CS345

Compact Skeletons

Compact Skeletons

- Assume tuples components are scattered over website
- We have a tagger that can tag all tuple components on website
  - Assume no noise for now
- Reconstruct relation

Compact Skeletons

Relation

↑

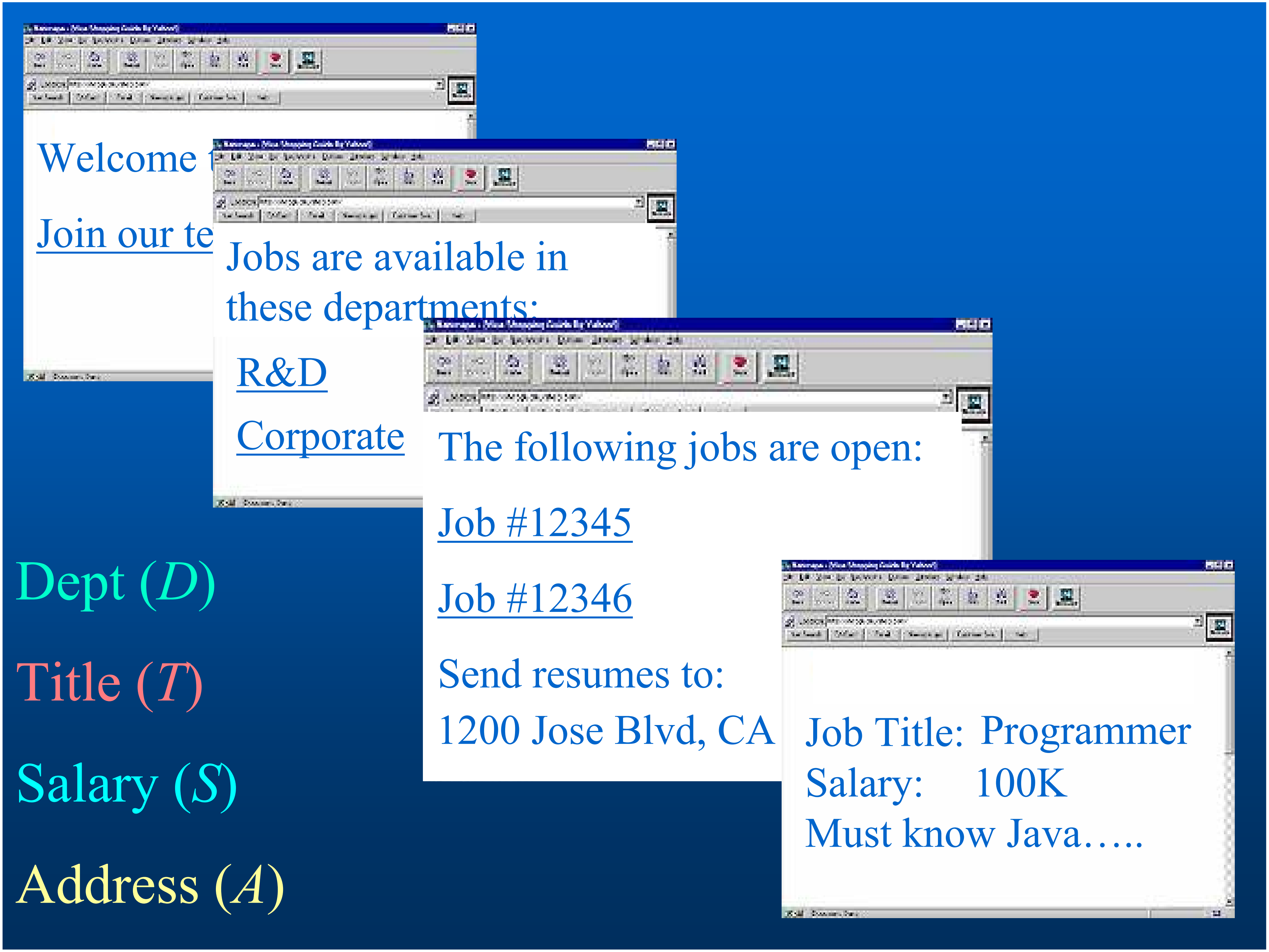
Skeleton

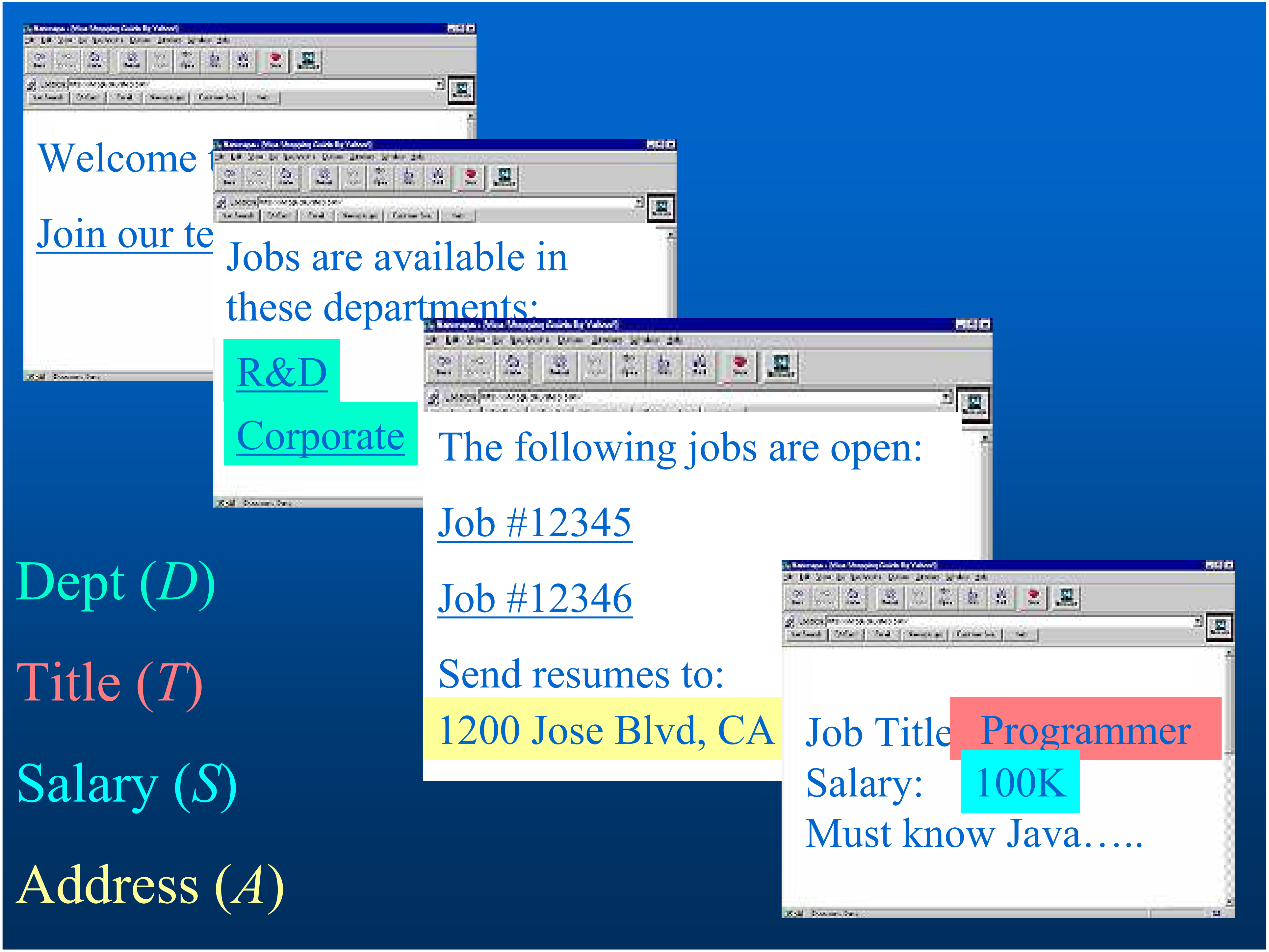
↑

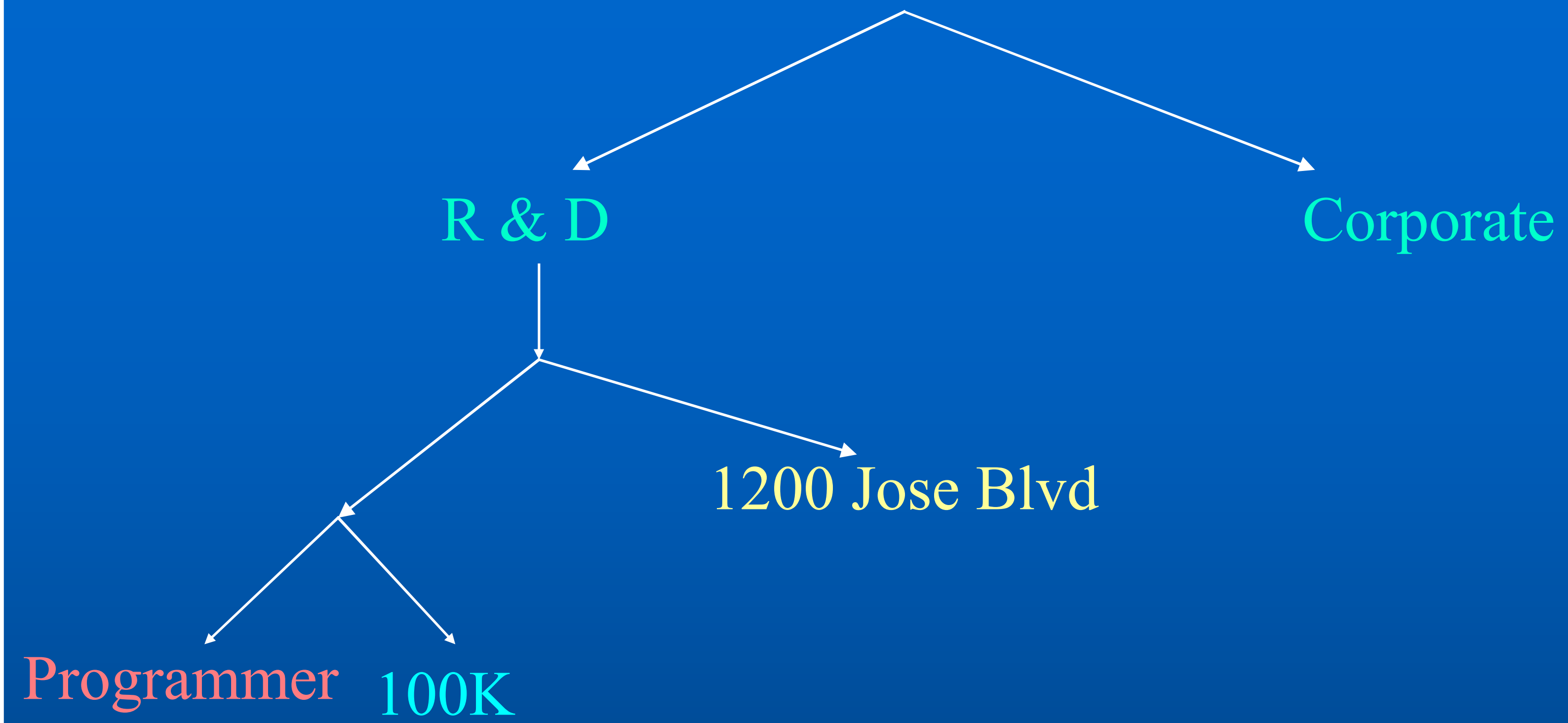
Data Graph

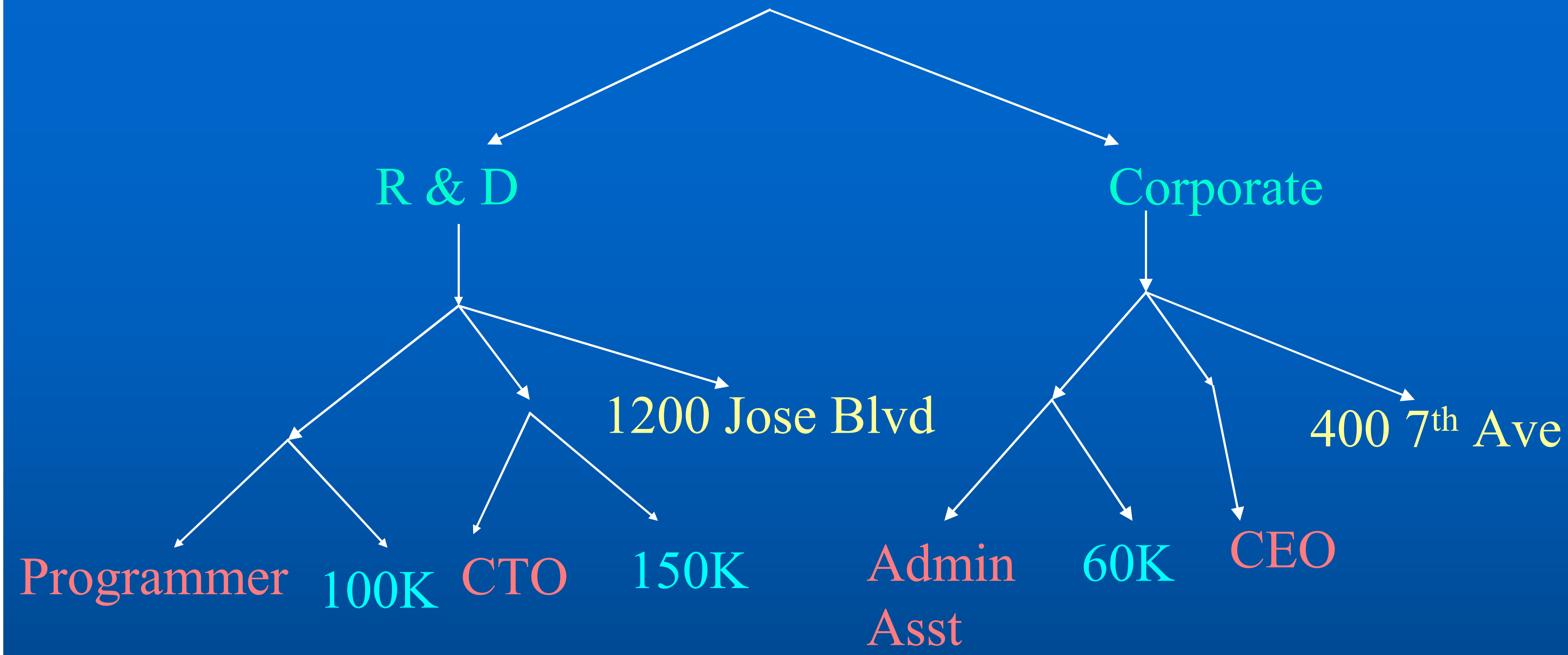
↑

Website

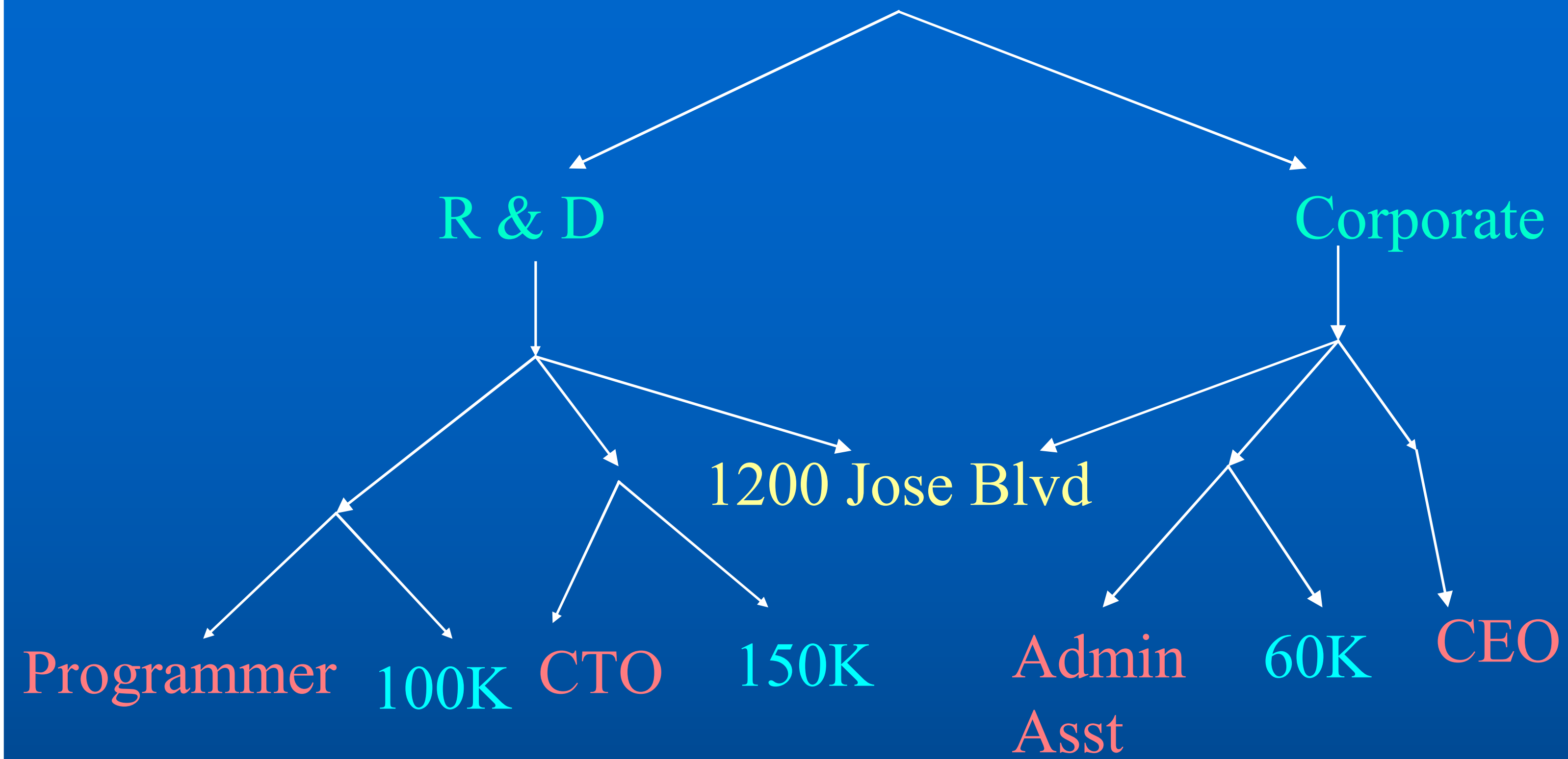

Welcome t
Join our te
Jobs are available in these departments:
R&D
Corporate
The following jobs are open:
Job #12345
Job #12346
Send resumes to:
1200 Jose Blvd, CA
Job Title: Programmer
Salary: 100K
Must know Java…..
Dept (D)
Title (T)
Salary (S)
Address (A)


Welcome t
Join our te
Jobs are available in these departments:
R&D
Corporate
The following jobs are open:
Job #12345
Job #12346
Send resumes to:
1200 Jose Blvd, CA
Job Title Programmer
Salary: 100K
Must know Java…..
Dept (D)
Title (T)
Salary (S)
Address (A)

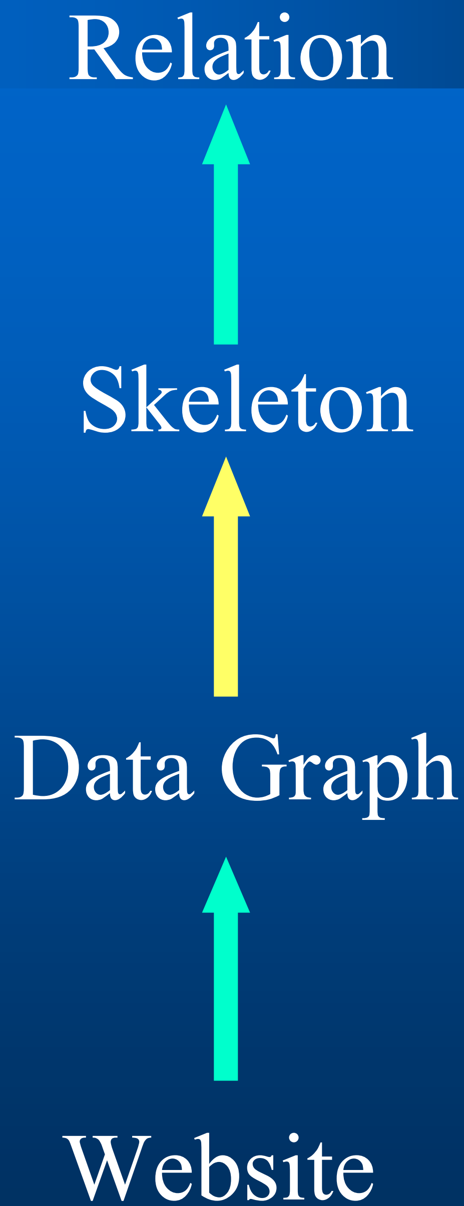

R & D
Corporate
1200 Jose Blvd
Programmer
100K



T	*S*	*D*	*A*
Programmer	100K	R &D	1200 Jose Blvd
CTO	150K	R & D	1200 Jose Blvd
Admin Asst	60K	Corporate	400 7th Ave
CEO	(null)	Corporate	400 7th Ave

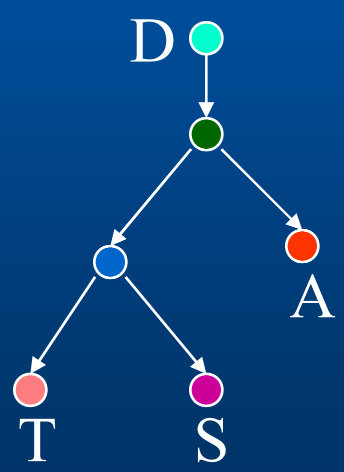


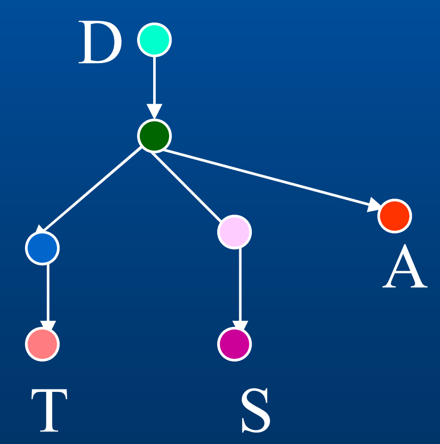
T	*S*	*D*	*A*
Programmer	100K	R &D	1200 Jose Blvd
CTO	150K	R & D	1200 Jose Blvd
Admin Asst	60K	Corporate	1200 Jose Blvd
CEO	(null)	Corporate	1200 Jose Blvd


Relation
Skeleton
Data Graph
Website

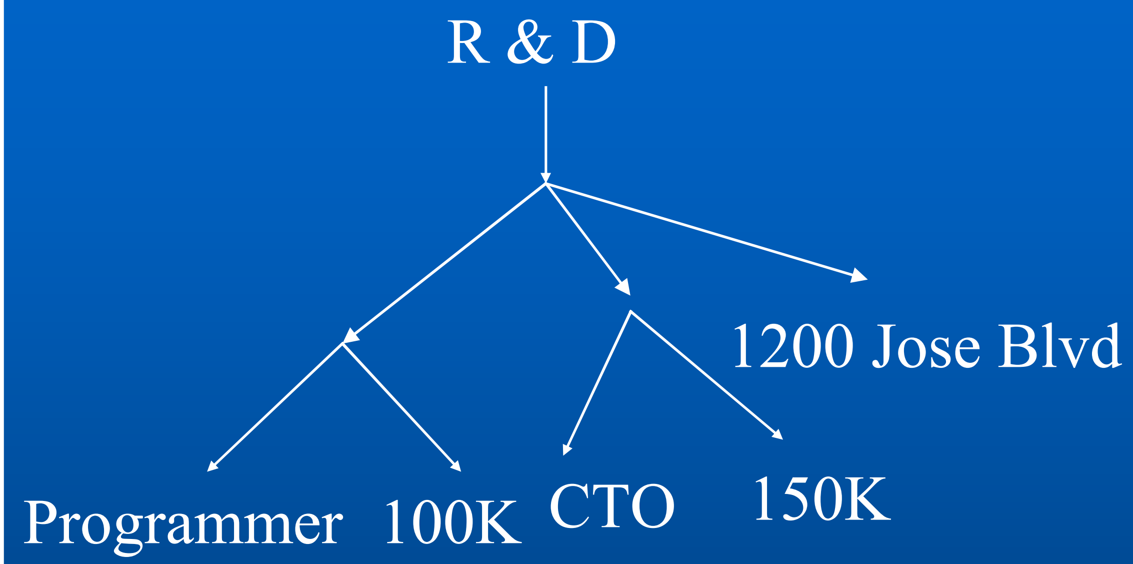
Skeletons

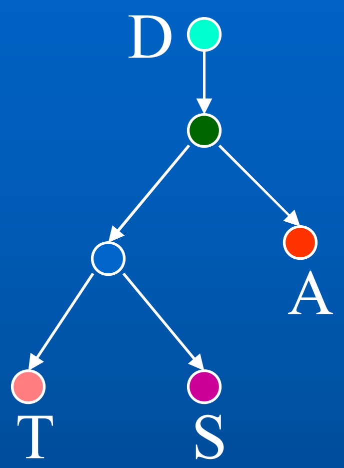
- Labeled trees
- Transformation from data graphs to relations



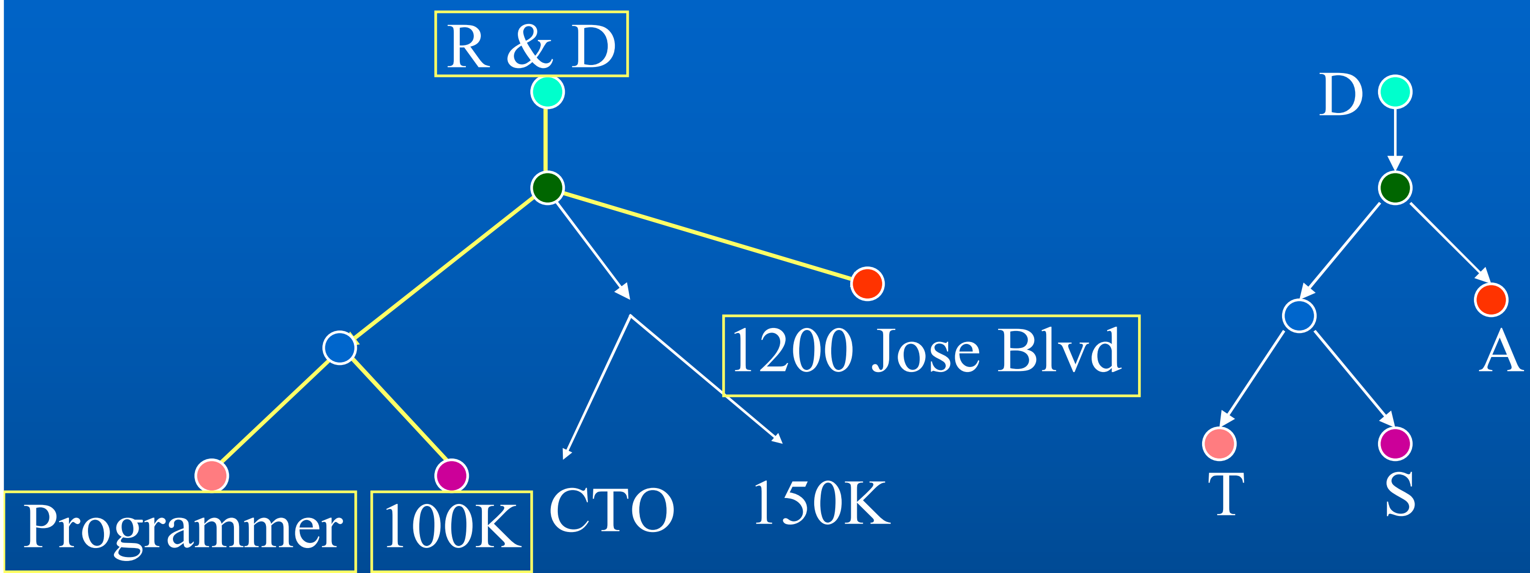


Overlays



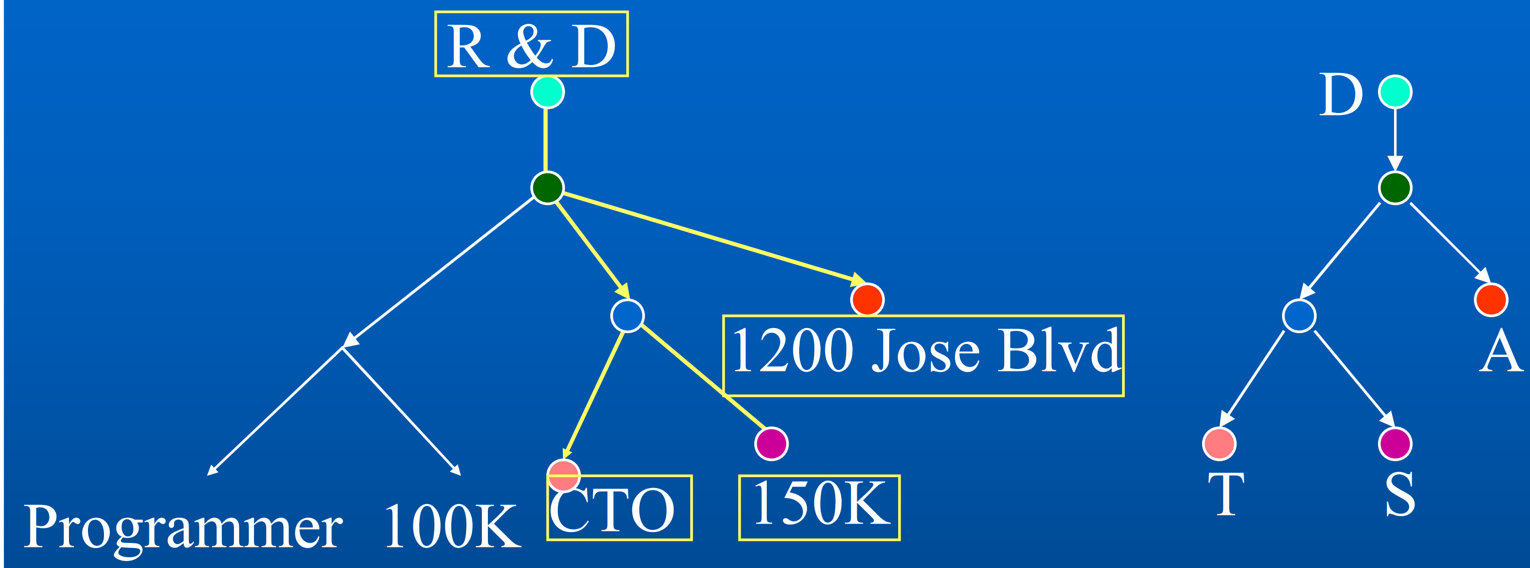


Overlays



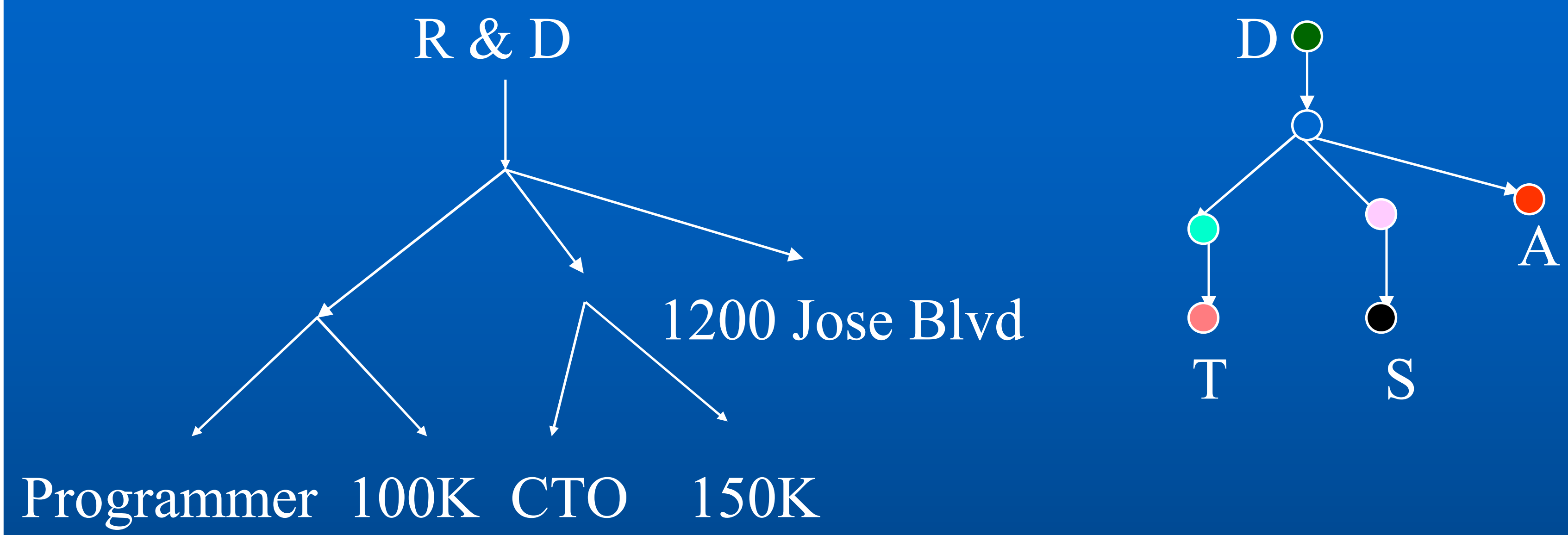
T	*S*	*D*	*A*
Programmer	100K	R &D	1200 Jose Blvd

Overlays

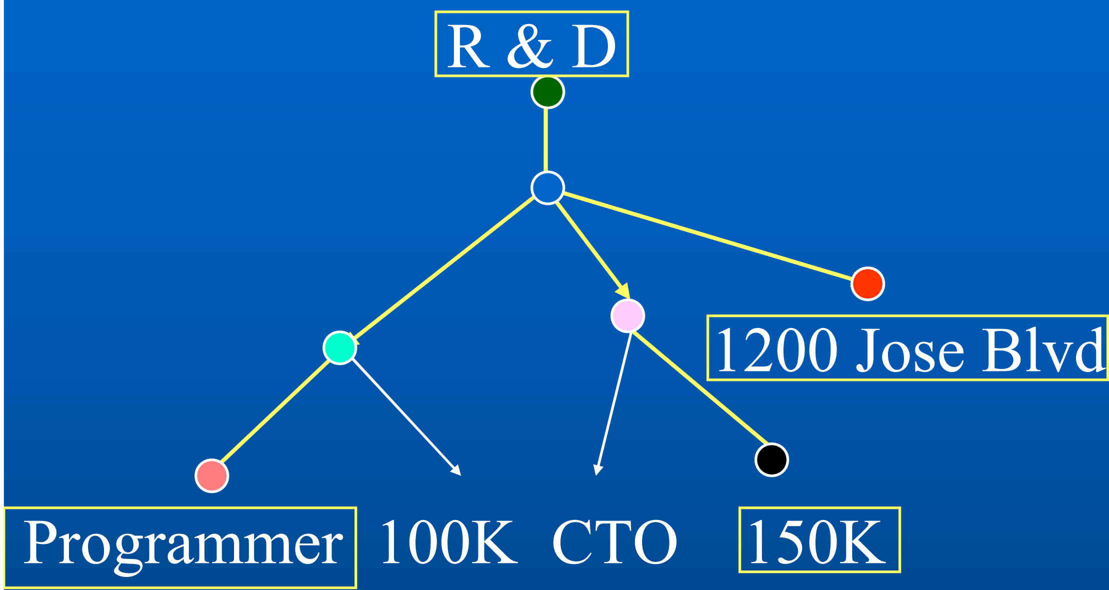


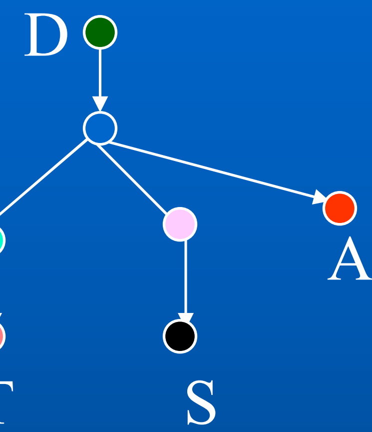
T	*S*	*D*	*A*
Programmer	100K	R &D	1200 Jose Blvd
CTO	150K	R &D	1200 Jose Blvd

Overlays



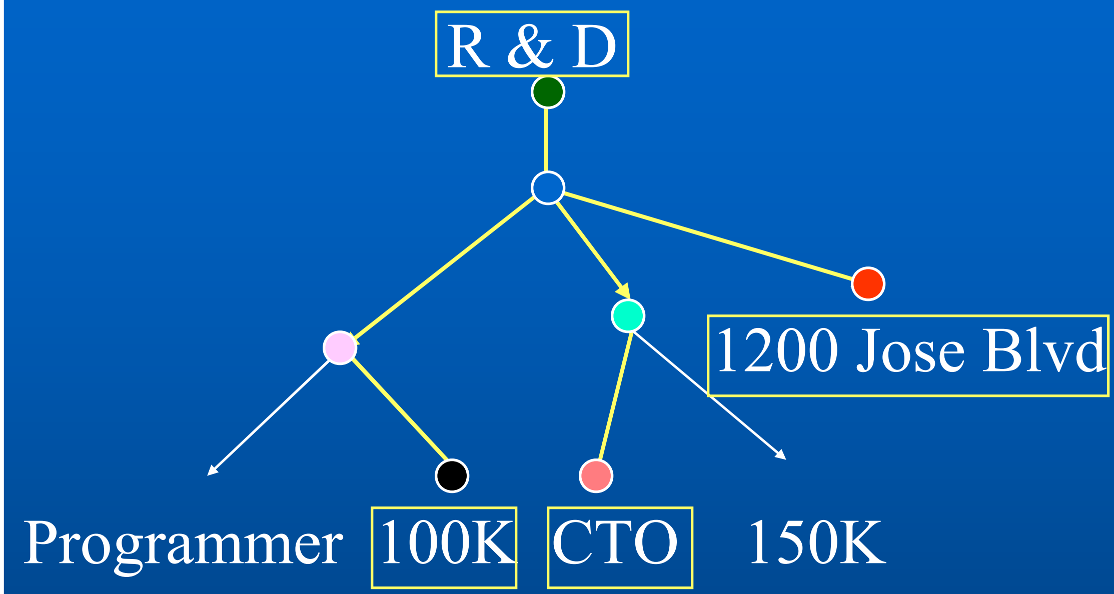
Overlays

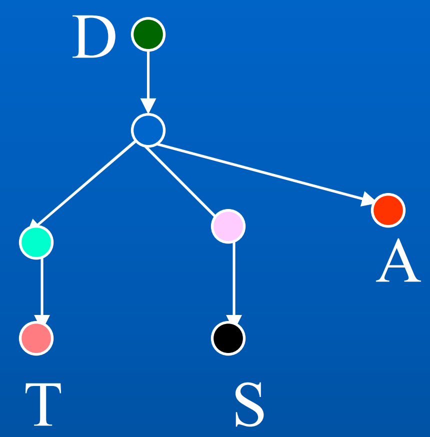




T	*S*	*D*	*A*
Programmer	150K	R &D	1200 Jose Blvd

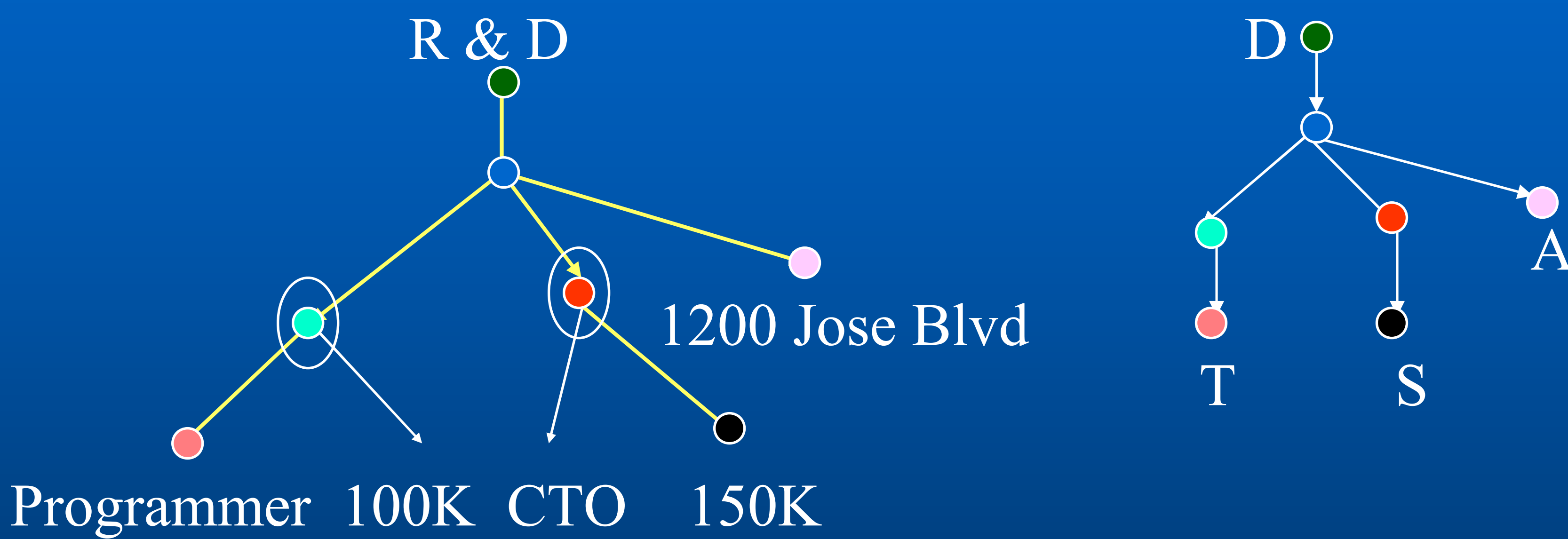
Overlays



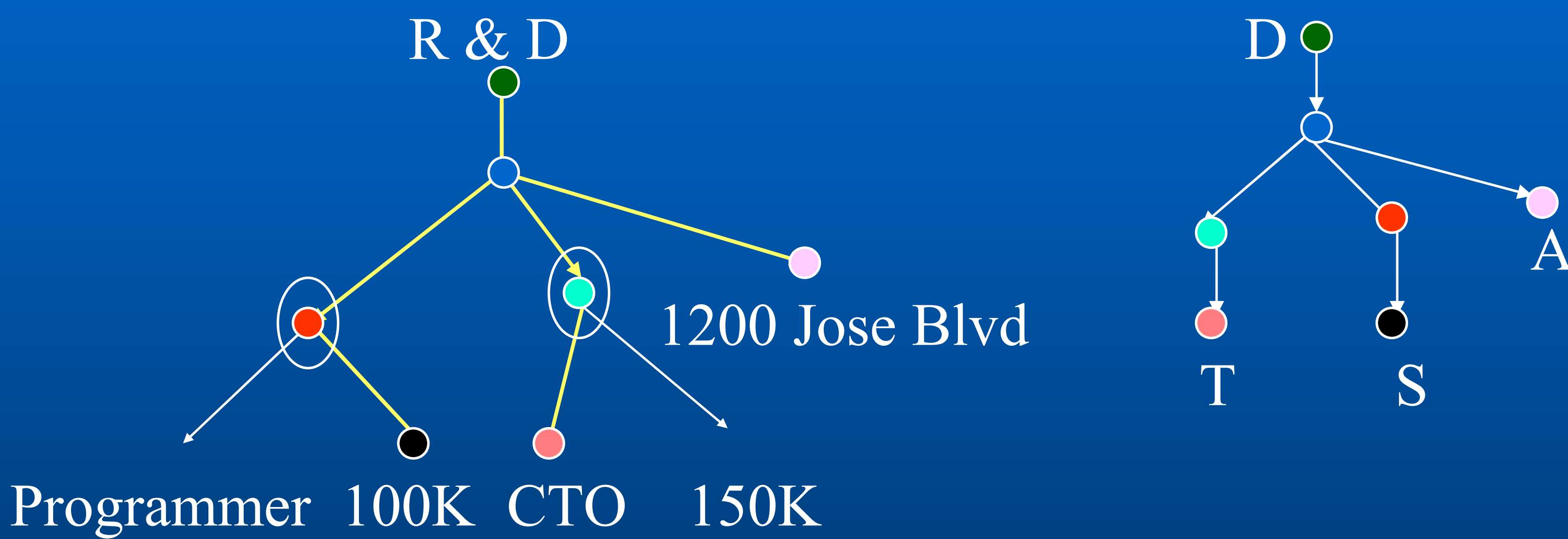


T	*S*	*D*	*A*
Programmer	150K	R &D	1200 Jose Blvd
CTO	100K	R & D	1200 Jose Blvd

Inconsistent Overlays



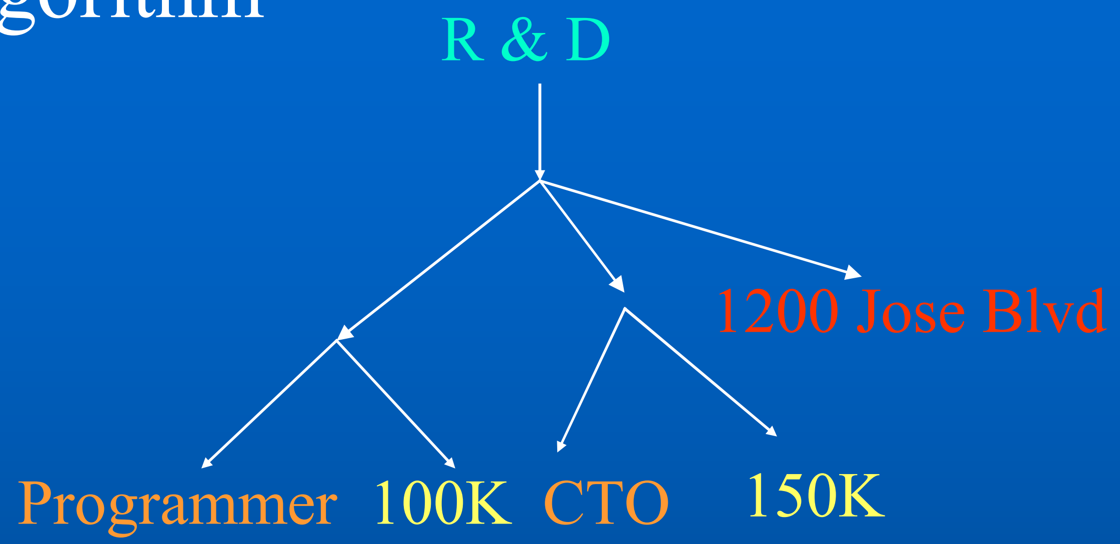
Inconsistent Overlays



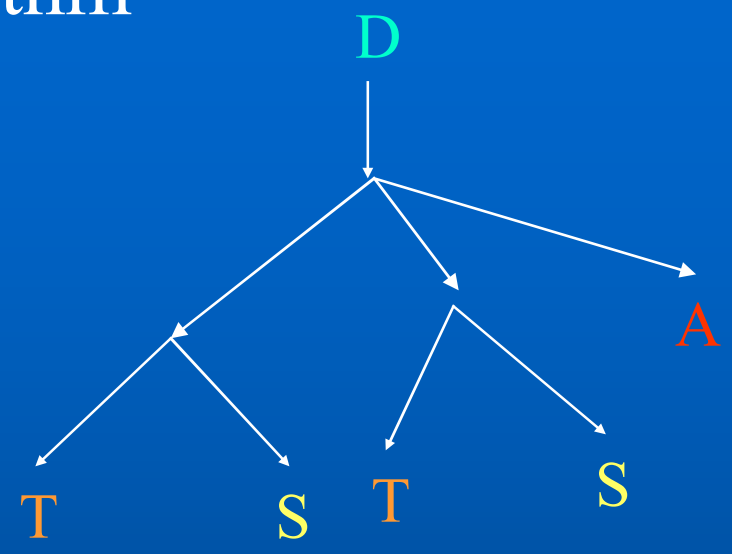
Compact Skeletons

- A skeleton is compact if all overlays are consistent
- Perfect if each node and edge of data graph is covered by at least one overlay
- Given a data graph G, does G have a Perfect Compact Skeleton (PCS)?
  - Not always
  - But if it exists it is unique

PCS Algorithm

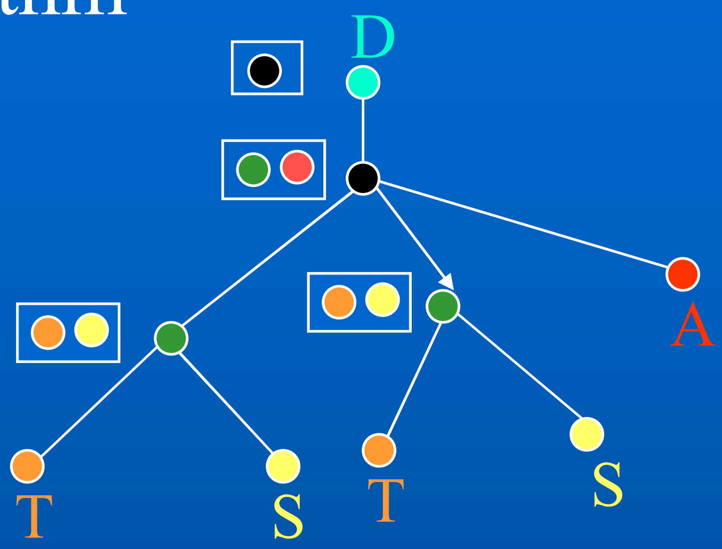


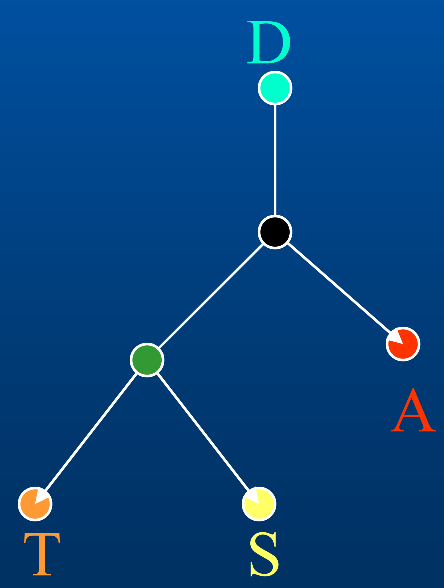
PCS Algorithm



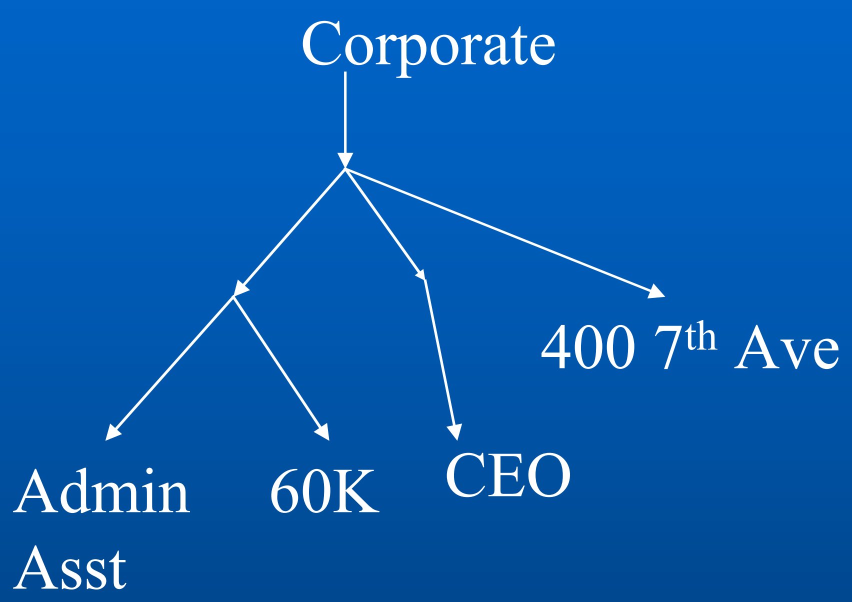
Work bottom-up:
Compute node signatures
Place nodes in equivalence classes based on signature
Construct skeleton from equivalence classes

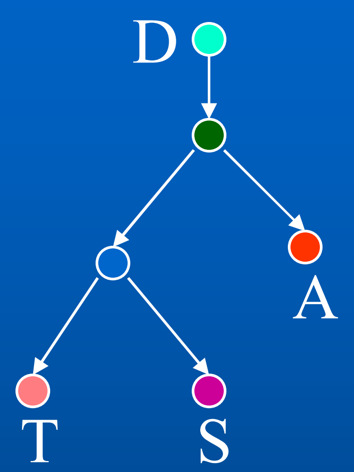
PCS Algorithm



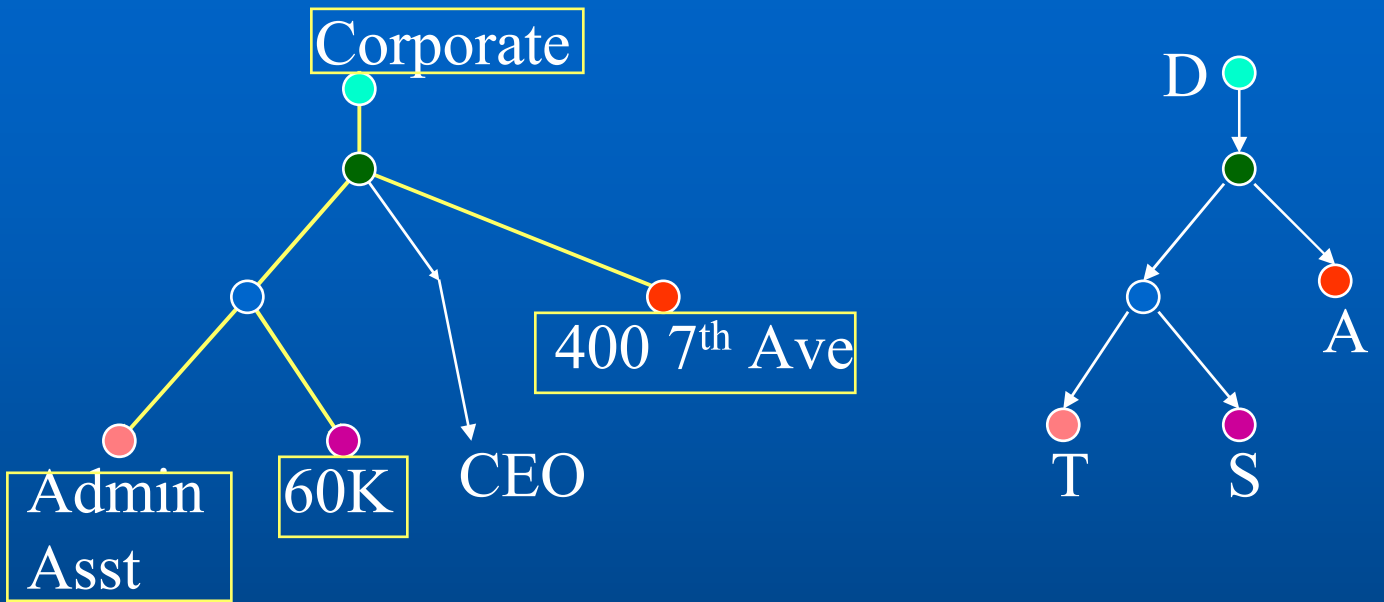


Incomplete information



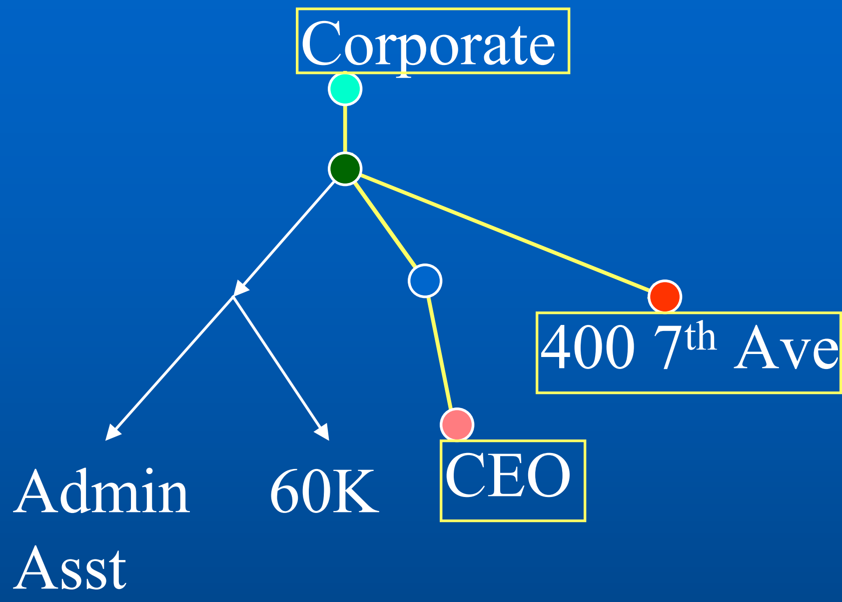


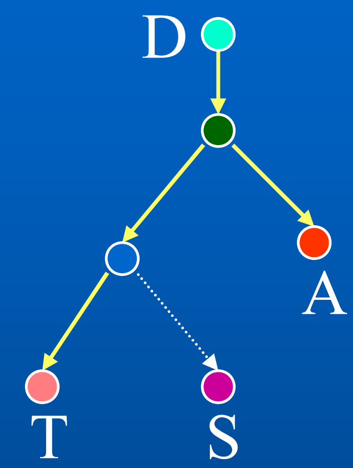
Incomplete information



T	*S*	*D*	*A*
Admin Asst	60K	Corporate	400 7th Ave

Incomplete information





T	*S*	*D*	*A*
Admin Asst	60K	Corporate	400 7th Ave
CEO	$\perp$	Corporate	400 7th Ave

Partial Compact Skeletons

- For data graphs with incomplete information, we allow partial overlays
  - Results in nulls in relation
- If we can use consistent partial overlays to cover every node and edge of the graph, we have a partially perfect compact skeleton (PPCS)

Tuple subsumption

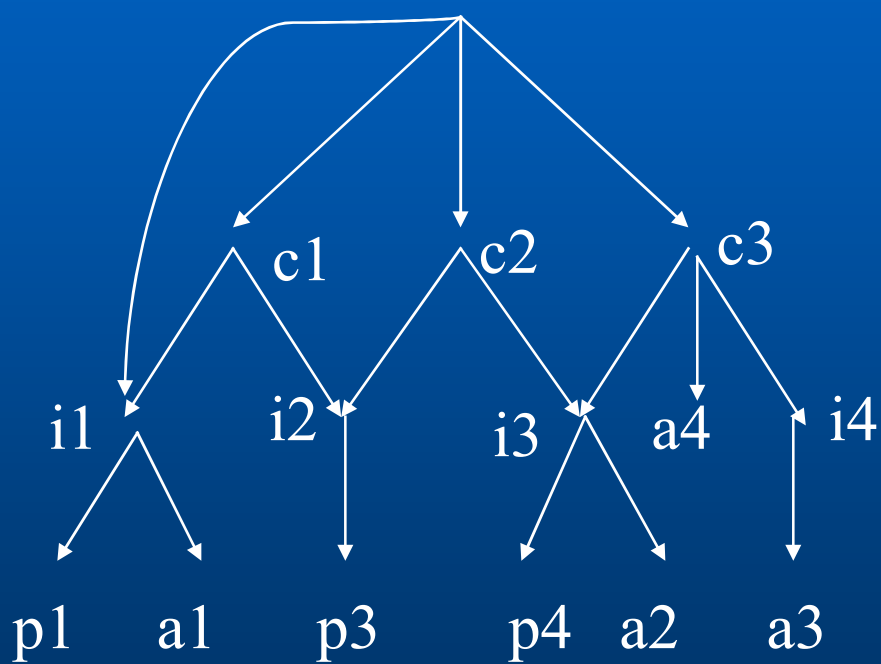
- **Tuple *t* subsumes tuple *u* if *t* and *u* agree on every component of *u* that is not null**

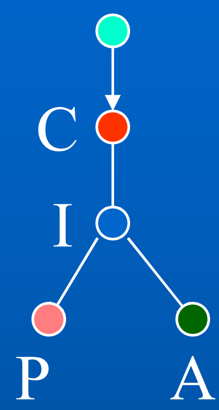
	T	S	D	A
t $\longrightarrow$	t_1	s_1	$\perp$	a_1
u $\longrightarrow$	t_1	$\perp$	$\perp$	a_1

Noisy Data Graphs

- Real-life websites are *noisy*
  - False positives e.g., MS = degree, state or Microsoft?
  - Non-skeleton links e.g., featured products

Data graph for a retail website



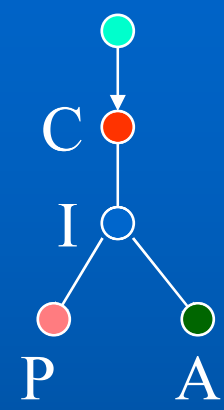


Skeleton K1

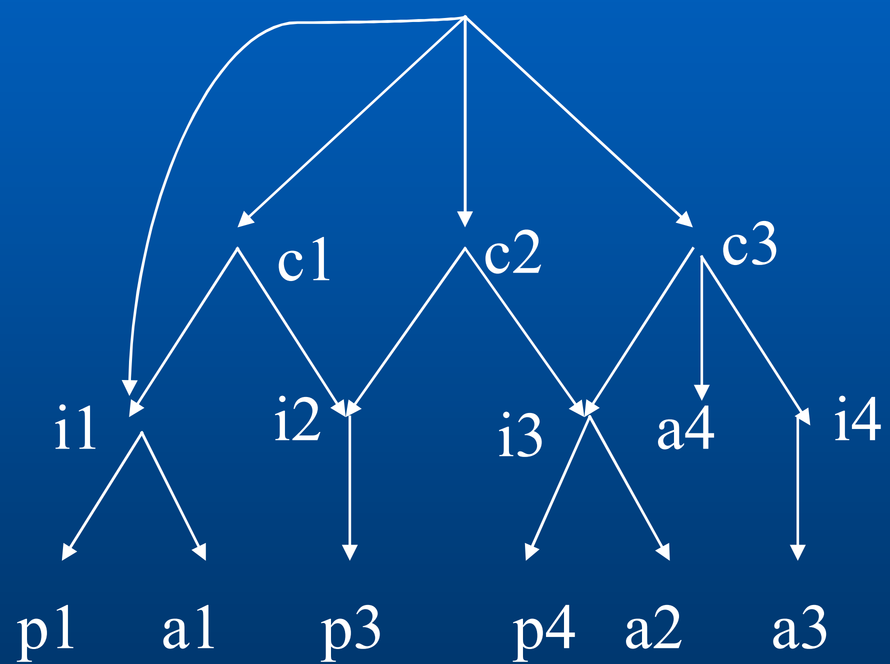
C: Category
I: Item
P: Price
A: Availability

For simplicity: assume all nodes have a label

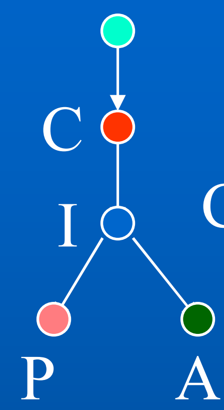
Coverage of a skeleton



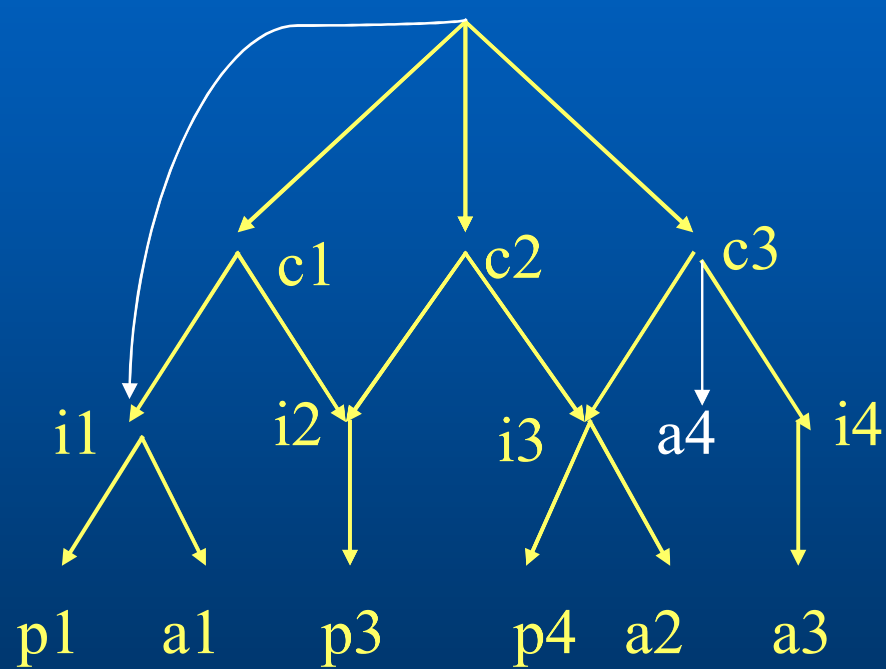
Skeleton K1



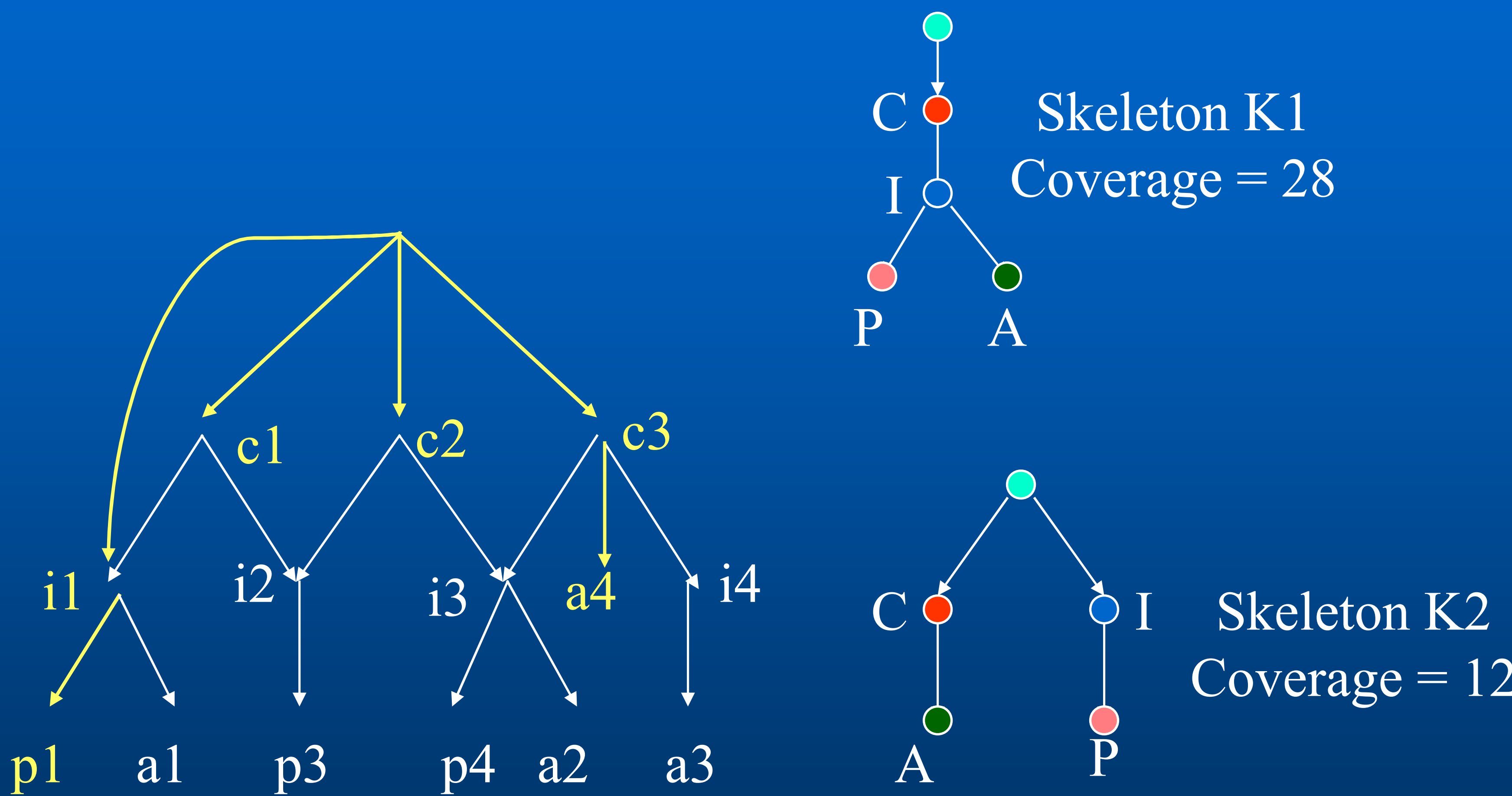
Coverage of a skeleton



Skeleton K1
Coverage = 28



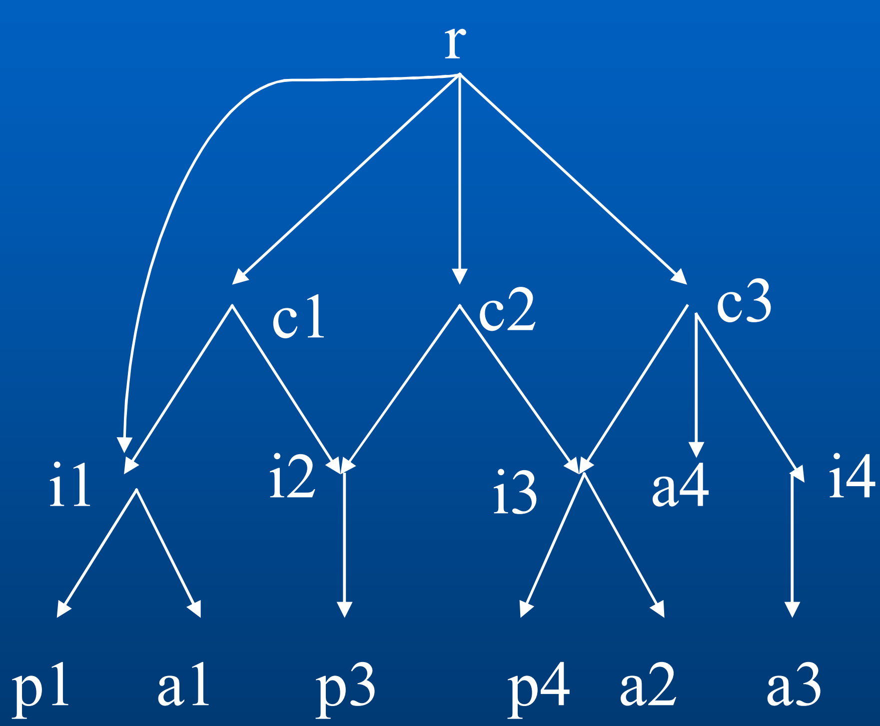
Coverage of a skeleton



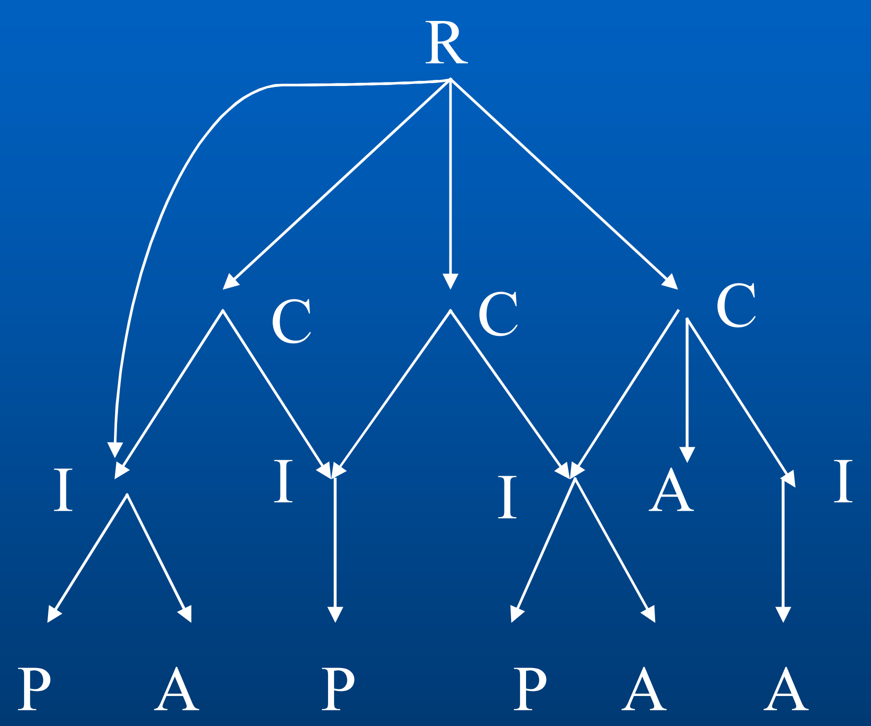
Skeletons for Noisy Data Graphs

- **Problem:**
  - **Find skeleton K with optimal coverage, called the best-fit skeleton (BFS)**
- **NP-complete**

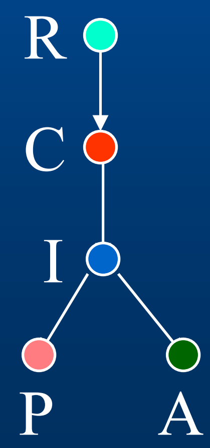
Greedy Heuristic for BFS

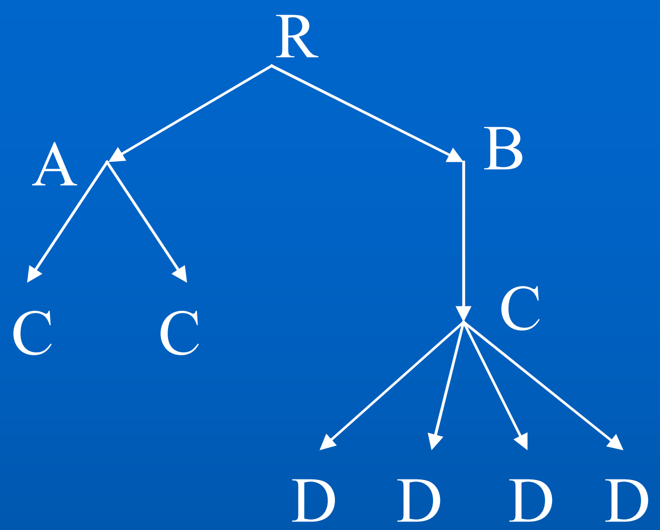


Greedy Heuristic for BFS

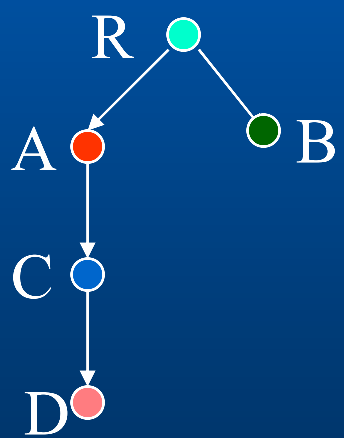


Label	Parent	Count
P	I	3
A	I	3
	C	1
I	C	4
	R	1
C	R	1

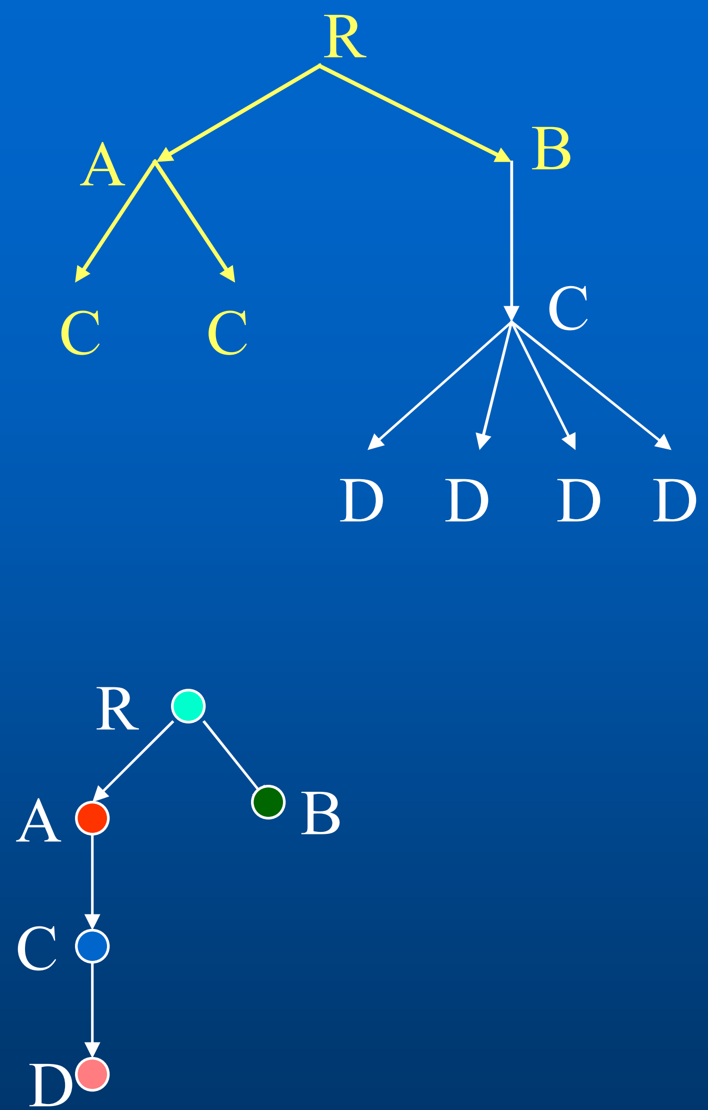




Label	Parent	Count
D	**C**	**4**
C	**A**	**2**
	B	**1**
A	**R**	**1**
B	**R**	**1**

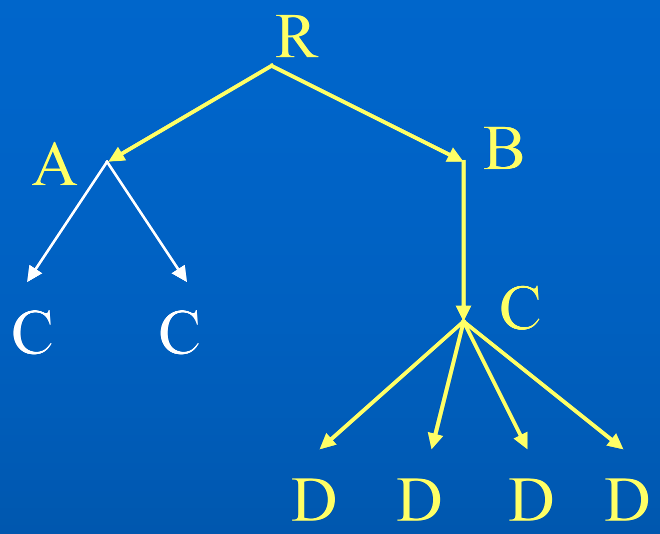


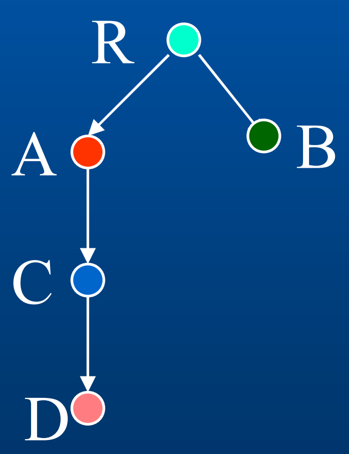
Greedy skeleton



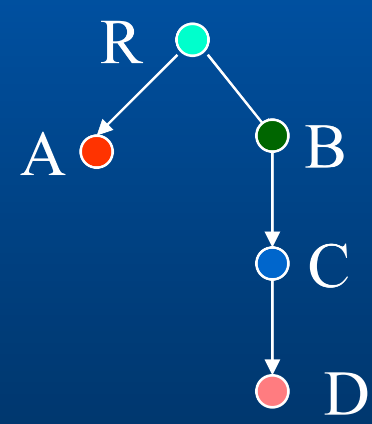
Greedy skeleton

Coverage = 9





Greedy skeleton
Coverage = 9

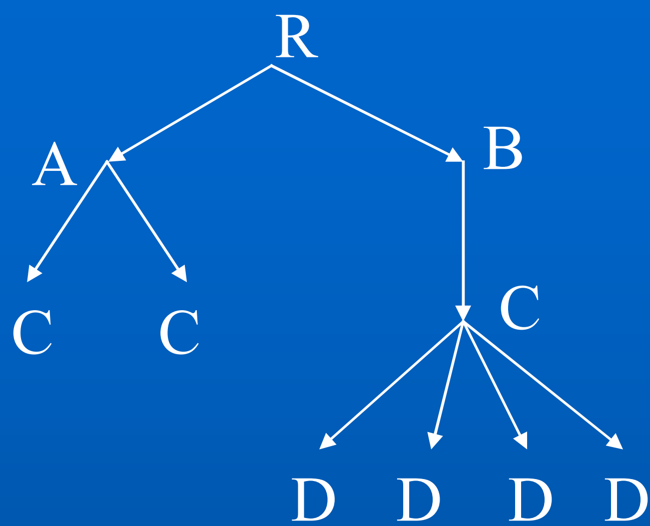


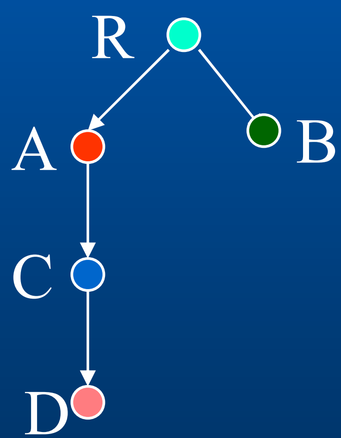
Optimal skeleton
Coverage = 15

Weighted Greedy Heuristic

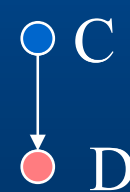
- Simple Greedy heuristic uses parent counts
  - "Memory-less"
- Weighted Greedy heuristic takes into account past selections to improve simple greedy selection
  - Computes "benefit" of each decision at every stage

Weighted Greedy



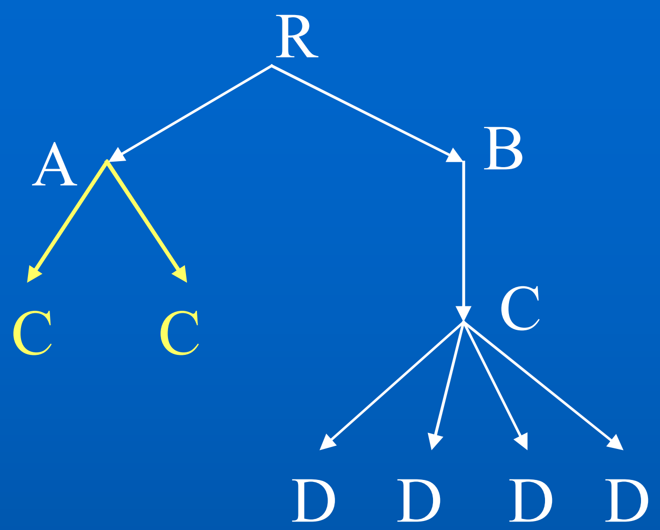


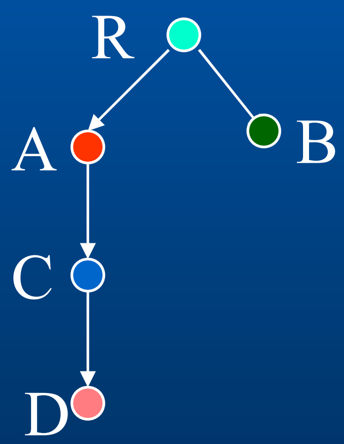
Greedy skeleton
Coverage = 9

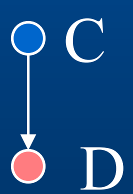


Weighted Greedy

benefit(A → C) = 4

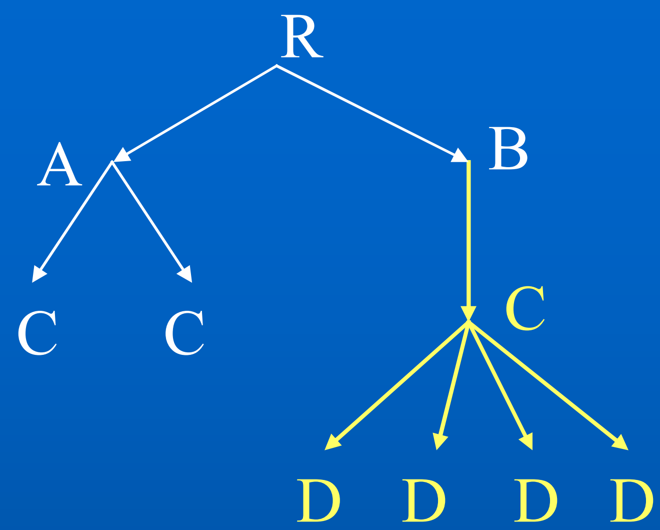




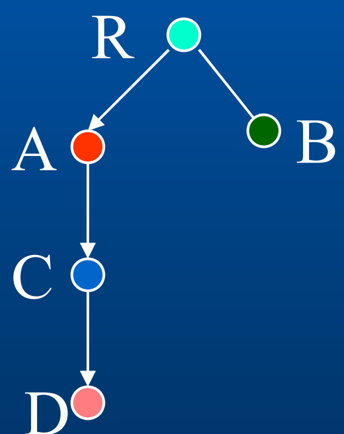


Greedy skeleton
Coverage = 9

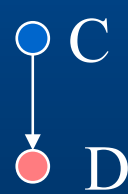
Weighted Greedy



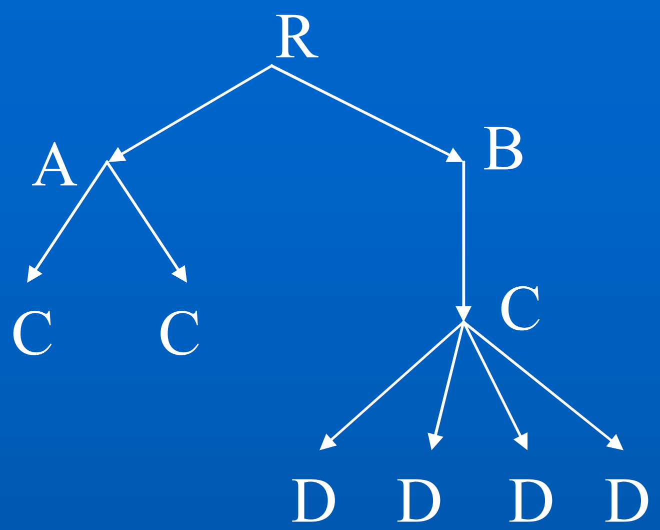
$$benefit(A \rightarrow C) = 4$$
$$benefit(B \rightarrow C) = 10$$

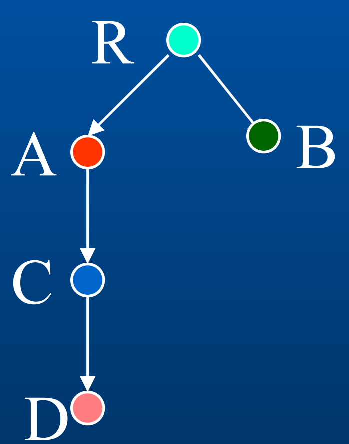


Greedy skeleton
Coverage = 9

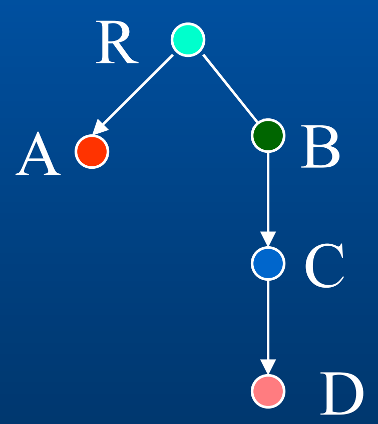


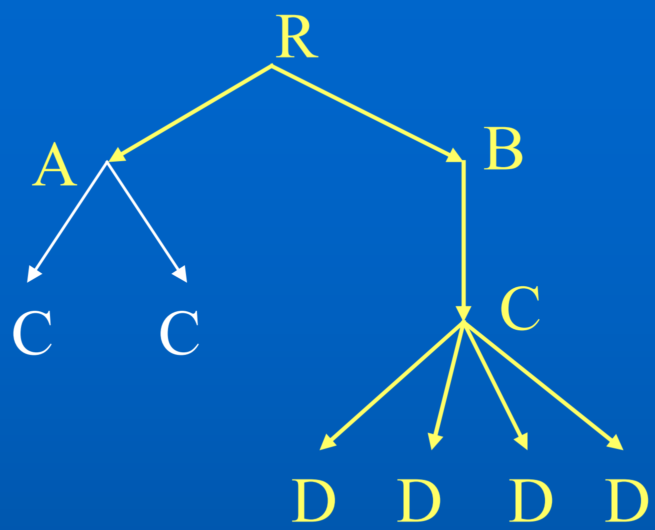
Weighted Greedy

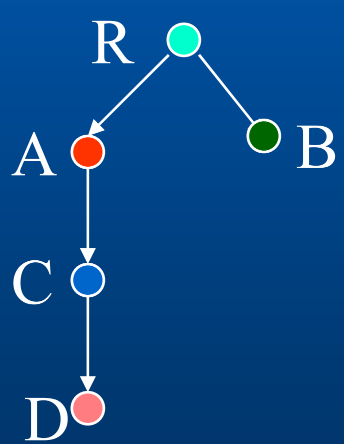




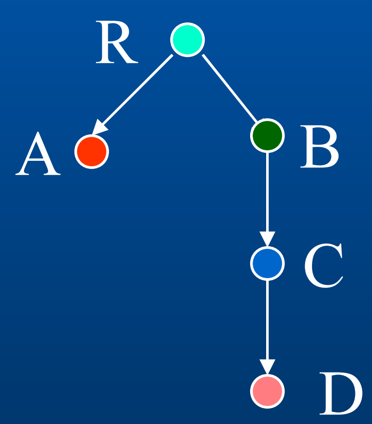
Greedy skeleton
Coverage = 9







Greedy skeleton
Coverage = 9



Weighted greedy skeleton
Coverage = 15

Summary

Relation

↑

Compact Skeleton

↑

Data Graph

↑

Website