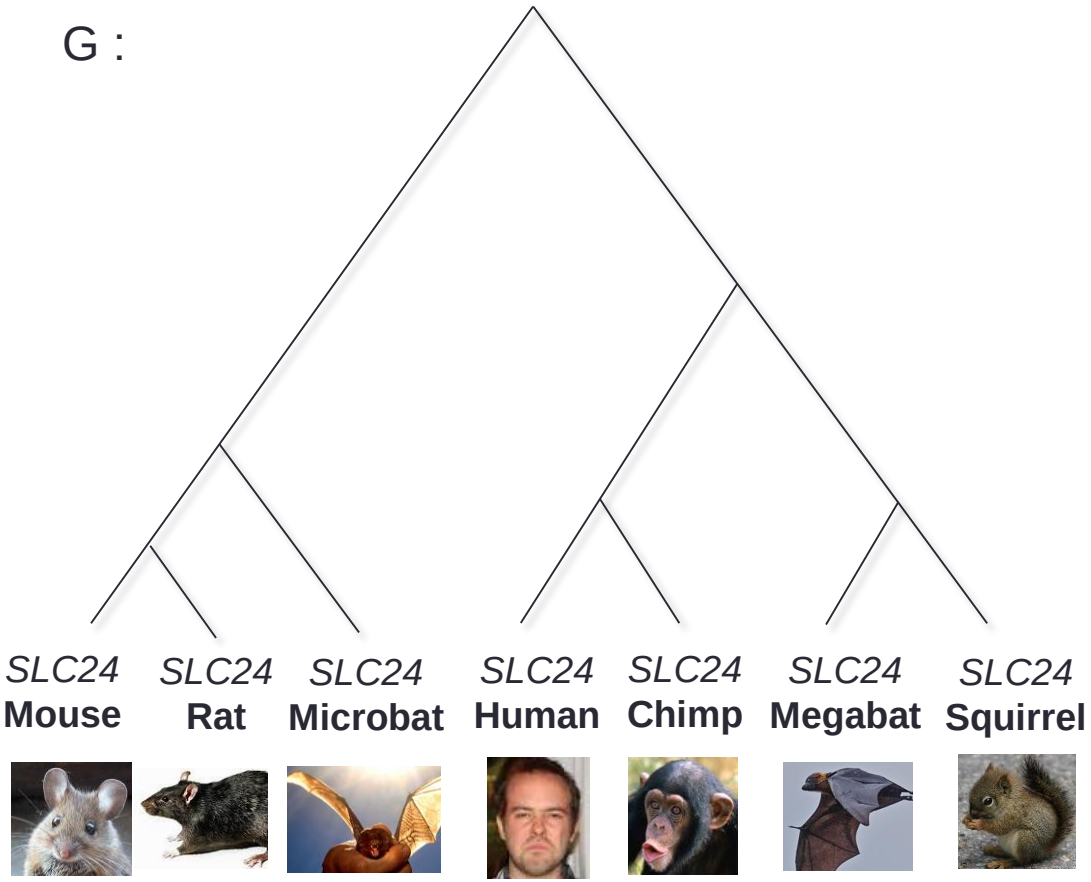
POLYTOMY REFINEMENT FOR THE CORRECTION OF DUBIOUS DUPLICATIONS IN GENE TREES

Manuel Lafond[1], Cedric Chauve[2,3], Riccardo Dondi[4], Nadia El-Mabrouk[1]

[1] Université de Montréal, Canada
[2] Université Bordeaux 1, France
[3] Simon Fraser University, Canada
[4] Universitá degli Studi di Bergamo, Italy

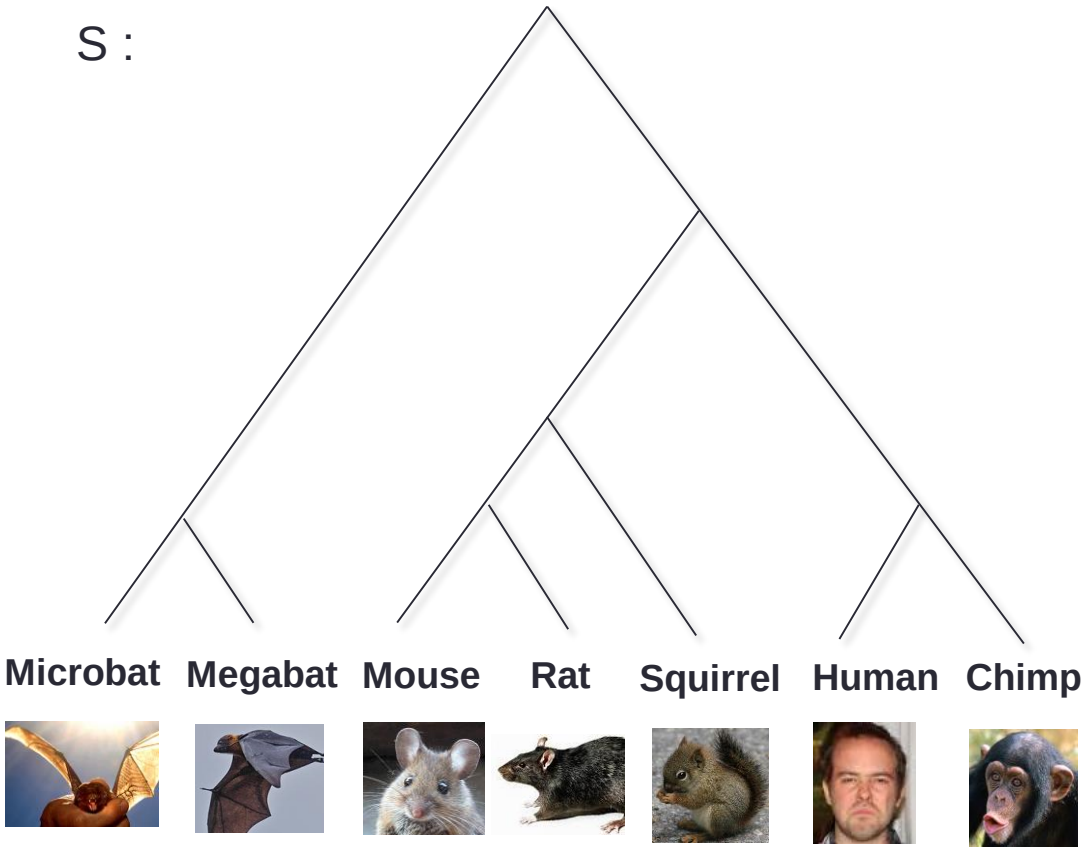
Introduction

- **Gene tree** for the SLC24a2 gene family (solute carrier 24)



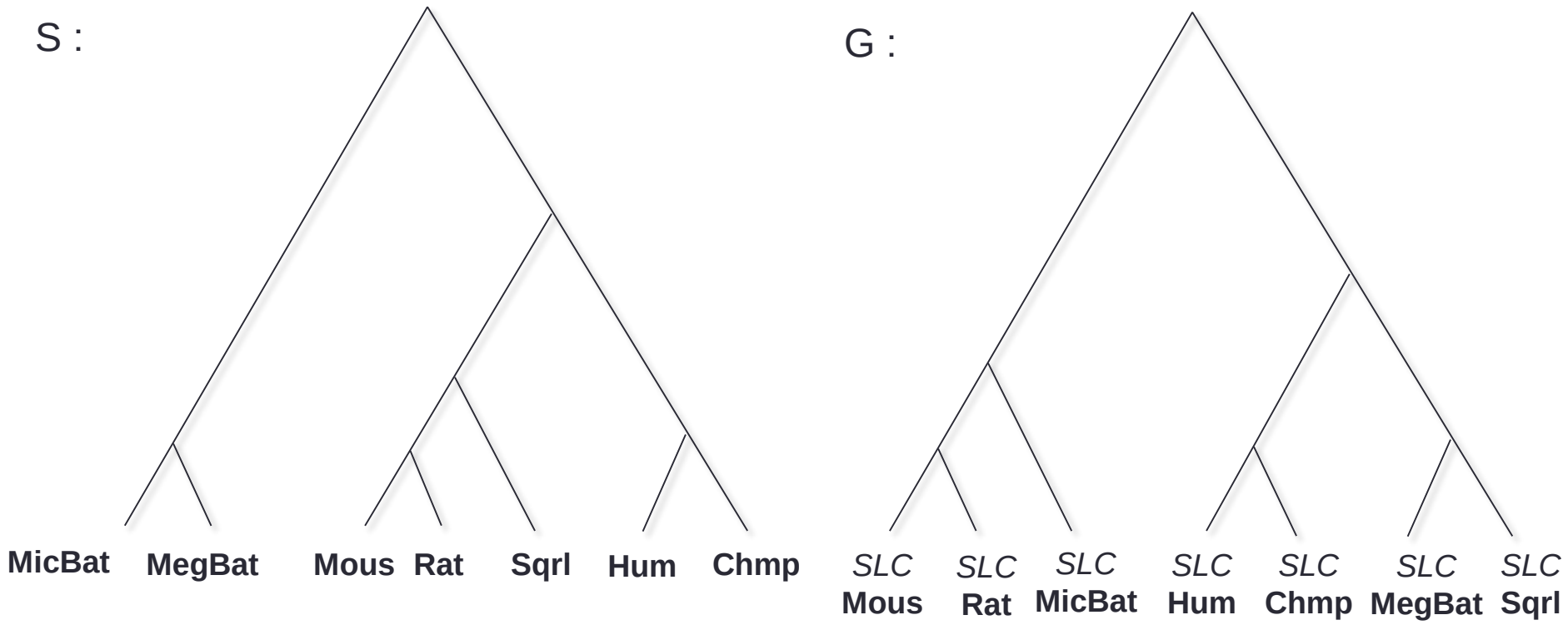
Introduction

- **Species tree** for the species having a gene in G.



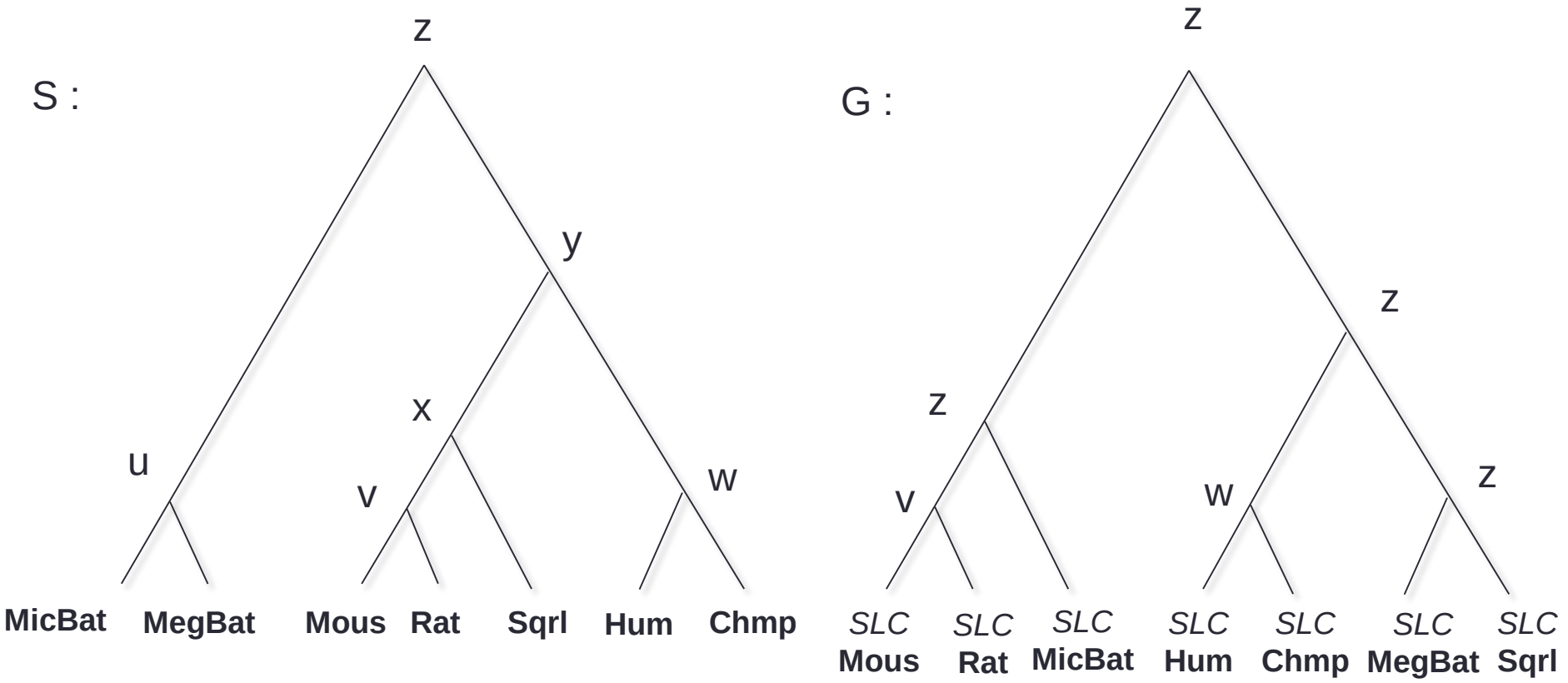
Introduction

- **G** and **S disagree**



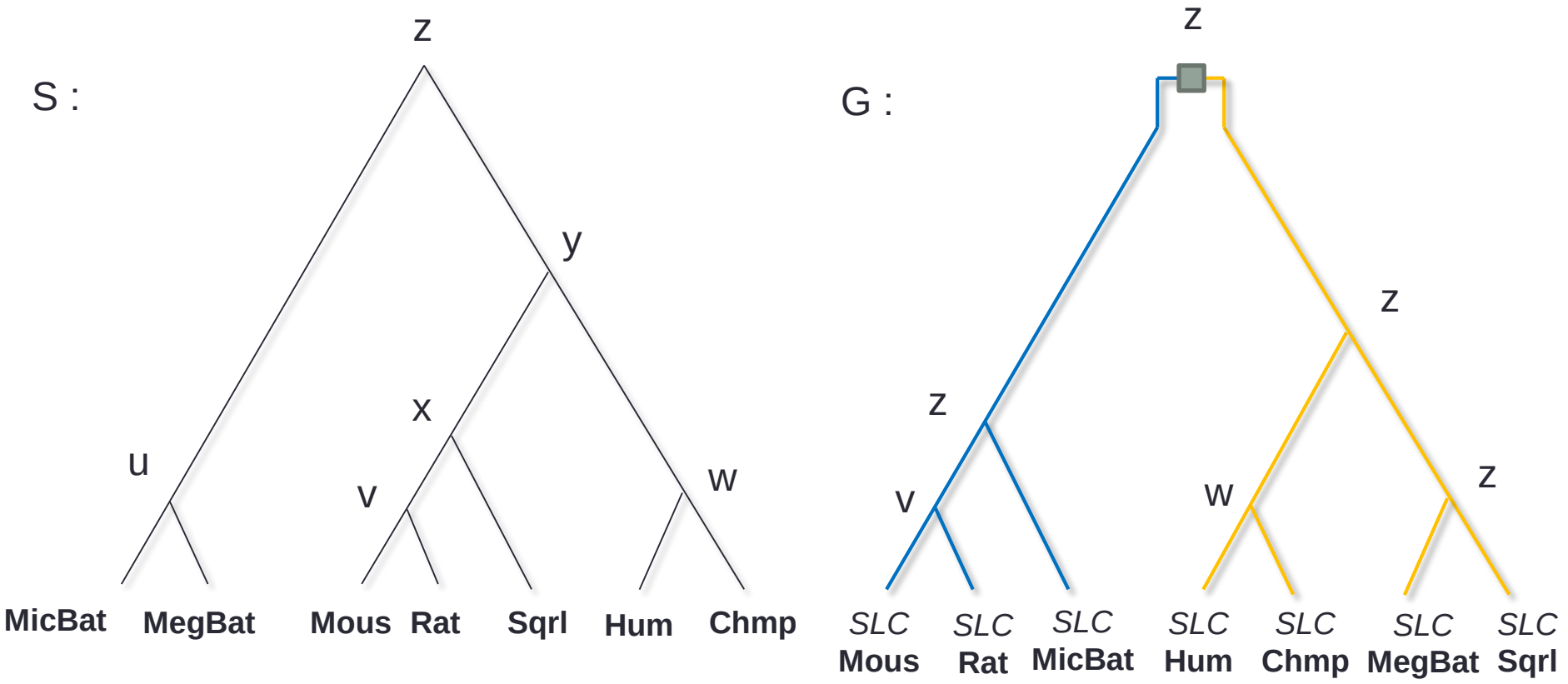
Introduction

- **LCA MAPPING :** associate each ancestral gene with the species it belonged to



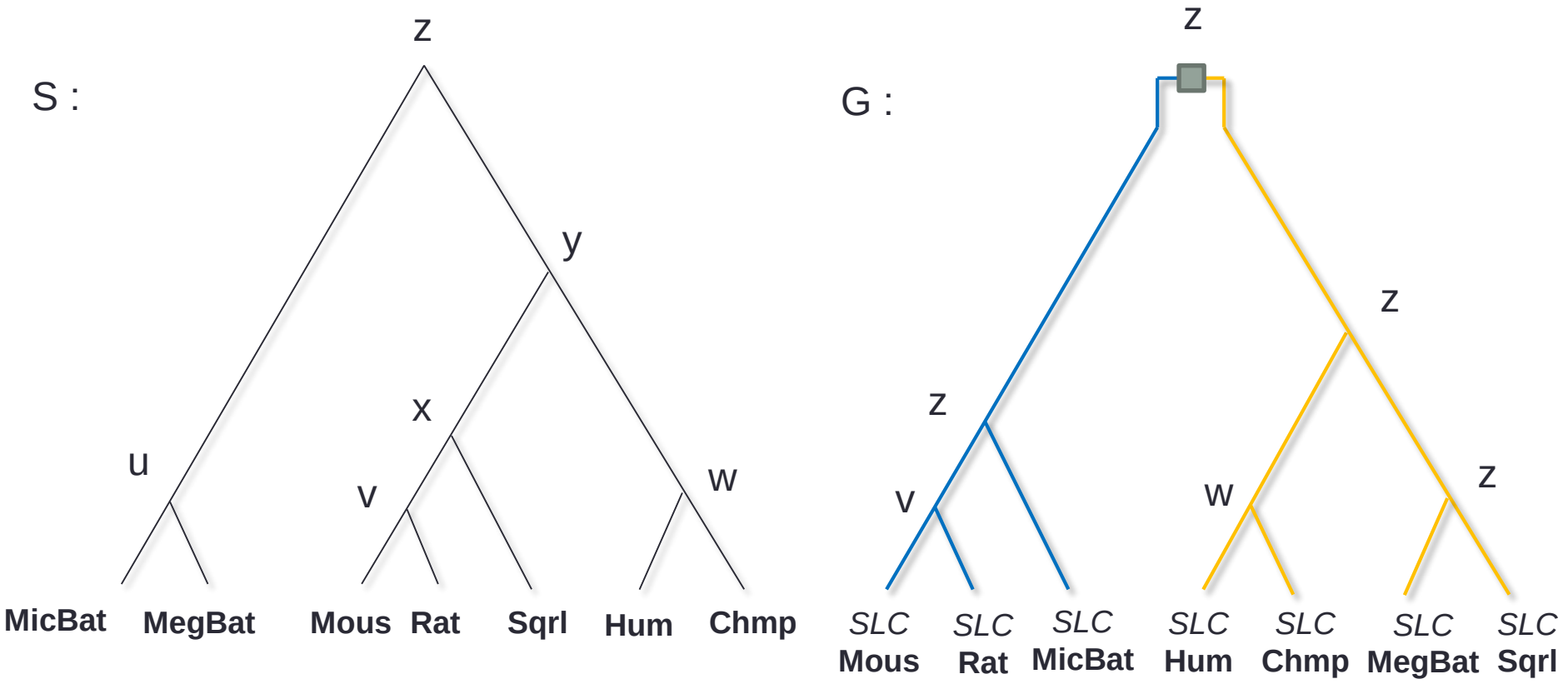
Introduction

- **G** and **S disagree => Duplication** of an ancestral gene



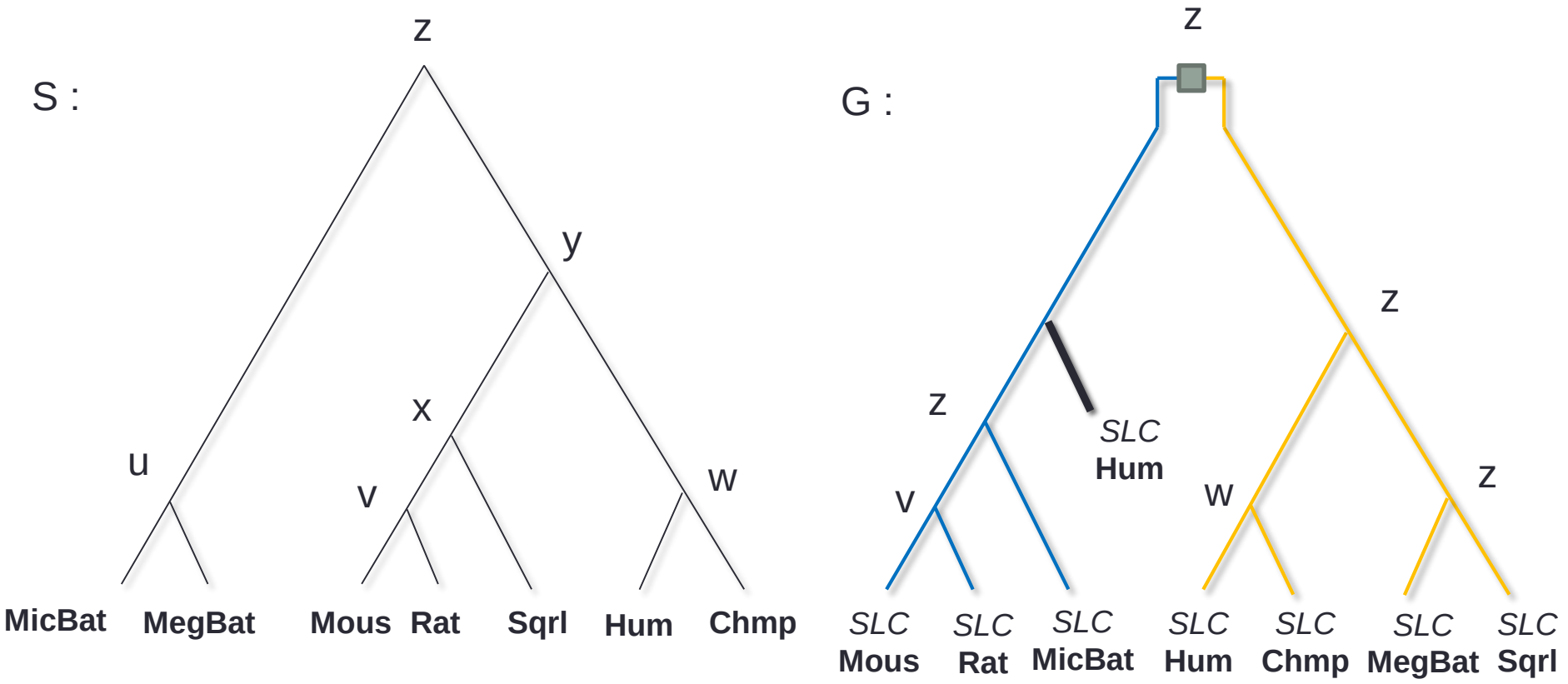
Introduction

- Extant species are expected to have **2 copies** of the gene
- None of them do. That's **dubious** !



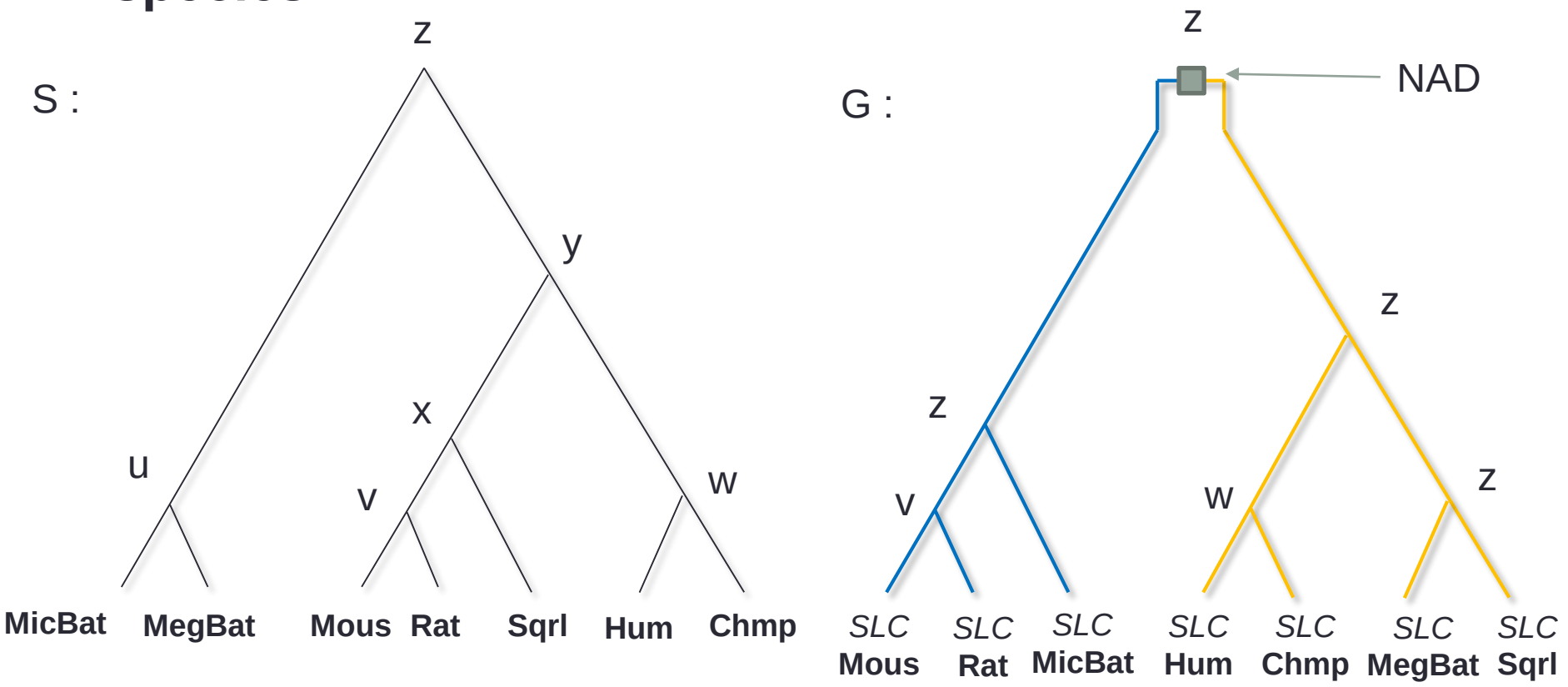
Introduction

- If some species was represented on both sides of the duplication, it would be an **Apparent Duplication (AD)**



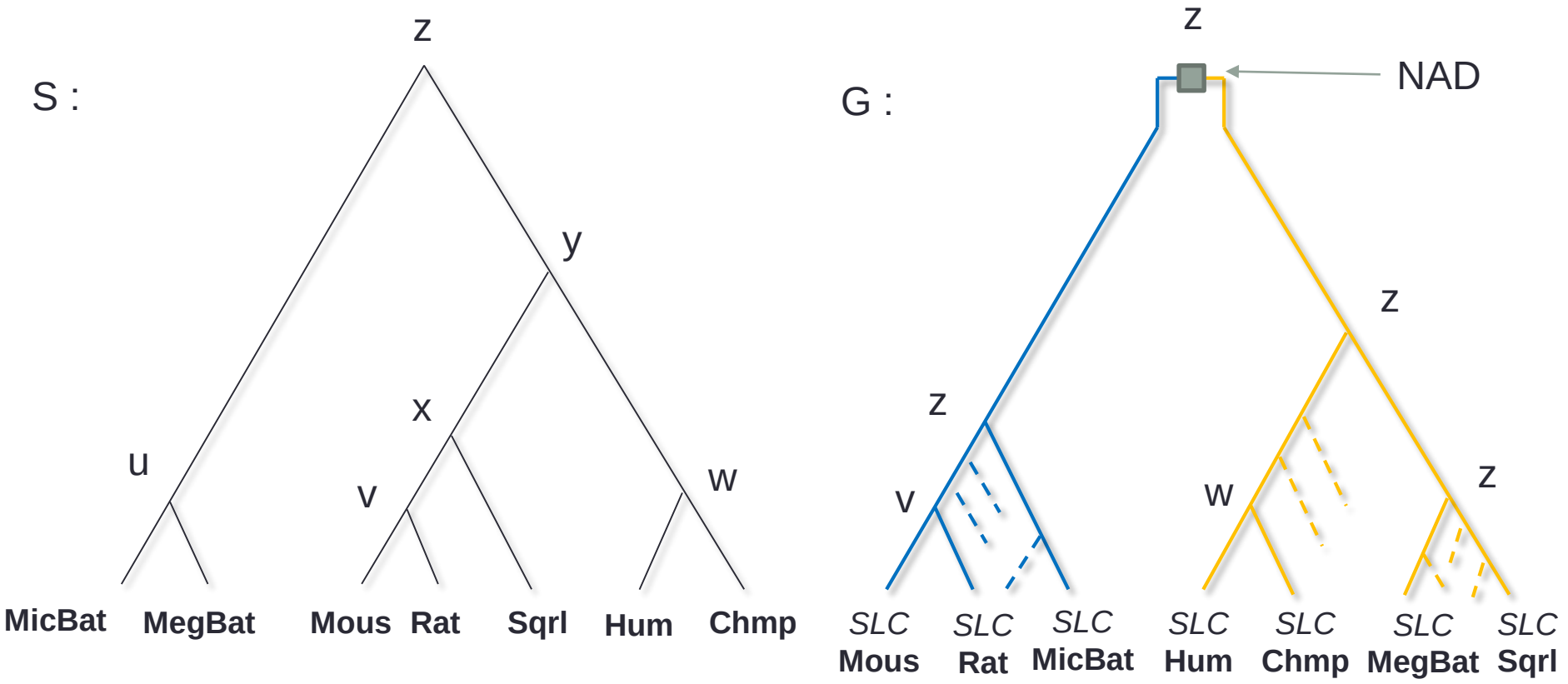
Introduction

- **Non-apparent duplication (NAD)** : the left and right subtrees of the duplication share no gene from the **same species**.



Introduction

- Missing gene copies must have been **lost** sometime ago.
- NADs usually imply a bunch of losses.



Introduction

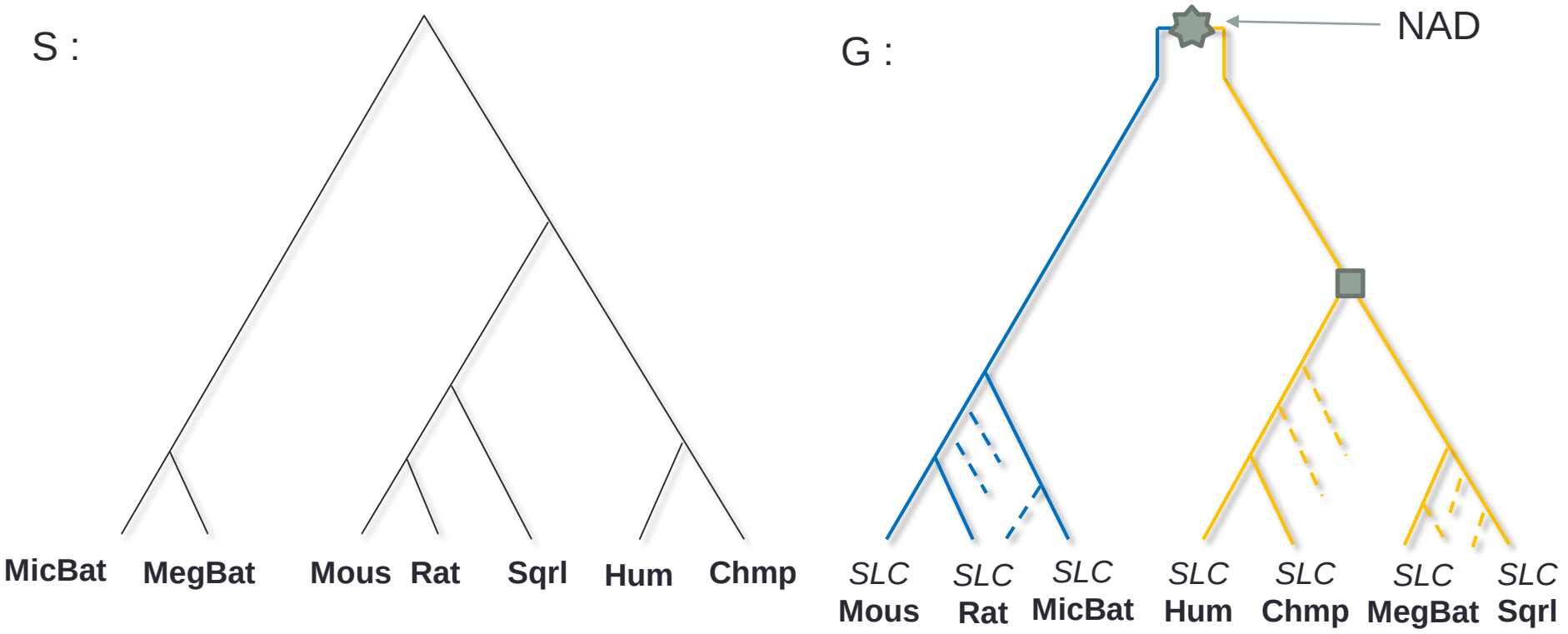
- NADs are called **dubious**, or **ambiguous** duplications in the Ensembl database.

- About **44%** of duplication nodes are dubious.
  - The SLC24 gene tree has **32** duplication nodes, **24** of which are dubious.

- Simulations showed that only **5%** percent of duplications were actually NADs (Chauve & Mabrouk, 2009).

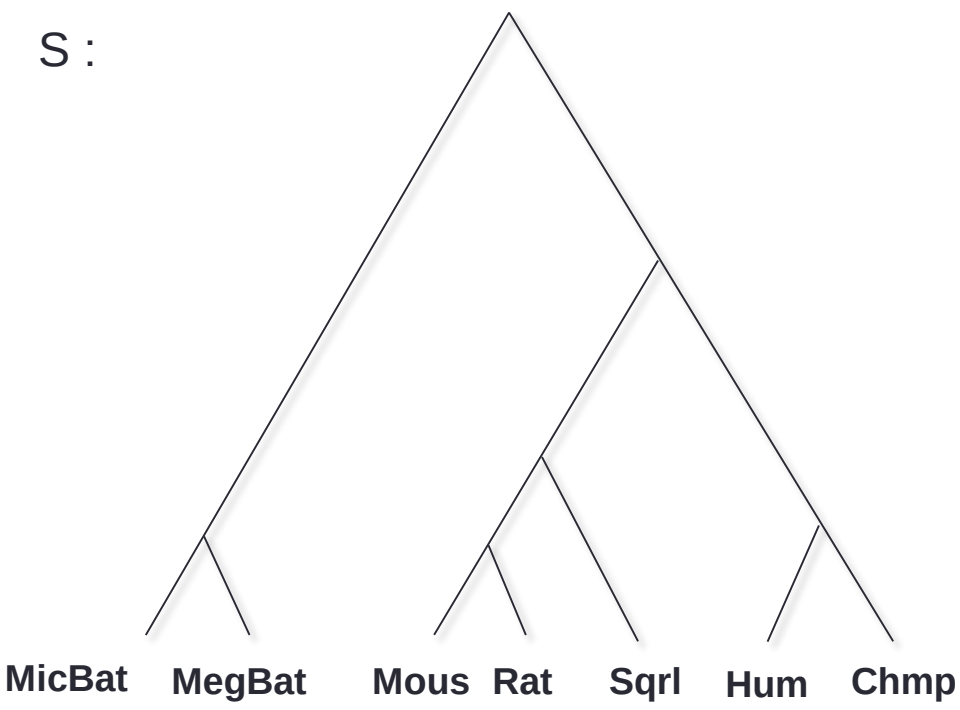
Introduction

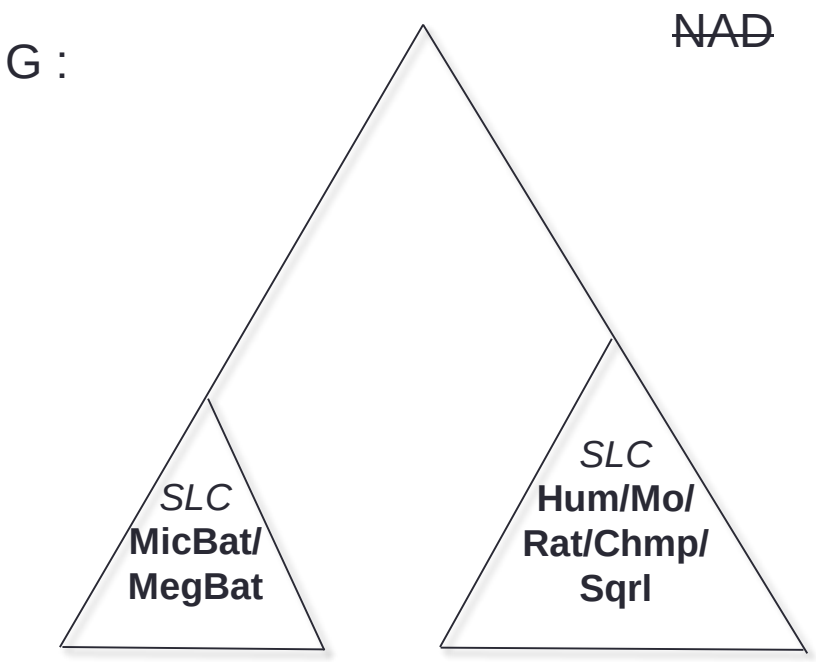
- Alternative scenario for the root of G : no duplication occurred.



Introduction

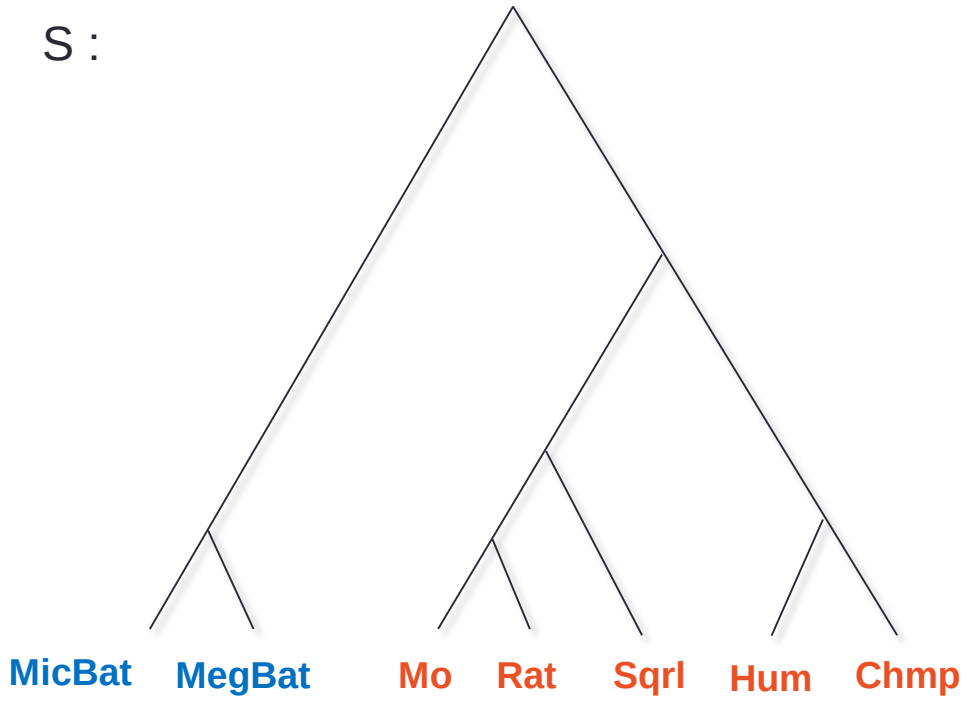
- Alternative scenario for the root of G : no duplication occurred => **speciation** => the bat genes should be separated from the others.

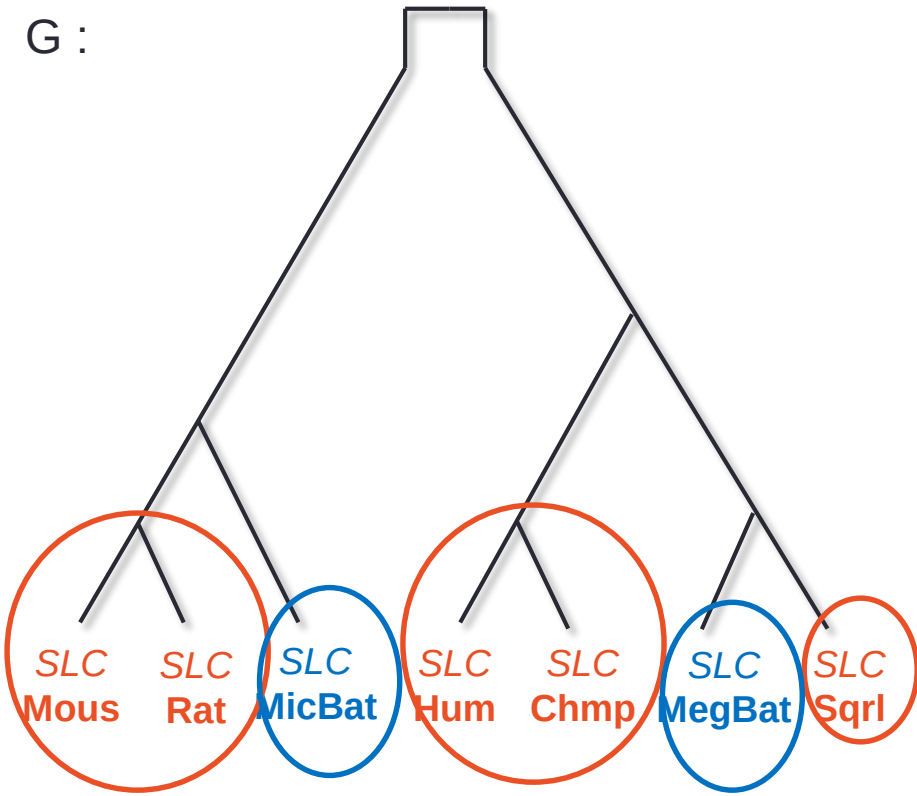




Introduction

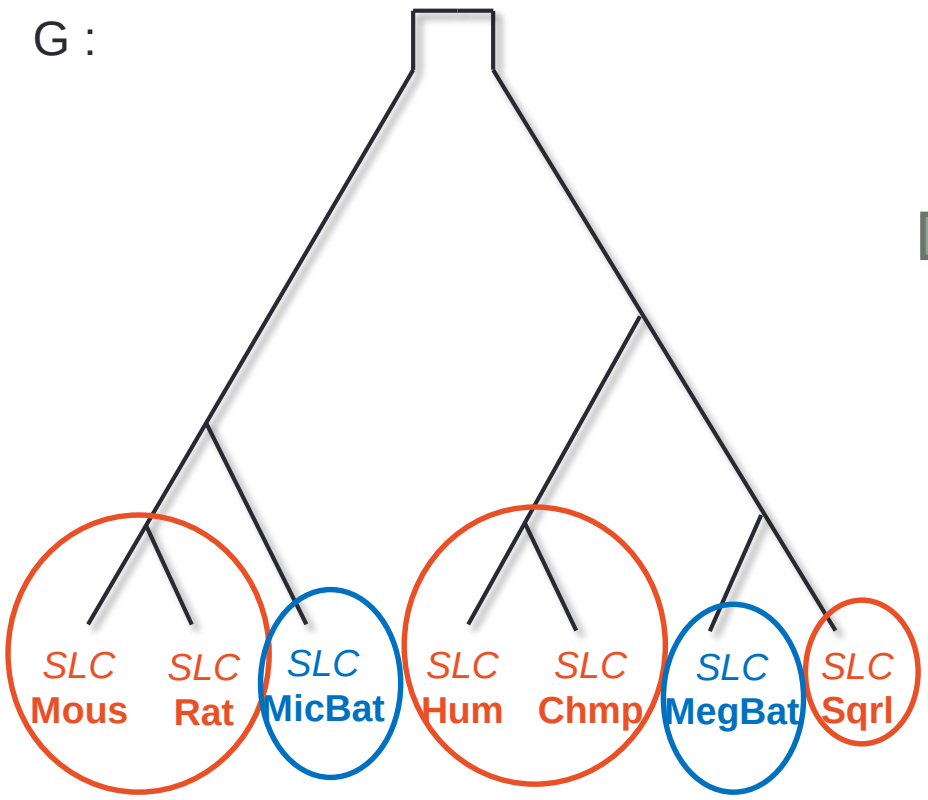
- Break G as least as possible : send the maximal bat subtrees **left**, and the maximal rodent/primate subtrees **right**

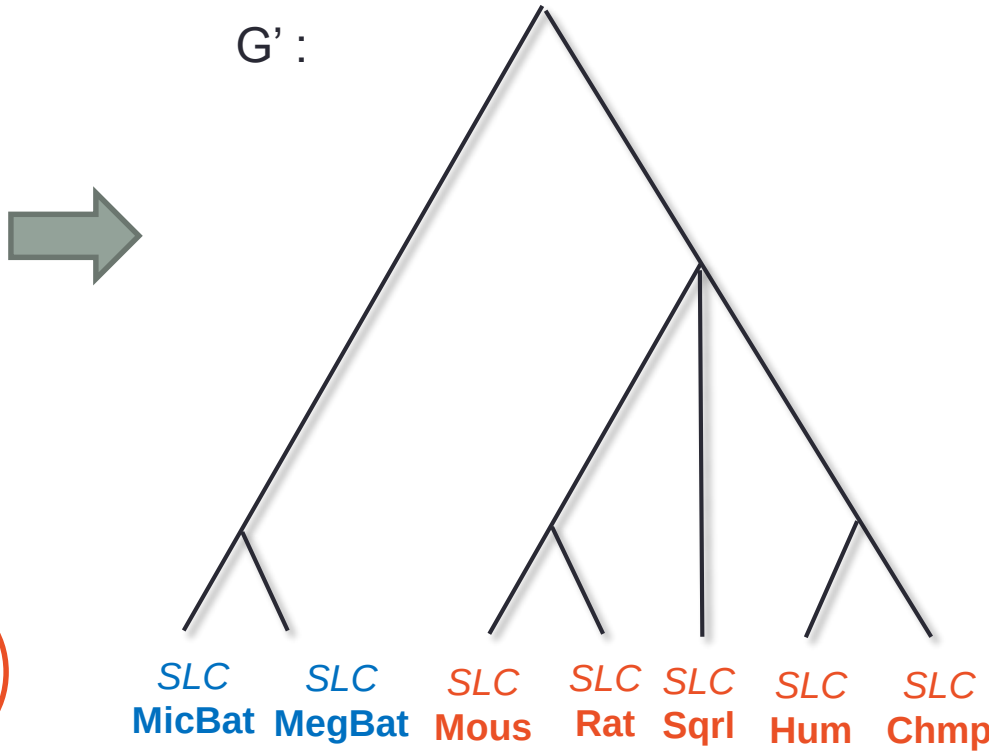




Introduction

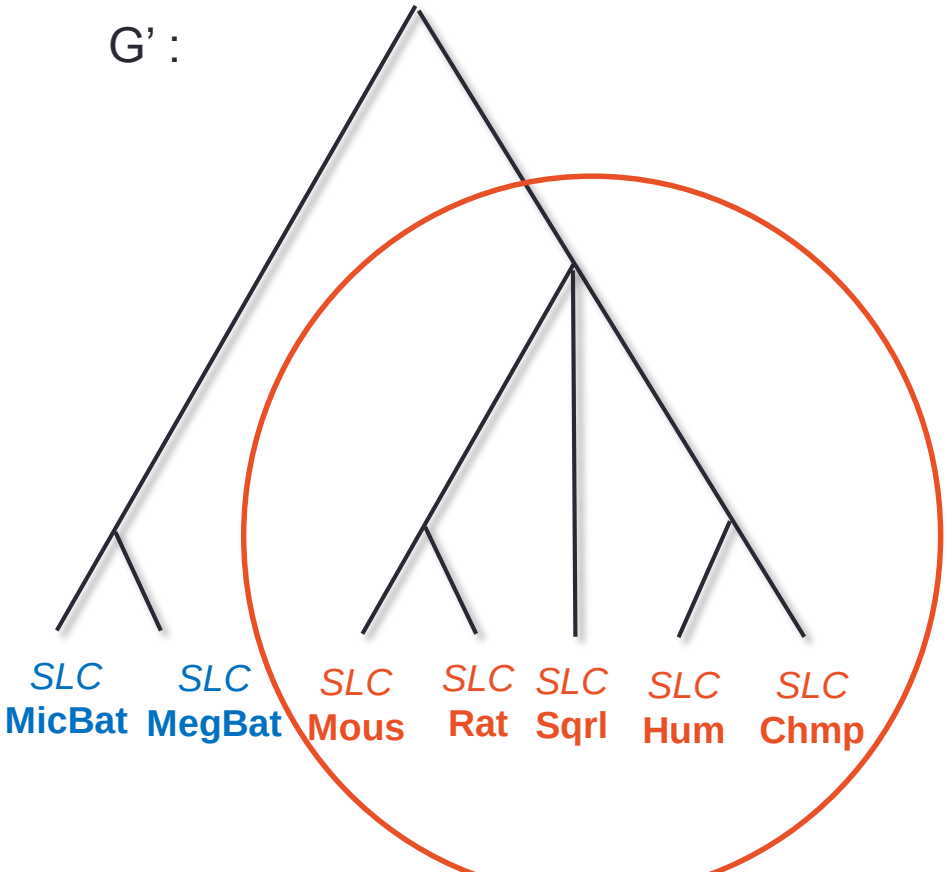
- Break G as least as possible : send the maximal bat subtrees **left**, and the maximal rodent/primate subtrees **right**





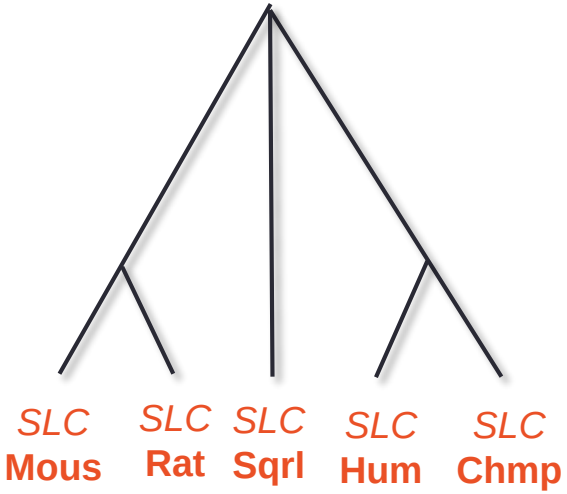
Introduction

- G’ ends up with possibly two unresolved **polytomies**.
- We are looking for a **binary refinement** of these polytomies.



Introduction

- Other sources of polytomies :
  - Lack of phylogenetic signal in the sequences, causing some gene tree construction algorithms to leave the gene tree partially unresolved.
  - Contraction of gene tree branches having low support (e.g. bootstrap values).



Previous works

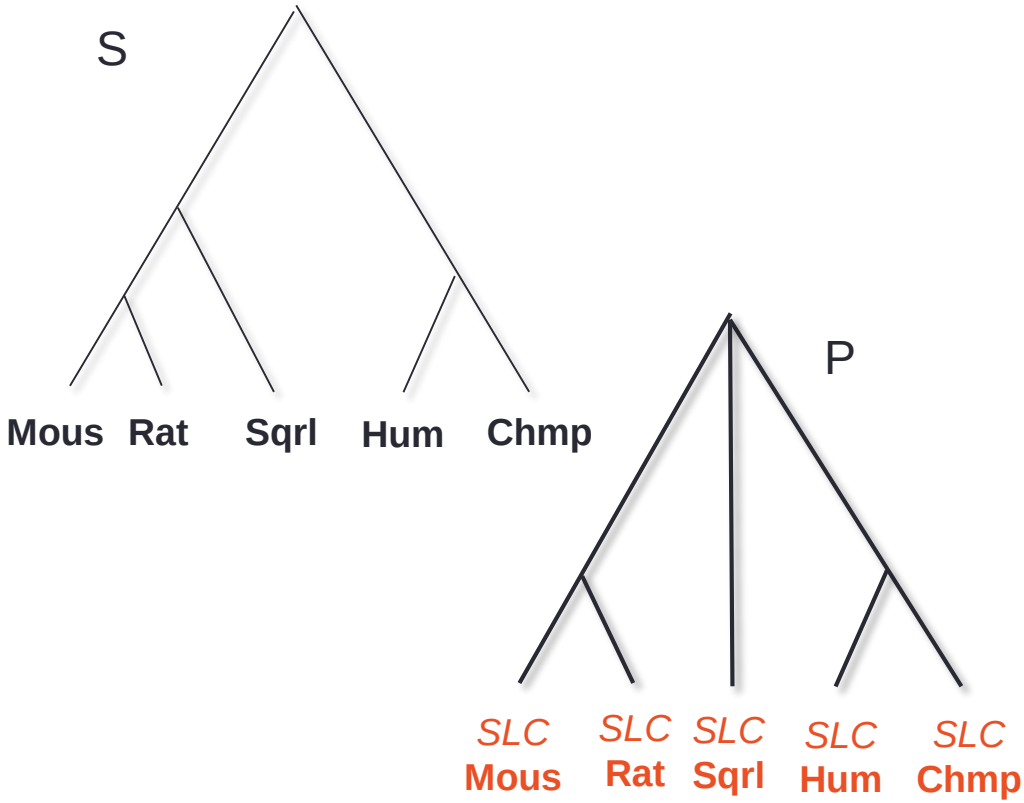
- Find a **binary refinement** minimizing:
  - **Duplications + losses** (Chang & Eulenstein, 2006, $O(n^3)$);
  - **Duplications + losses** (Lafond & Swenson & El-Mabrouk, 2012, O(n))
  - **Duplications and then losses** (Zheng, Wu, Zhang, 2012, O(n))
  - **Losses**: It's a linear problem.
- Our problem here:

Minimize NAD nodes

- For all these optimization criteria, polytomies can be refined independantly. Thus we reduce the problem to a single polytomy.

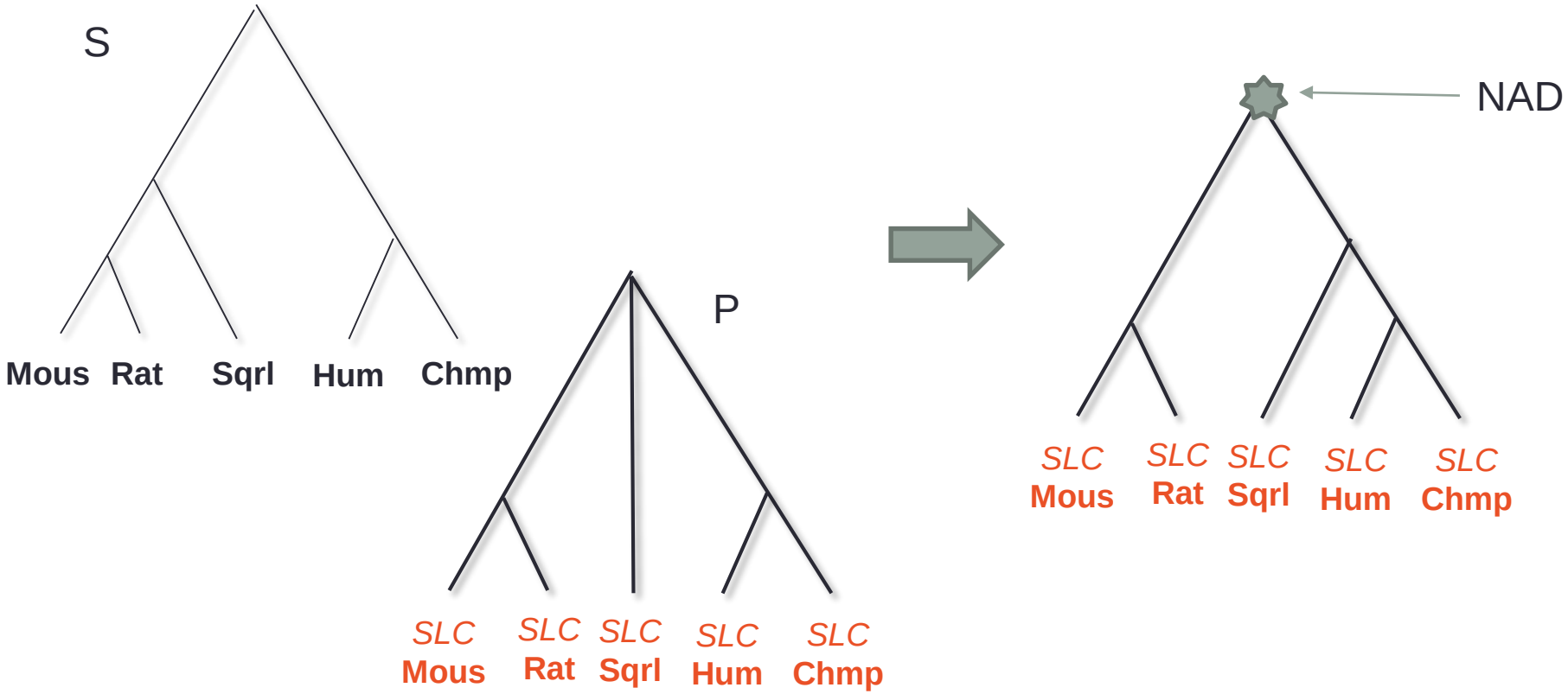
Introduction

- **Given** : a polytomy P and a species tree S
- **Find** : a binary refinement of P that minimizes the number of NADs created.



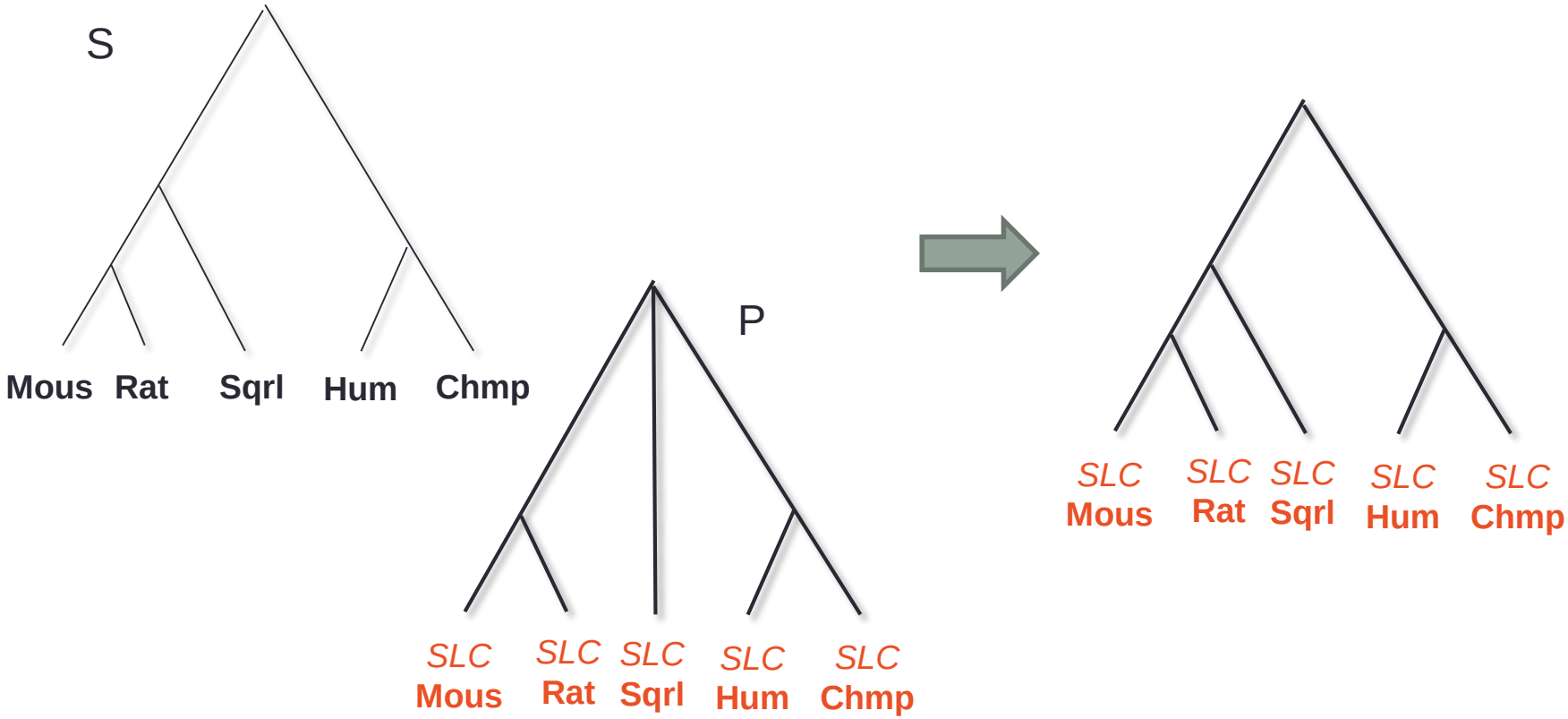
Introduction

- Given : a polytomy P and a species tree S
- Objective : find a binary refinement of P that minimizes the number of NADs created.

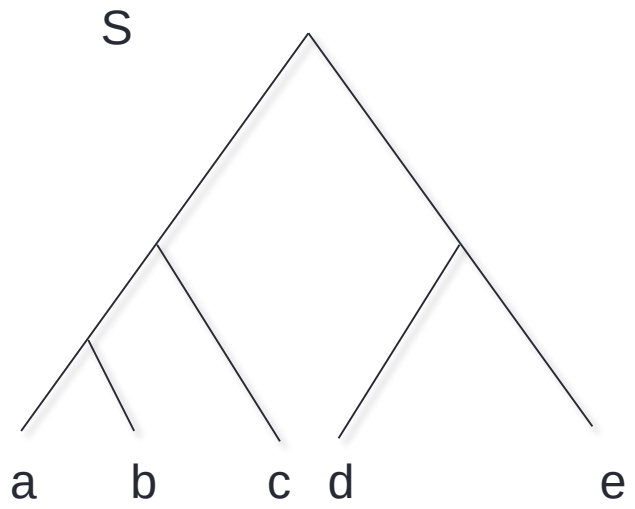


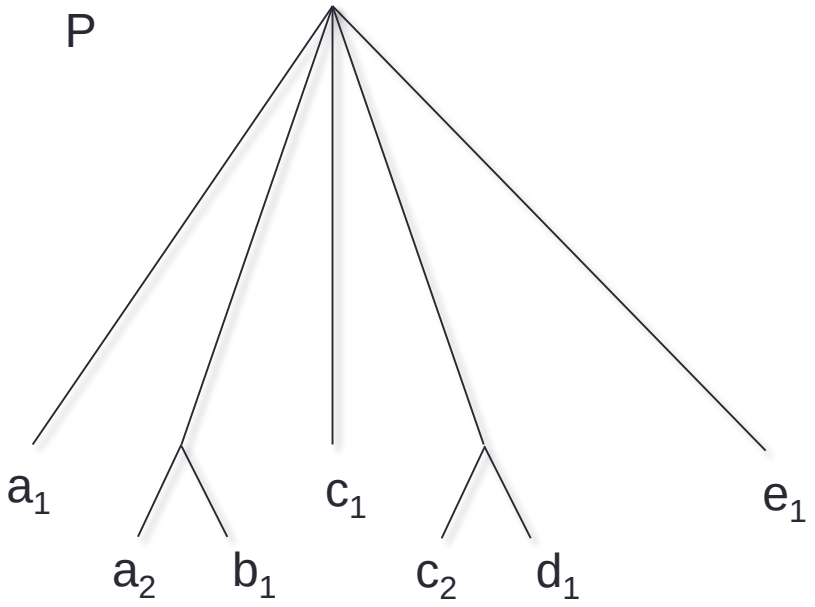
Introduction

- Given : a polytomy P and a species tree S
- Objective : find a binary refinement of P that minimizes the number of NADs created.

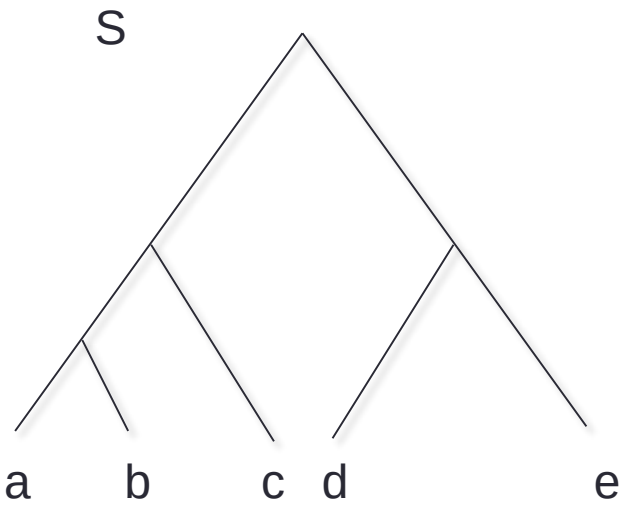


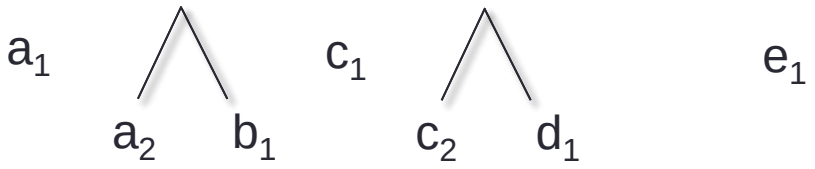
A simple example



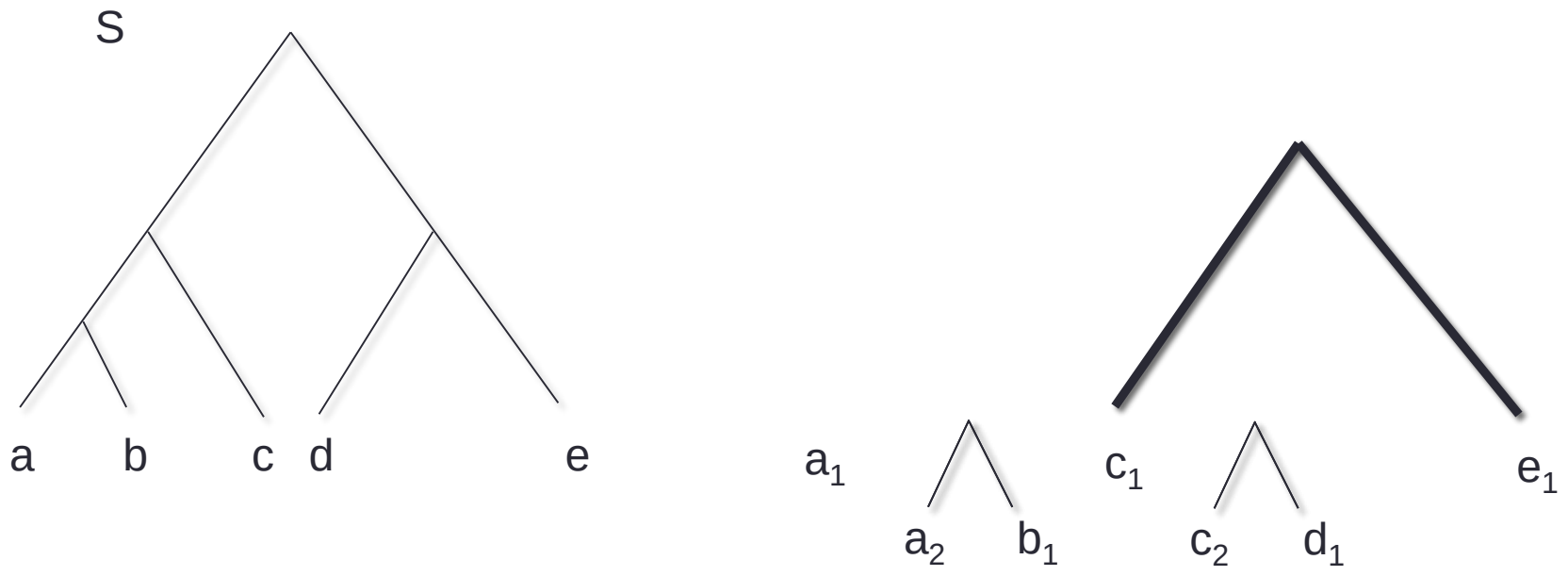


Reconnecting subtrees

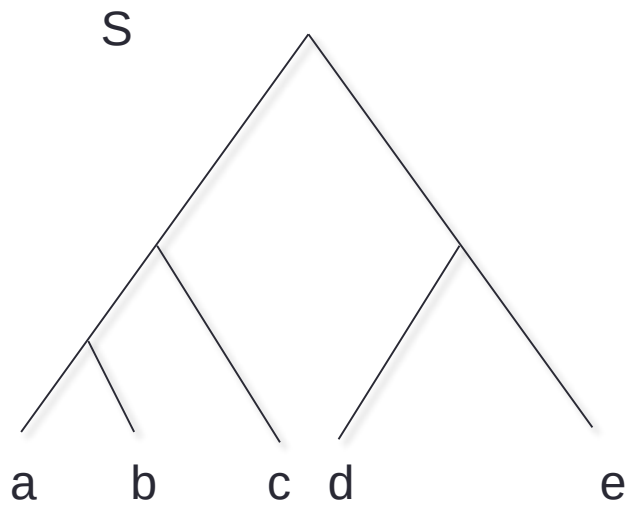


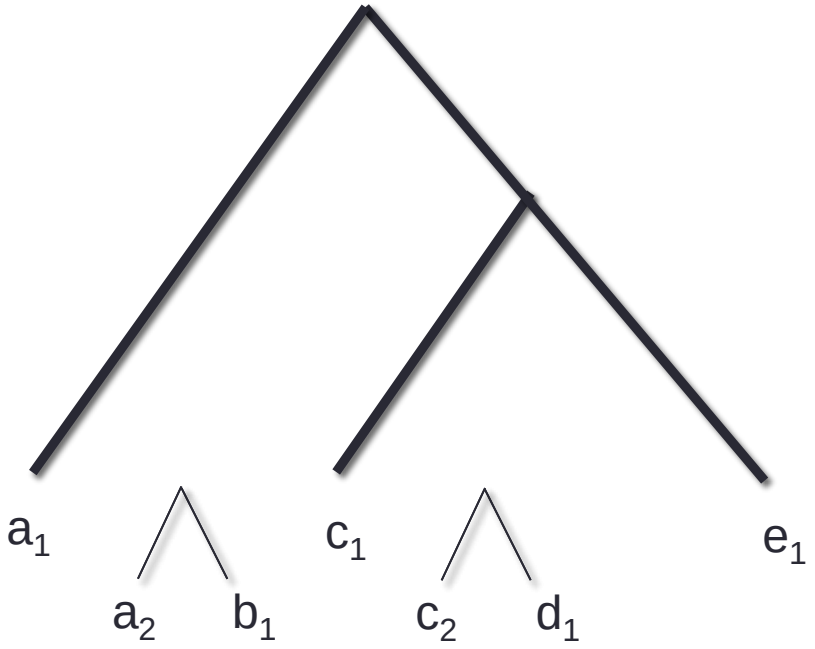


Reconnecting subtrees

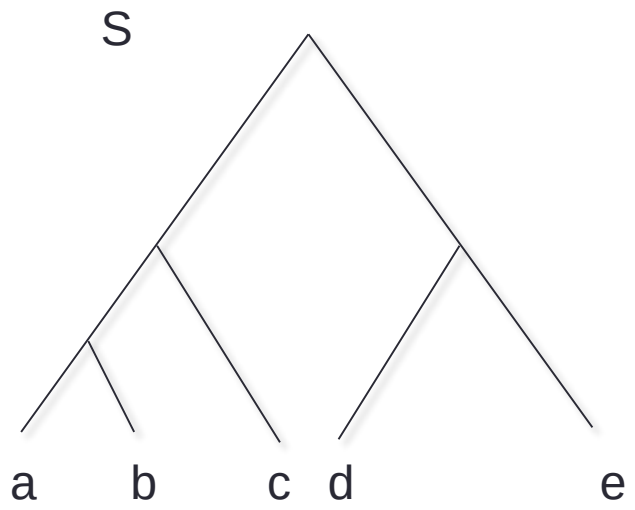


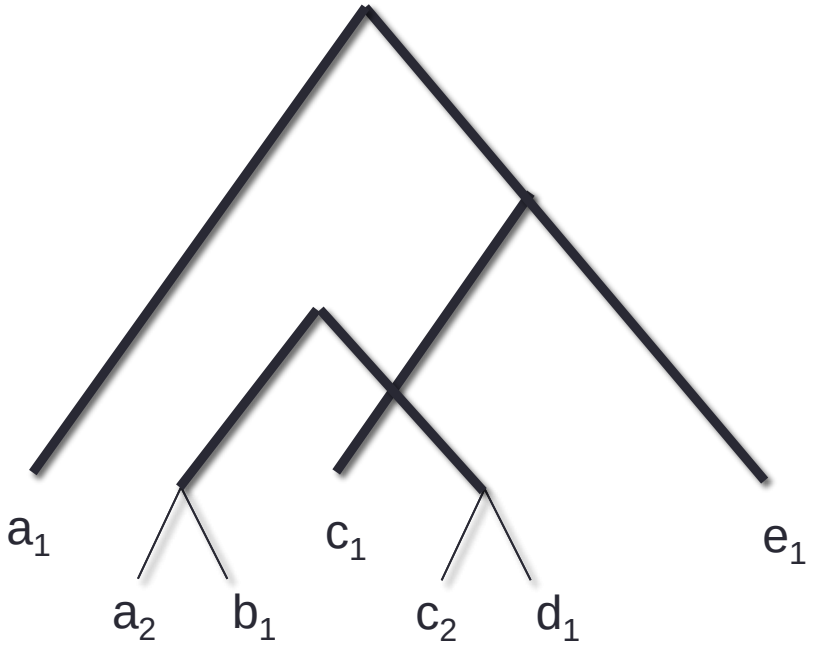
Reconnecting subtrees



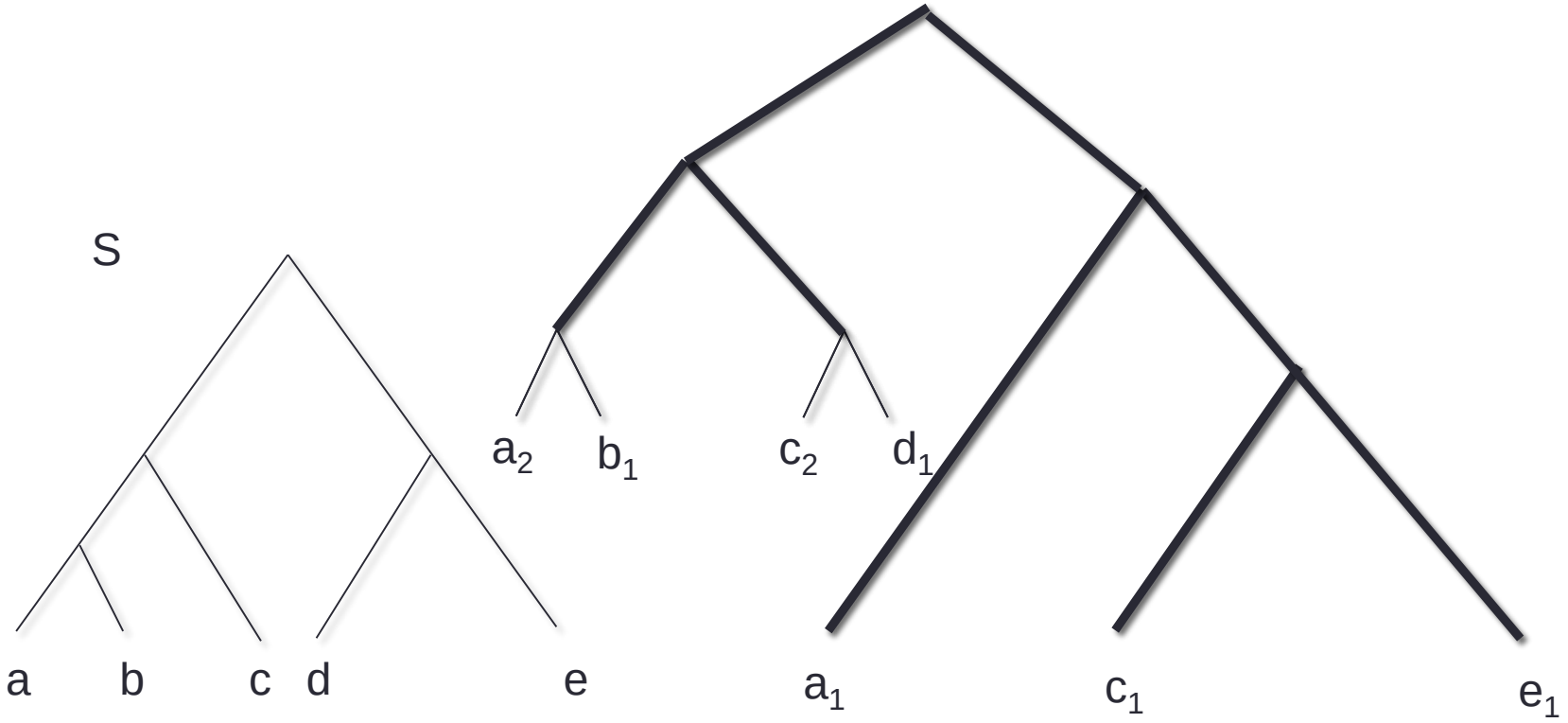


Reconnecting subtrees

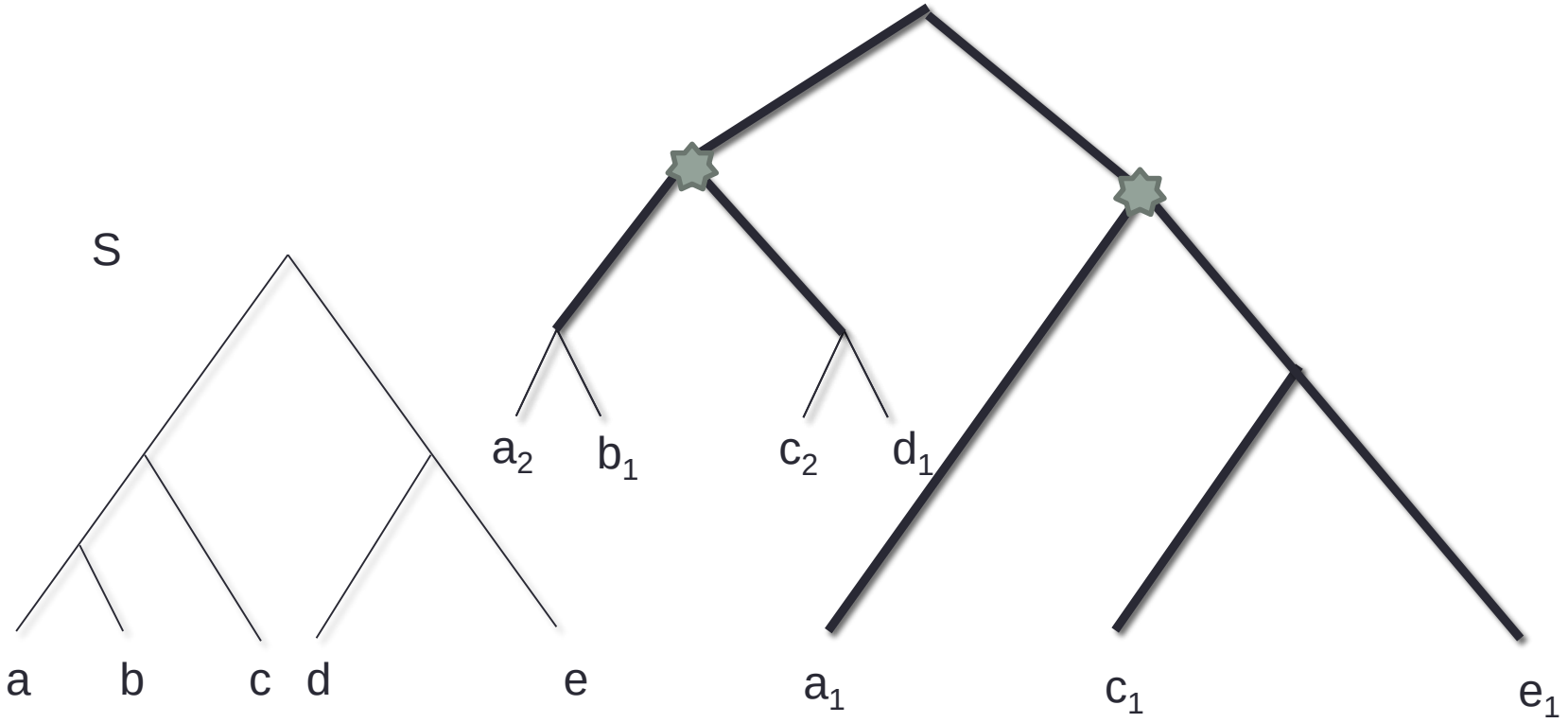




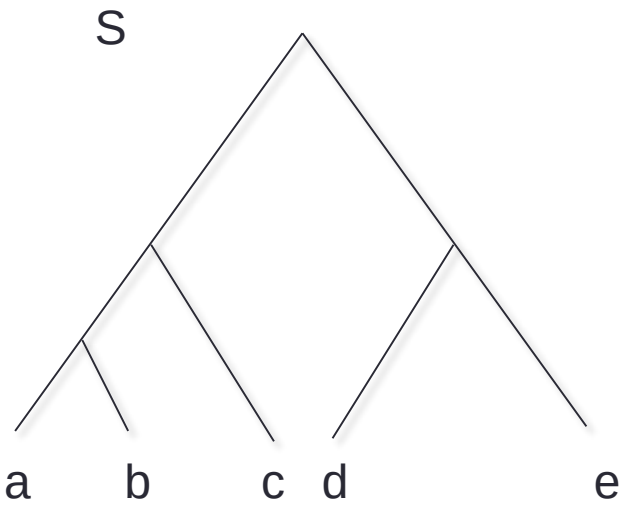
Reconnecting subtrees

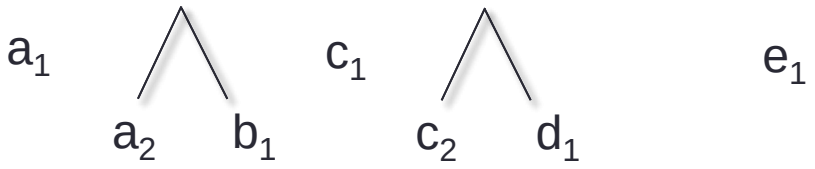


Reconnecting subtrees

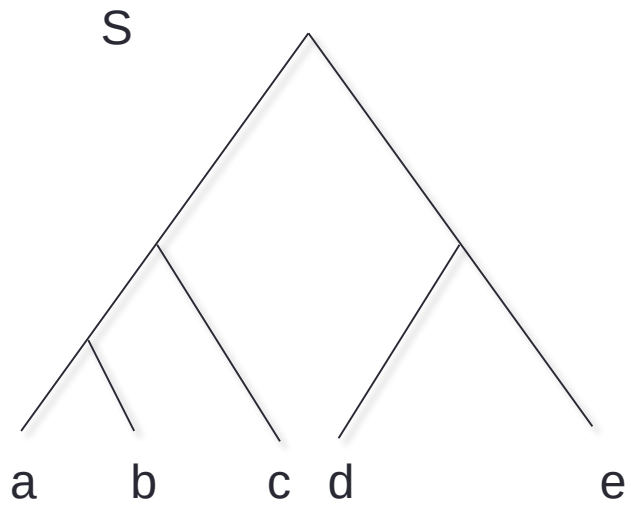


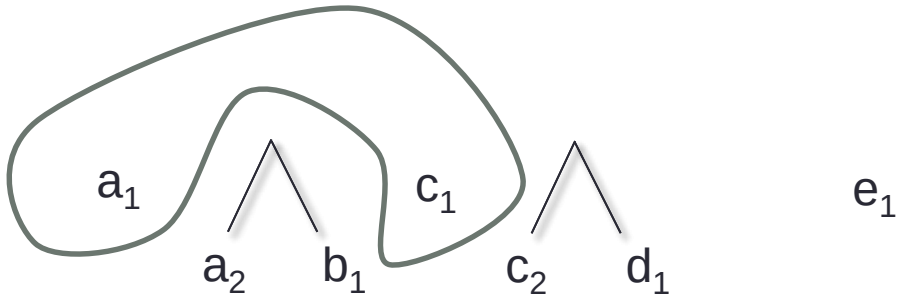
Reconnecting subtrees





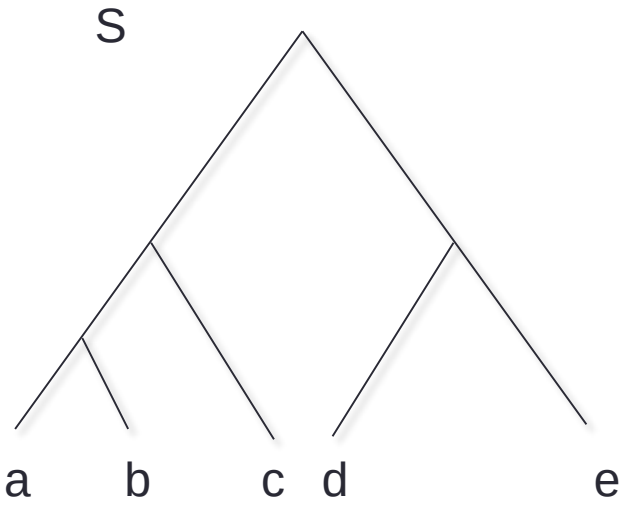
Reconnecting subtrees

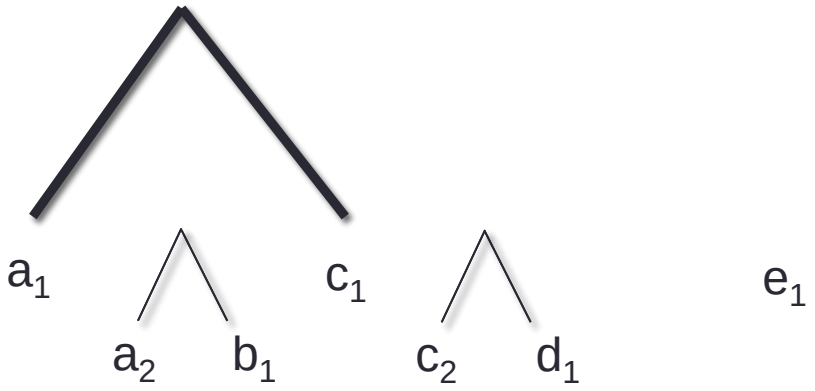




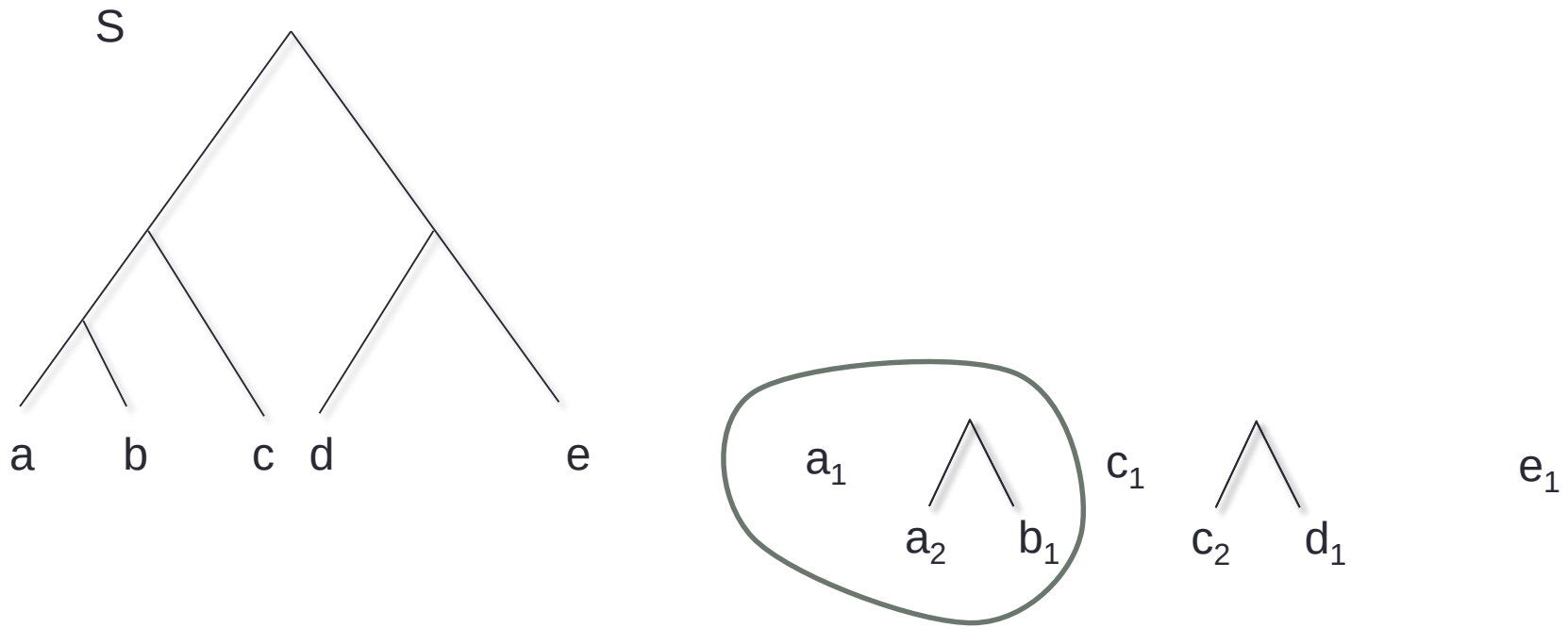
Reconnecting subtrees

a_1,c_1 are connected by Speciation (S)

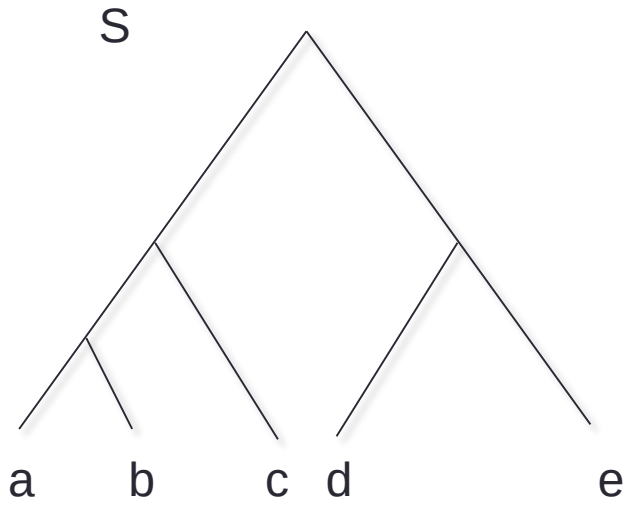




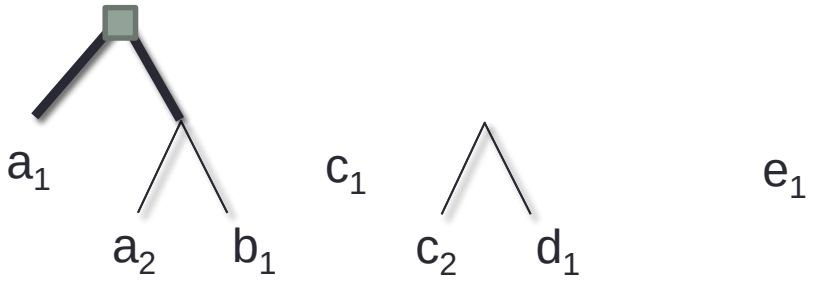
Reconnecting subtrees



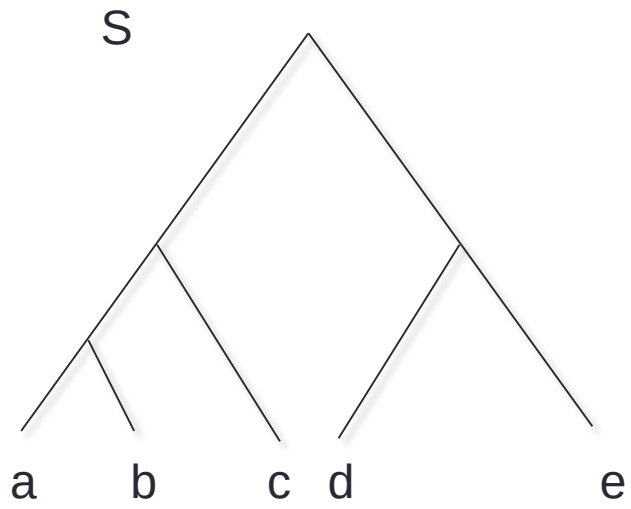
Reconnecting subtrees

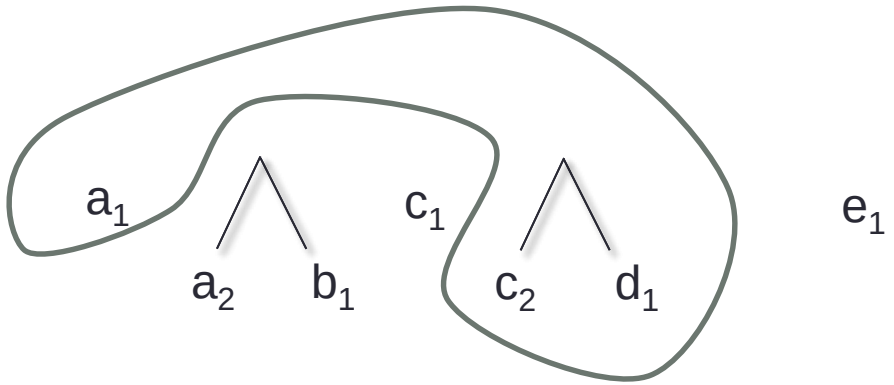


a_1,(a2, b1) are connected by Apparent Duplication (AD)

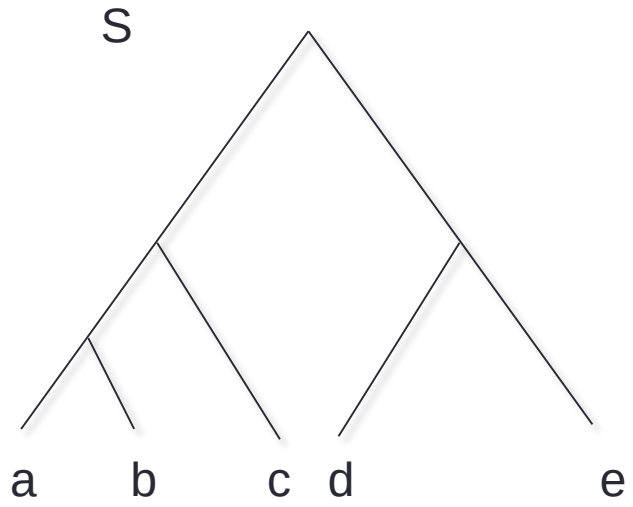


Reconnecting subtrees

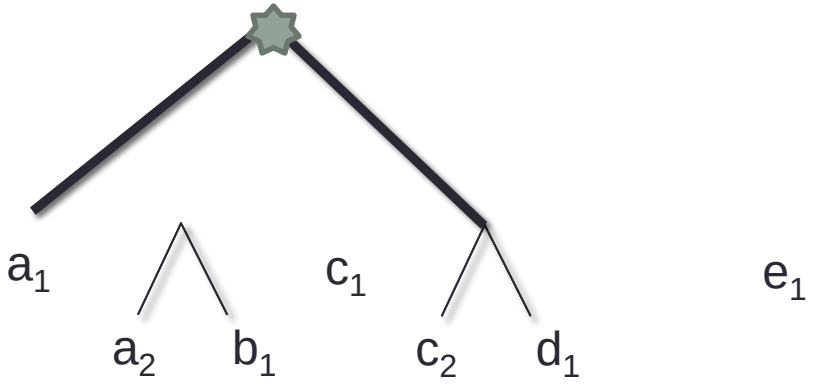




Reconnecting subtrees



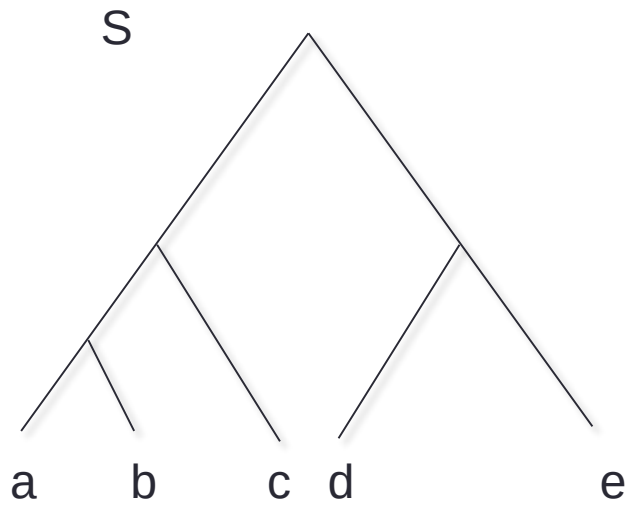
a_1,(a2, b1) are connected by Non-Apparent Duplication (NAD)



Relationship graph

Each subtree is a vertex.

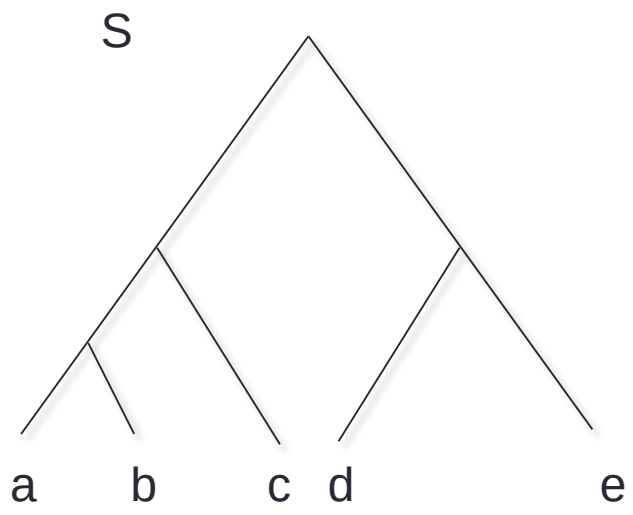
Each pair of vertices (x,y) is connected by an edge labeled by the connection type of x and y.

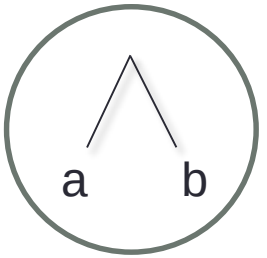


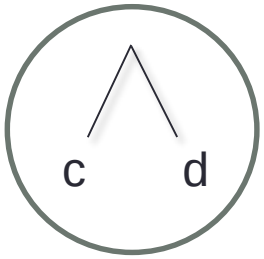
Relationship graph

Each subtree is a vertex.

Each pair of vertices (x,y) is connected by an edge labeled by the connection type of x and y.



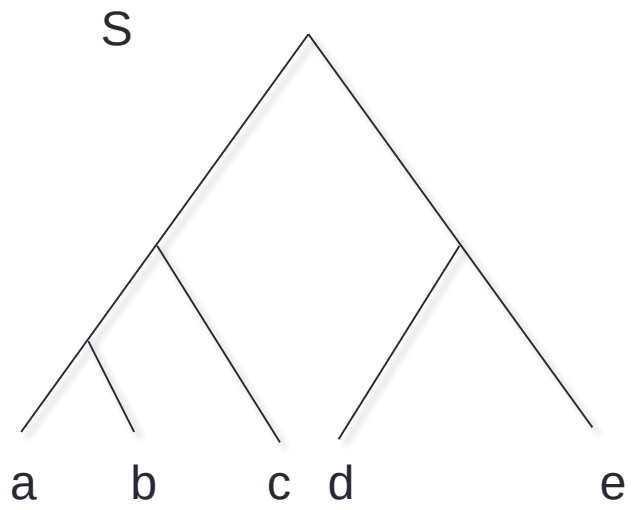


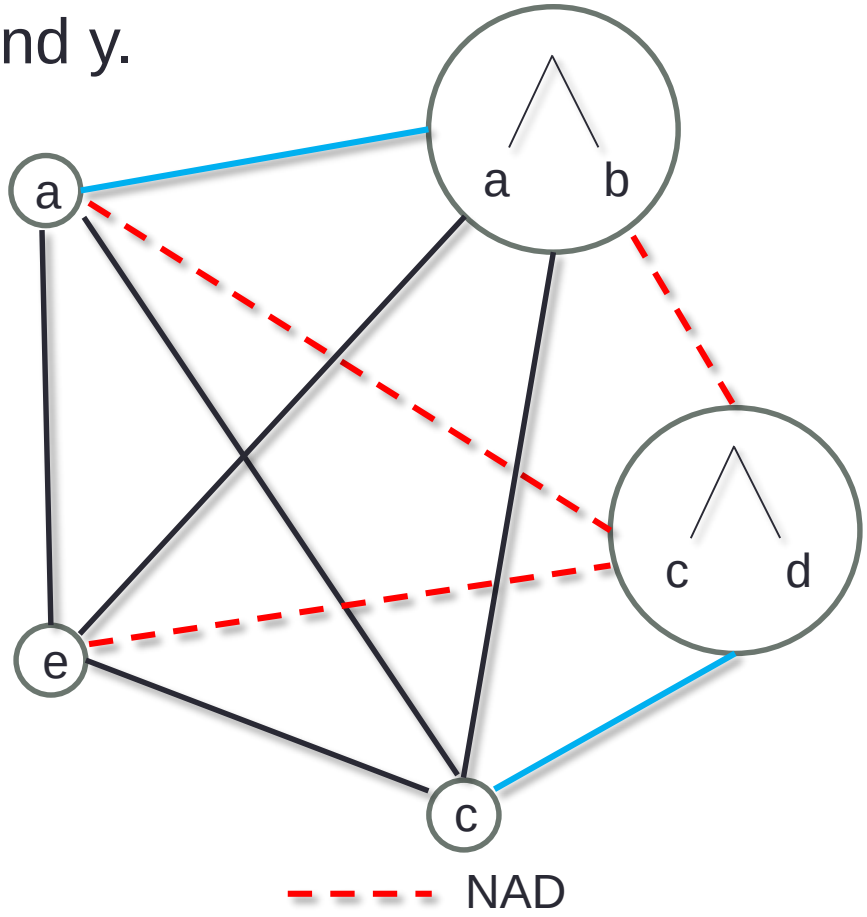


Relationship graph

Each subtree is a vertex.

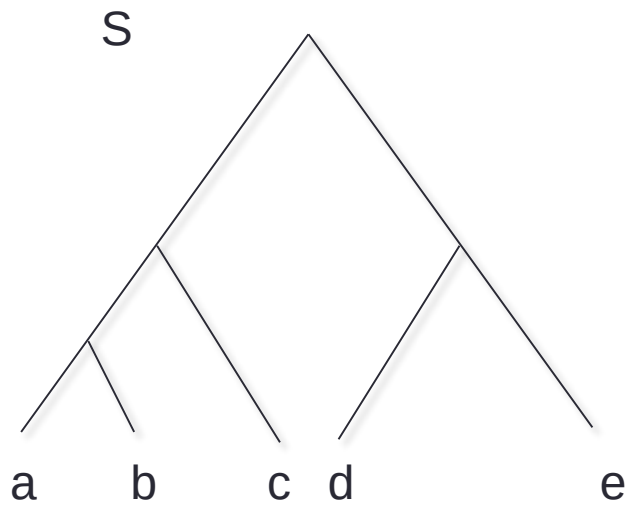
Each pair of vertices (x,y) is connected by an edge labeled by the connection type of x and y.

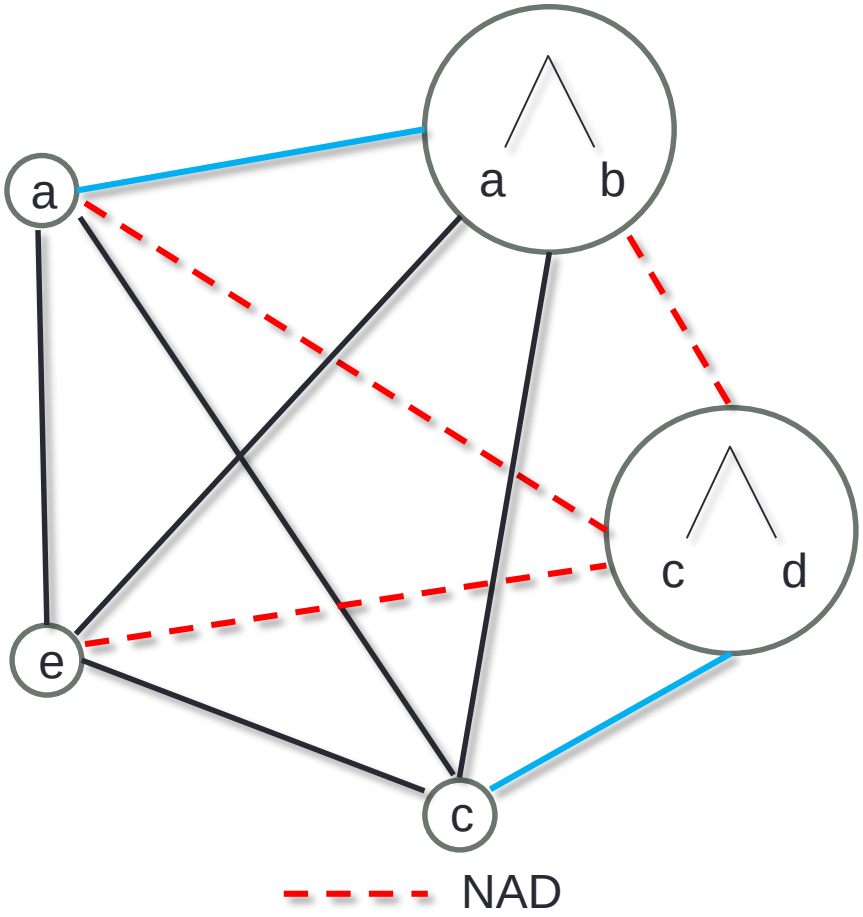




Relationship graph

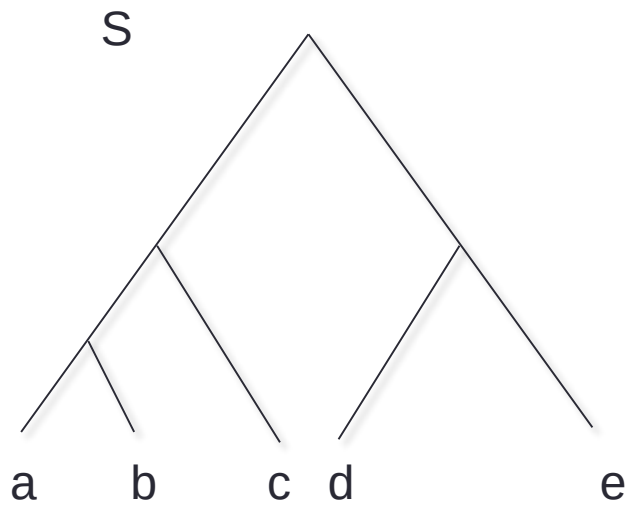
Speciation clique : a clique exclusively made up of “Spec” edges.

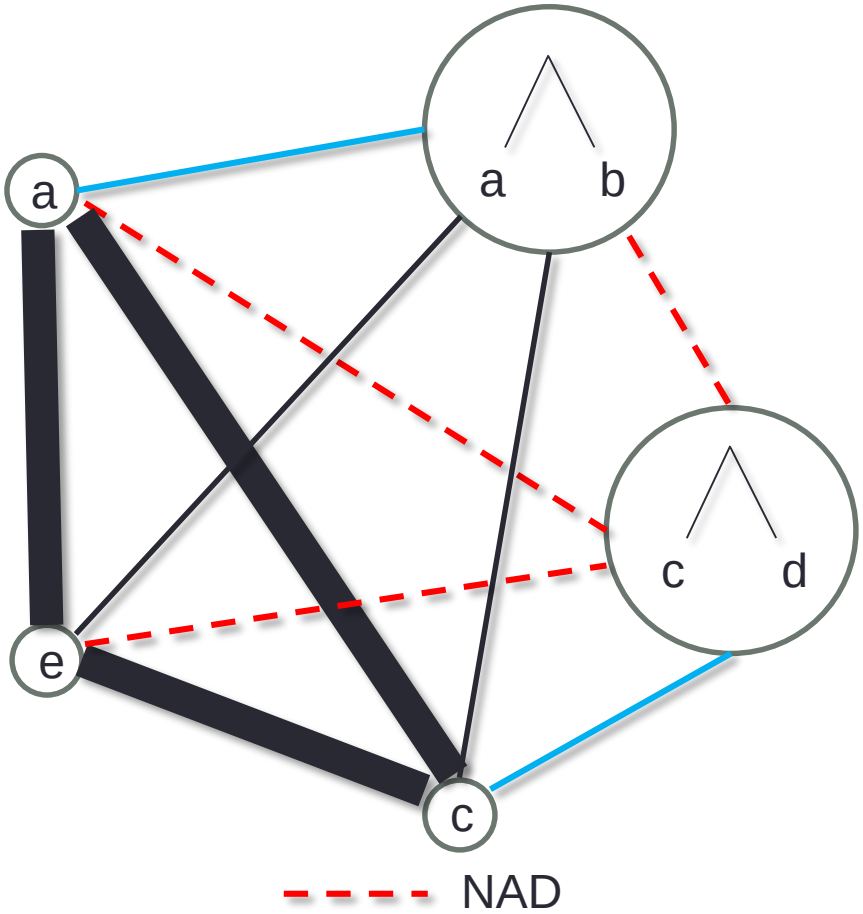




Relationship graph

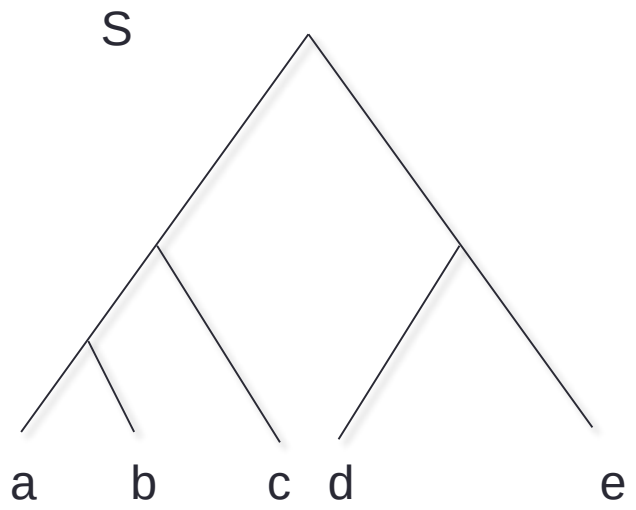
Speciation clique : a clique exclusively made up of “Spec” edges.

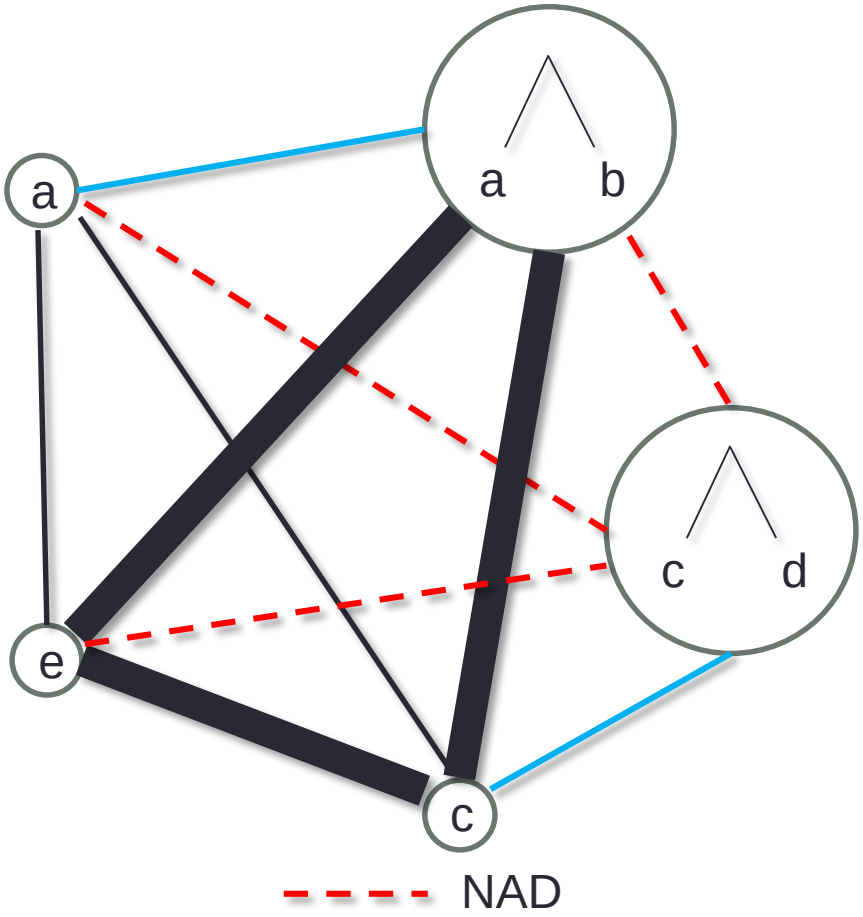




Relationship graph

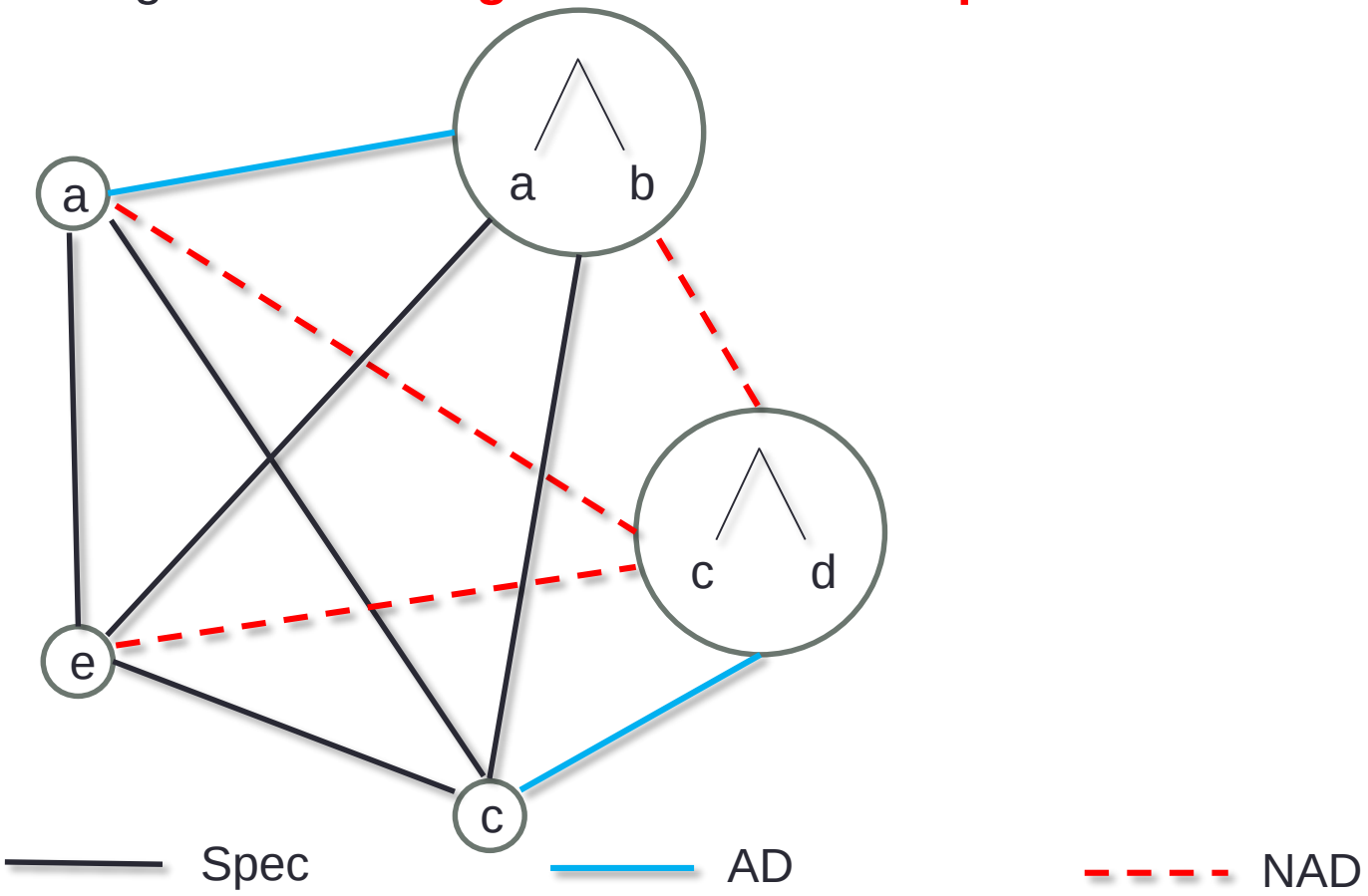
Speciation clique : a clique exclusively made up of “Spec” edges.





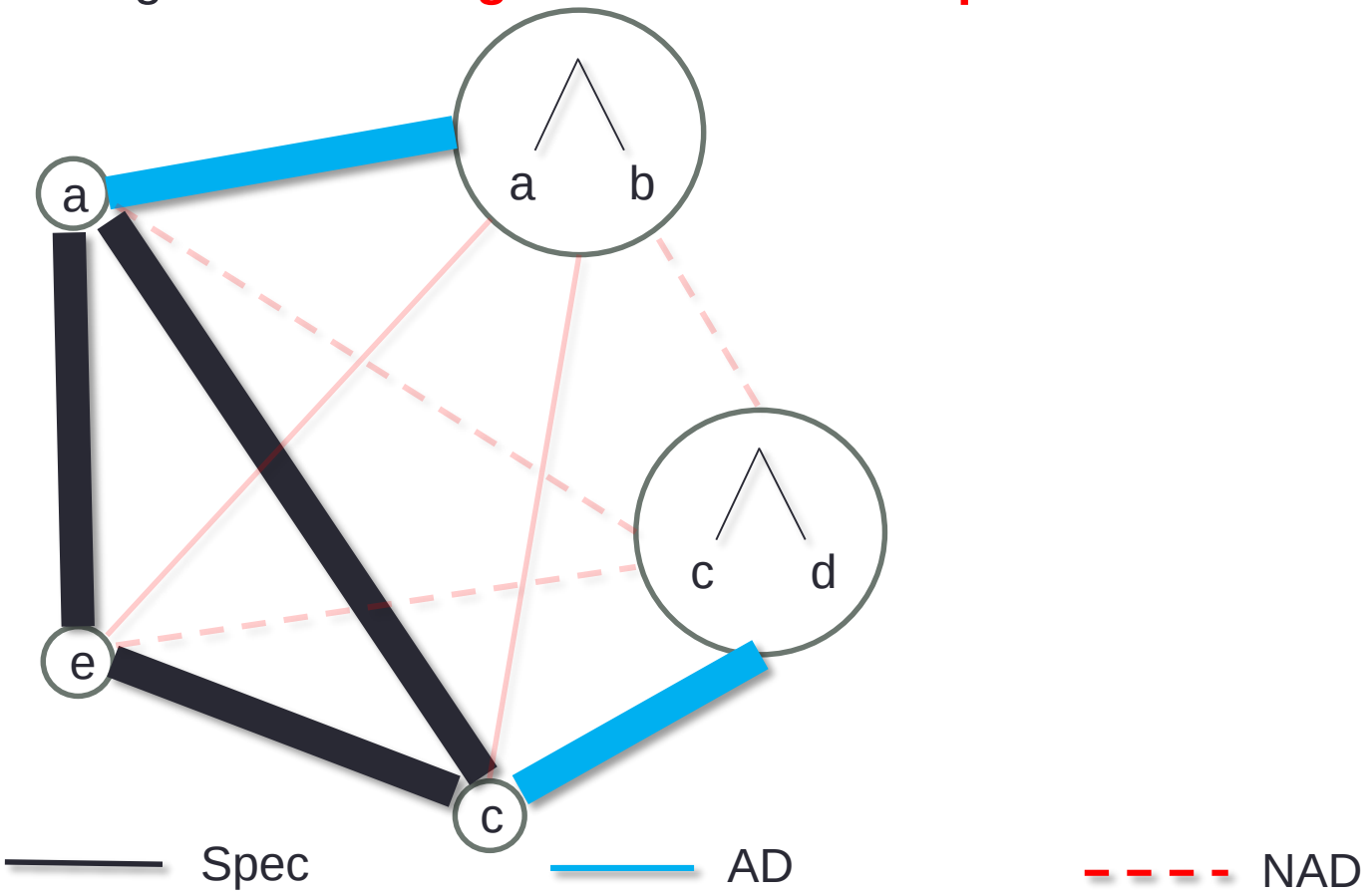
Theorem

There exists a binary refinement with **zero** NADs iff there exists a set of disjoint **speciation cliques** W in the relationship graph such that W + the AD edges form a **single connected component.**



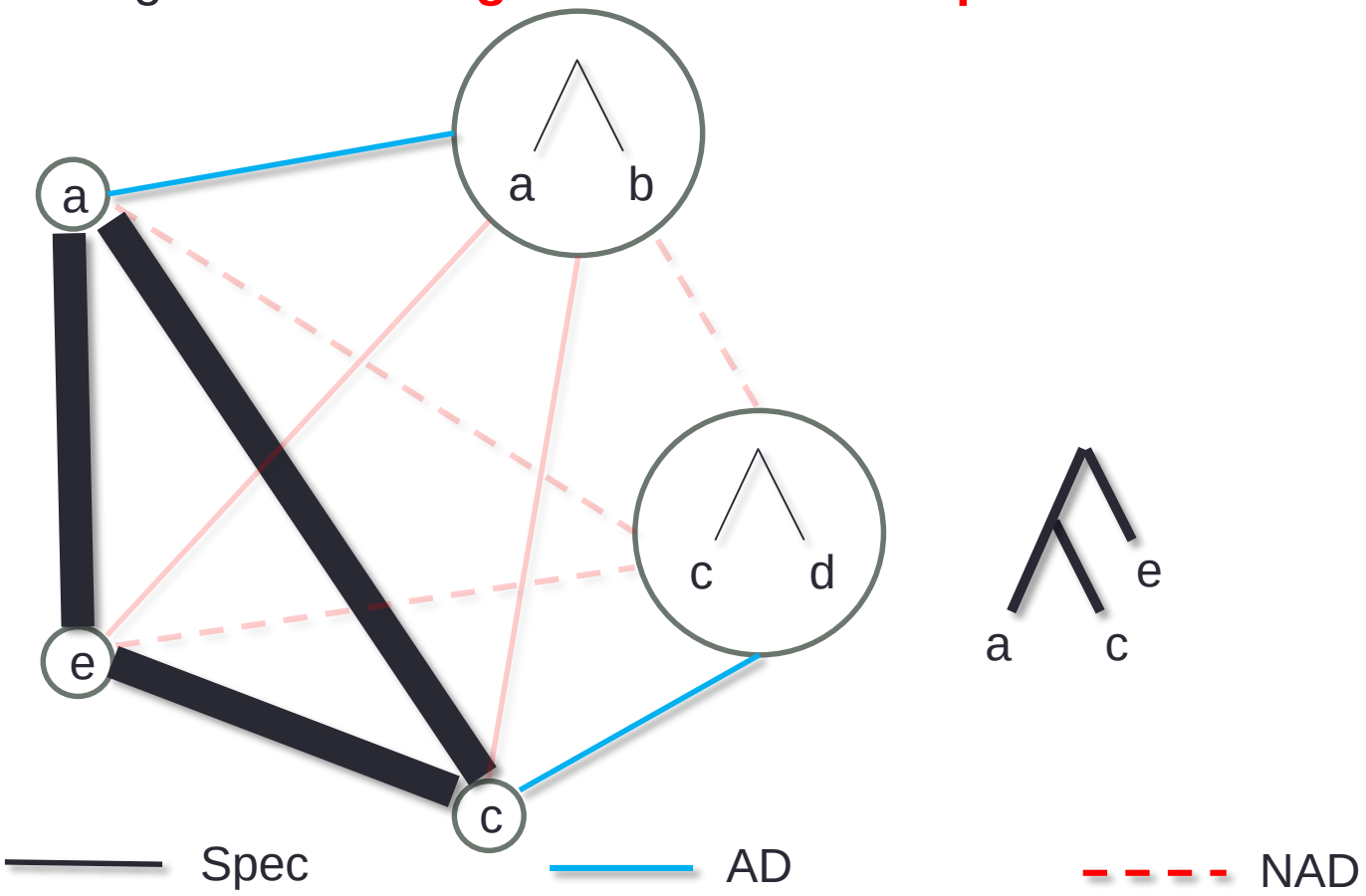
Theorem

There exists a binary refinement with **zero** NADs iff there exists a set of disjoint **speciation cliques** W in the relationship graph such that W + the AD edges form a **single connected component.**



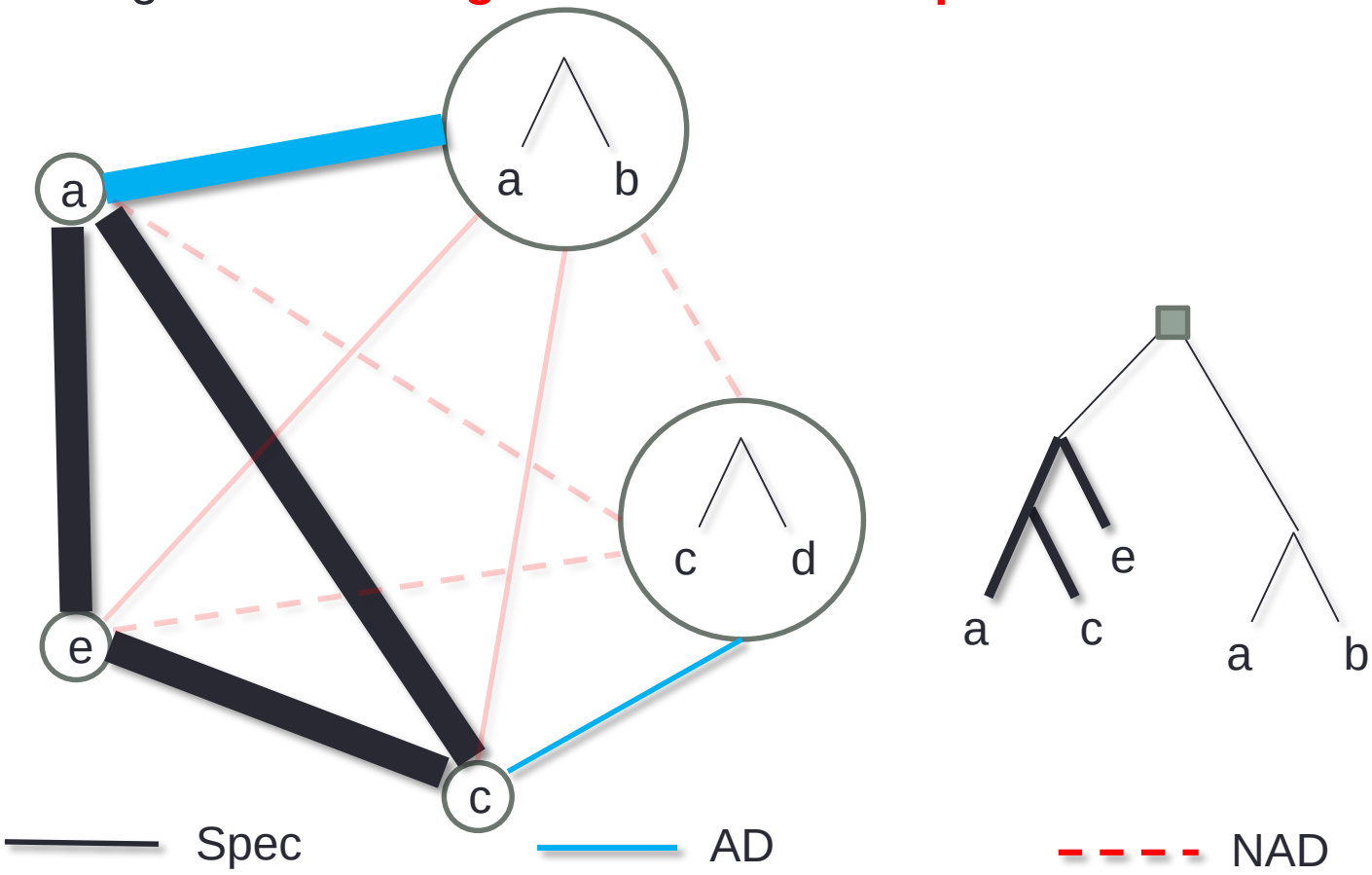
Theorem

There exists a binary refinement with **zero** NADs iff there exists a set of disjoint **speciation cliques** W in the relationship graph such that W + the AD edges form a **single connected component.**



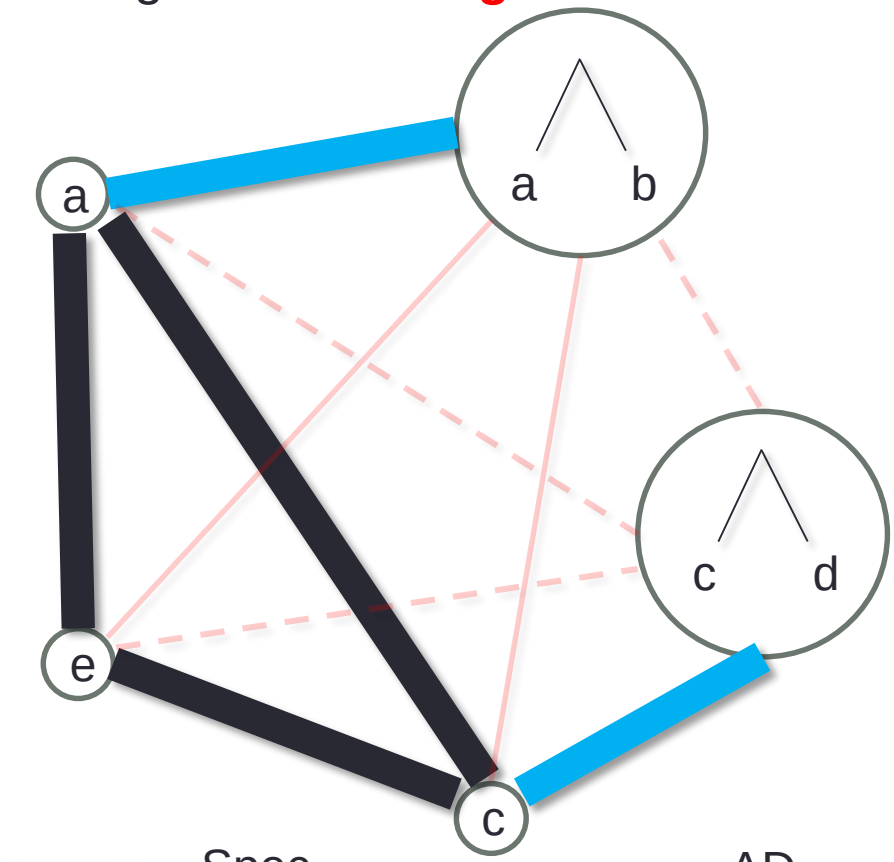
Theorem

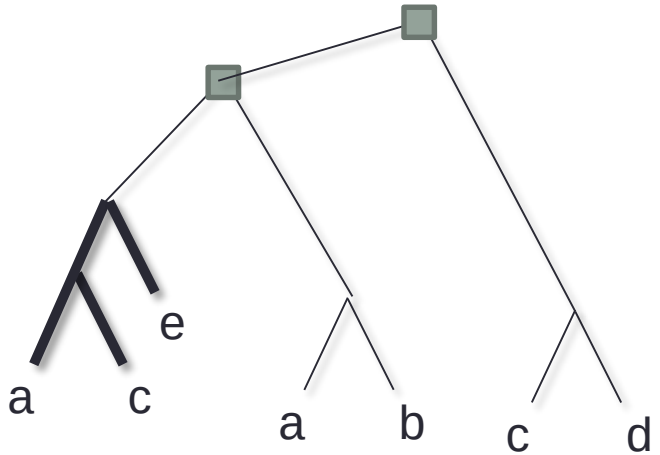
There exists a binary refinement with **zero** NADs iff there exists a set of disjoint **speciation cliques** W in the relationship graph such that W + the AD edges form a **single connected component.**



Theorem

There exists a binary refinement with **zero** NADs iff there exists a set of disjoint **speciation cliques** W in the relationship graph such that W + the AD edges form a **single connected component.**

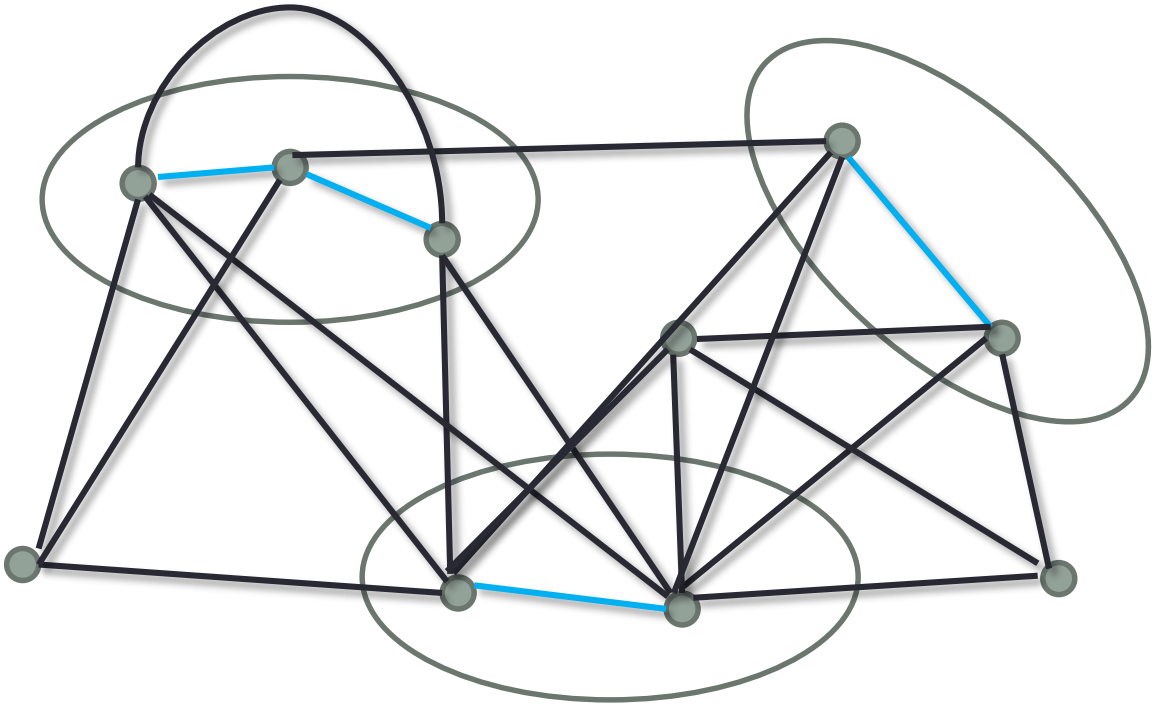




NAD

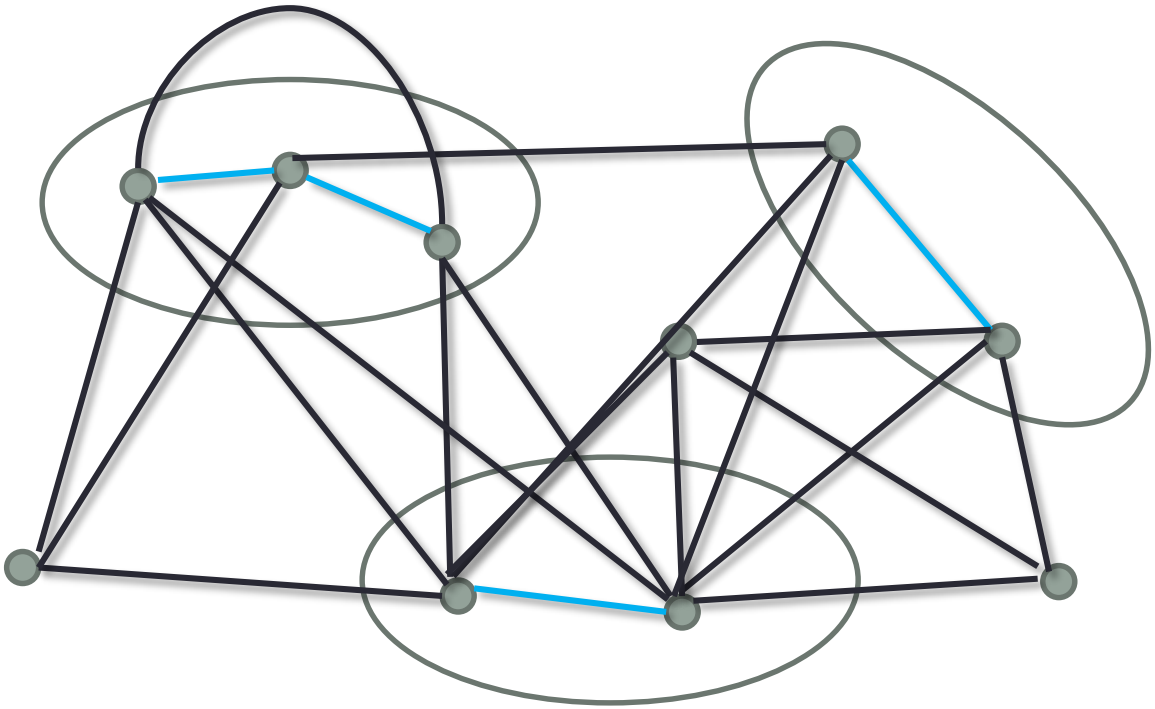
Theorem

There exists a binary refinement with a **minimum of d** NADs iff there exists a set of disjoint **speciation cliques** W in the relationship graph such that W + the AD edges have a **minimum of d + 1 connected components.**



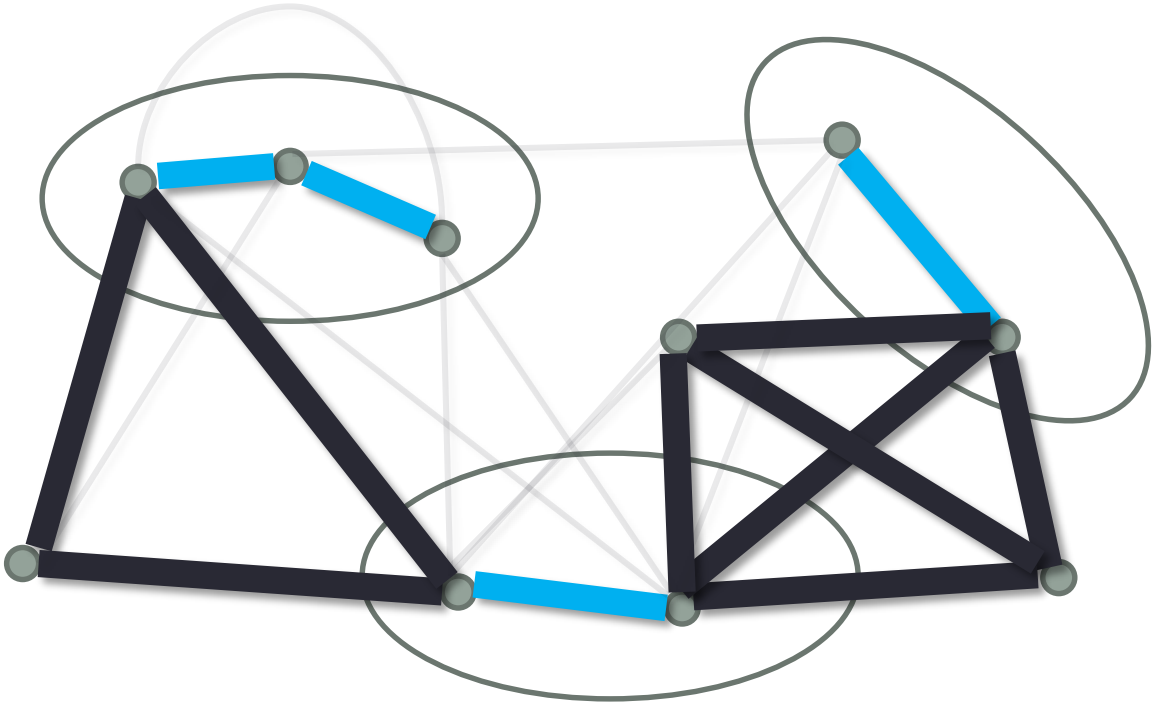
Problem reformulation

Given a graph with Spec and AD edges, find a set of cliques W such that W + the AD edges has the **minimum number of connected components.**



Problem reformulation

Given a graph with Spec and AD edges, find a set of cliques W such that W + the AD edges has the **minimum number of connected components.**



Problem reformulation

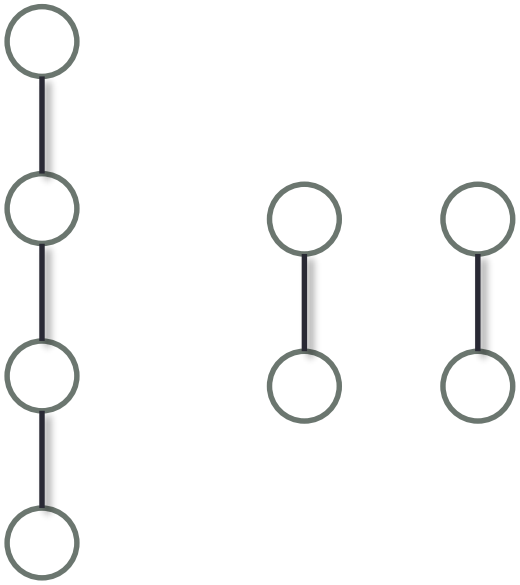
Given a graph R with Spec and AD edges, find a set of cliques W such that R restricted to W + the AD edges has the **minimum number of connected components.**

In general, finding W is an NP-Hard problem.

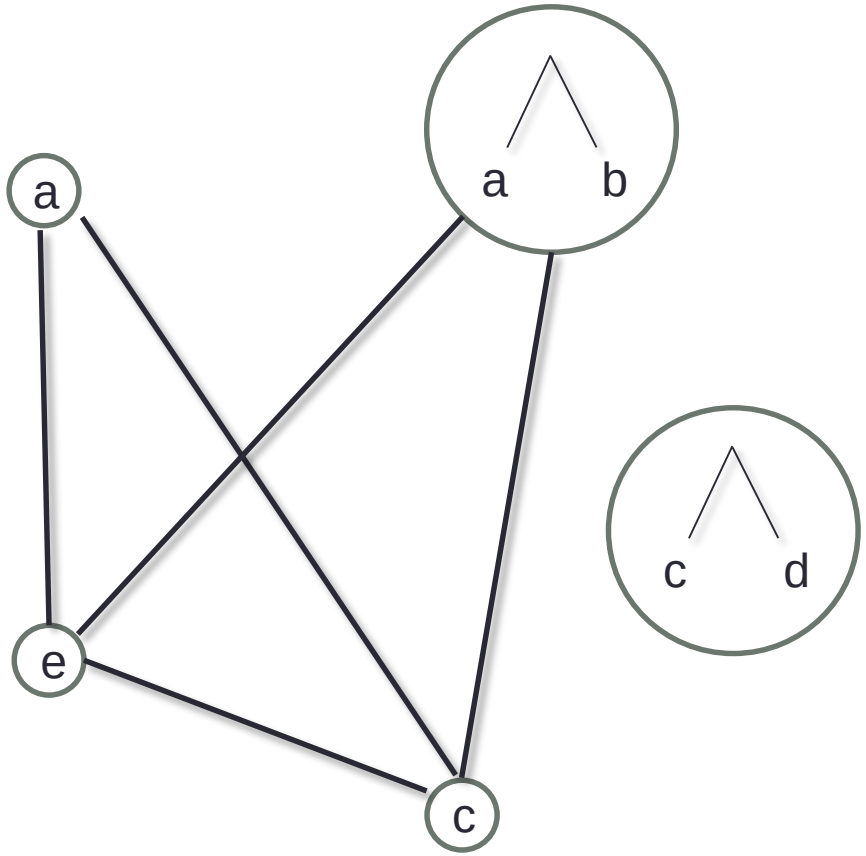
But R is not just any graph !

Characterization of the relationships

The relationship graph restricted to the Spec edges is **{P_4, $2K_2$}-free**.

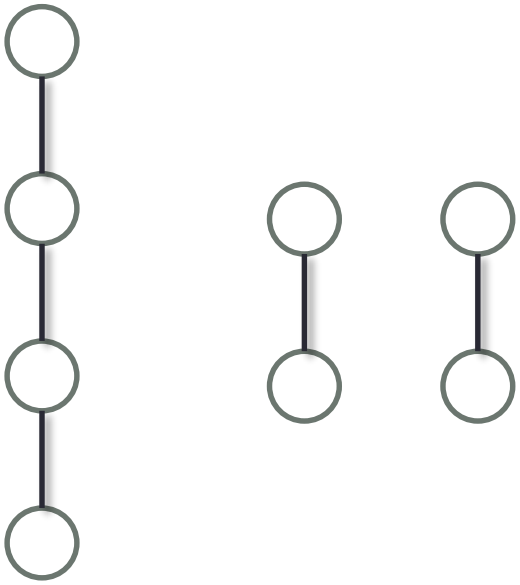


P_4 $2K_2$

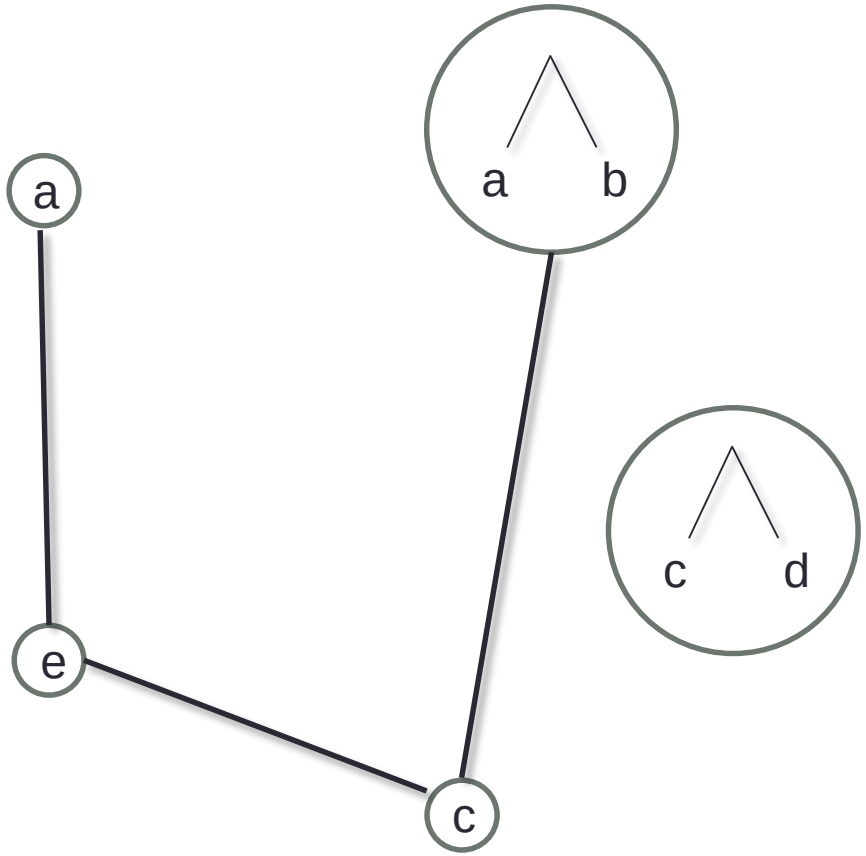


Characterization of the relationships

The relationship graph restricted to the Spec edges is **{P_4, $2K_2$}-free**.

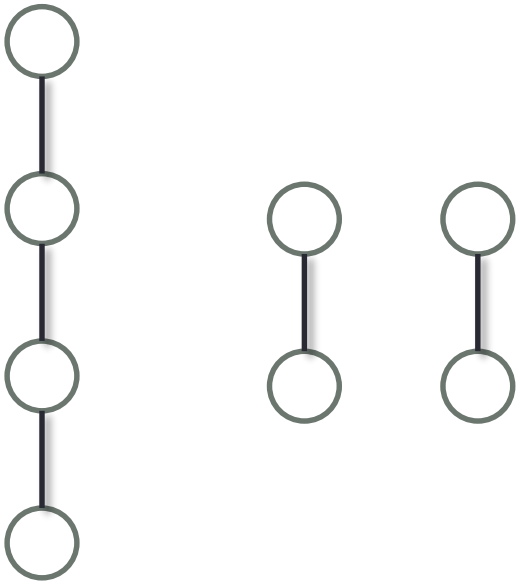


P_4 $2K_2$



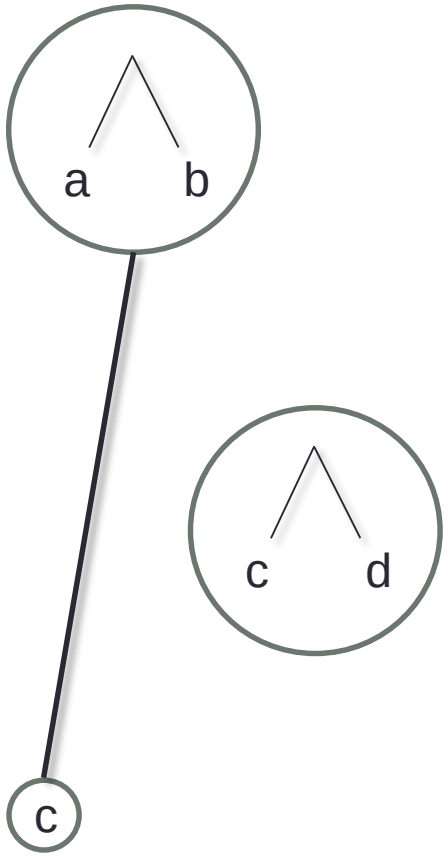
Characterization of the relationships

The relationship graph restricted to the Spec edges is **{P_4, $2K_2$}-free**.



P_4

$2K_2$



Problem reformulation

Given a graph R with Spec and AD edges, find a set of cliques W such that R restricted to W + the AD edges has the **minimum number of connected components.**

In general, finding W is an NP-Hard problem.

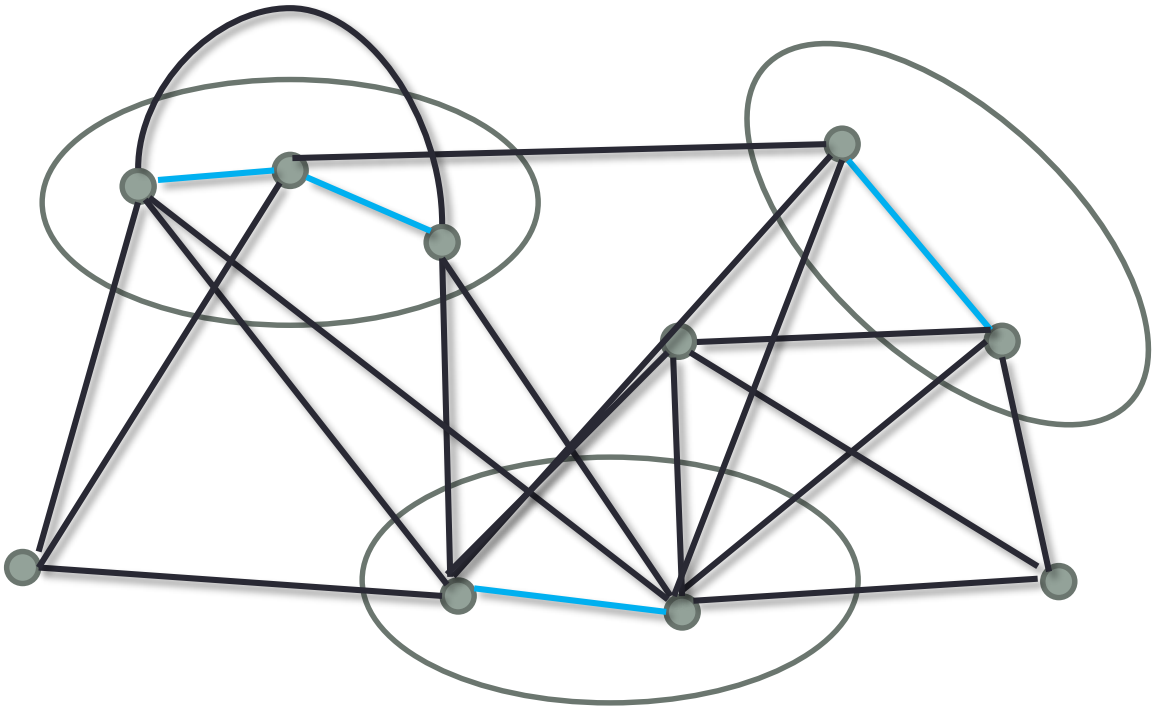
But R is not just any graph !

R is {P4, 2k2}-free.

Complexity for this class of graphs : **who knows?**

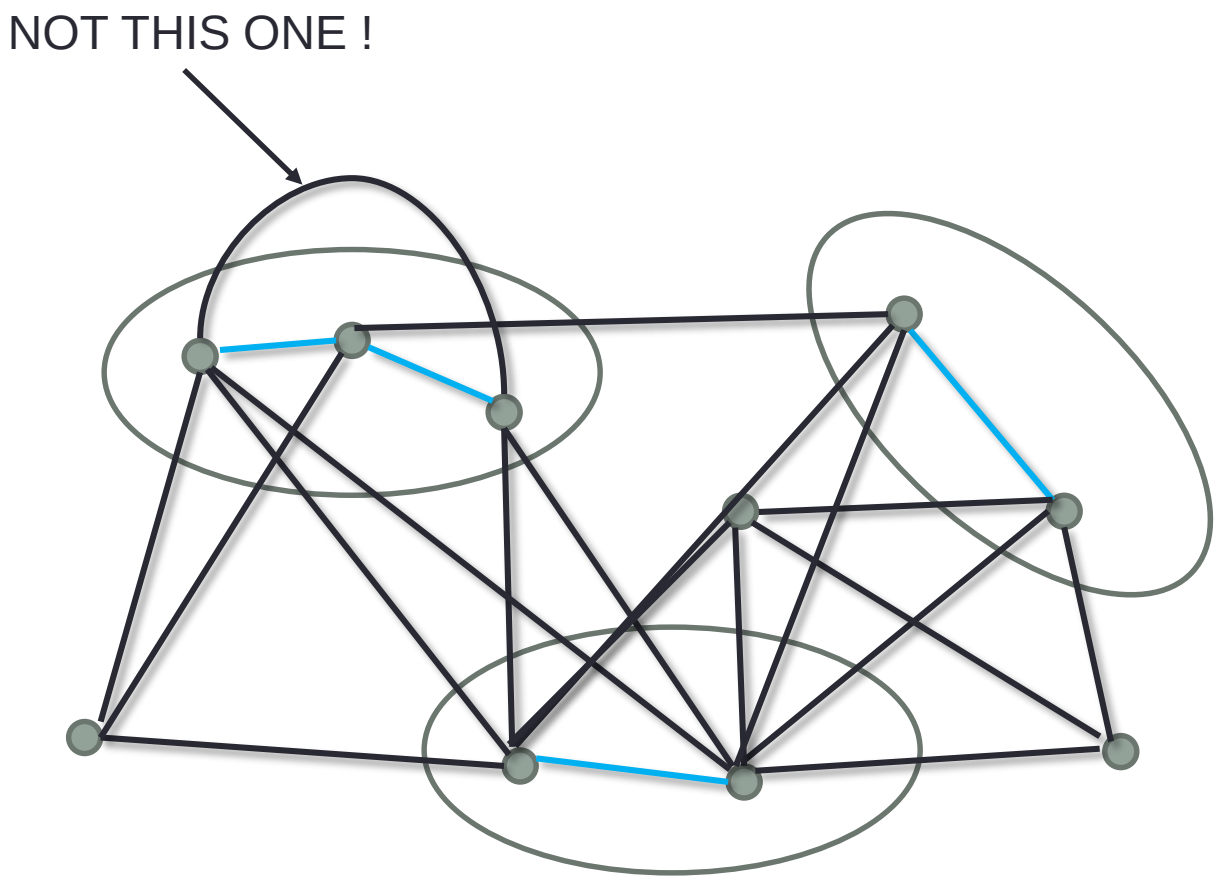
Heuristic

Since our goal is to connect AD connected components using Spec edges, take Spec edges that “link” two AD-components until there is possible choice left.



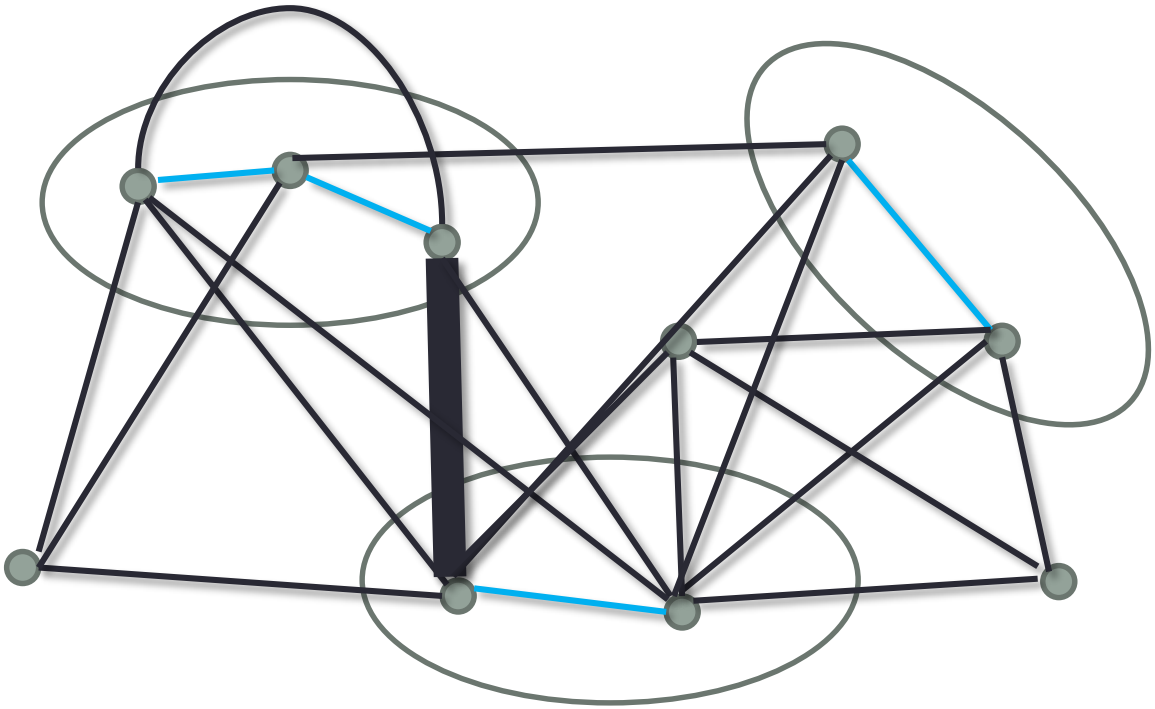
Heuristic

Since our goal is to connect AD connected components using Spec edges, take Spec edges that “link” two AD-components until there is possible choice left.



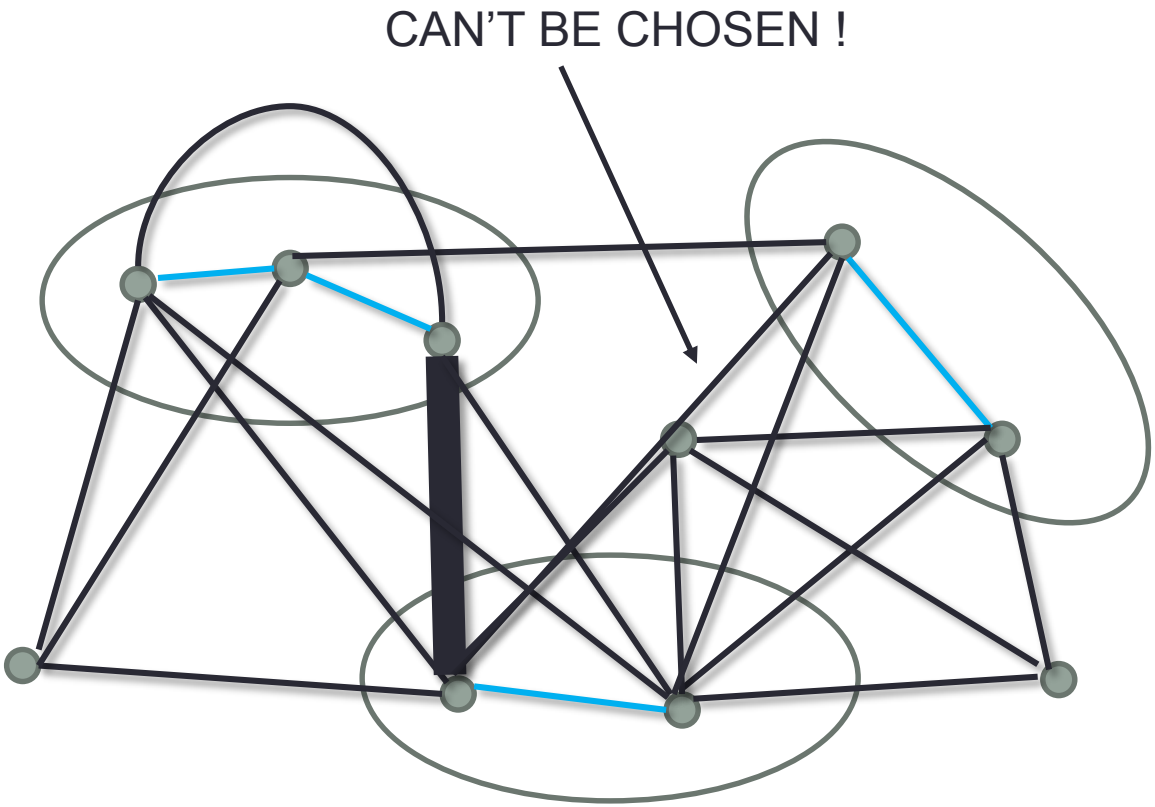
Heuristic

Since our goal is to connect AD connected components using Spec edges, take Spec edges that “link” two AD-components until there is possible choice left.



Heuristic

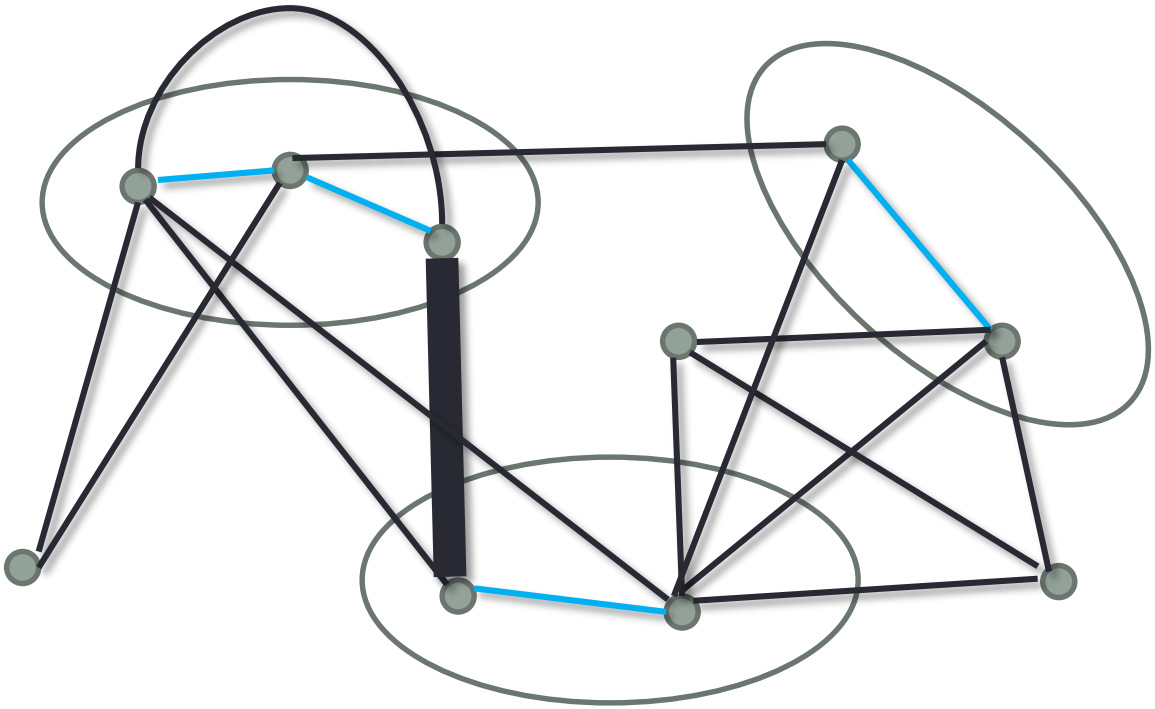
And since we are looking for cliques, remove edges that couldn’t form a clique with our chosen edge.



Heuristic

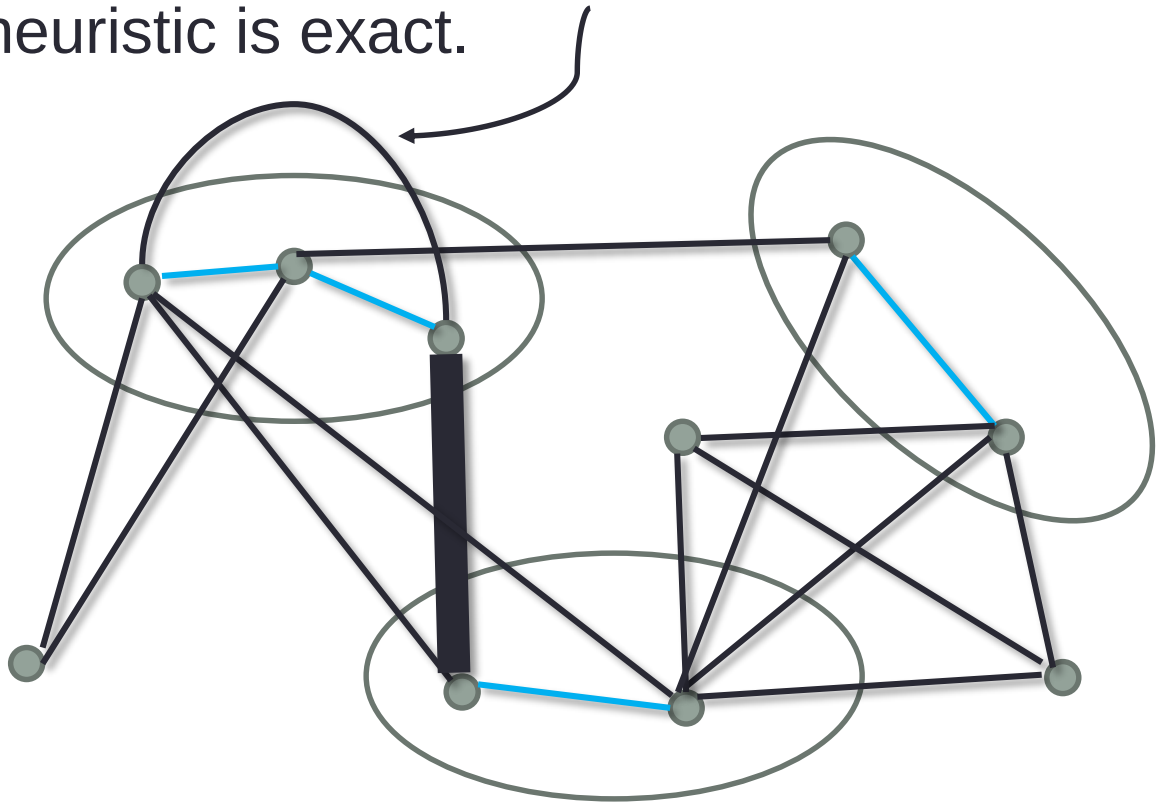
And since we are looking for cliques, remove edges that couldn't form a clique with our chosen edge.

Update the graph, and repeat.



Heuristic

- Bounds : using this idea, we developed a heuristic that can be **at most twice as bad** as the best solution (in terms of AD components connected)
- If the graph has no **Spec edge inside an AD-Component**, the heuristic is exact.



Random polytomy/species tree

We generated 1000 random polytomies having n subtrees for each n = 4..14 (along with a random species tree)

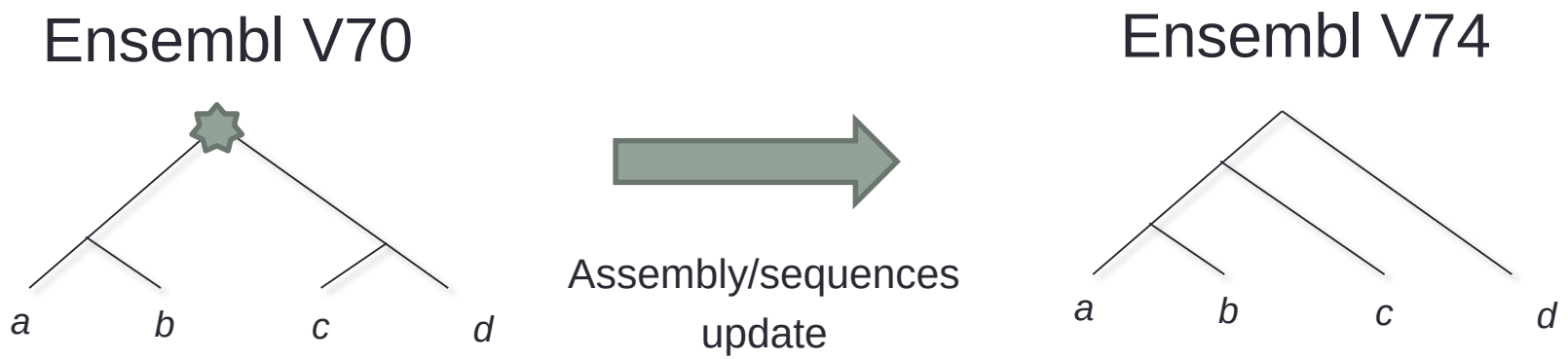
Heuristic vs Brute force

The heuristic **always** found a refinement with the minimum number of NADs.

Minimizing NADs vs # of duplications + losses

In 39,7% of random trees, finding a binary refinement the minimizes dups + losses **does not minimize the number of NADs created.**

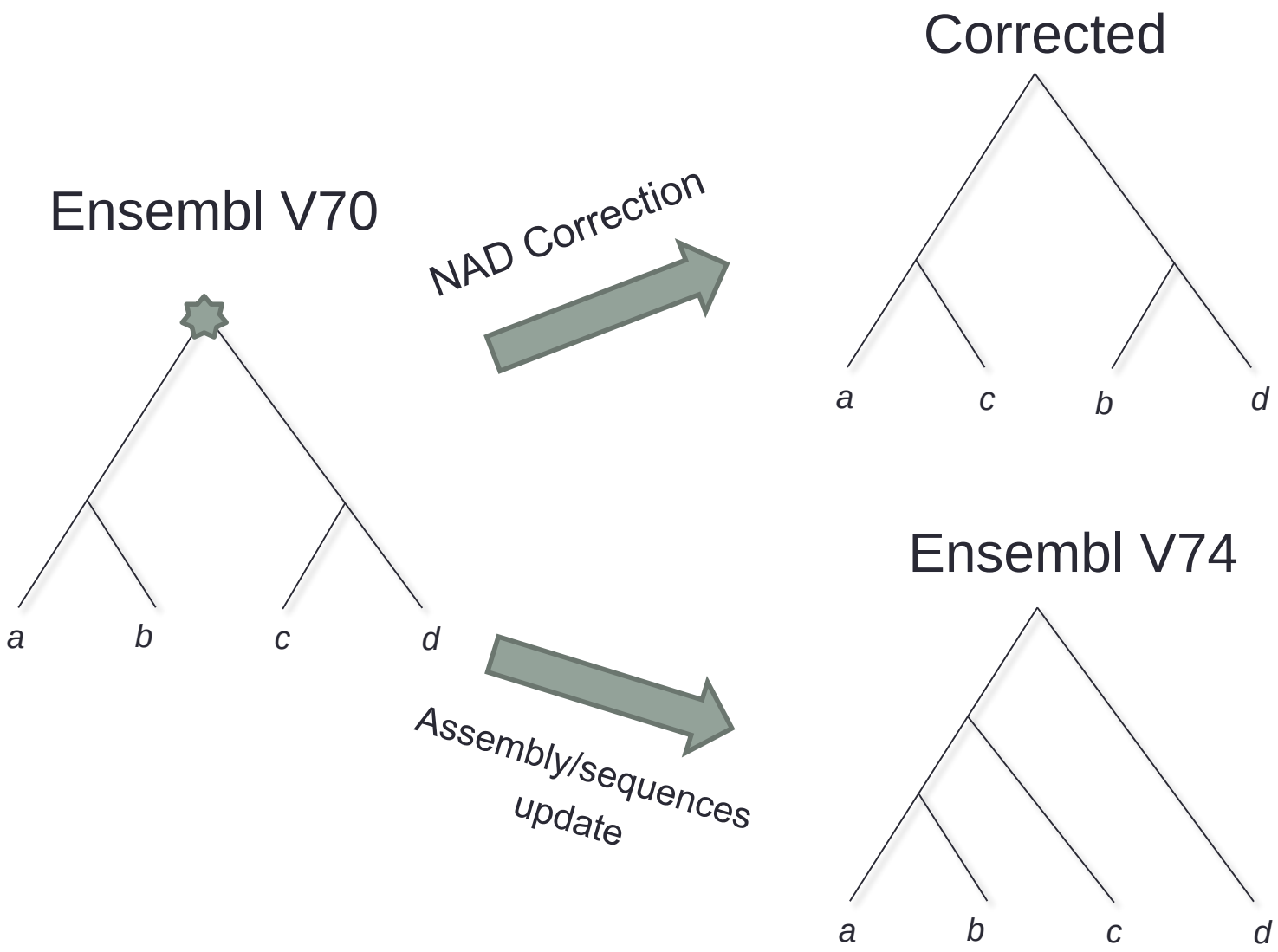
Ensembl updates : what happens to NADs ?



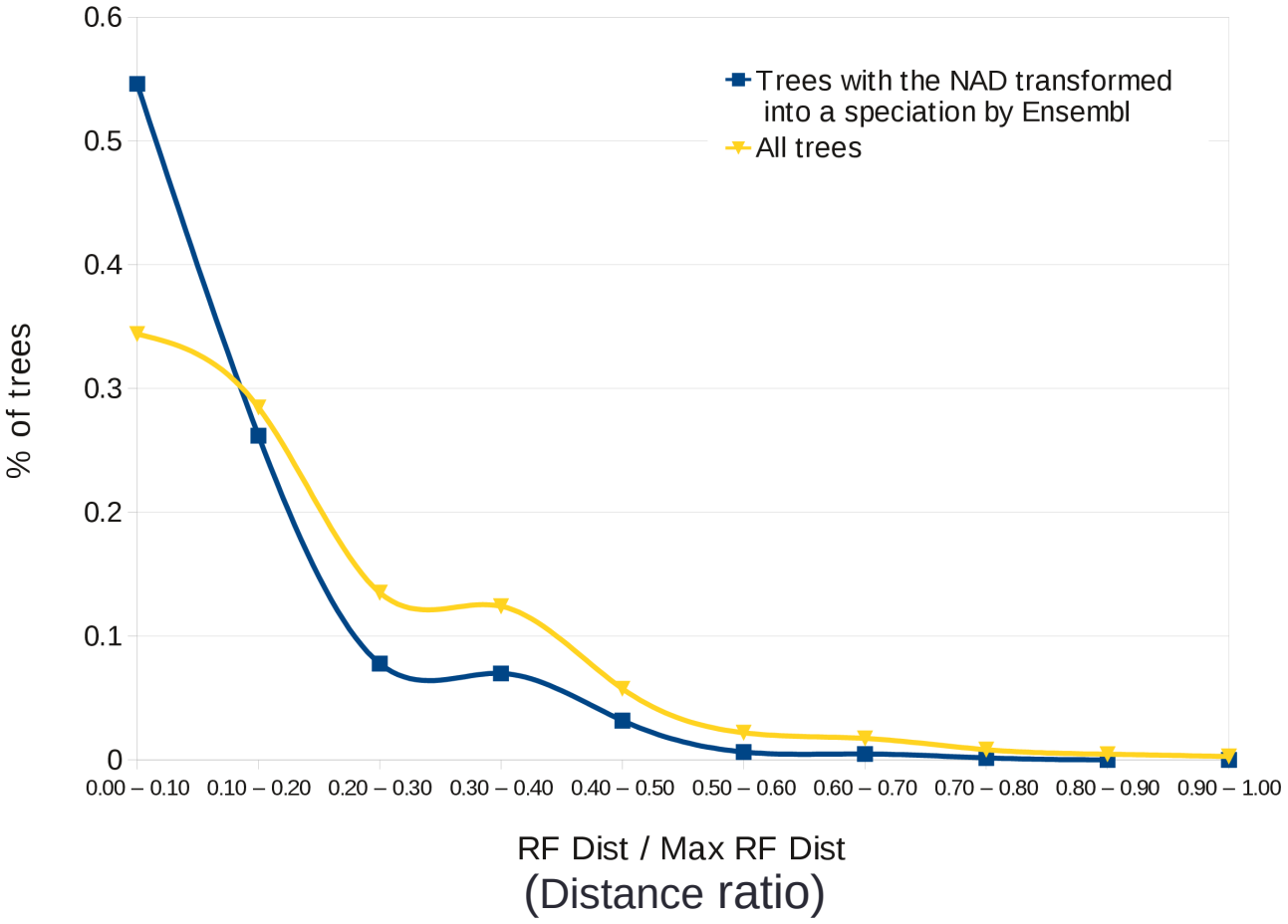
Events inferred at the root of NAD clades after Ensembl update
(993 trees of fish genes, highest NAD only)

NAD fate (v70 => v74)	% of NADs
NAD => Speciation	63.4 % (630 trees)
NAD => NAD	35.5 % (352 trees)
NAD => Apparent Duplication	1.1 % (11 trees)

Comparison with Ensembl updates



RF-Distance between Our correction vs Ensembl Updated Tree



65% of corrected trees share > 80% clades with Ensembl updated tree.

For those trees that Ensembl made

NAD => Spec,

44% are identical to our corrected tree.

Likelihood

We found 4454 NAD nodes in 1896 Ensembl fish gene trees.

For each tree T and each NAD node x

T_x is the tree obtained by correcting NAD node x

$R(x) = LogLH(T) / LogLH(T_x)$

43.9% of NAD nodes yielded a **better likelihood** ($R(x) > 1$) after correction

62.4% of the trees contained **at least one NAD** yielding a better likelihood after correction

Conclusion

- **When** does NAD correction/minimization apply ?

- Our heuristic builds a resolution that places duplications **as "high" as possible**. We should consider exploring other (or all) solutions.

- Is the problem **NP-Hard** ? Is there a polynomial time algorithm that solves it ?