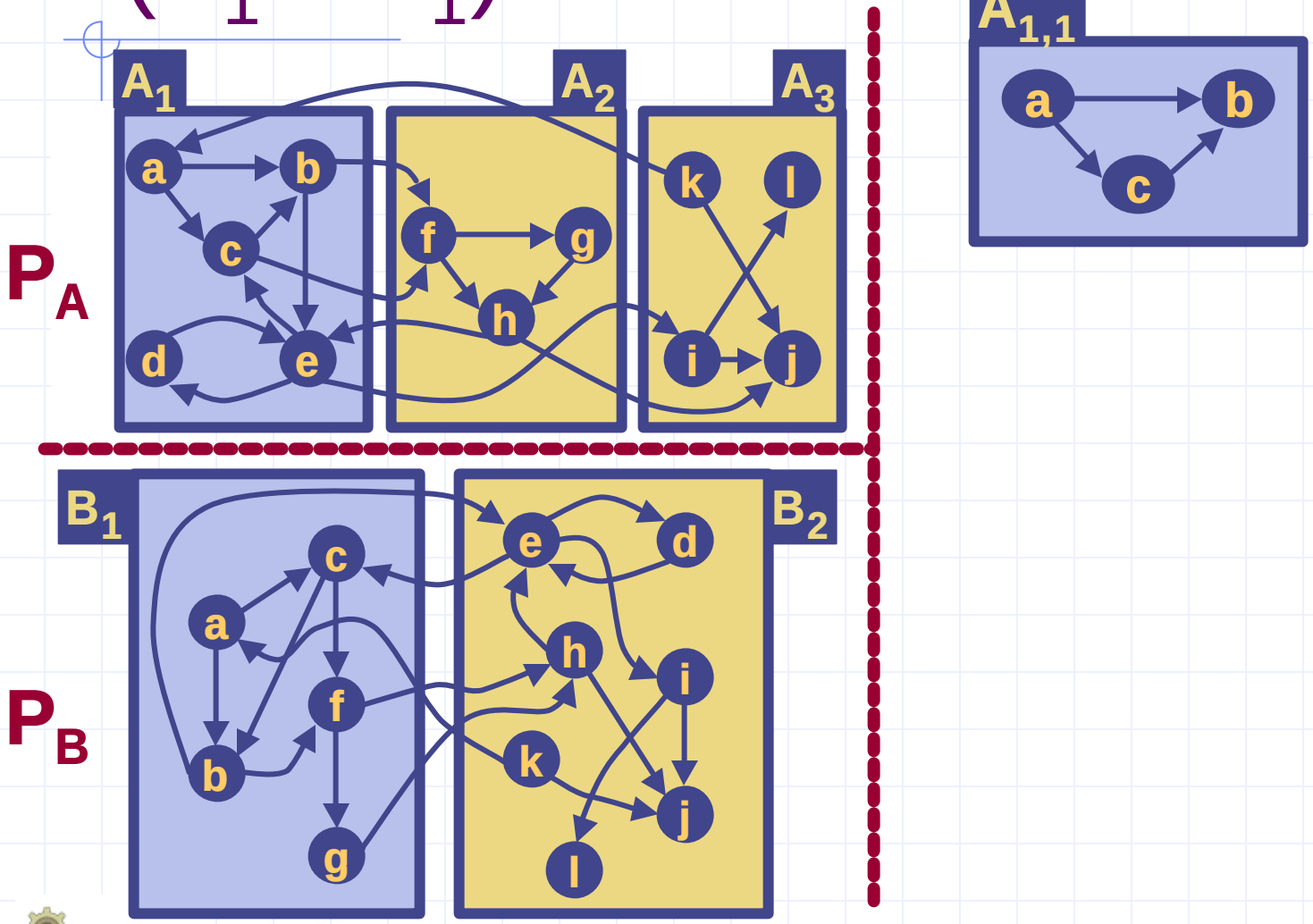
P_B



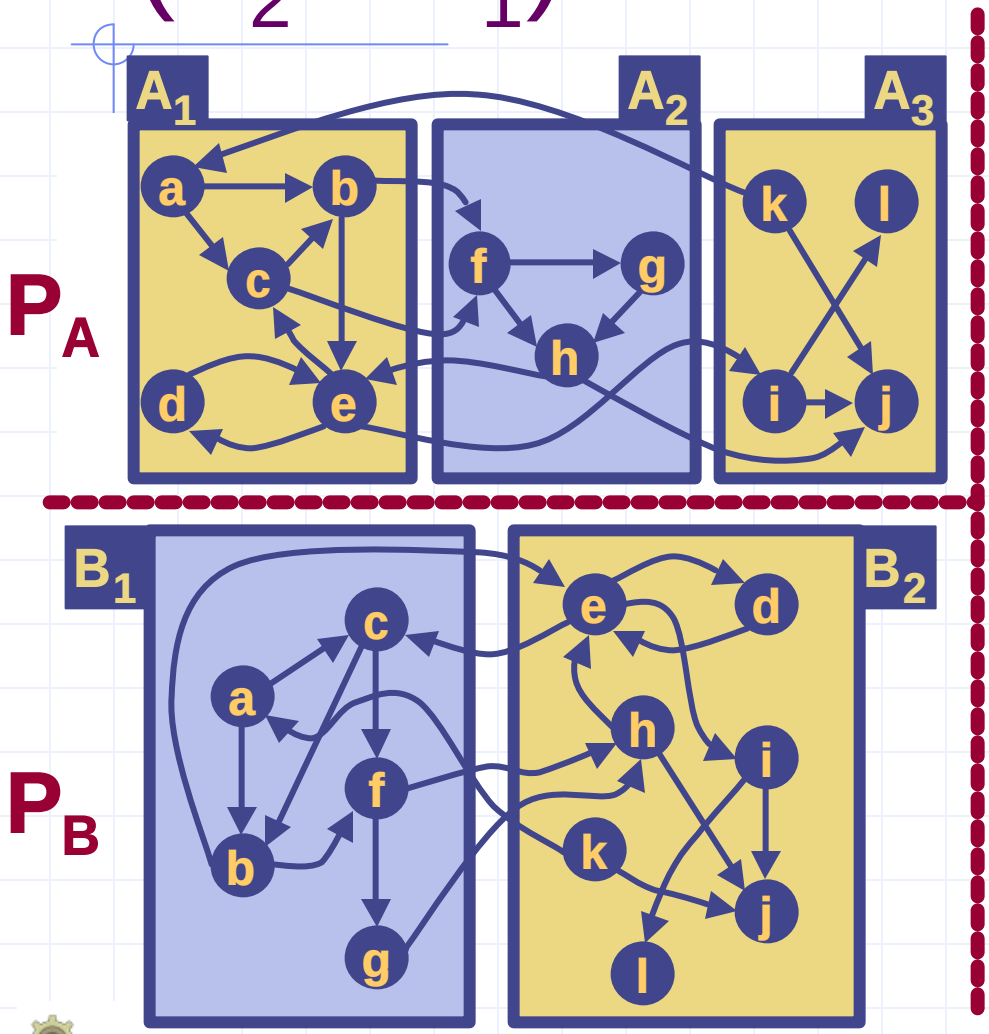
MeCI Example (A→B)

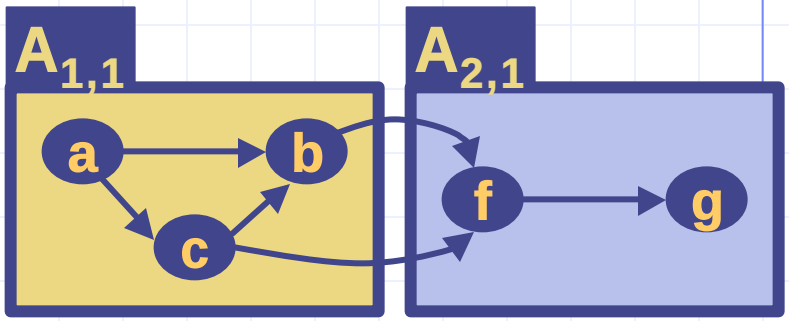


MeCI Example ($A_1 \cap B_1$)

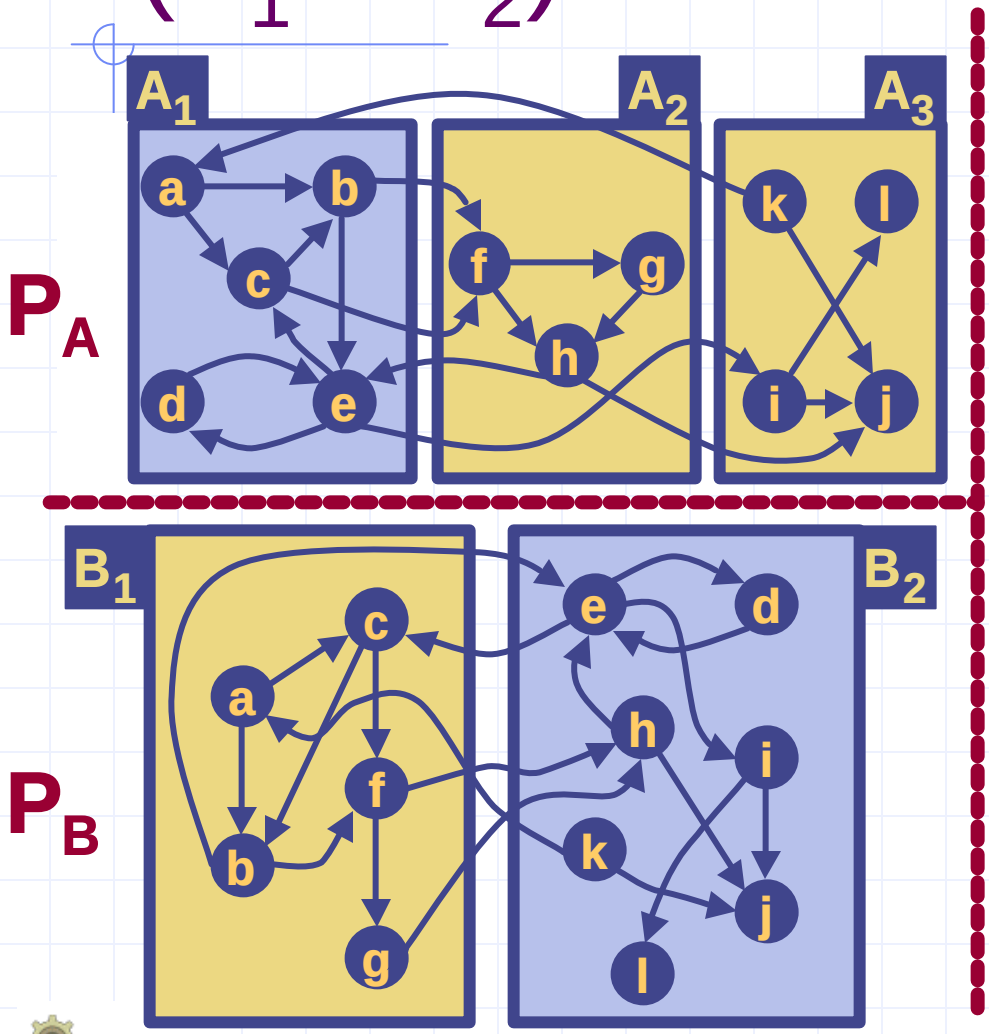


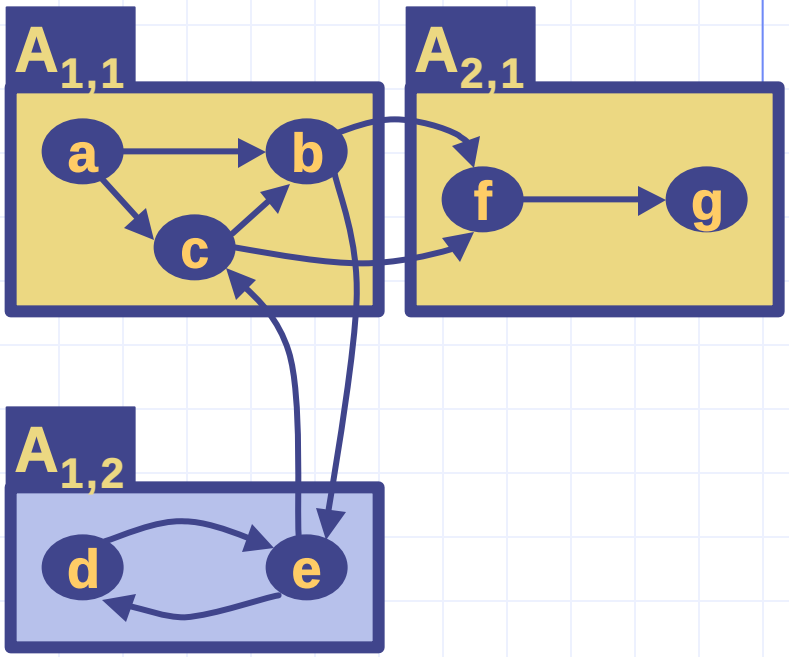
MeCI Example ($A_2 \cap B_1$)



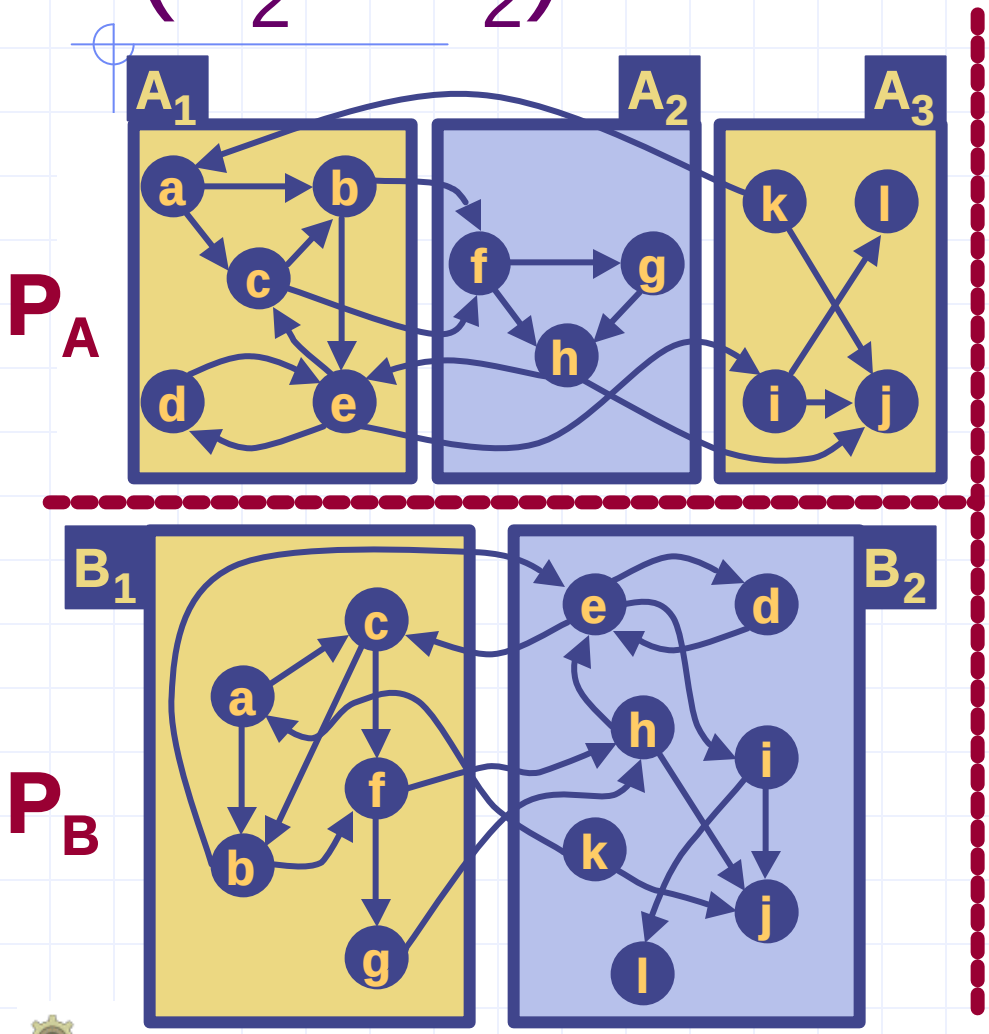


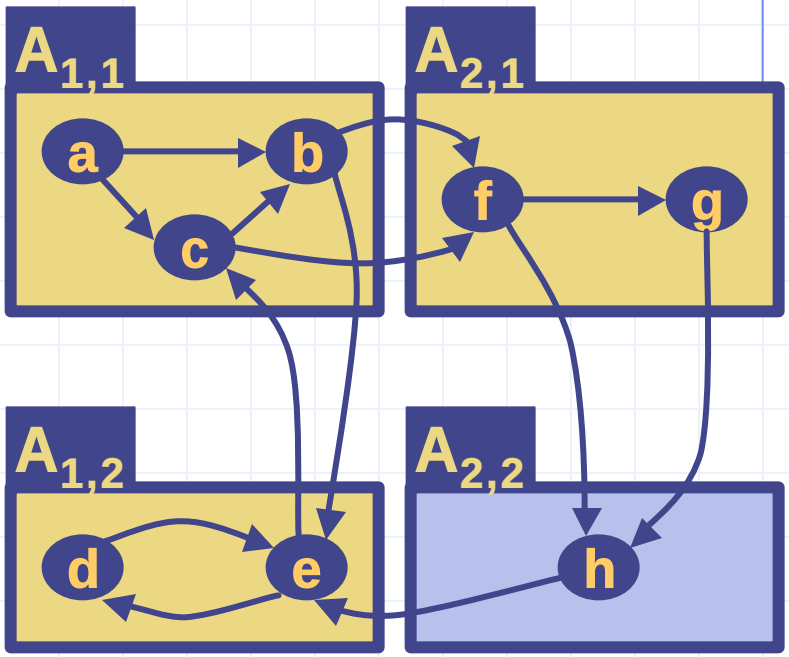
MeCI Example ($A_1 \cap B_2$)



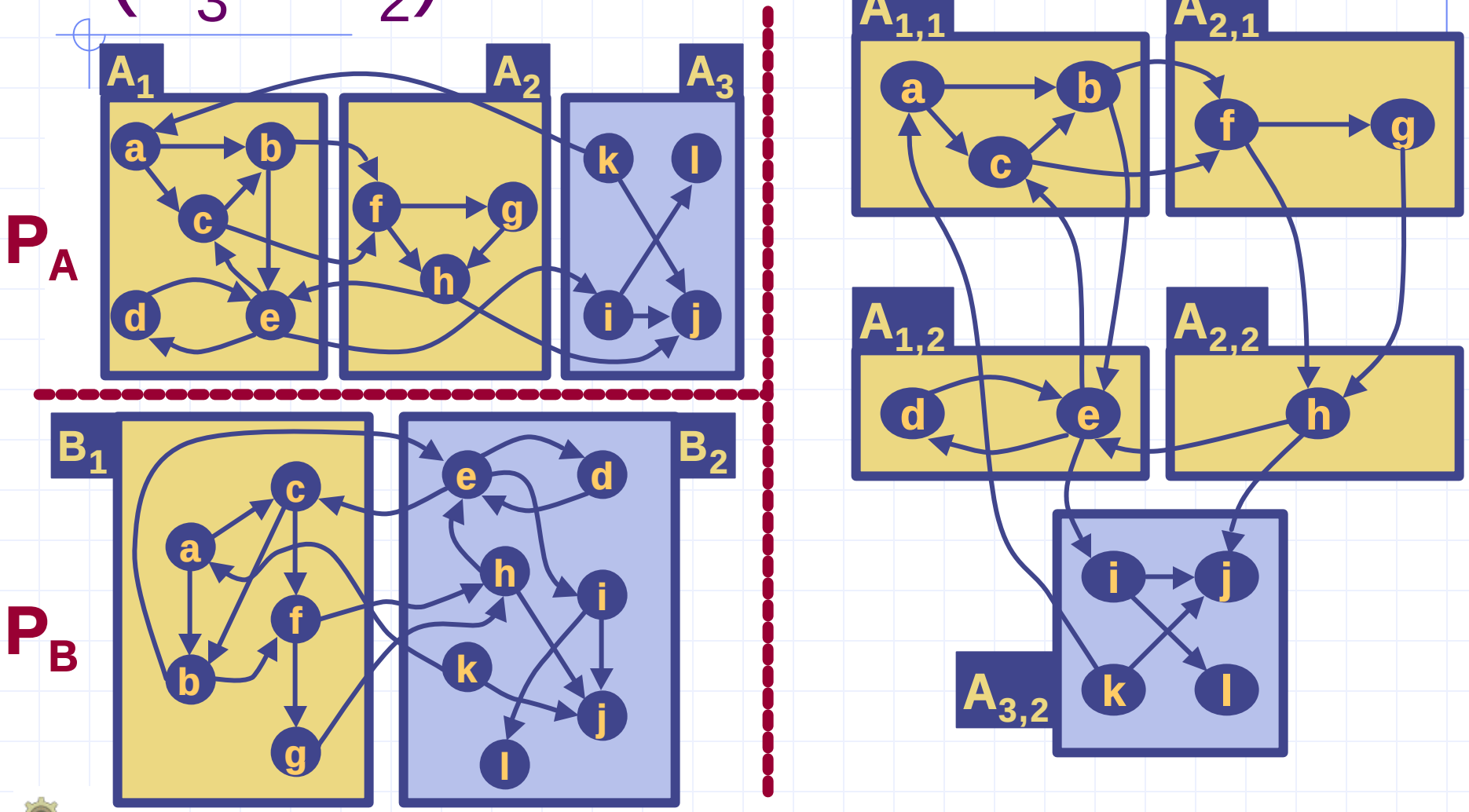


MeCI Example ($A_2 \cap B_2$)

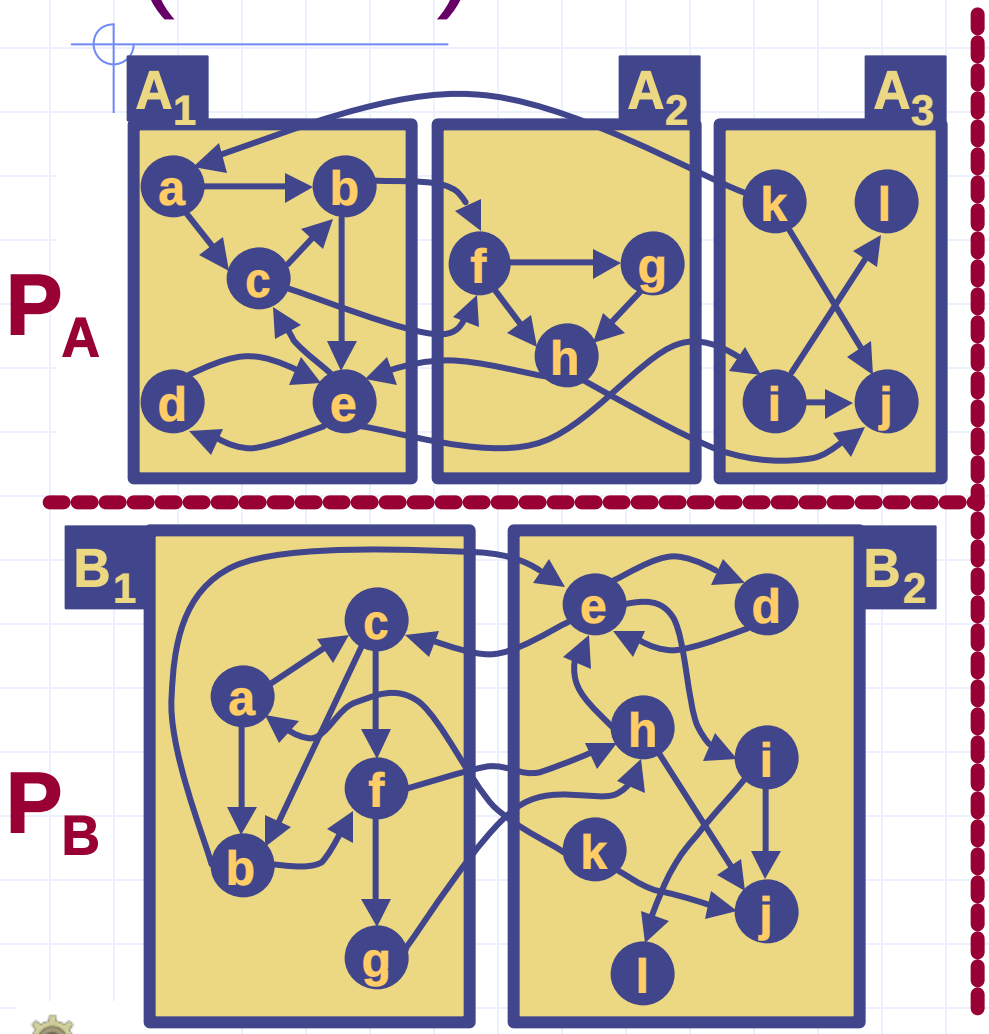


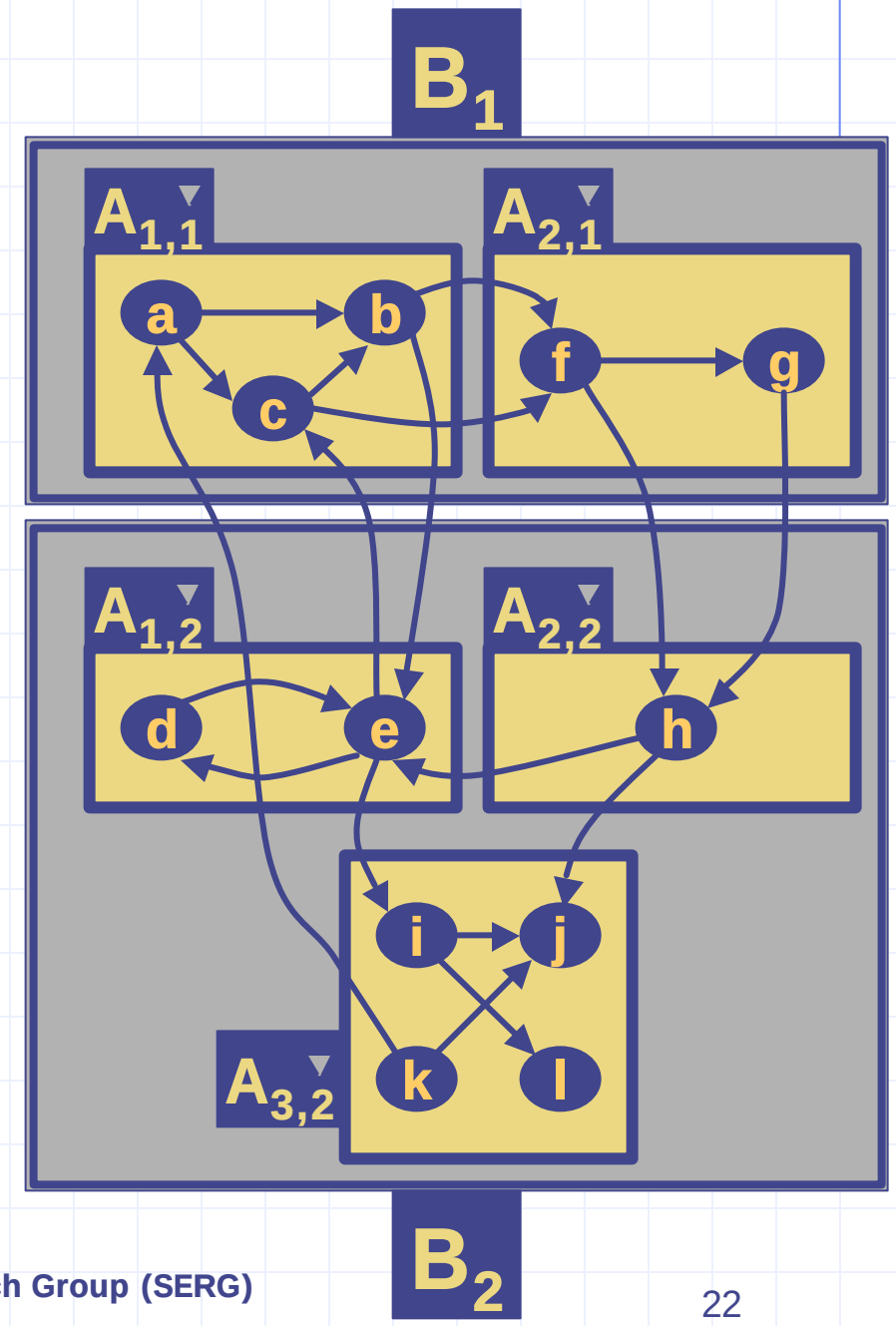


MeCI Example ($A_3 \cap B_2$)



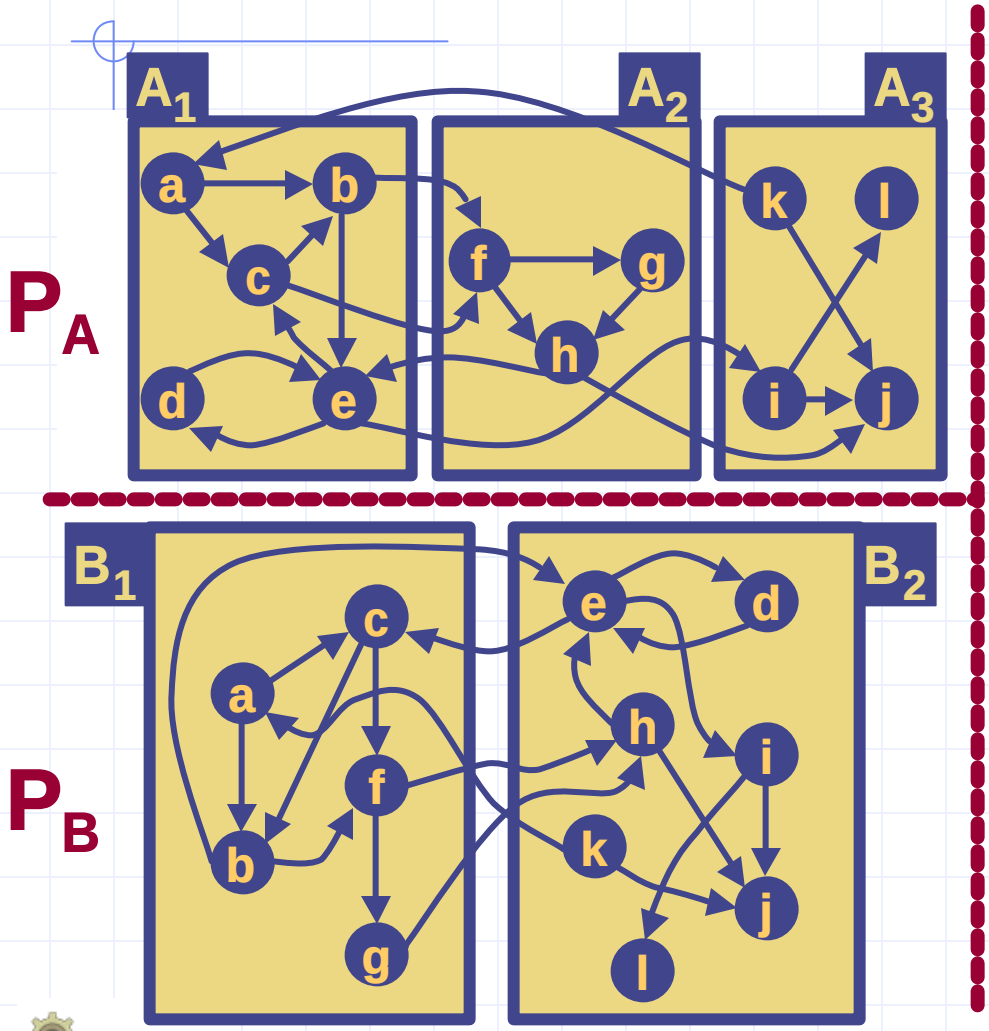
MeCI Example (A→B)

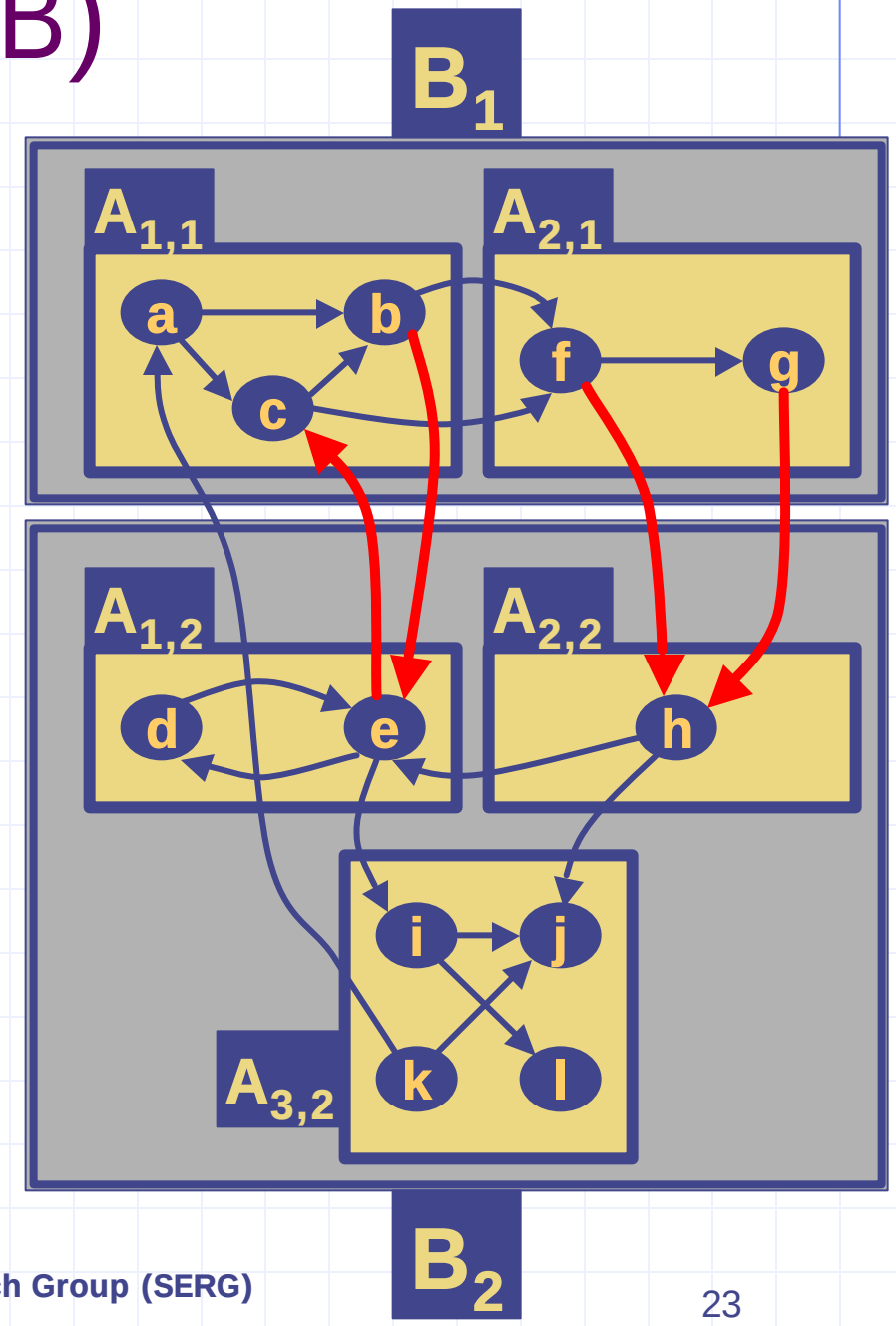




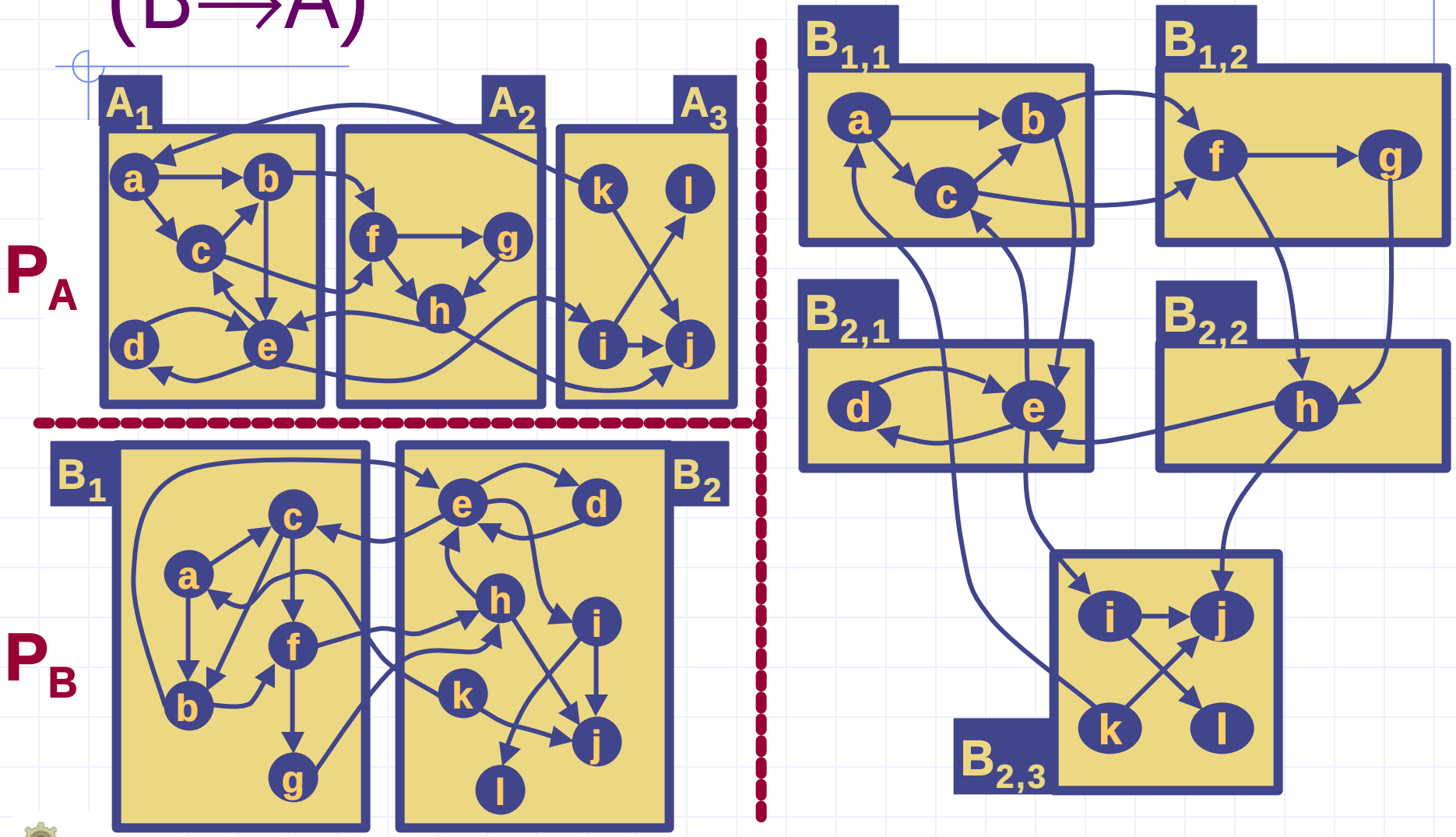
MeCl Example (A→B)

Newly Introduced Inter-Edges

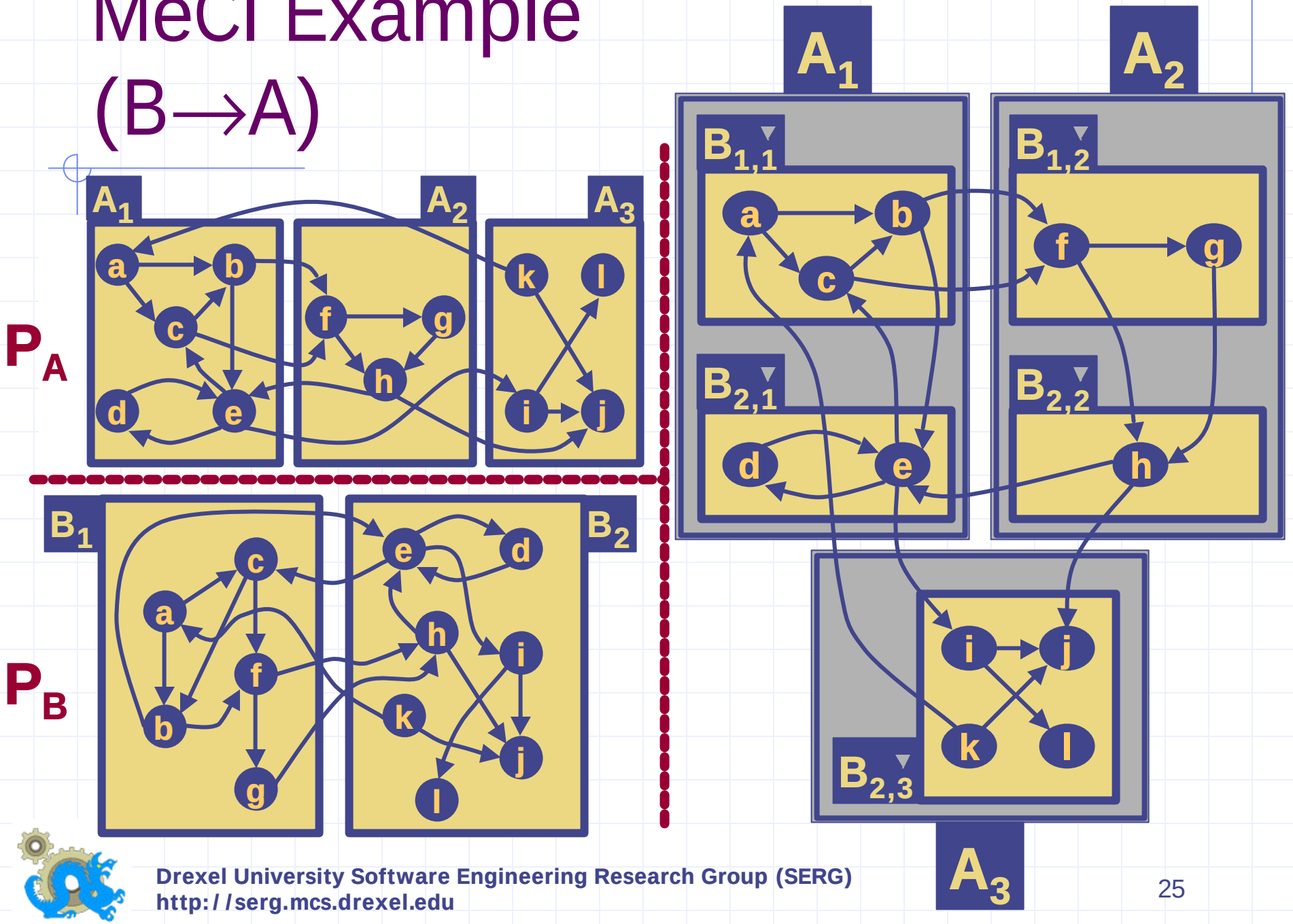




MeCI Example (B→A)

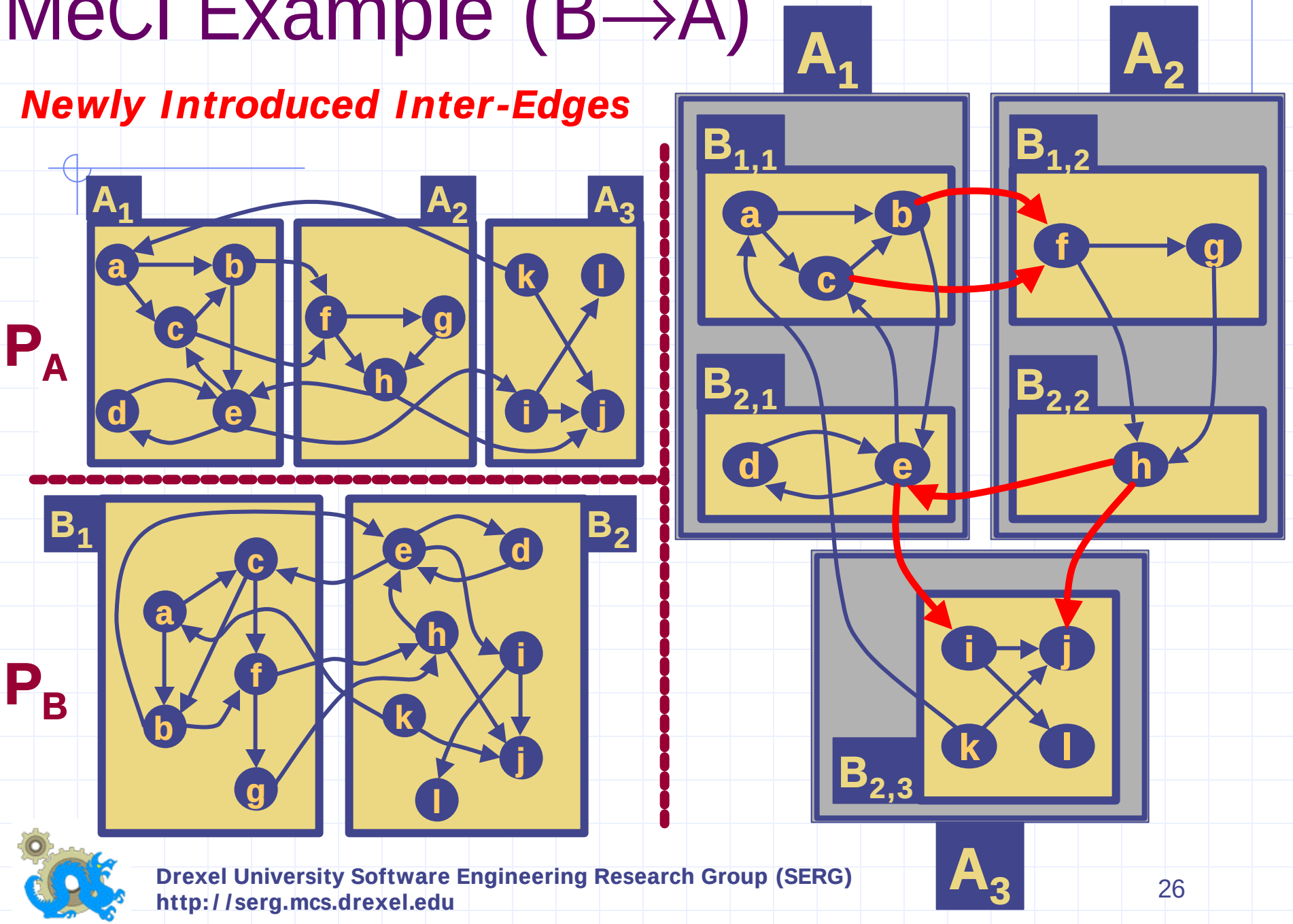


MeCI Example (B→A)

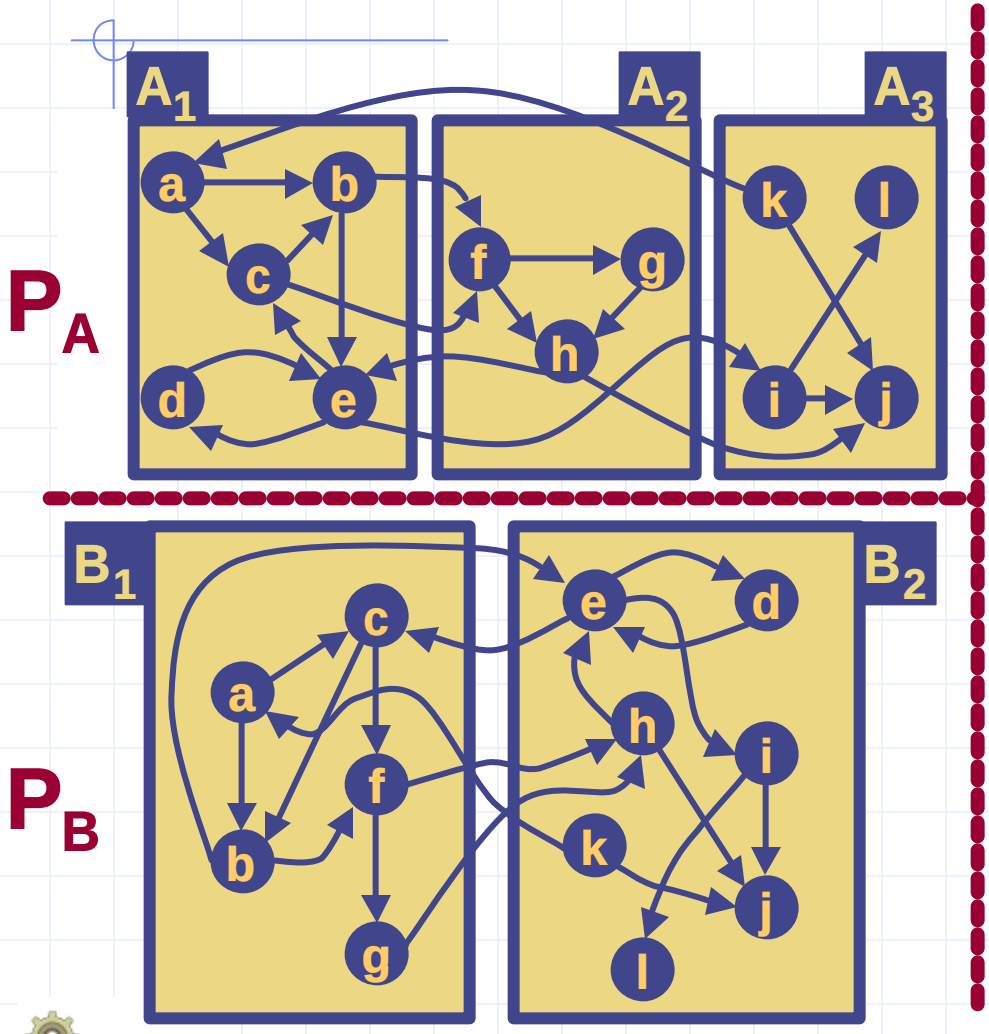


MeCl Example (B→A)

Newly Introduced Inter-Edges



MeCl Calculation



Inter-Edges Introduced

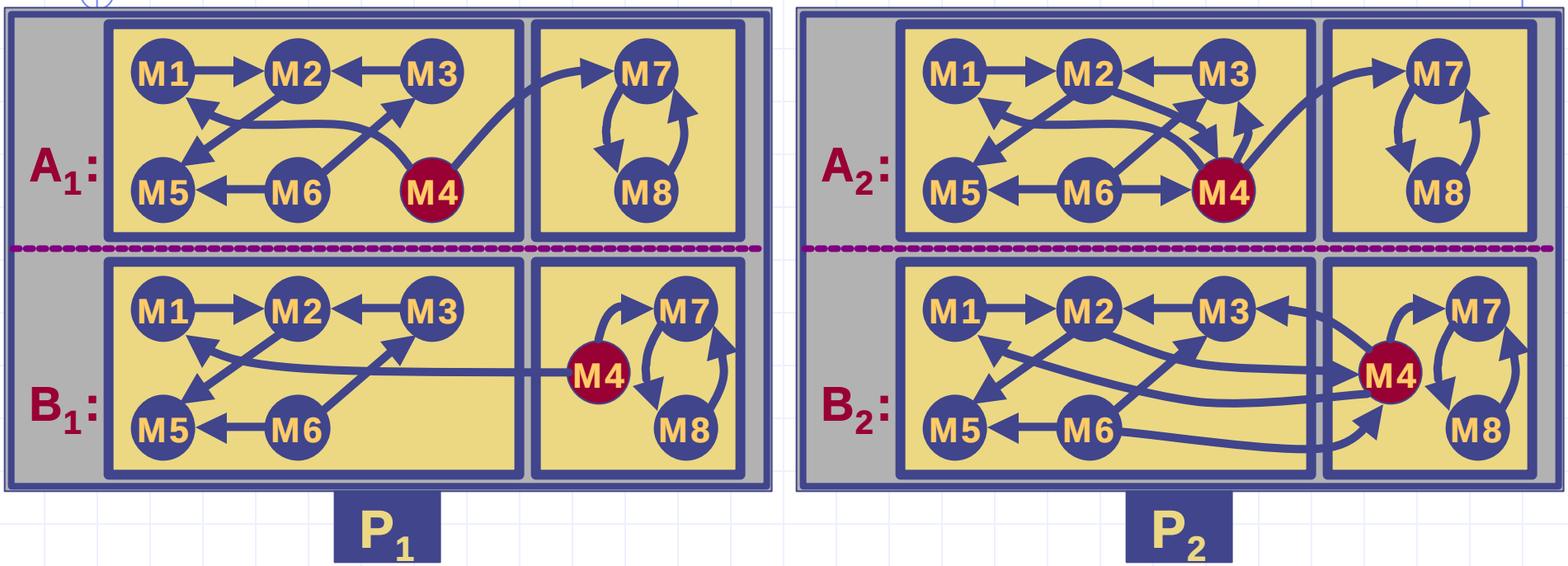
MeCl(A→B):
({b,e},{e,c},{g,h},{f,h})

MeCl(B→A):
({e,i},{h,j},{b,f},{c,f},{h,e})

$$\textbf{MeCl} = \left[1 - \frac{\max_W(M_{A\to B}, M_{B\to A})}{\text{Total Edge Weight}}\right]$$

$$\textbf{MeCl} = \left[1 - \frac{5}{19}\right] = \textbf{73.7\%}$$

Similarity Measurement Recap

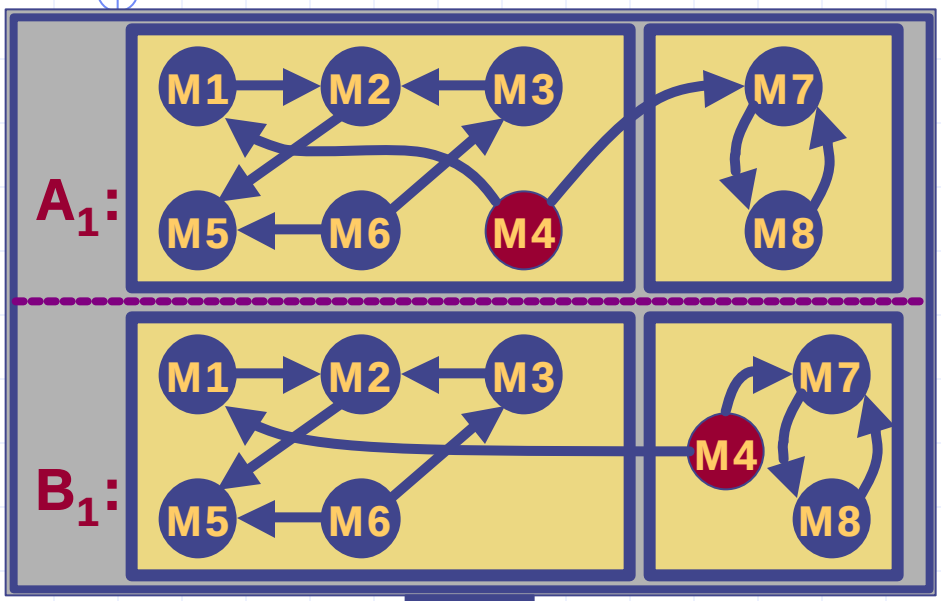


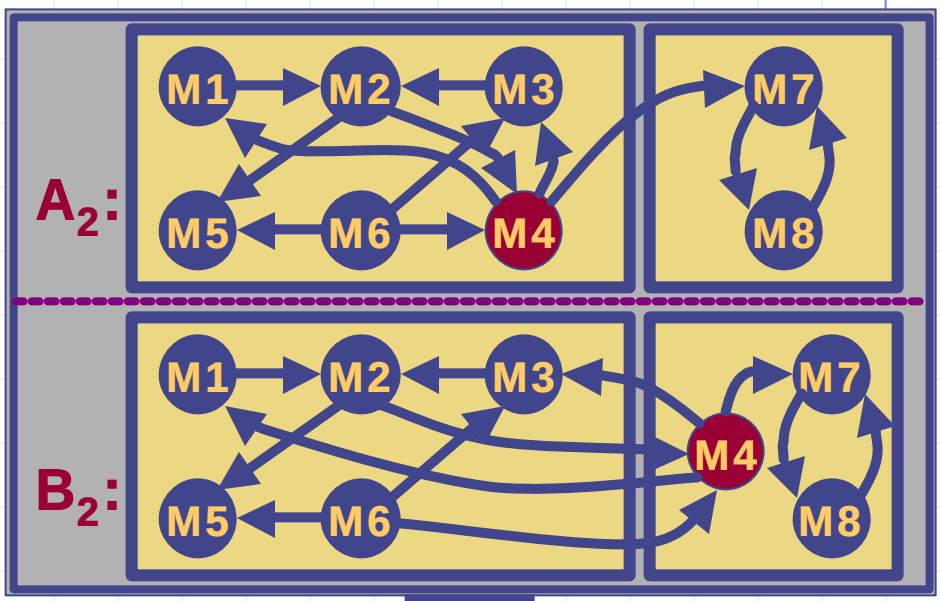
$MoJo(P_1) = MoJo(P_2) = 87.5\%$

$PR(P_1) = PR(P_2) = P{:}84.6\%, R{:}68.7\%, AVG_{PR}{=}76.7\%$

Conclusion… P1 is equally similar to P2

Similarity Measurement Recap





$EdgeSim(P_1) = 77.8\%$ $EdgeSim(P_2) = 58.3\%$

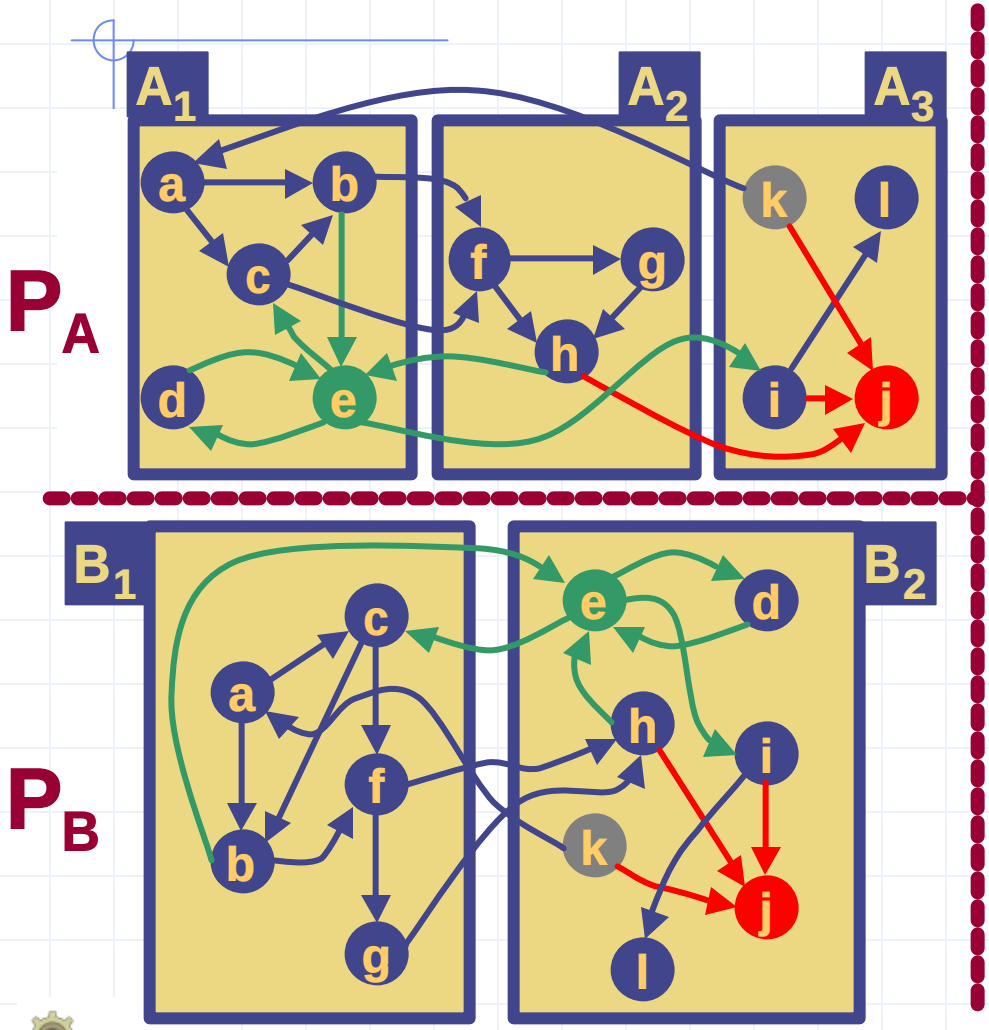
$MeCl(P_1) = 88.9\%$ $MeCl(P_2) = 66.7\%$

Conclusion… P1 is more similar than P2

Summary: EdgeSim & MeCl

- EdgeSim:
  - Rewards clustering algorithms for preserving the edge types
  - Penalizes clustering algorithms for changing the edge types
- MeCl:
  - Rewards the clustering algorithm for creating cohesive “subclusters”

Special Modules

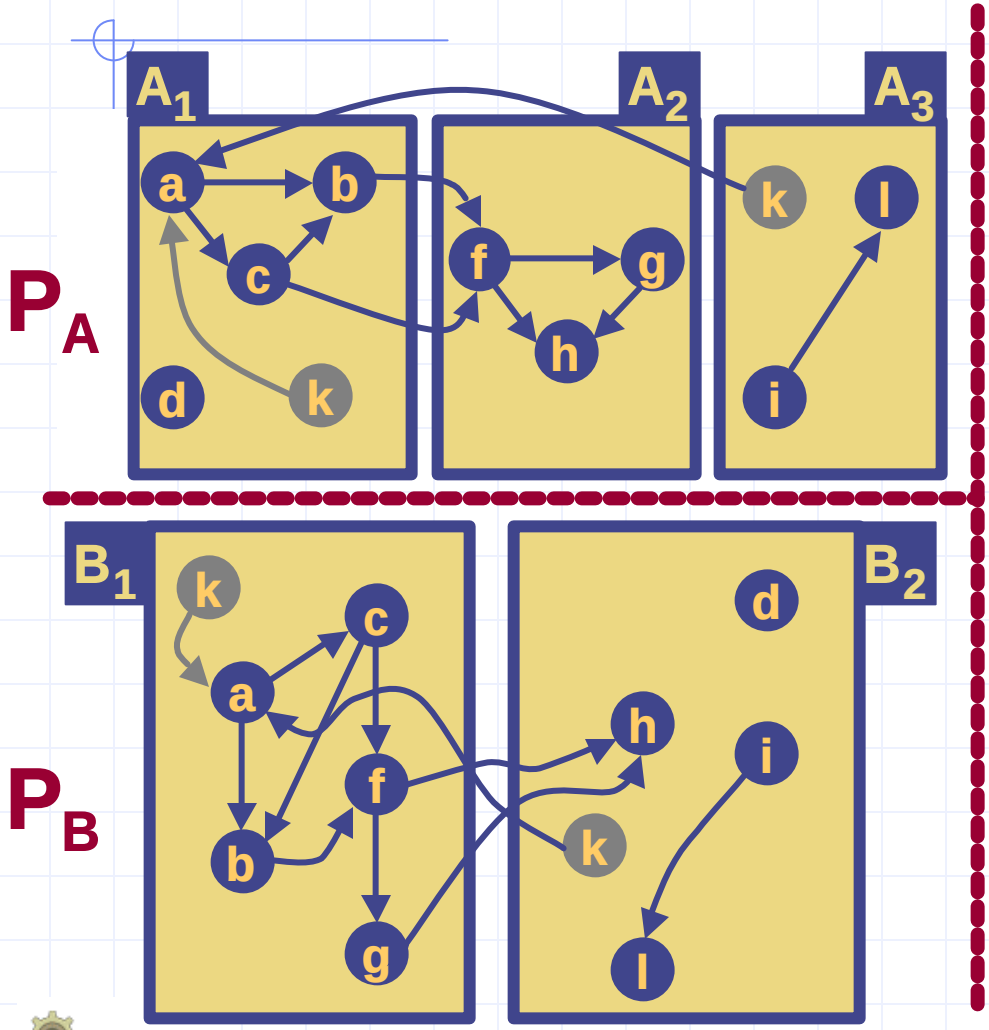


Omnipresent Modules:
"Strong" Connection to other Modules

Library Modules:
Always used by other modules, never use other modules

Isomorphic Modules:
Modules equally connected to other subsystems

Special Modules



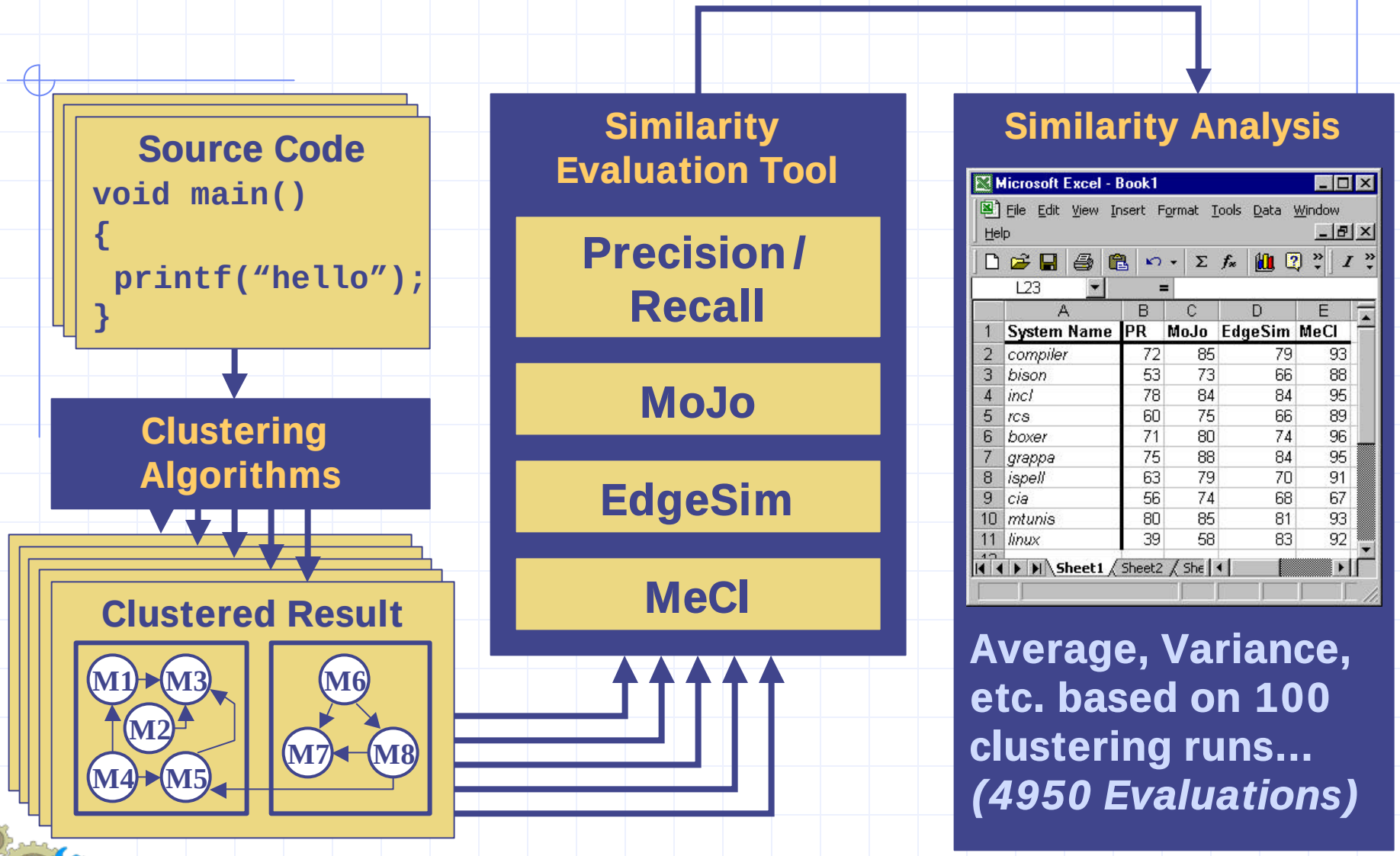
Special Treatment of Special Modules helps to determine the Similarity

Omnipresent Modules:
Removed

Library Modules:
Removed

Isomorphic Modules:
Replicated

Case Study Overview



Case Study Observations

- All similarity measurements exhibit consistent behavior for the systems studied

  <u>For all systems examined:</u>
  If $MeCl(S_A) < MeCl(S_B)$ then $MoJo(S_A) < MoJo(S_B)$, $PR(S_A) < PR(S_B)$, and $EdgeSim(S_A) < EdgeSim(S_B)$

- Removal of “special” modules improved all similarity measurements
- Treating isomorphic modules specially only improved similarity slightly
- EdgeSim and MeCl produced higher and less variable similarity values then Precision/Recall and MoJo

Questions

- Special Thanks To:
  - AT&T Research
  - Sun Microsystems
  - DARPA
  - NSF
  - US Army