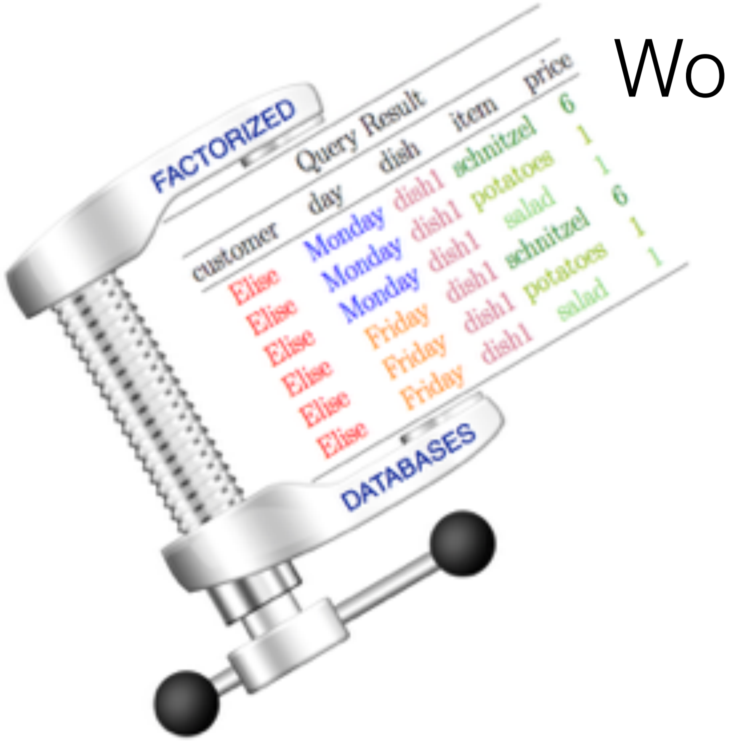
Covers of Query Results

Ahmet Kara

Dan Olteanu



customer	day	dish	item	price
Elise	Monday	dish1	schnitzel	6
Elise	Monday	dish1	potatoes	1
Elise	Monday	dish1	salad	1
Elise	Friday	dish1	schnitzel	6
Elise	Friday	dish1	potatoes	1
Elise	Friday	dish1	salad	1

Workshop on Aggregate Queries
Lille, June 2017

UNIVERSITY OF OXFORD

erc European Research Council

The Database of a Fast Food Restaurant

Orders

Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog
joe	friday	baguette

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4
cheese	3

The Database of a Fast Food Restaurant

Orders

Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog
joe	friday	baguette

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4
cheese	3

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

Redundancies in the Query Result

Orders

Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog
joe	friday	baguette

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4
cheese	3

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

Redundancies in the Query Result

Orders

Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog
joe	friday	baguette

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4
cheese	3

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

Searching for a Core of the Query Result

$$Q(\mathrm{Customer}, \mathrm{Day}, \mathrm{Dish}, \mathrm{Item}, \mathrm{Price}) = \mathrm{Orders} \bowtie \mathrm{Dishes} \bowtie \mathrm{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2

⋮ ⋮ ⋮ ⋮ ⋮

The Hypergraph of (a part of) the Query Result

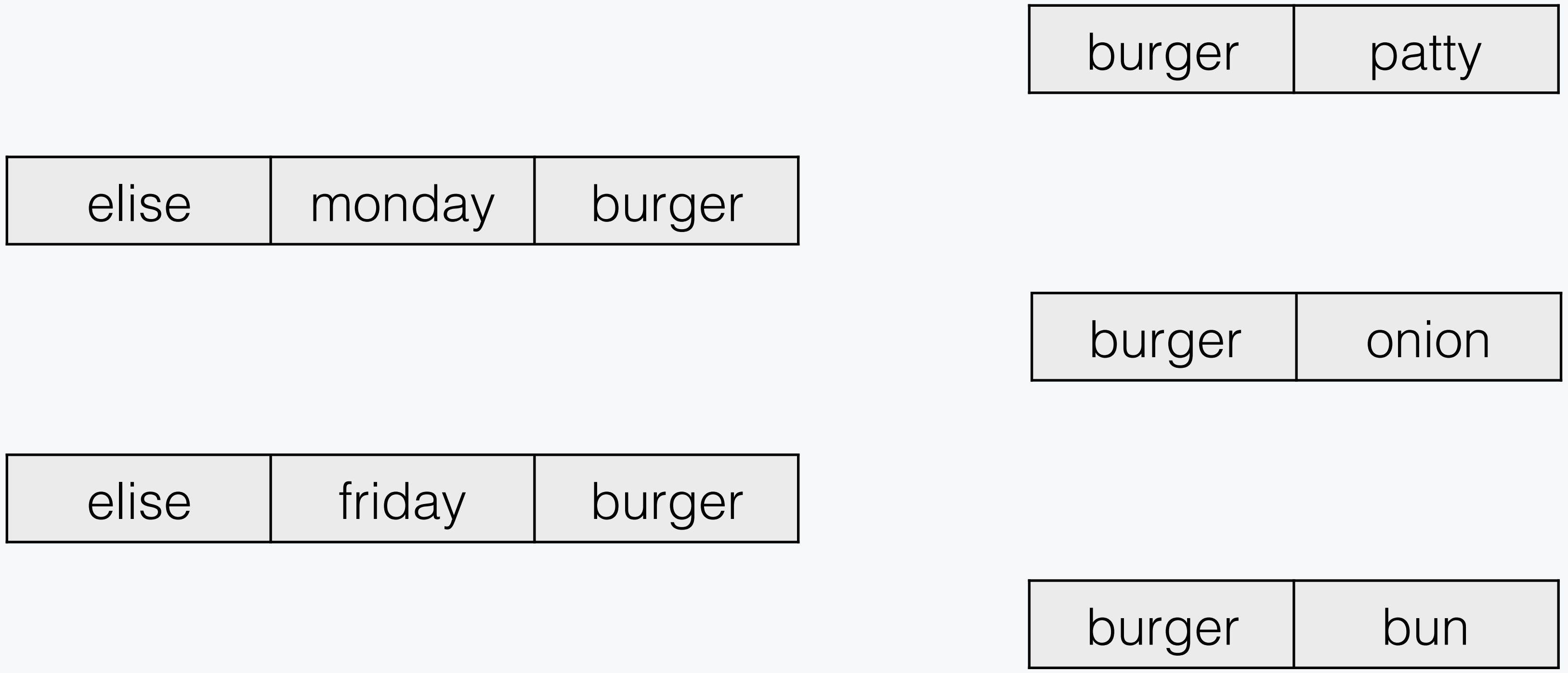
elise	monday	burger

elise	friday	burger

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

The Hypergraph of (a part of) the Query Result



burger | patty

elise | monday | burger

burger | onion

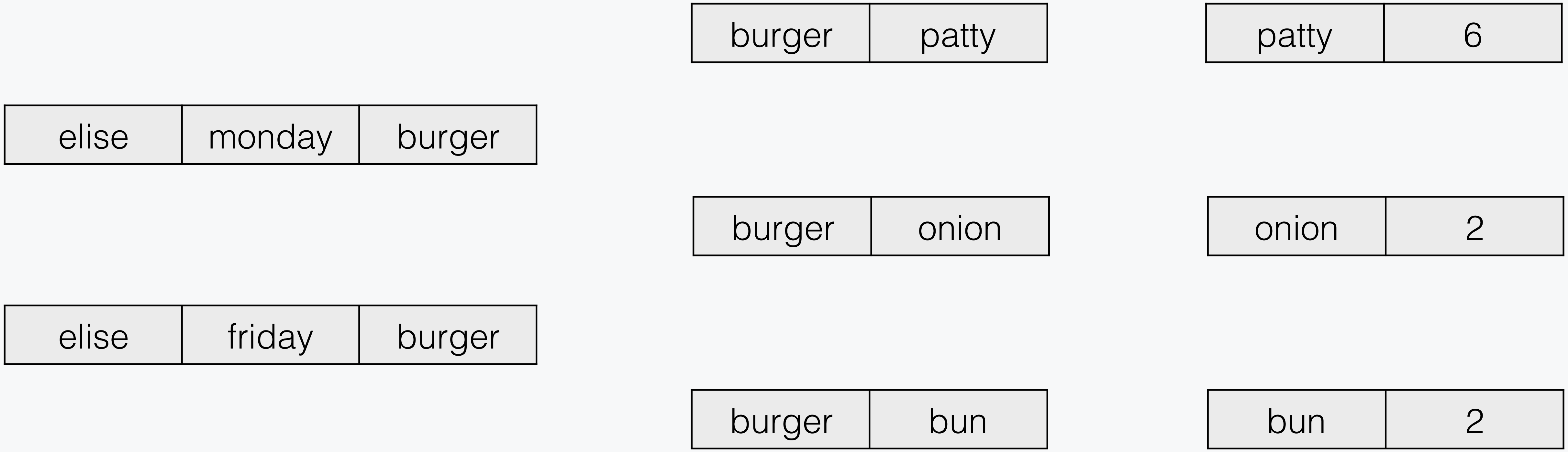
elise | friday | burger

burger | bun

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

The Hypergraph of (a part of) the Query Result



burger | patty

patty | 6

elise | monday | burger

burger | onion

onion | 2

elise | friday | burger

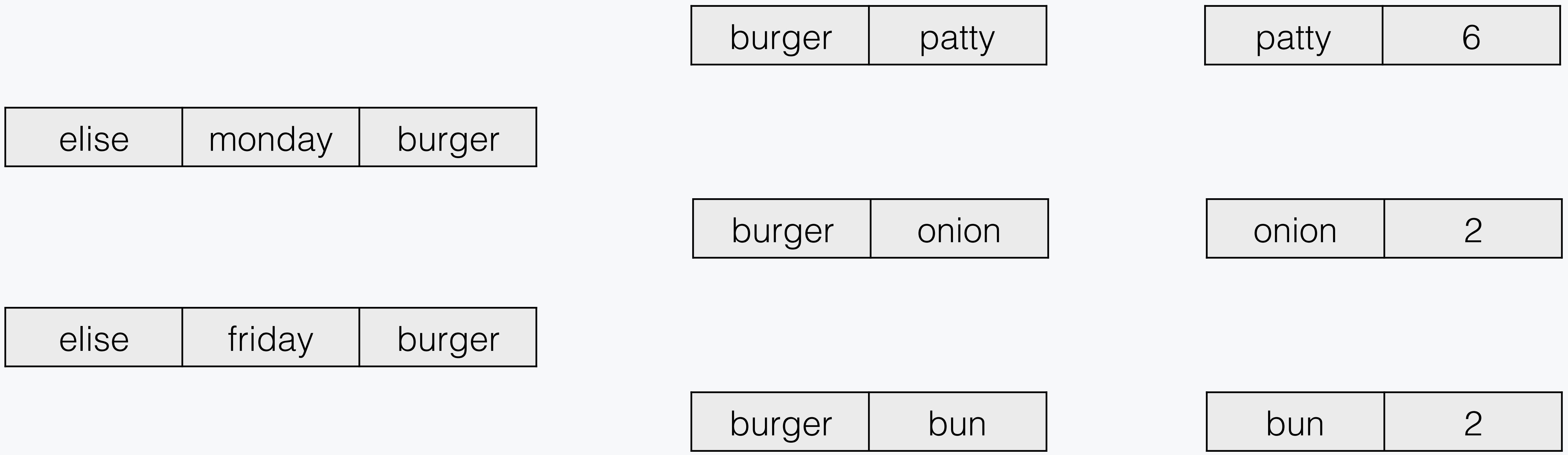
burger | bun

bun | 2

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

The Hypergraph of (a part of) the Query Result



burger | patty

patty | 6

elise | monday | burger

burger | onion

onion | 2

elise | friday | burger

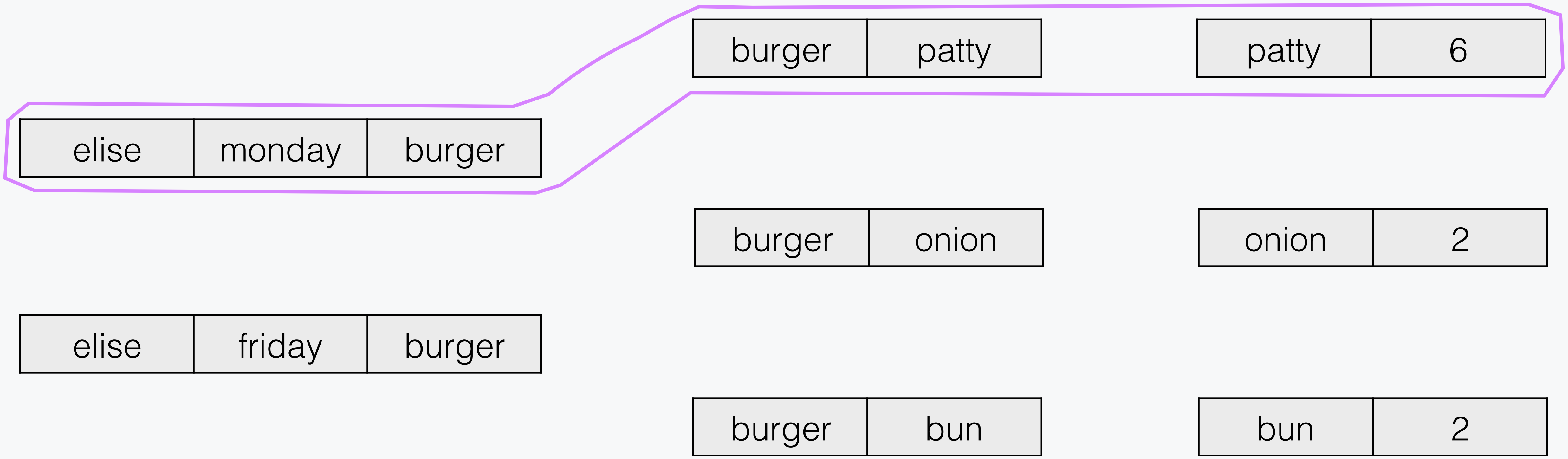
burger | bun

bun | 2

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

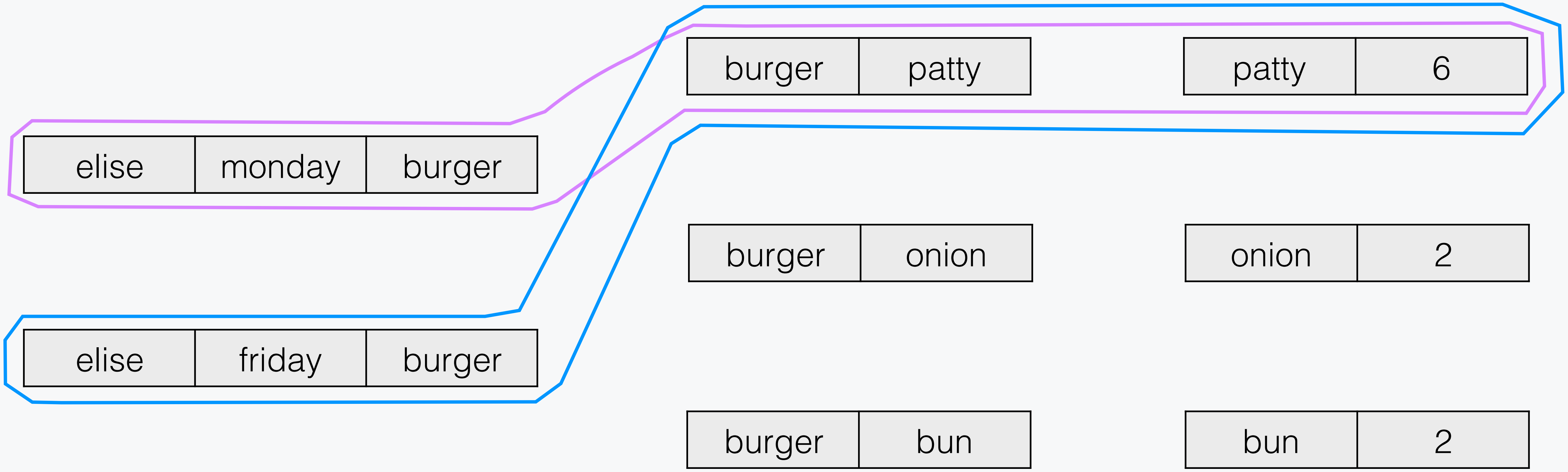
The Hypergraph of (a part of) the Query Result



$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

The Hypergraph of (a part of) the Query Result



elise | monday | burger

elise | friday | burger

burger | patty

burger | onion

burger | bun

patty | 6

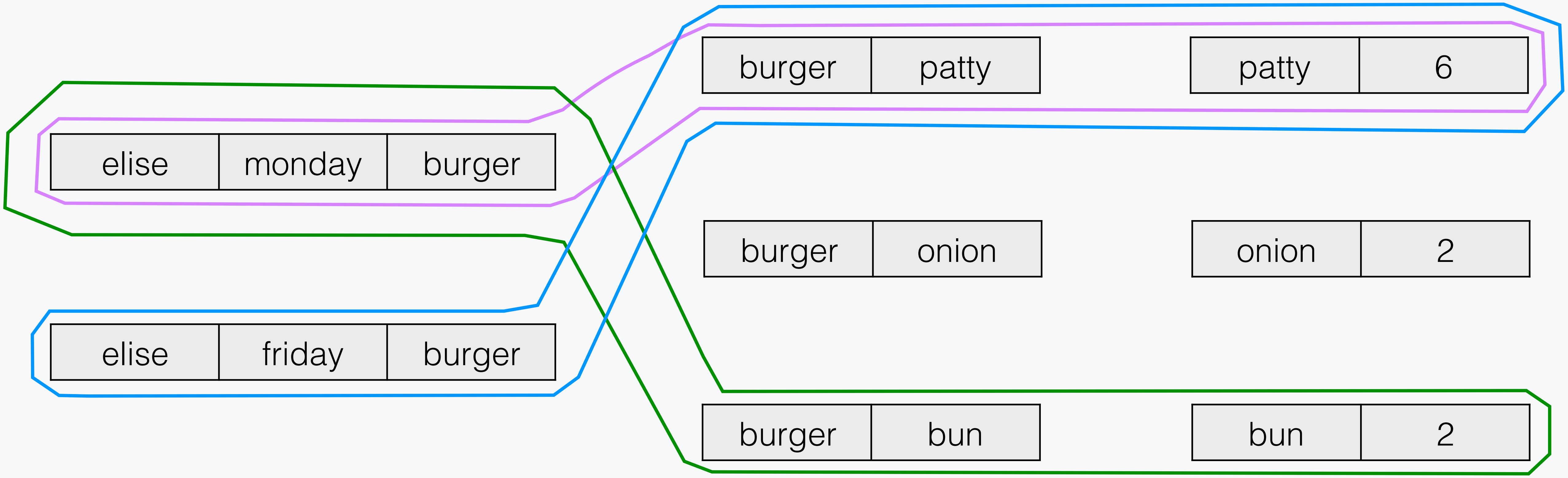
onion | 2

bun | 2

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

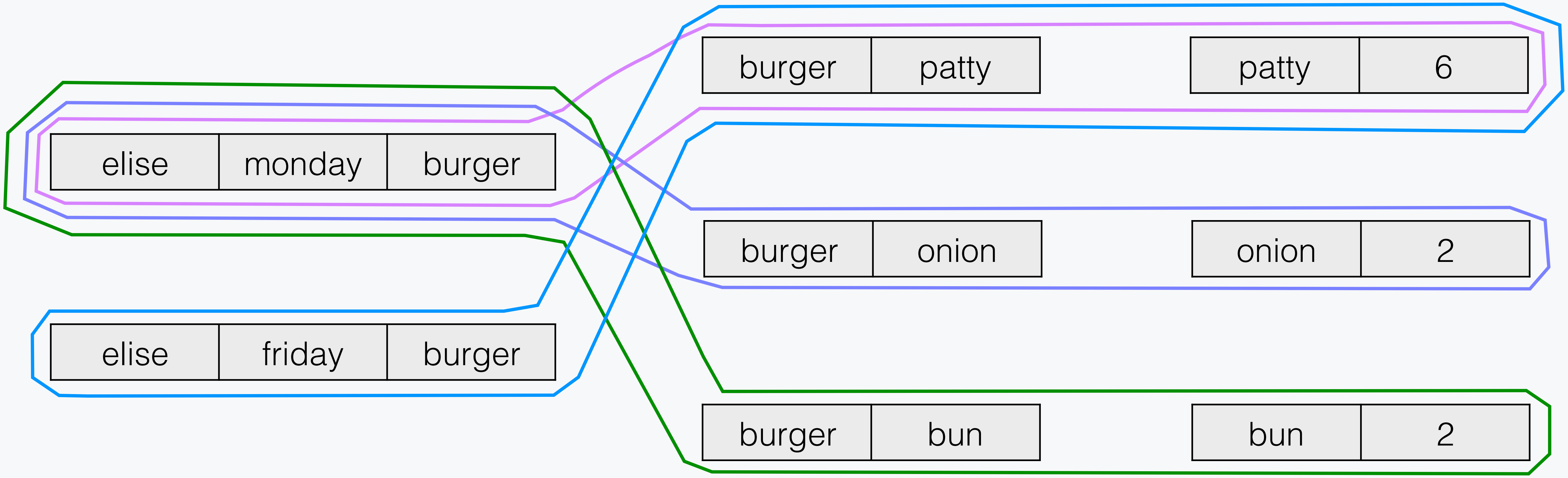
The Hypergraph of (a part of) the Query Result



$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

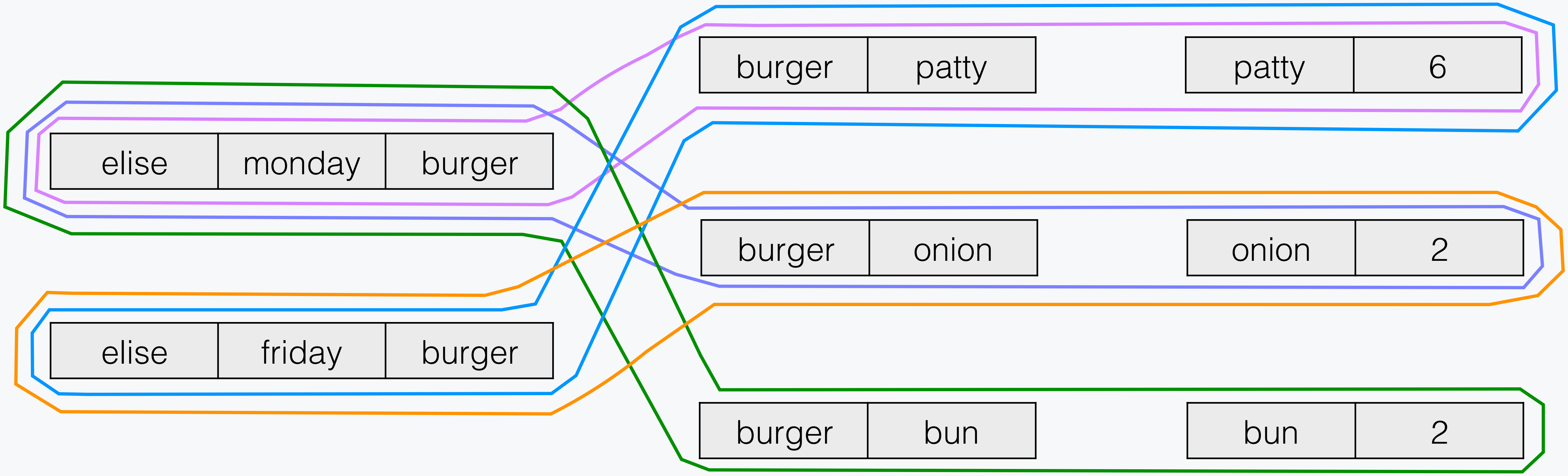
The Hypergraph of (a part of) the Query Result



$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

The Hypergraph of (a part of) the Query Result



elise | monday | burger

elise | friday | burger

burger | patty

burger | onion

burger | bun

patty | 6

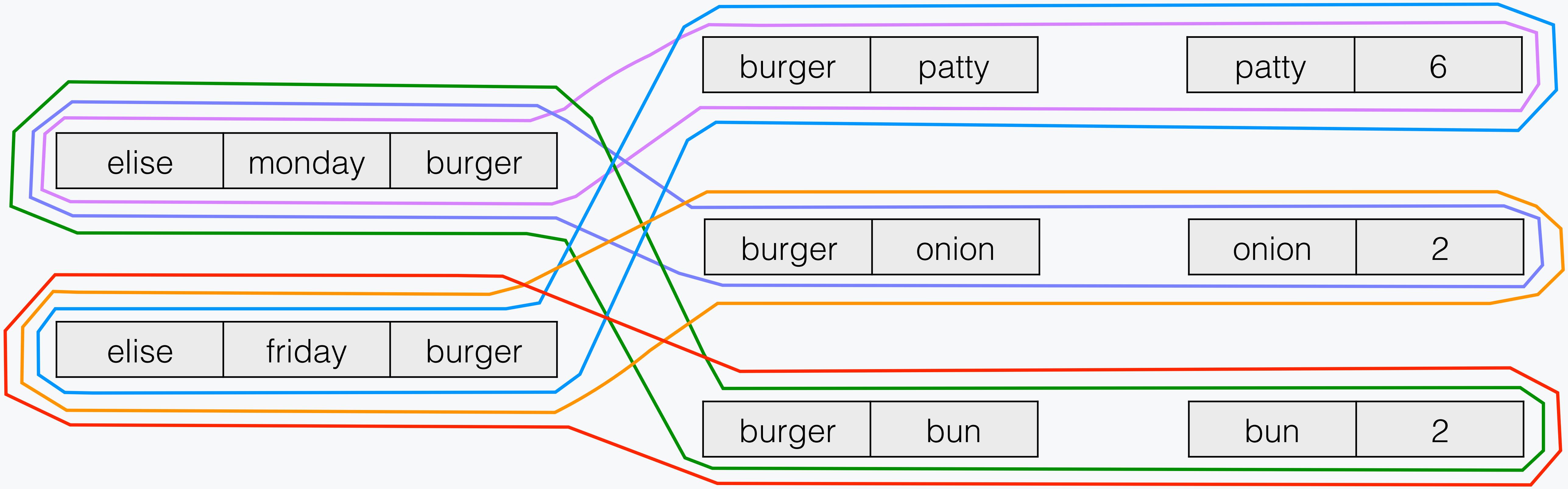
onion | 2

bun | 2

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

The Hypergraph of (a part of) the Query Result



elise | monday | burger

elise | friday | burger

burger | patty

burger | onion

burger | bun

patty | 6

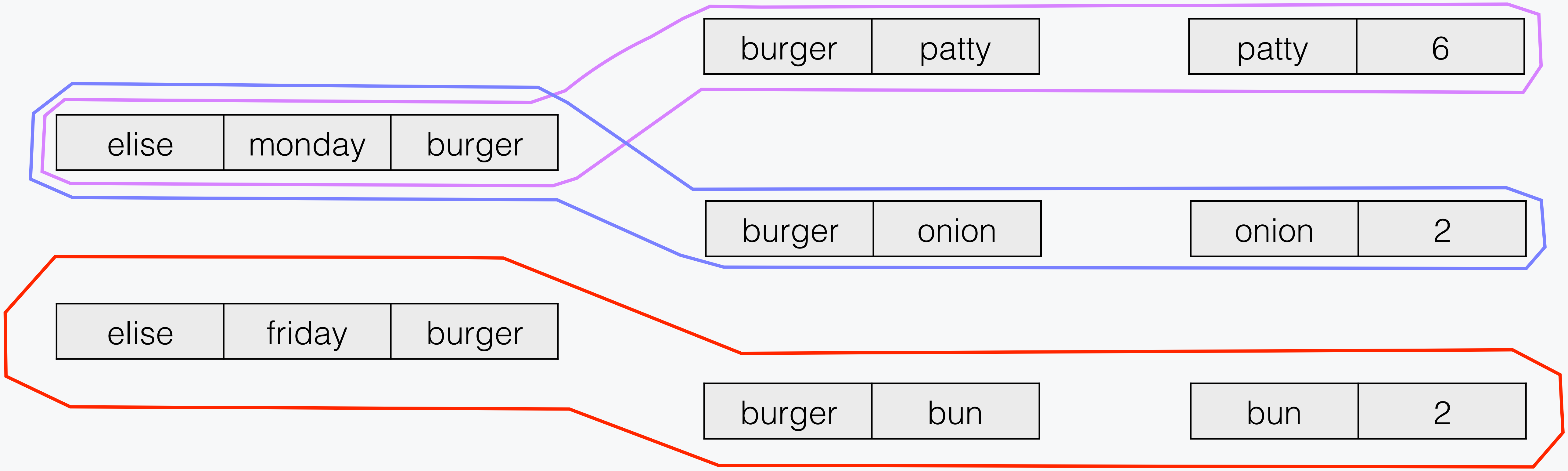
onion | 2

bun | 2

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

A Minimal Edge Cover of the Hypergraph



elise | monday | burger

elise | friday | burger

burger | patty

burger | onion

burger | bun

patty | 6

onion | 2

bun | 2

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

A Cover of (a part of) the Query Result

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

$$Q(\text{Customer}, \text{Day}, \text{Dish}, \text{Item}, \text{Price}) = \text{Orders} \bowtie \text{Dishes} \bowtie \text{Items}$$

Customer	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	monday	burger	onion	2
elise	monday	burger	bun	2
elise	friday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
⋮	⋮	⋮	⋮	⋮

Why Covers?

Observation on Query Results

- Query results can contain redundancies in computation and representation

Benefits of Covers

- A succinct relational representation system for query results
- The full query result can be enumerated from a cover with linearithmic time pre-computation and constant delay
- Allow for the succinct representation of the results of a wide range of computational problems
- Support subsequent operations:
  - aggregates
  - learning regression models
  - parallel computation

Previous Work

Foundations of Factorised Databases

- *Size Bounds for Factorised Representations of Query Results*
  [Olteanu and Závodný, TODS 2015]
- *FDB: A Query Engine for Factorised Relational Databases*
  [Bakibayev, Olteanu and Závodný, PVLDB, 2012]
- *Factorised Representations of Query Results: Size Bounds and Readability*
  [Olteanu and Závodný, ICDT 2012]

Aggregation in Factorised Databases

- *Aggregation and Ordering in Factorised Databases*
  [Bakibayev, Kočiský, Olteanu, and Závodný, PVLDB 2013]

Regression Models in Factorised Databases

- *Learning Linear regression Models over Factorised Joins*
  [Schleich, Olteanu and Ciucanu, SIGMOD 2016]
- *F: Regression Models over Factorised Views*
  [Olteanu and Schleich, PVLDB 2016]

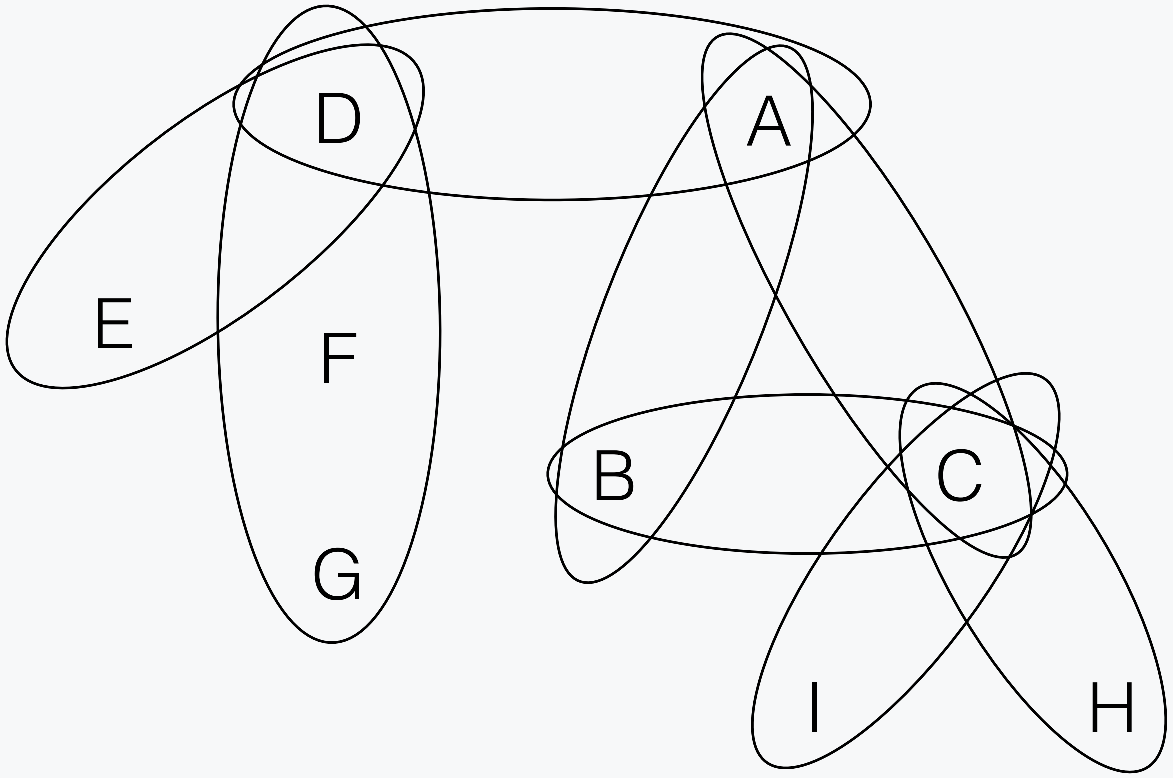
Outline of the Talk

- Covers of Query Results
- Computation of Covers
- Covers of more general Computational Problems

Hypertree Decompositions

$$Q(A,B,C,D,E,F,G,H,I) = R(A,B) \bowtie S(A,C) \bowtie T(B,C) \bowtie U(D,A) \bowtie V(E,D) \bowtie W(D,F,G) \bowtie X(C,I) \bowtie Y(C,H)$$

Hypergraph of Query Q

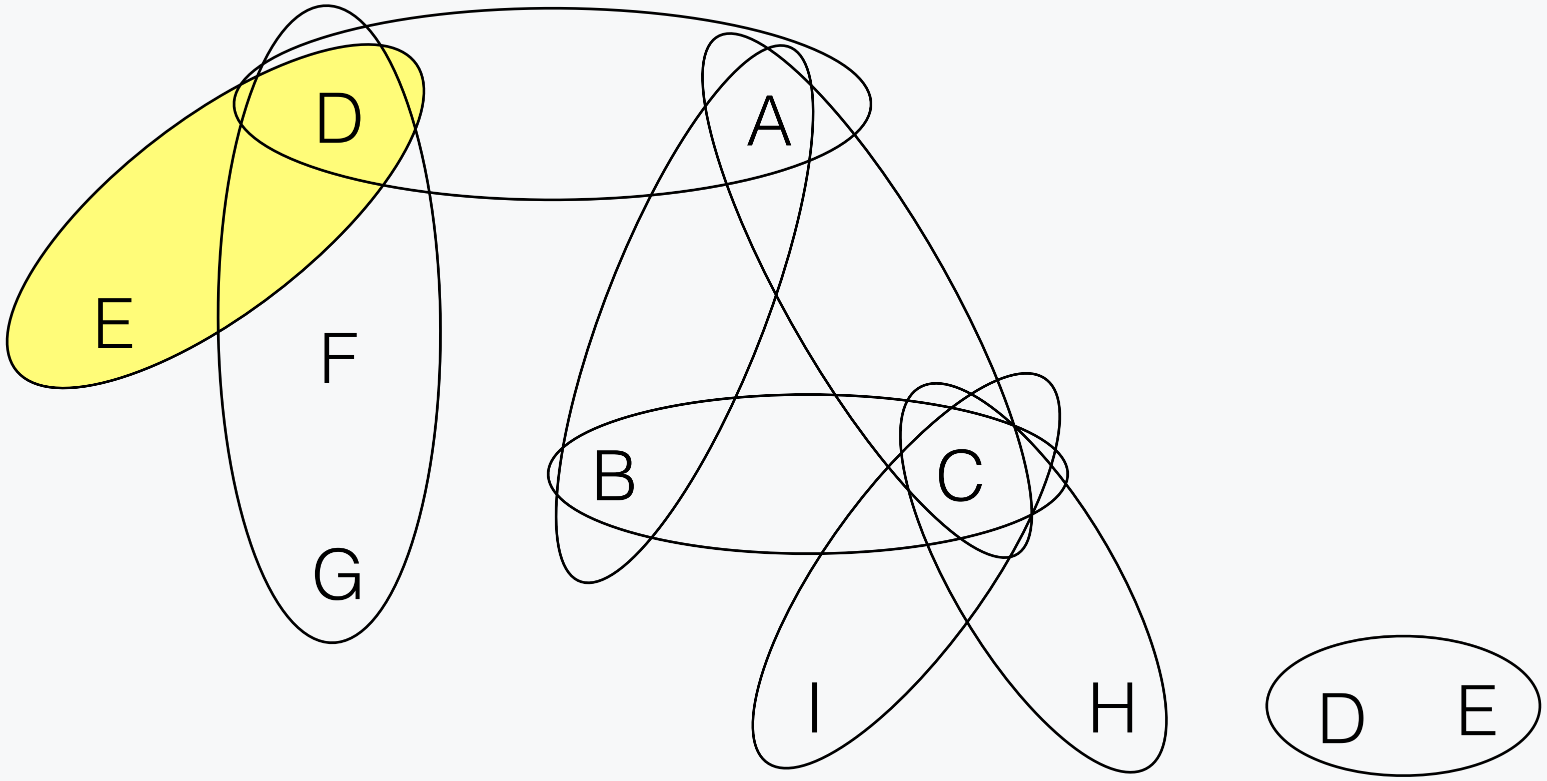


Hypertree Decompositions

$$Q(A,B,C,D,E,F,G,H,I) = R(A,B) \bowtie S(A,C) \bowtie T(B,C) \bowtie U(D,A) \bowtie$$
$$V(E,D) \bowtie W(D,F,G) \bowtie X(C,I) \bowtie Y(C,H)$$

Hypergraph of Query Q

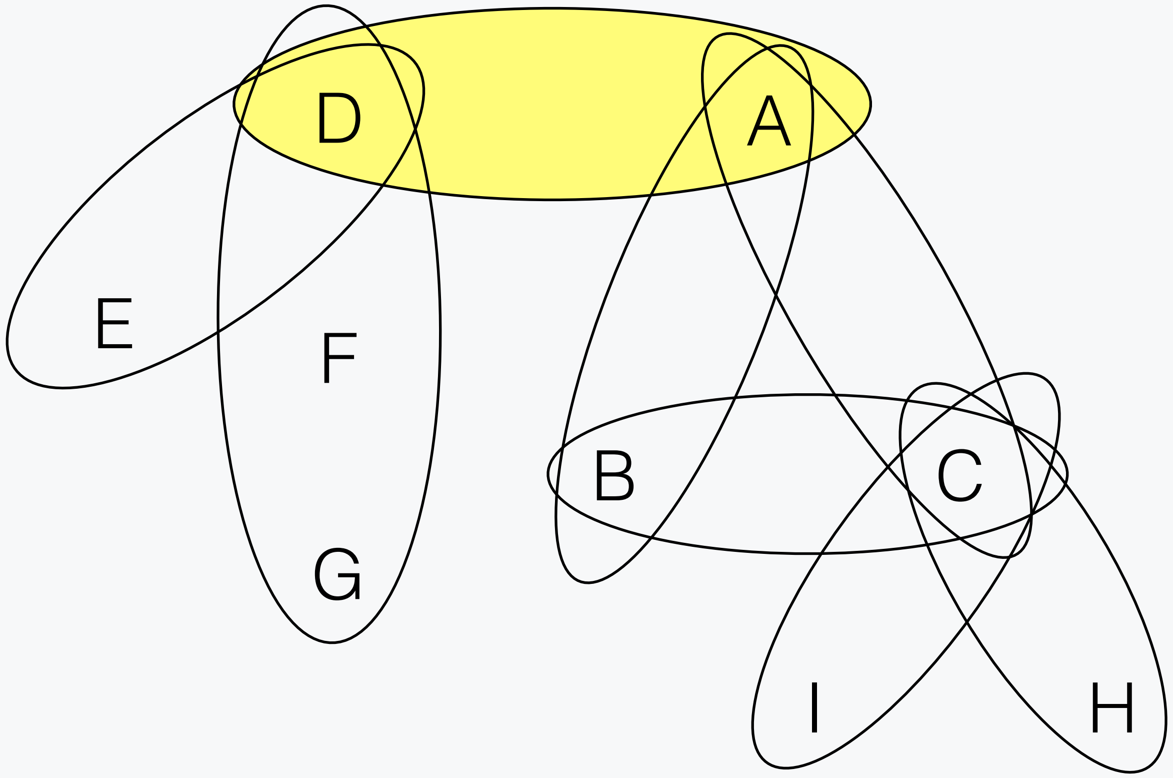
A Hypertree Decomposition of Q



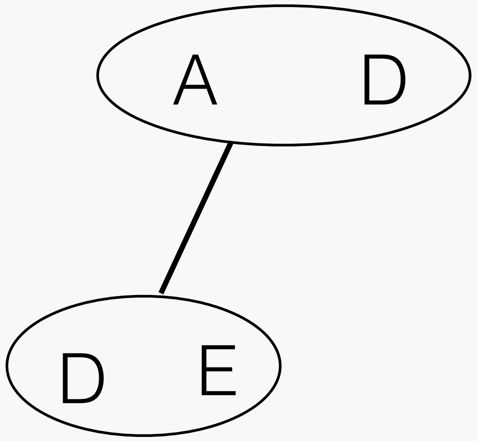
Hypertree Decompositions

$$Q(A,B,C,D,E,F,G,H,I) = R(A,B) \bowtie S(A,C) \bowtie T(B,C) \bowtie U(D,A) \bowtie V(E,D) \bowtie W(D,F,G) \bowtie X(C,I) \bowtie Y(C,H)$$

Hypergraph of Query Q



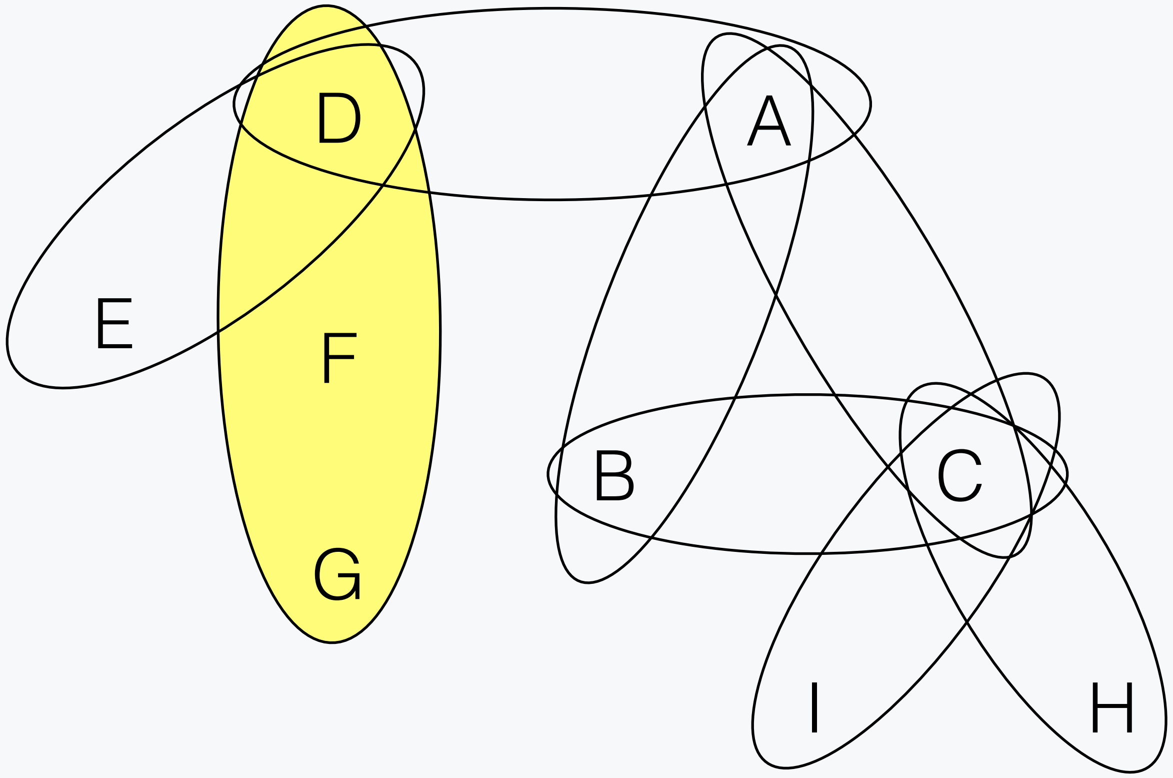
A Hypertree Decomposition of Q



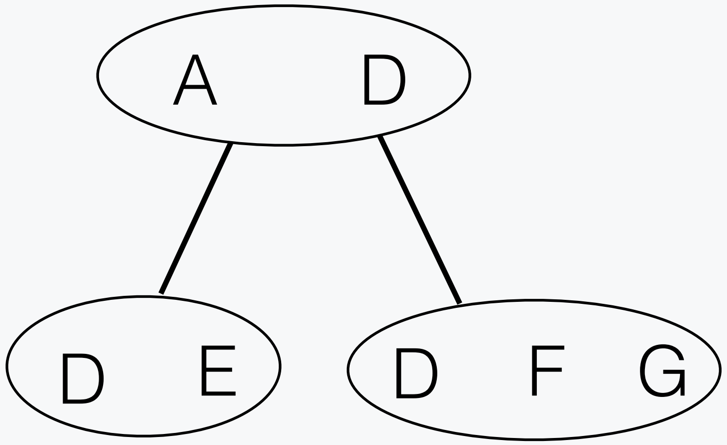
Hypertree Decompositions

$$Q(A,B,C,D,E,F,G,H,I) = R(A,B) \bowtie S(A,C) \bowtie T(B,C) \bowtie U(D,A) \bowtie$$
$$V(E,D) \bowtie W(D,F,G) \bowtie X(C,I) \bowtie Y(C,H)$$

Hypergraph of Query Q



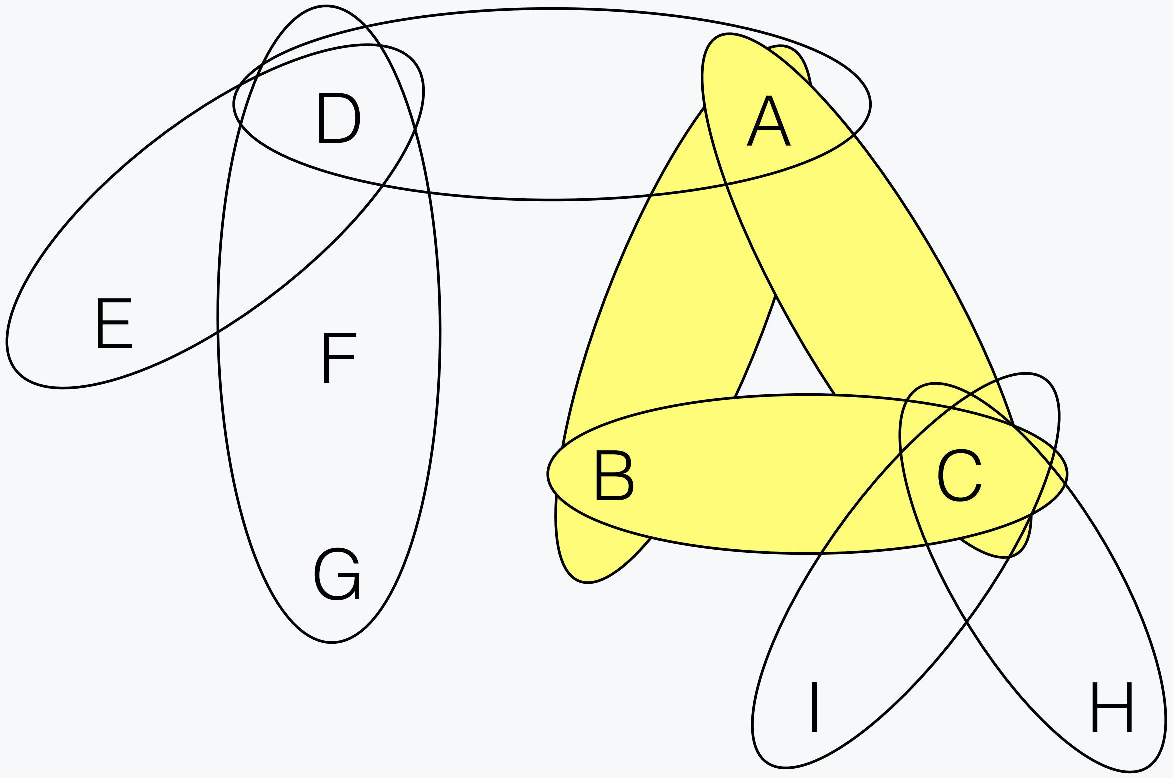
A Hypertree Decomposition of Q



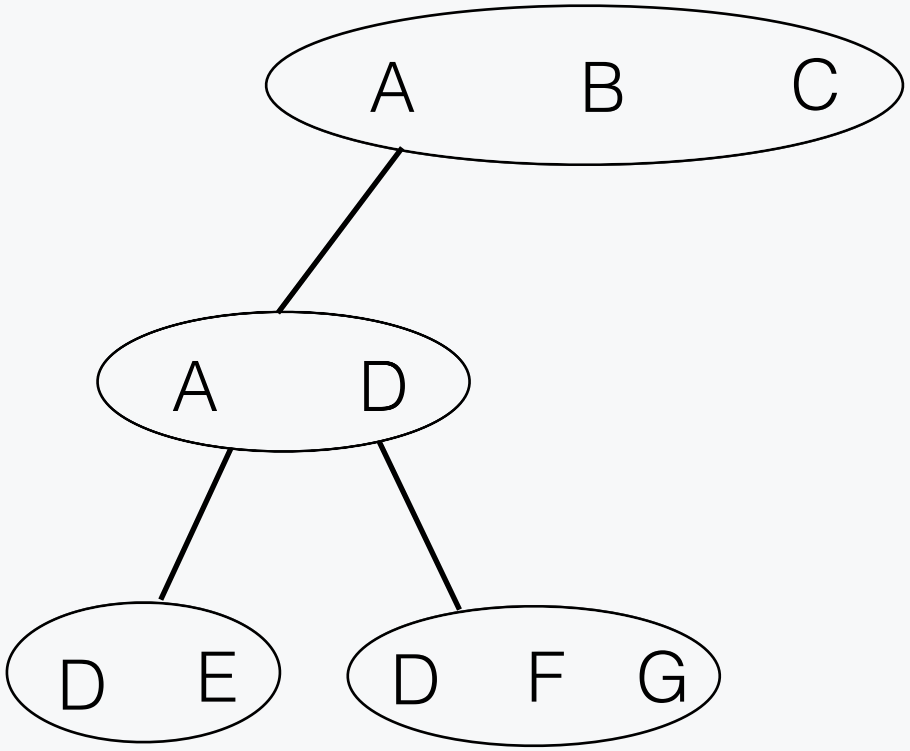
Hypertree Decompositions

$$Q(A,B,C,D,E,F,G,H,I) = R(A,B) \bowtie S(A,C) \bowtie T(B,C) \bowtie U(D,A) \bowtie$$
$$V(E,D) \bowtie W(D,F,G) \bowtie X(C,I) \bowtie Y(C,H)$$

Hypergraph of Query Q



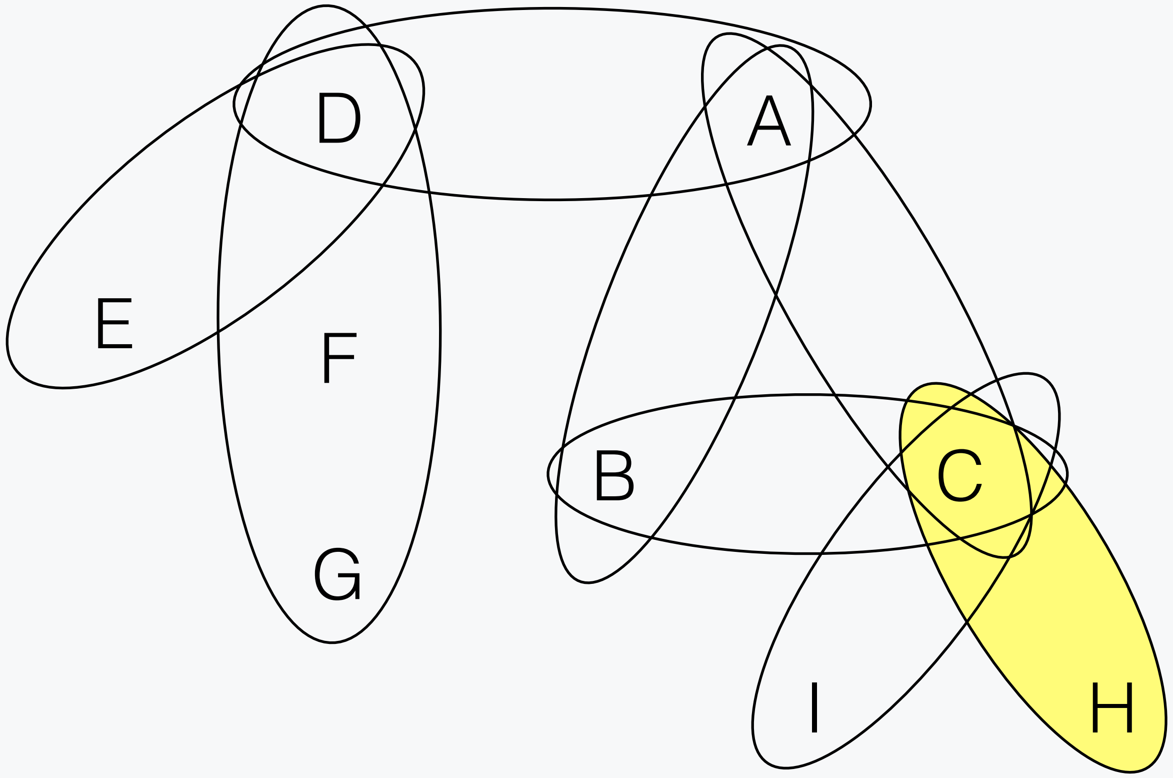
A Hypertree Decomposition of Q



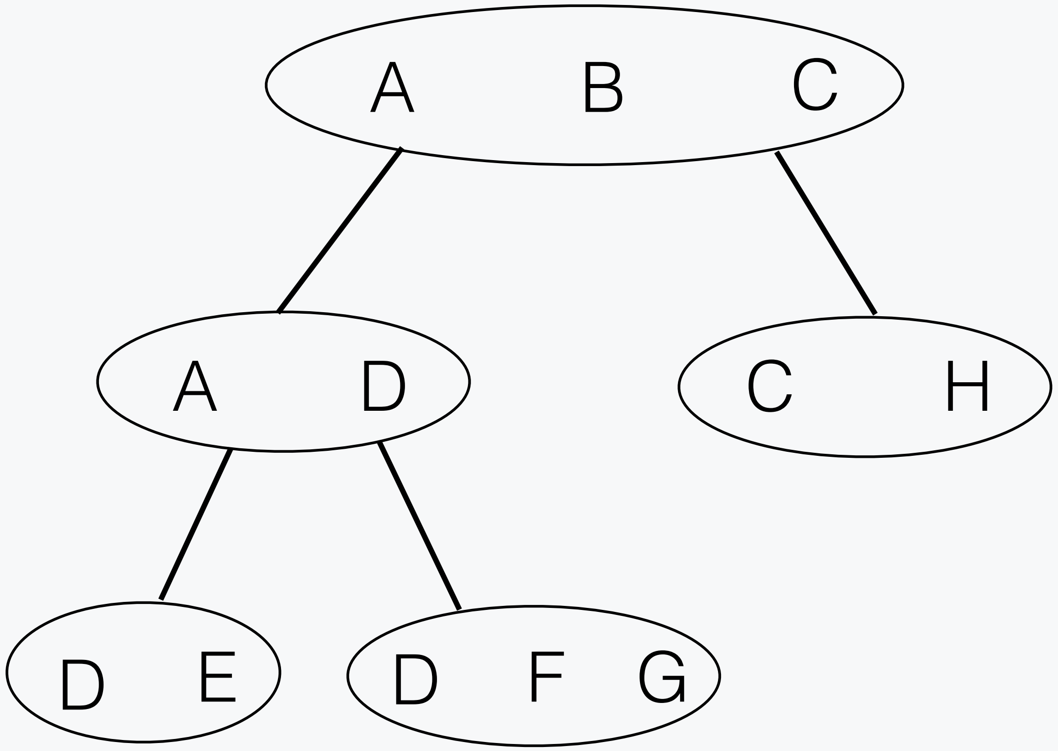
Hypertree Decompositions

$$Q(A,B,C,D,E,F,G,H,I) = R(A,B) \bowtie S(A,C) \bowtie T(B,C) \bowtie U(D,A) \bowtie V(E,D) \bowtie W(D,F,G) \bowtie X(C,I) \bowtie Y(C,H)$$

Hypergraph of Query Q



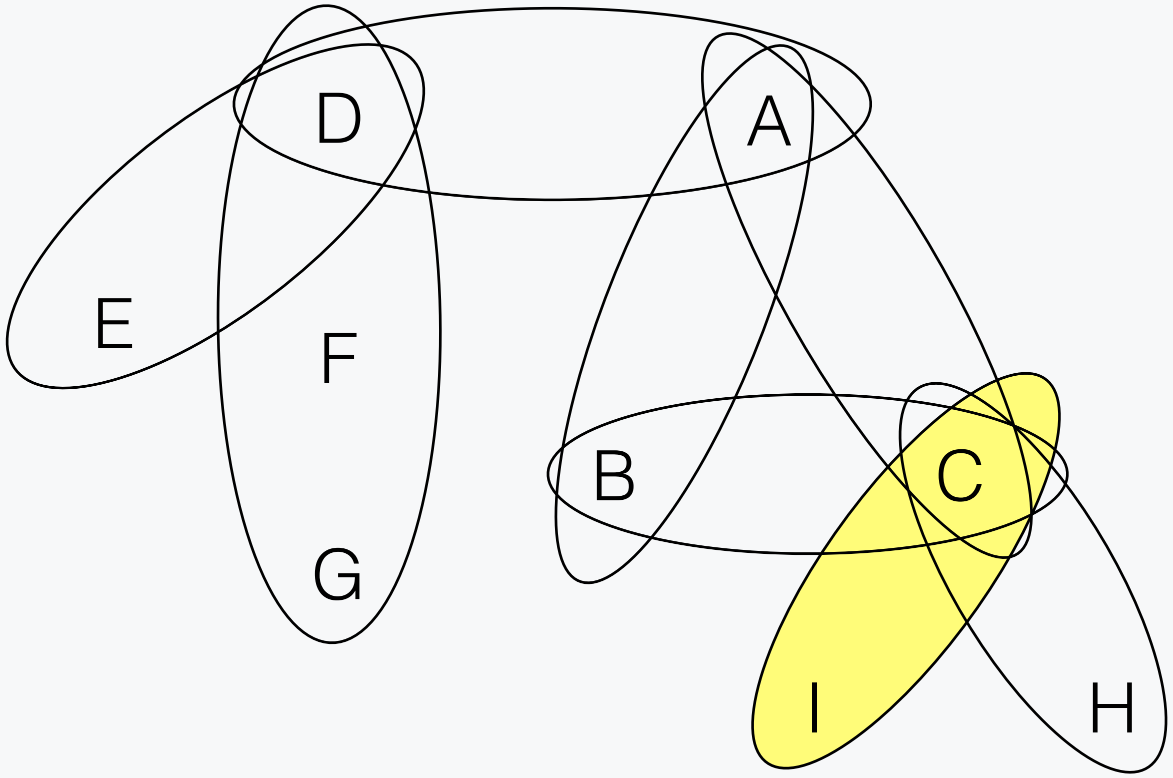
A Hypertree Decomposition of Q



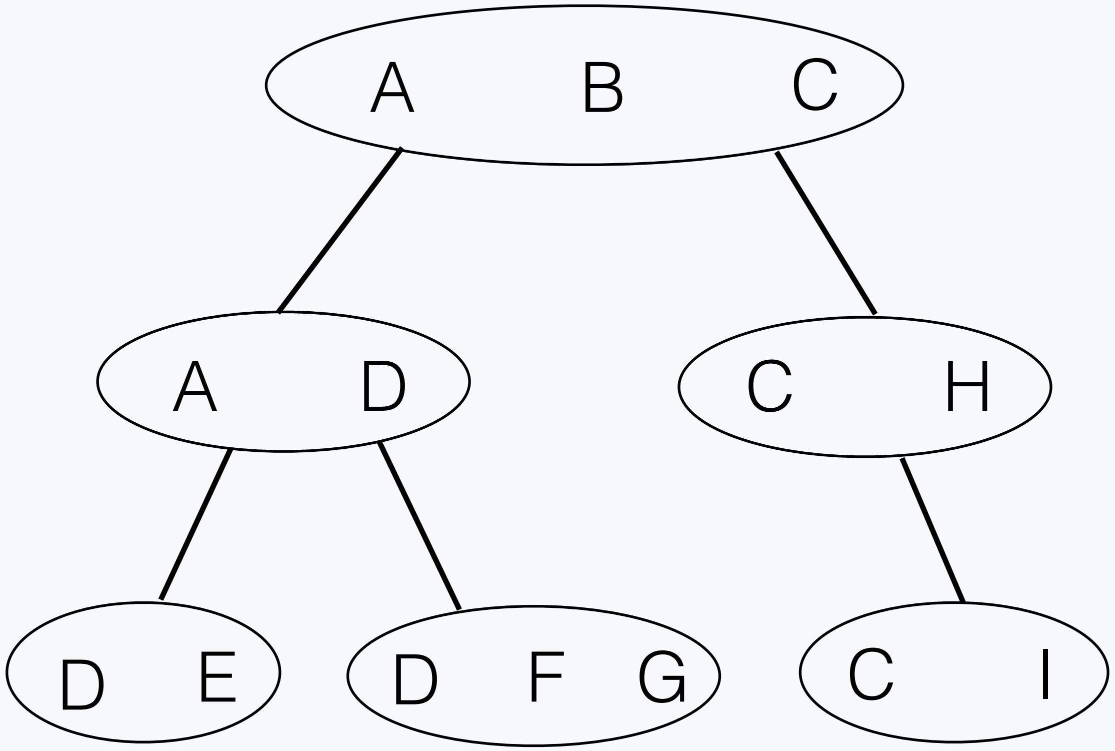
Hypertree Decompositions

$$Q(A, B, C, D, E, F, G, H, I) = R(A, B) \bowtie S(A, C) \bowtie T(B, C) \bowtie U(D, A) \bowtie$$
$$V(E, D) \bowtie W(D, F, G) \bowtie X(C, I) \bowtie Y(C, H)$$

Hypergraph of Query Q



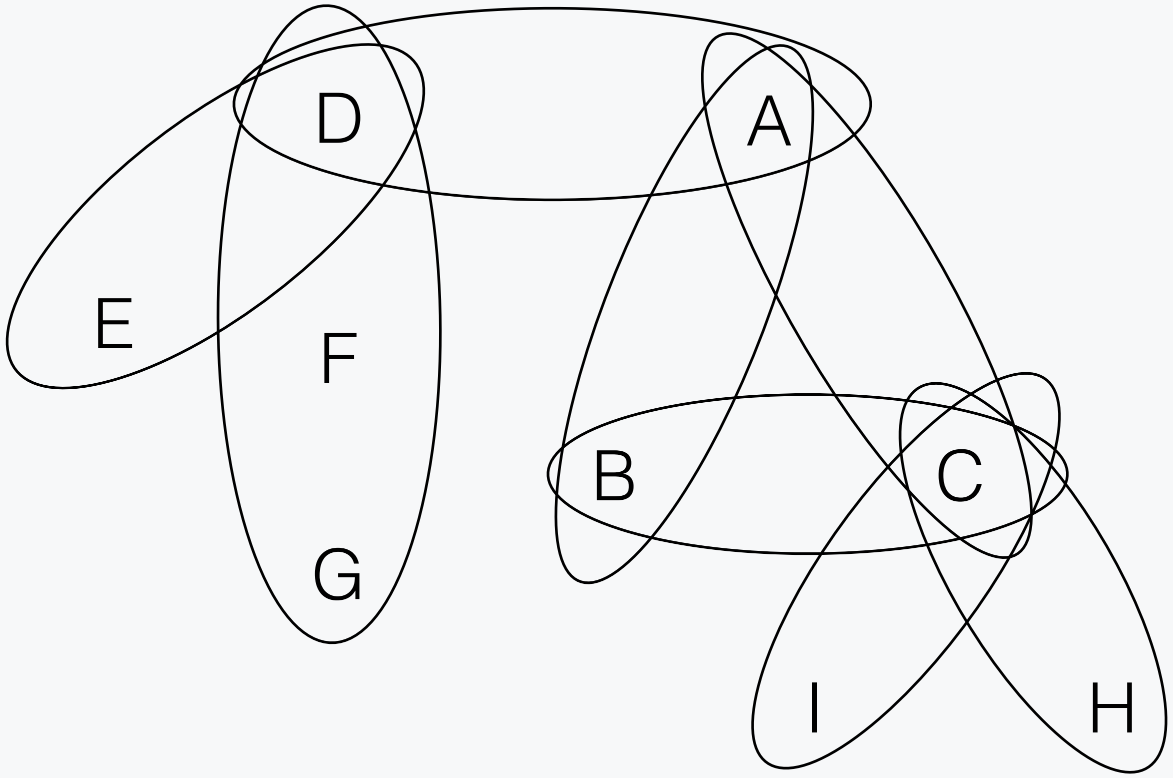
A Hypertree Decomposition of Q



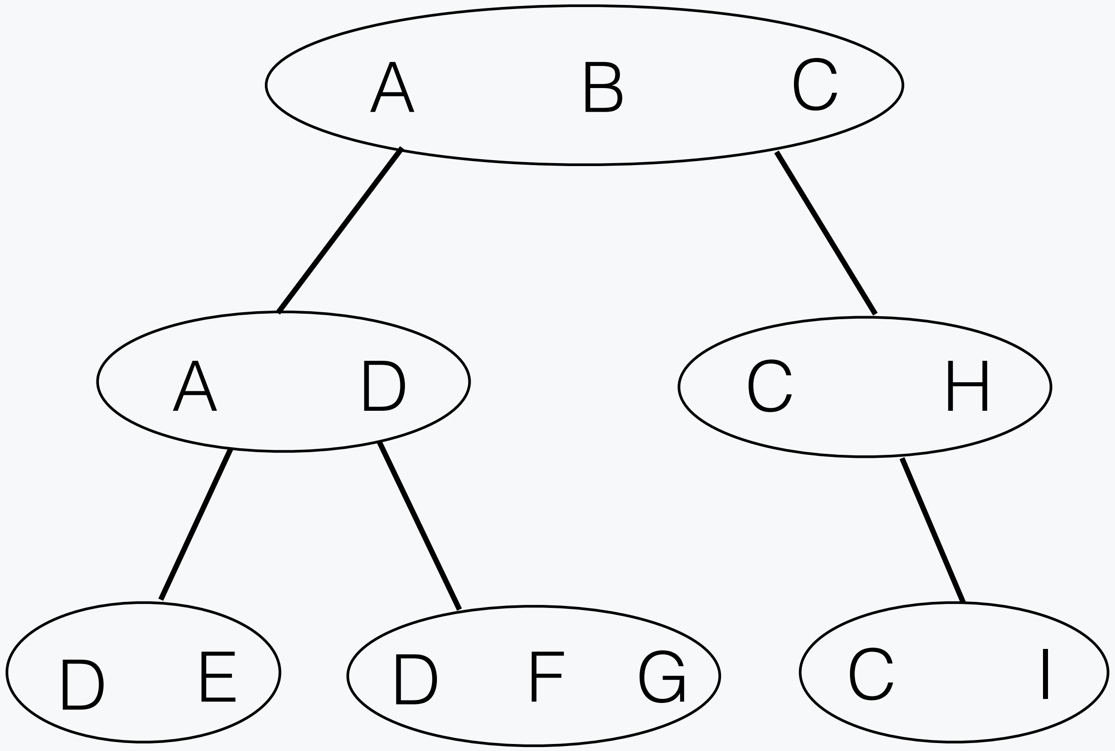
Hypertree Decompositions

$$Q(A, B, C, D, E, F, G, H, I) = R(A, B) \bowtie S(A, C) \bowtie T(B, C) \bowtie U(D, A) \bowtie$$
$$V(E, D) \bowtie W(D, F, G) \bowtie X(C, I) \bowtie Y(C, H)$$

Hypergraph of Query Q



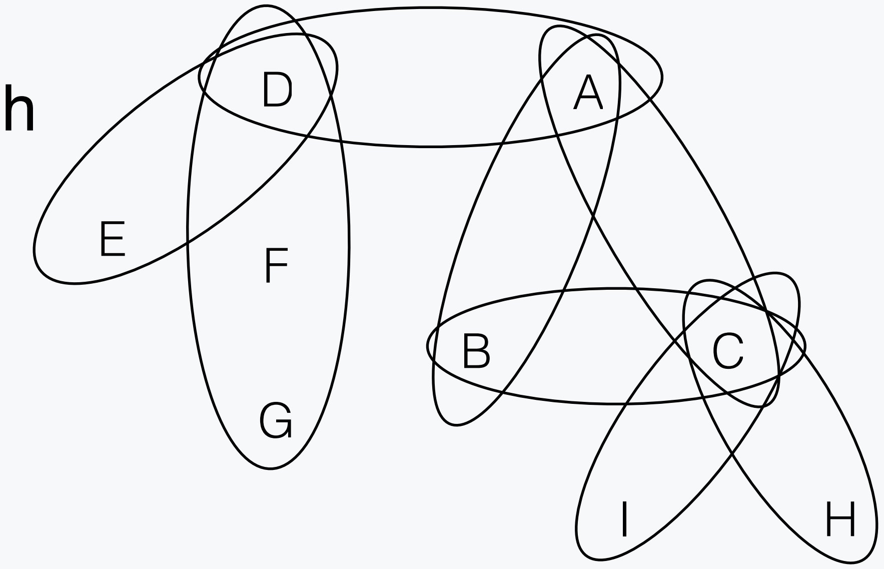
A Hypertree Decomposition of Q

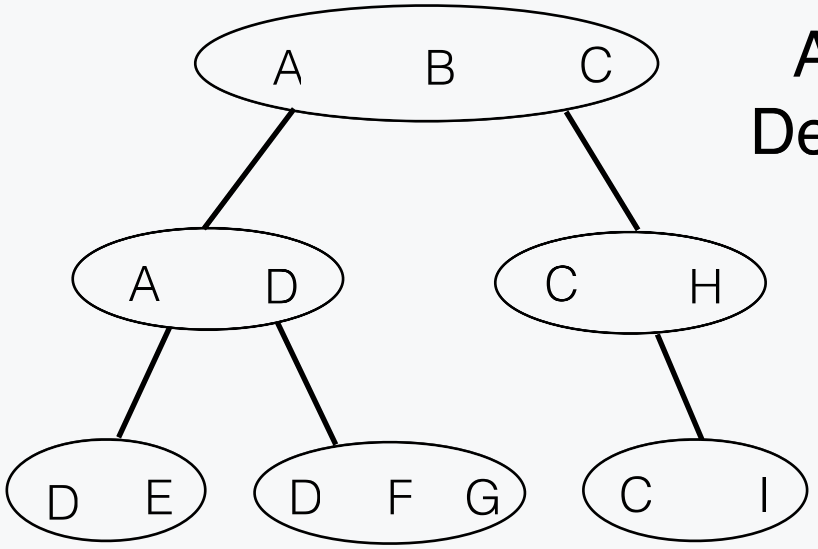


Coverage, Connectivity

Result Preservation

Hypergraph of Q:





A Hypertree Decomposition of Q

$\mathbf{D}$

A	B		A	C		B	C		A	D		E	D		D	F	G		C	I		C	H

$Q(\mathbf{D})$

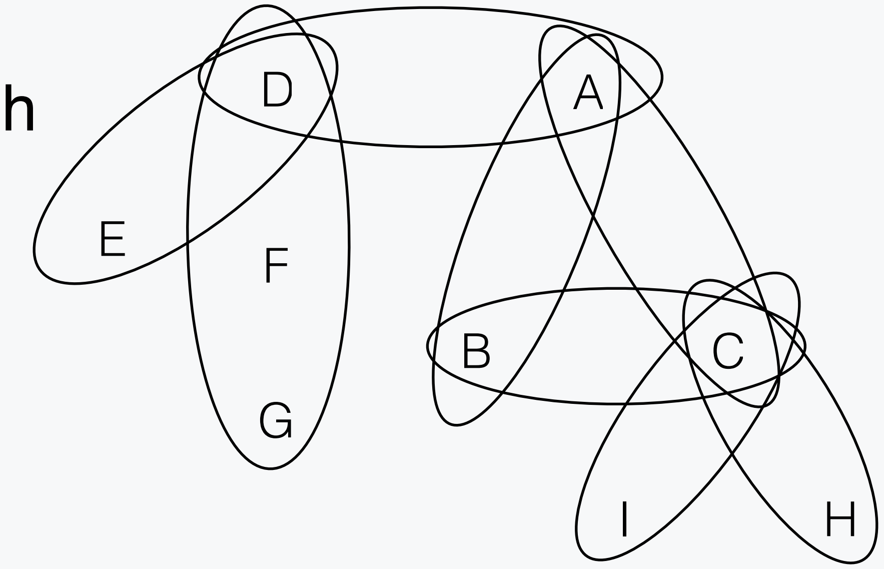
A	B	C	D	E	F	G	H	I

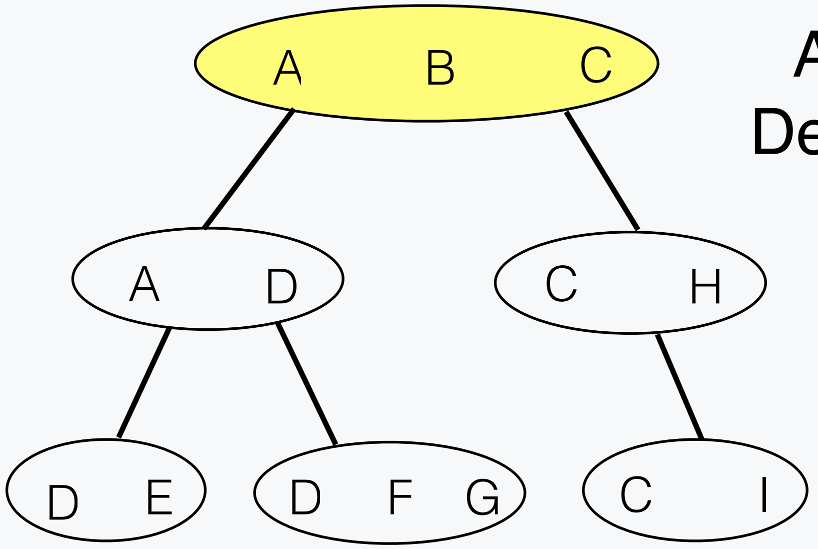
$\mathbf{C}$

A	B	C	D	E	F	G	H	I

Result Preservation

Hypergraph of Q:





A Hypertree Decomposition of Q

$\mathbf{D}$

A B | A C | B C | A D | E D | D F G | C I | C H

$Q(\mathbf{D})$

A B C D E F G H I

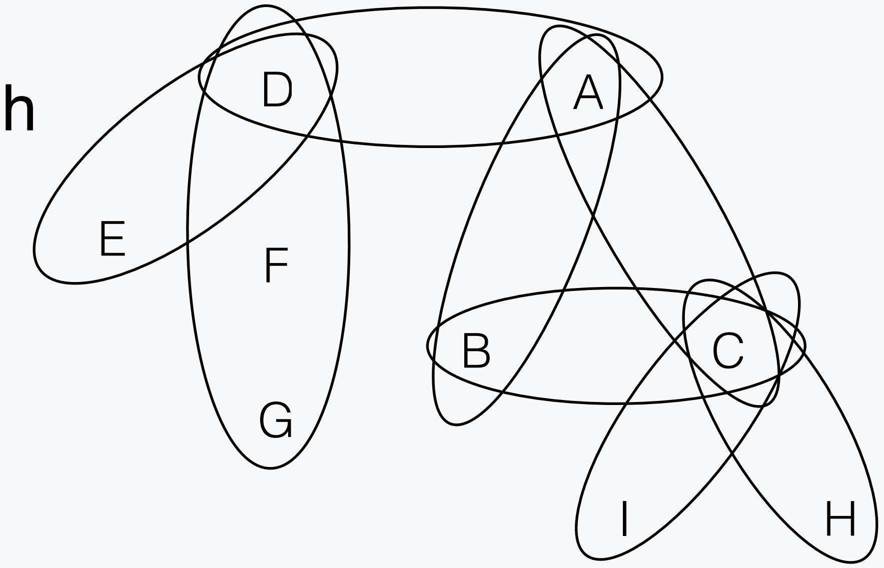
$\mathbf{C}$

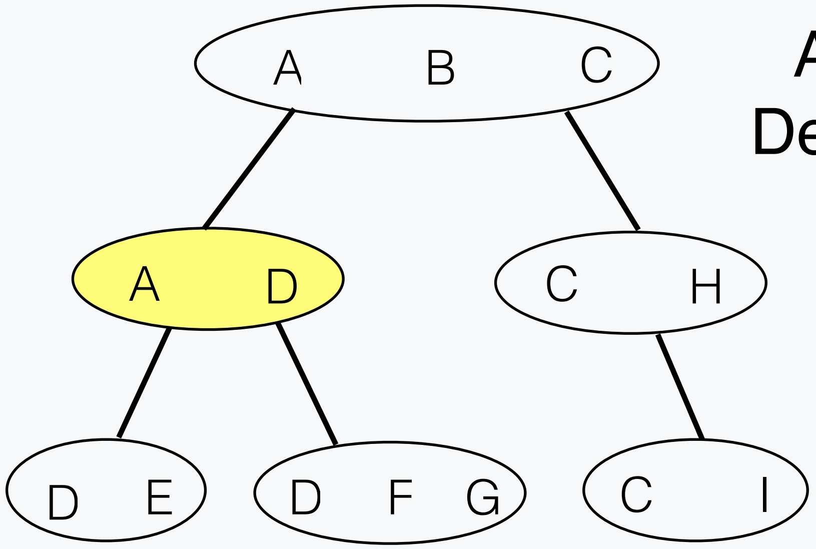
A B C D E F G H I

$$\pi_{\{A,B,C\}} Q(\mathbf{D}) = \pi_{\{A,B,C\}} \mathbf{C}$$

Result Preservation

Hypergraph of Q:





A Hypertree Decomposition of Q

$\mathbf{D}$

A	B		A	C		B	C		A	D		E	D		D	F	G		C	I		C	H

$Q(\mathbf{D})$

A	B	C	D	E	F	G	H	I

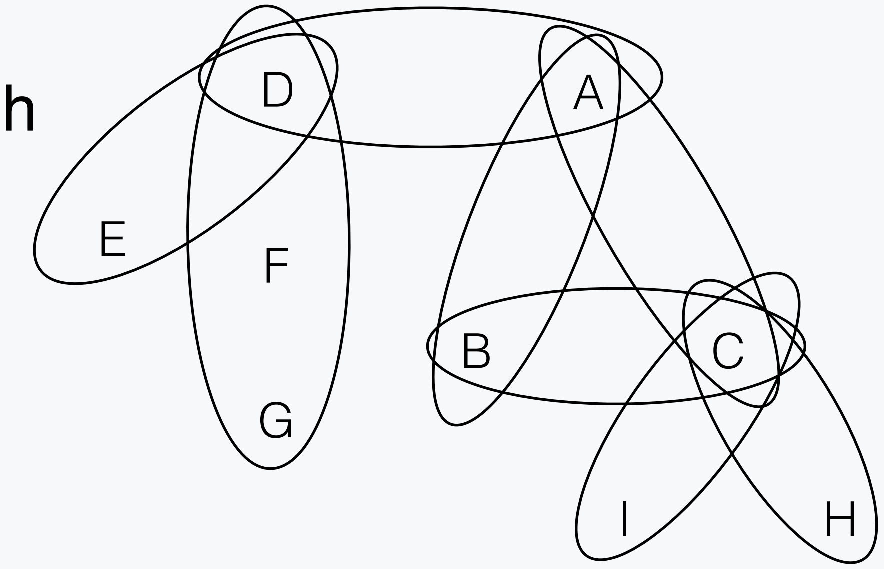
$\mathbf{C}$

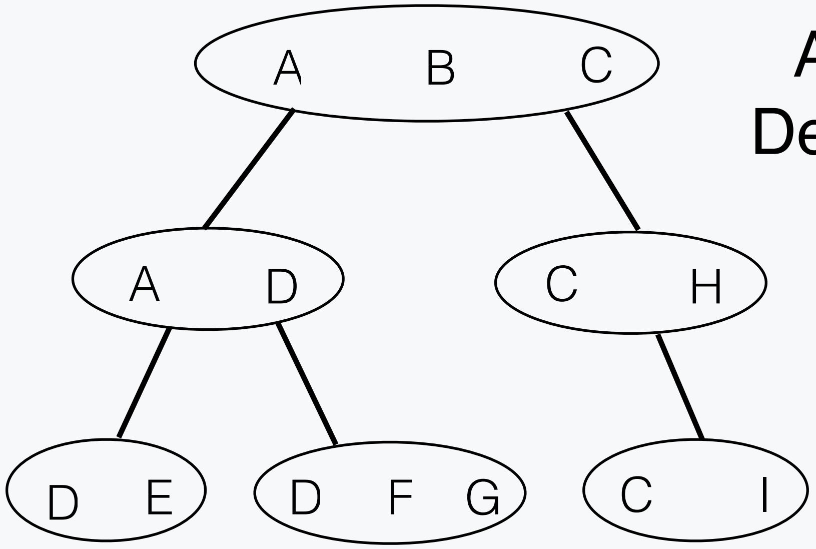
A	B	C	D	E	F	G	H	I

$$\pi_{\{A,D\}} Q(\mathbf{D}) = \pi_{\{A,D\}} \mathbf{C}$$

Result Preservation

Hypergraph of Q:





A Hypertree Decomposition of Q

$\mathbf{D}$

A	B	A	C	B	C	A	D	E	D	D	F	G	C	I	C	H

$Q(\mathbf{D})$

A	B	C	D	E	F	G	H	I

$\mathbf{C}$

A	B	C	D	E	F	G	H	I

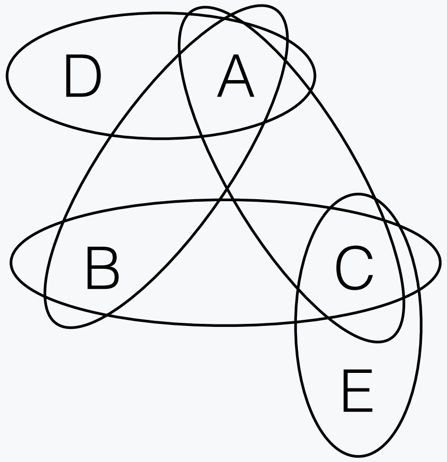
Given a query Q, a hypertree decomposition $\mathcal{T}$ of Q and a database $\mathbf{D}$, a relation $\mathbf{C}$ is result-preserving with respect to $(Q, \mathcal{T}, \mathbf{D})$ if $\pi_B \mathbf{C} = \pi_B Q(\mathbf{C})$ for each bag B of $\mathcal{T}$.

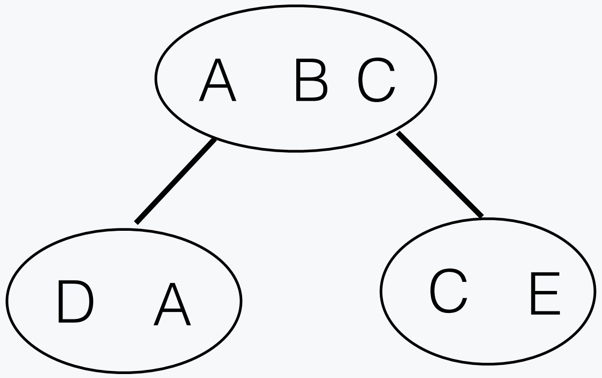
Covers of Query Results

Given a query Q, a hypertree decomposition $\mathcal{T}$ of Q and a database $\mathbf{D}$, a cover of the query result $Q(\mathbf{D})$ over $\mathcal{T}$ is a minimal result-preserving relation with respect to $(Q, \mathcal{T}, \mathbf{D})$.

Covers and Non-Covers

$$Q(A, B, C, D, E) = R \bowtie S \bowtie T \bowtie U \bowtie V$$





R

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

S

A	B
a_1	b_1
a_1	b_2
a_1	b_3
a_2	b_4
a_2	b_1

T

A	C
a_1	c_1
a_1	c_2
a_2	c_2
a_1	c_3

U

B	C
b_1	c_1
b_2	c_1
b_3	c_2
b_4	c_2

V

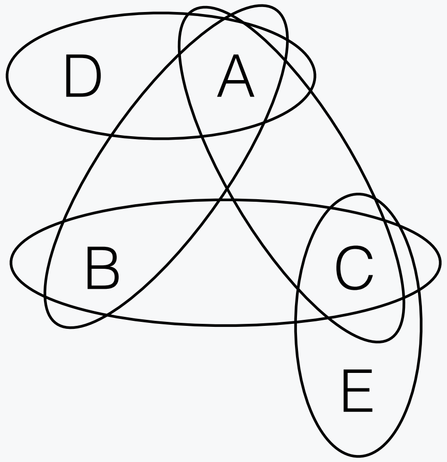
C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2
c_3	e_1

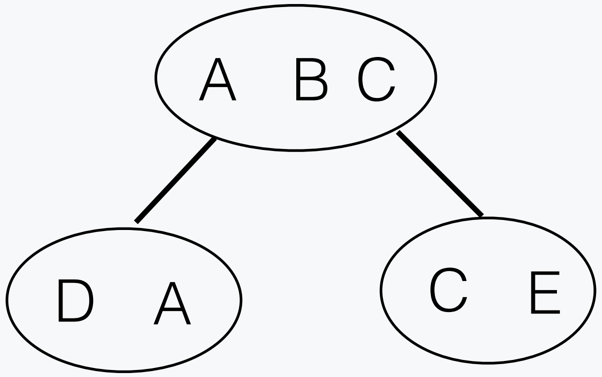
$Q(\mathbf{D})$

D	A	B	C	E
d_1	a_1	b_1	c_1	e_1
d_2	a_1	b_1	c_1	e_1
d_3	a_1	b_1	c_1	e_1
d_1	a_1	b_1	c_1	e_2
d_2	a_1	b_1	c_1	e_1
d_1	a_1	b_1	c_1	e_1
⋮	⋮	⋮	⋮	⋮

Covers and Non-Covers

$$Q(A, B, C, D, E) = R \bowtie S \bowtie T \bowtie U \bowtie V$$





R

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

S

A	B
a_1	b_1
a_1	b_2
a_1	b_3
a_2	b_4
a_2	b_1

T

A	C
a_1	c_1
a_1	c_2
a_2	c_2
a_1	c_3

U

B	C
b_1	c_1
b_2	c_1
b_3	c_2
b_4	c_2

V

C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2
c_3	e_1

$\pi_{\{D,A\}} Q(\mathbf{D})$

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

$\pi_{\{A,B,C\}} Q(\mathbf{D})$

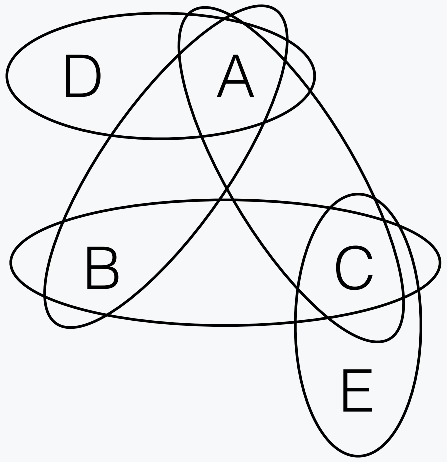
A	B	C
a_1	b_1	c_1
a_2	b_2	c_1
a_1	b_3	c_2
a_2	b_4	c_2

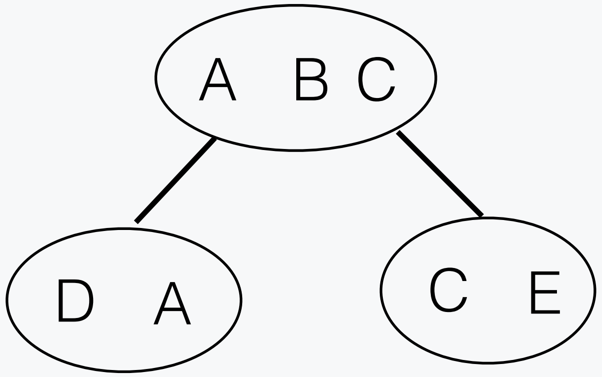
$\pi_{\{C,E\}} Q(\mathbf{D})$

C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2

Covers and Non-Covers

$$Q(A, B, C, D, E) = R \bowtie S \bowtie T \bowtie U \bowtie V$$





R

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

S

A	B
a_1	b_1
a_1	b_2
a_1	b_3
a_2	b_4
a_2	b_1

T

A	C
a_1	c_1
a_1	c_2
a_2	c_2
a_1	c_3

U

B	C
b_1	c_1
b_2	c_1
b_3	c_2
b_4	c_2

V

C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2
c_3	e_1

not result-preserving

D	A	B	C	E
d_1	a_1	b_1	c_1	e_1
d_2	a_1	b_1	c_1	e_1
d_3	a_1	b_3	c_2	e_1
d_1	a_2	b_4	c_2	e_1
d_2	a_2	b_4	c_2	e_1

$\pi_{\{D,A\}}Q(\mathbf{D})$

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

$\pi_{\{A,B,C\}}Q(\mathbf{D})$

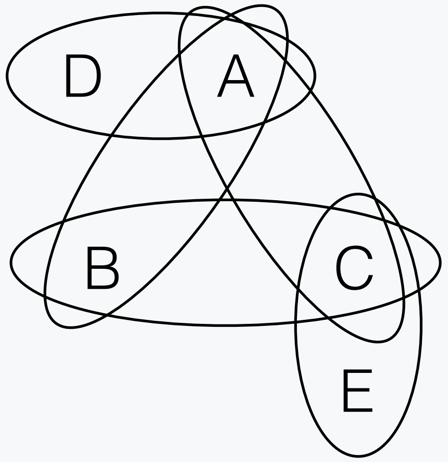
A	B	C
a_1	b_1	c_1
a_2	b_2	c_1
a_1	b_3	c_2
a_2	b_4	c_2

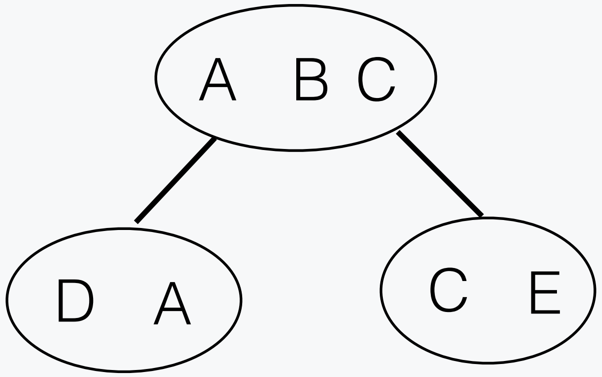
$\pi_{\{C,E\}}Q(\mathbf{D})$

C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2

Covers and Non-Covers

$$Q(A, B, C, D, E) = R \bowtie S \bowtie T \bowtie U \bowtie V$$





R

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

S

A	B
a_1	b_1
a_1	b_2
a_1	b_3
a_2	b_4
a_2	b_1

T

A	C
a_1	c_1
a_1	c_2
a_2	c_2
a_1	c_3

U

B	C
b_1	c_1
b_2	c_1
b_3	c_2
b_4	c_2

V

C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2
c_3	e_1

not minimal

D	A	B	C	E
d_1	a_1	b_1	c_1	e_1
d_2	a_1	b_2	c_1	e_2
d_3	a_1	b_3	c_2	e_1
d_1	a_2	b_4	c_2	e_2
d_2	a_2	b_4	c_2	e_2
d_2	a_2	b_4	c_2	e_1

$\pi_{\{D,A\}} Q(\mathbf{D})$

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

$\pi_{\{A,B,C\}} Q(\mathbf{D})$

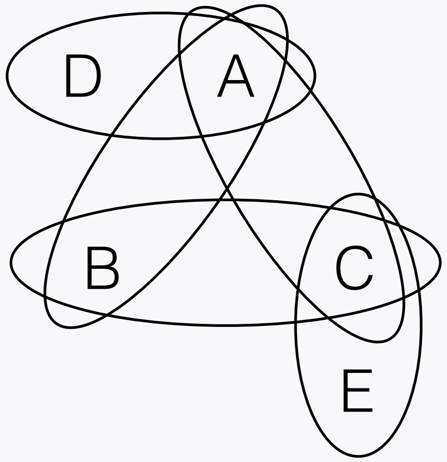
A	B	C
a_1	b_1	c_1
a_2	b_2	c_1
a_1	b_3	c_2
a_2	b_4	c_2

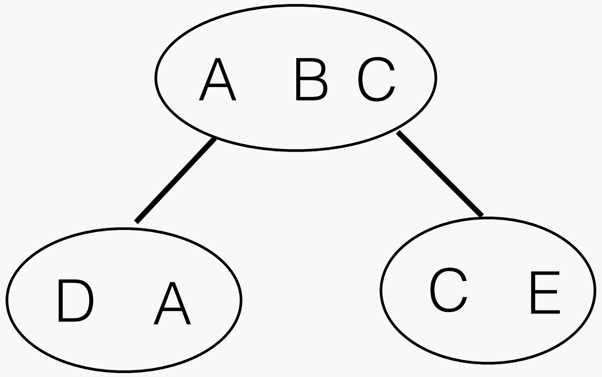
$\pi_{\{C,E\}} Q(\mathbf{D})$

C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2

Covers and Non-Covers

$$Q(A, B, C, D, E) = R \bowtie S \bowtie T \bowtie U \bowtie V$$





R

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

S

A	B
a_1	b_1
a_1	b_2
a_1	b_3
a_2	b_4
a_2	b_1

T

A	C
a_1	c_1
a_1	c_2
a_2	c_2
a_1	c_3

U

B	C
b_1	c_1
b_2	c_1
b_3	c_2
b_4	c_2

V

C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2
c_3	e_1

a cover

D	A	B	C	E
d_1	a_1	b_1	c_1	e_1
d_2	a_1	b_2	c_1	e_2
d_3	a_1	b_3	c_2	e_1
d_1	a_2	b_4	c_2	e_2
d_2	a_2	b_4	c_2	e_2

$\pi_{\{D,A\}}Q(\mathbf{D})$

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

$\pi_{\{A,B,C\}}Q(\mathbf{D})$

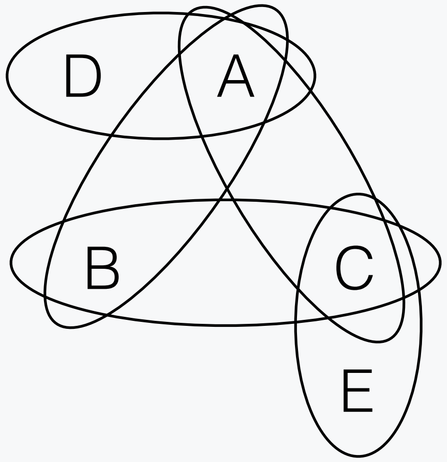
A	B	C
a_1	b_1	c_1
a_2	b_2	c_1
a_1	b_3	c_2
a_2	b_4	c_2

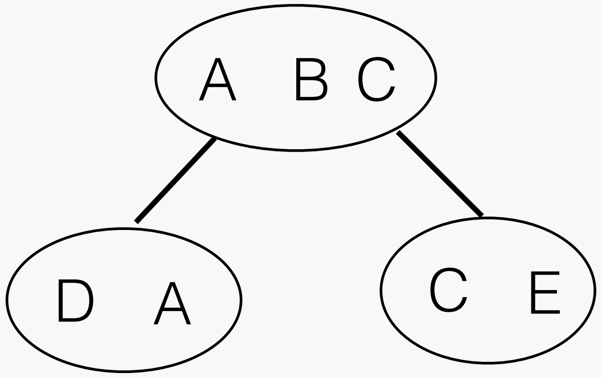
$\pi_{\{C,E\}}Q(\mathbf{D})$

C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2

Covers and Non-Covers

$$Q(A, B, C, D, E) = R \bowtie S \bowtie T \bowtie U \bowtie V$$





R

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

S

A	B
a_1	b_1
a_1	b_2
a_1	b_3
a_2	b_4
a_2	b_1

T

A	C
a_1	c_1
a_1	c_2
a_2	c_2
a_1	c_3

U

B	C
b_1	c_1
b_2	c_1
b_3	c_2
b_4	c_2

V

C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2
c_3	e_1

another cover

D	A	B	C	E
d_1	a_1	b_1	c_1	e_1
d_2	a_1	b_1	c_1	e_1
d_3	a_1	b_2	c_1	e_2
d_3	a_1	b_3	c_2	e_1
d_1	a_2	b_4	c_2	e_2
d_2	a_2	b_4	c_2	e_2

$\pi_{\{D,A\}}Q(\mathbf{D})$

D	A
d_1	a_1
d_2	a_1
d_3	a_1
d_1	a_2
d_2	a_2

$\pi_{\{A,B,C\}}Q(\mathbf{D})$

A	B	C
a_1	b_1	c_1
a_2	b_2	c_1
a_1	b_3	c_2
a_2	b_4	c_2

$\pi_{\{C,E\}}Q(\mathbf{D})$

C	E
c_1	e_1
c_1	e_2
c_2	e_1
c_2	e_2

Insights on Covers

Let Q be a query, $\mathcal{T}$ a hypertree decomposition of Q, $\mathbf{D}$ a database and
$\mathbf{C}$ a cover of $Q(\mathbf{D})$ over $\mathcal{T}$.

Covers are included in Query Result

$\mathbf{C}$ is a subset of $Q(\mathbf{D})$.

Covers allow Recovery of Query Result

$$\Join_{B \in \mathcal{S}(\mathcal{T})} \pi_B \mathbf{C} = Q(\mathbf{D}).$$

Characterization by Minimal Edge Covers in Hypergraph of Query Result

$\mathbf{C}$ corresponds to a minimal edge cover in the hypergraph of the join of the projections of $Q(\mathbf{D})$ onto the bags of $\mathcal{T}$.

Insights on Covers

Let Q be a query, $\mathcal{T}$ a hypertree decomposition of Q, $\mathbf{D}$ a database and
$\mathbf{C}$ a cover of $Q(\mathbf{D})$ over $\mathcal{T}$.

Size of Covers

The size of $\mathbf{C}$ is $\mathcal{O}(|\mathbf{D}|^{\mathrm{fhtw}(\mathcal{T})})$.

There are arbitrarily large databases $\mathbf{D}$ such that each cover of $Q(\mathbf{D})$ has size $\Omega(|\mathbf{D}|^{\mathrm{fhtw}(\mathcal{T})})$.

Constant-delay Enumeration from Covers

$Q(\mathbf{D})$ can be enumerated from $\mathbf{C}$ with $\widetilde{\mathcal{O}}(|\mathbf{C}|)$ pre-computation time and constant delay.

Covers of Results of Acyclic Queries

Representation System: Acyclic Query + Input Database

- Input database has asymptotically the same size as a minimum-sized cover of the query result.
- Enumeration of the query result from the input database has the same time complexity as covers [Bagan et al., 07].

So, why use a cover instead of the input database
in case of acyclic queries?

Covers of Results of Acyclic Queries

Representation System: Acyclic Query + Input Database

- Input database has asymptotically the same size as a minimum-sized cover of the query result.
- Enumeration of the query result from the input database has the same time complexity as covers [Bagan et al., 07].

So, why use a cover instead of the input database
in case of acyclic queries?

Practical Benefits of Covers

- Covers provide cache and access locality.
- A cover is a sample of the query result.
- Subsequent processing may require a single relation (e.g, machine learning over joins).

Computation of Covers

Two possible ways to compute covers in worst-case optimal time:

Monolithic Approach

- Given a query, a hypertree decomposition and a database, construct a cover by considering all joins at once.

Compositional Approach

- Given a query, a hypertree decomposition and a database, construct relations for the bags of the decomposition.
- Turn bag relations into a globally consistent database.
- Use a cover-join plan to compute the overall cover by computing partial covers.

Computing a Cover of our Fast Food Database

Orders

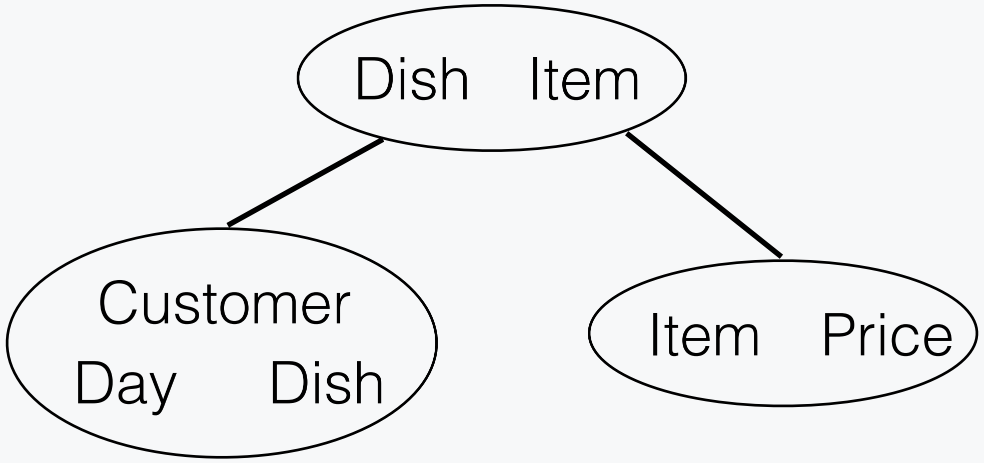
Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4



Computing a Cover of our Fast Food Database

Orders

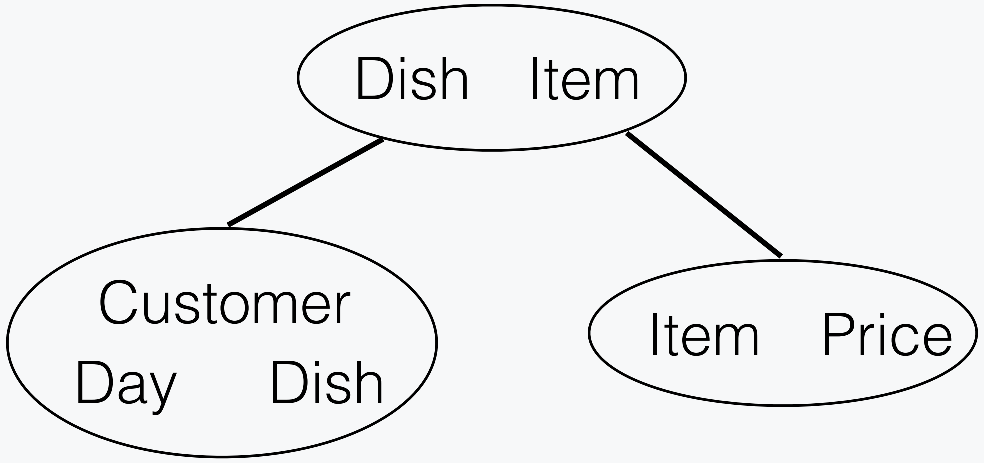
Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4



Valid cover-join plans

$$(\text{Orders} \overset{\circ}{\bowtie} \text{Dishes}) \overset{\circ}{\bowtie} \text{Items}$$

$$\text{Orders} \overset{\circ}{\bowtie} (\text{Dishes} \overset{\circ}{\bowtie} \text{Items})$$

A non-valid plan

$$(\text{Orders} \overset{\circ}{\bowtie} \text{Items}) \overset{\circ}{\bowtie} \text{Dishes}$$

Computing a Cover of our Fast Food Database

Orders

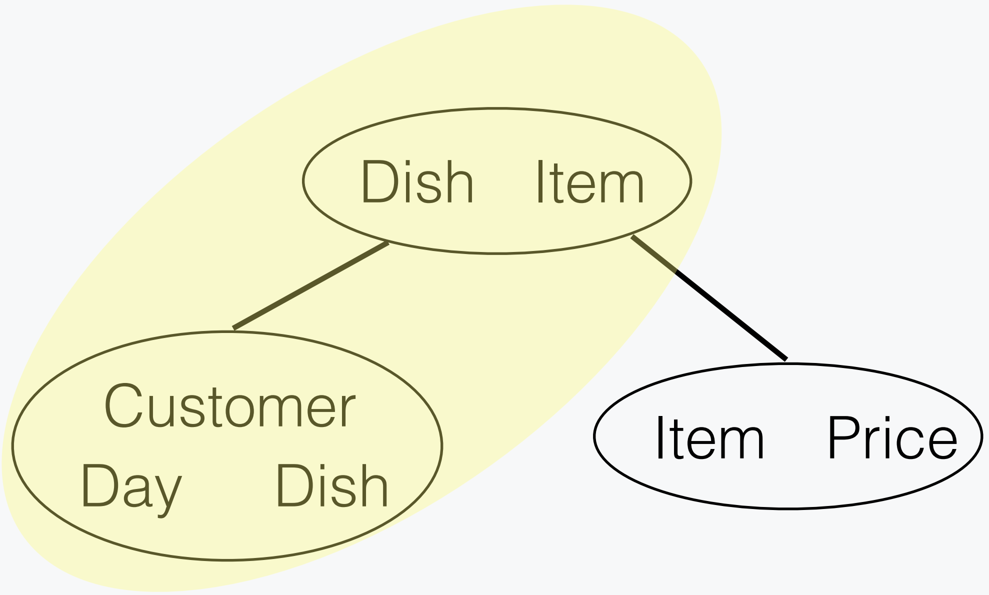
Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4



Valid cover-join plans

→ $(\text{Orders} \overset{\circ}{\bowtie} \text{Dishes}) \overset{\circ}{\bowtie} \text{Items}$

$\text{Orders} \overset{\circ}{\bowtie} (\text{Dishes} \overset{\circ}{\bowtie} \text{Items})$

A non-valid plan

$(\text{Orders} \overset{\circ}{\bowtie} \text{Items}) \overset{\circ}{\bowtie} \text{Dishes}$

Computing a Cover of our Fast Food Database

Orders

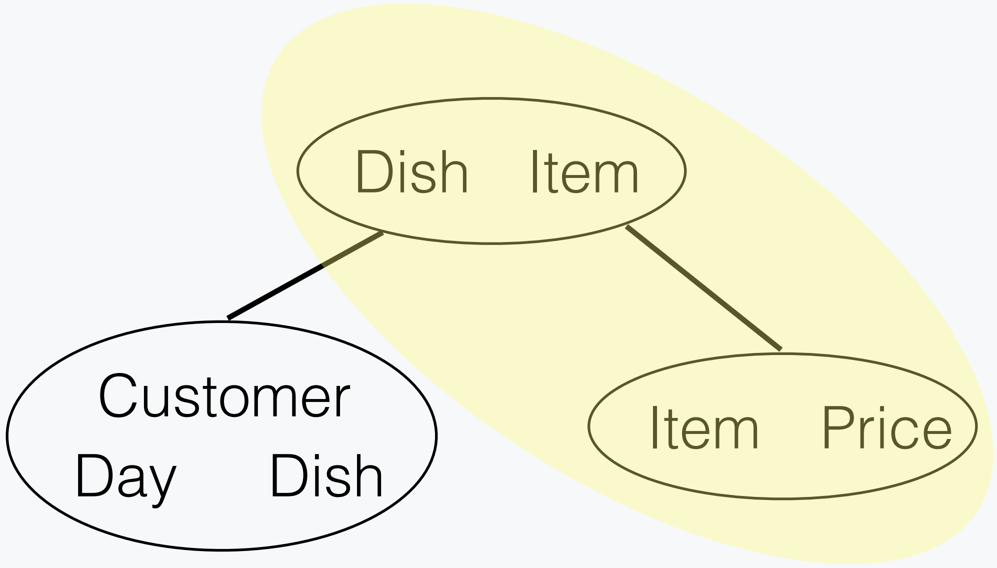
Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4



Valid cover-join plans

$(\text{Orders} \overset{\circ}{\bowtie} \text{Dishes}) \overset{\circ}{\bowtie} \text{Items}$

→ $\text{Orders} \overset{\circ}{\bowtie} (\text{Dishes} \overset{\circ}{\bowtie} \text{Items})$

A non-valid plan

$(\text{Orders} \overset{\circ}{\bowtie} \text{Items}) \overset{\circ}{\bowtie} \text{Dishes}$

Computing a Cover of our Fast Food Database

Orders

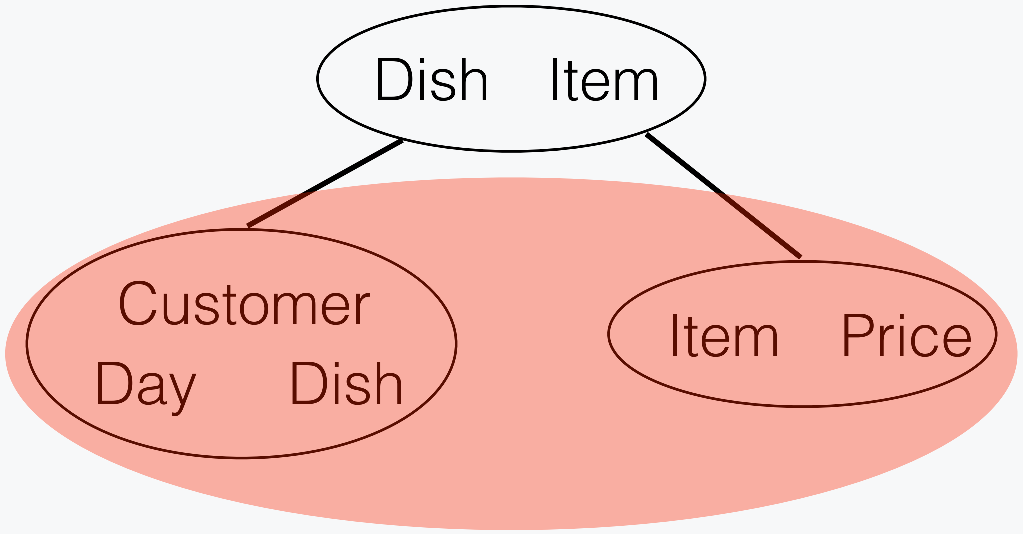
Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4



Valid cover-join plans

$$(\text{Orders} \overset{\circ}{\bowtie} \text{Dishes}) \overset{\circ}{\bowtie} \text{Items}$$

$$\text{Orders} \overset{\circ}{\bowtie} (\text{Dishes} \overset{\circ}{\bowtie} \text{Items})$$

A non-valid plan

→ $(\text{Orders} \overset{\circ}{\bowtie} \text{Items}) \overset{\circ}{\bowtie} \text{Dishes}$

Computing a Cover of our Fast Food Database

Orders

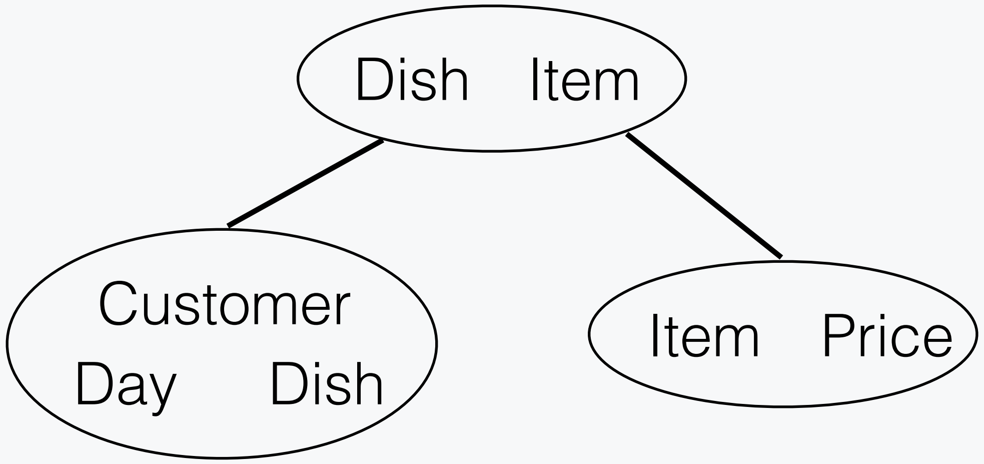
Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4



Valid cover-join plans

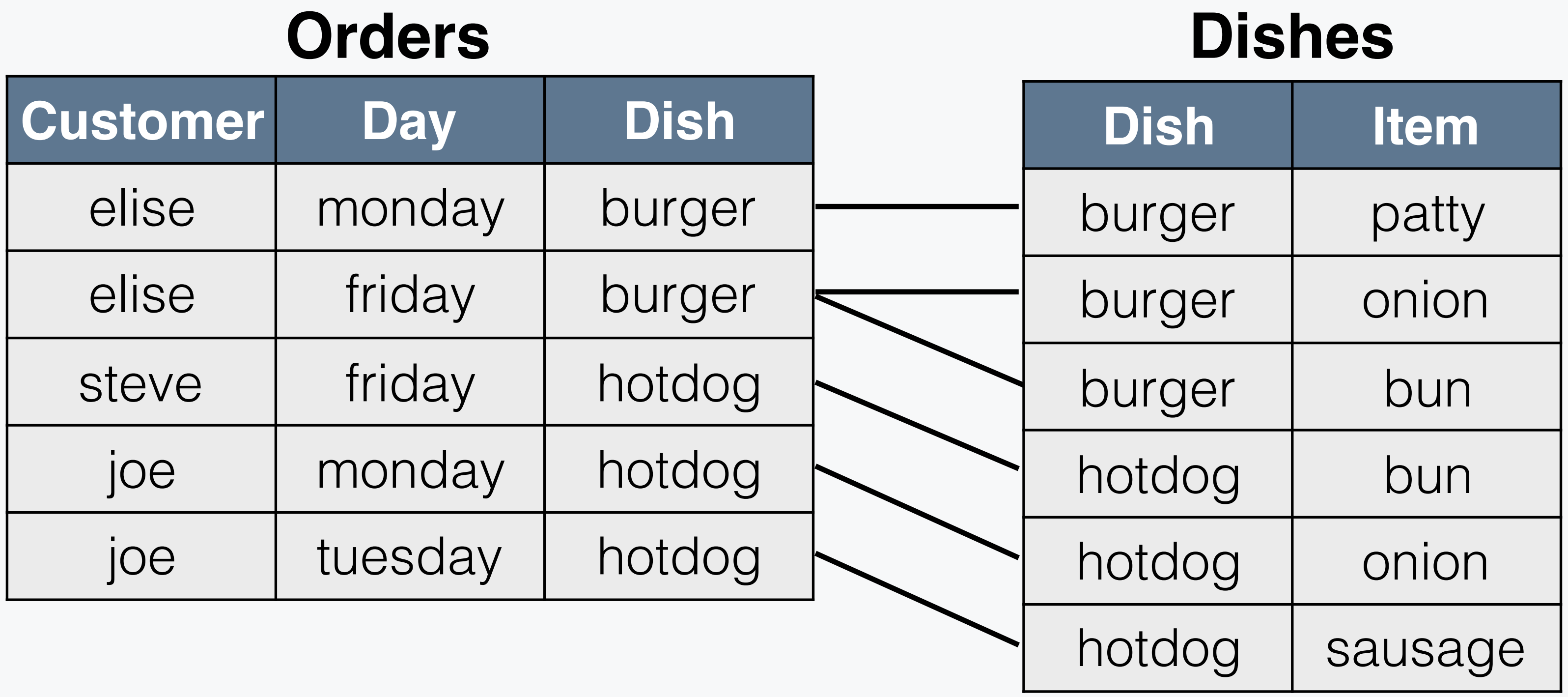
→ $(\text{Orders} \overset{\circ}{\bowtie} \text{Dishes}) \overset{\circ}{\bowtie} \text{Items}$

$\text{Orders} \overset{\circ}{\bowtie} (\text{Dishes} \overset{\circ}{\bowtie} \text{Items})$

A non-valid plan

$(\text{Orders} \overset{\circ}{\bowtie} \text{Items}) \overset{\circ}{\bowtie} \text{Dishes}$

Computing a Cover of our Fast Food Database



Orders

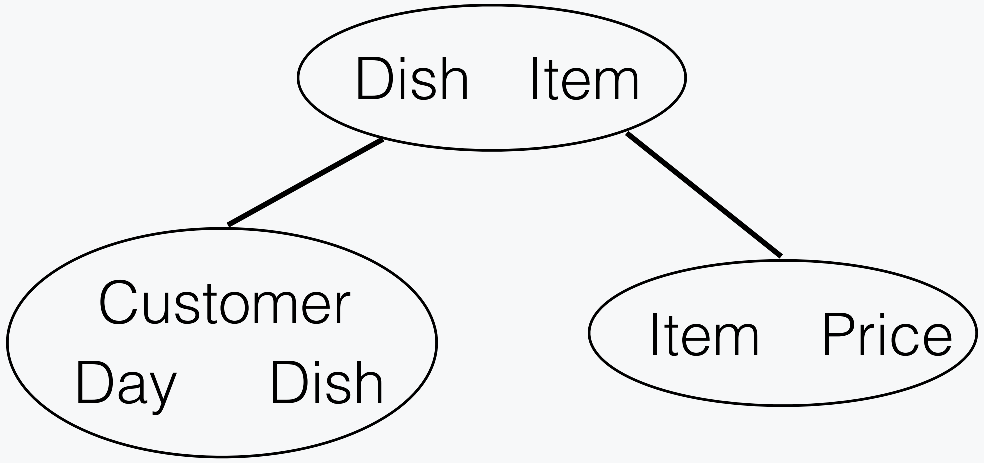
Customer	Day	Dish
elise	monday	burger
elise	friday	burger
steve	friday	hotdog
joe	monday	hotdog
joe	tuesday	hotdog

Dishes

Dish	Item
burger	patty
burger	onion
burger	bun
hotdog	bun
hotdog	onion
hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4



Valid cover-join plans

→ $(\text{Orders} \overset{\circ}{\bowtie} \text{Dishes}) \overset{\circ}{\bowtie} \text{Items}$

$\text{Orders} \overset{\circ}{\bowtie} (\text{Dishes} \overset{\circ}{\bowtie} \text{Items})$

A non-valid plan

$(\text{Orders} \overset{\circ}{\bowtie} \text{Items}) \overset{\circ}{\bowtie} \text{Dishes}$

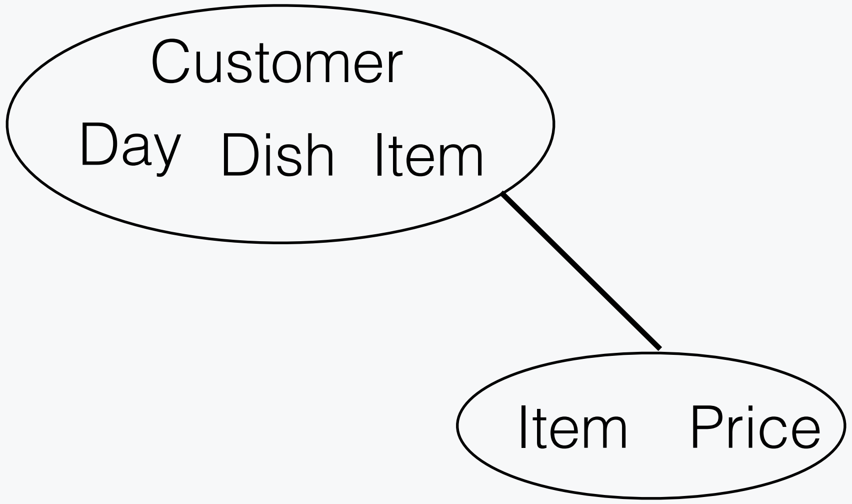
Computing a Cover of our Fast Food Database

Orders $\overset{\circ}{\bowtie}$ Dishes

Customer	Day	Dish	Item
elise	monday	burger	patty
elise	friday	burger	onion
elise	friday	burger	bun
steve	friday	hotdog	bun
joe	monday	hotdog	onion
joe	tuesday	hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4



Valid cover-join plans

→ $(\text{Orders} \overset{\circ}{\bowtie} \text{Dishes}) \overset{\circ}{\bowtie} \text{Items}$

$\text{Orders} \overset{\circ}{\bowtie} (\text{Dishes} \overset{\circ}{\bowtie} \text{Items})$

A non-valid plan

$(\text{Orders} \overset{\circ}{\bowtie} \text{Items}) \overset{\circ}{\bowtie} \text{Dishes}$

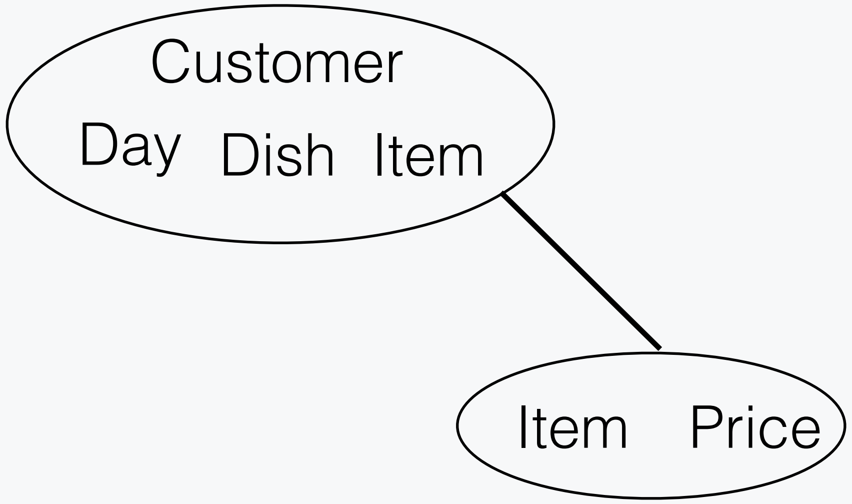
Computing a Cover of our Fast Food Database

Orders $\overset{\circ}{\bowtie}$ Dishes

Customer	Day	Dish	Item
elise	monday	burger	patty
elise	friday	burger	onion
elise	friday	burger	bun
steve	friday	hotdog	bun
joe	monday	hotdog	onion
joe	tuesday	hotdog	sausage

Items

Item	Price
patty	6
onion	2
bun	2
sausage	4



Customer
Day Dish Item

Item Price

Valid cover-join plans

→ (Orders $\overset{\circ}{\bowtie}$ Dishes) $\overset{\circ}{\bowtie}$ Items

Orders $\overset{\circ}{\bowtie}$ (Dishes $\overset{\circ}{\bowtie}$ Items)

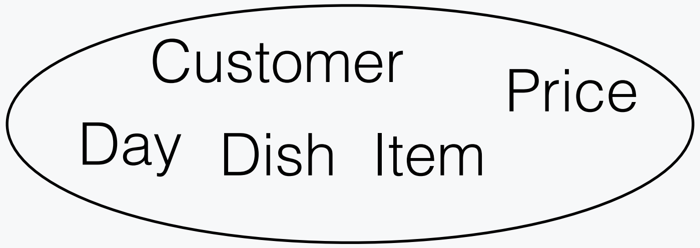
A non-valid plan

(Orders $\overset{\circ}{\bowtie}$ Items) $\overset{\circ}{\bowtie}$ Dishes

Computing a Cover of our Fast Food Database

$$(\textbf{Orders} \overset{\circ}{\bowtie} \textbf{Dishes}) \overset{\circ}{\bowtie} \textbf{Items}$$

Custome	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
steve	friday	hotdog	bun	2
joe	monday	hotdog	onion	2
joe	tuesday	hotdog	sausage	4



Valid cover-join plans

$\longrightarrow$ $(\text{Orders} \overset{\circ}{\bowtie} \text{Dishes}) \overset{\circ}{\bowtie} \text{Items}$

$\text{Orders} \overset{\circ}{\bowtie} (\text{Dishes} \overset{\circ}{\bowtie} \text{Items})$

A non-valid plan

$(\text{Orders} \overset{\circ}{\bowtie} \text{Items}) \overset{\circ}{\bowtie} \text{Dishes}$

Computation Time for Covers

$(\textbf{Orders} \overset{\circ}{\bowtie} \textbf{Dishes}) \overset{\circ}{\bowtie} \textbf{Items}$

Custome	Day	Dish	Item	Price
elise	monday	burger	patty	6
elise	friday	burger	onion	2
elise	friday	burger	bun	2
steve	friday	hotdog	bun	2
joe	monday	hotdog	onion	2
joe	tuesday	hotdog	sausage	4

Computation Time for Covers

Given a query Q, a hypertree decomposition $\mathcal{T}$ of Q and a database $\mathbf{D}$, any cover-join plan for Q over $\mathcal{T}$ computes a cover of $Q(\mathbf{D})$ over $\mathcal{T}$ in time $\widetilde{\mathcal{O}}(|\mathbf{D}|^{\mathrm{fhtw}(\mathcal{T})})$.

Covering the Results of Functional Aggregate Queries (FAQs)

Example

- φ counts for each tuple (a,b,d), the number of tuples in the result containing this tuple:

$$\varphi(a,b,d) = \sum_{c,e,f,g,h} \psi_1(a,b,c) \cdot \psi_2(b,d,e) \cdot \psi_3(d,e,f) \cdot \psi_4(f,h) \cdot \psi_5(e,g)$$

where

$\psi_1(a,b,c) = 1 \Leftrightarrow (a,b,c) \in R_1$ and so on …

Covering the Results of Functional Aggregate Queries (FAQs)

Example

- φ counts for each tuple (a,b,d), the number of tuples in the result containing this tuple:

$$\varphi(a,b,d) = \sum_{c,e,f,g,h} \psi_1(a,b,c) \cdot \psi_2(b,d,e) \cdot \psi_3(d,e,f) \cdot \psi_4(f,h) \cdot \psi_5(e,g)$$

where

$\psi_1(a,b,c) = 1 \Leftrightarrow (a,b,c) \in R_1$ and so on …

- push aggregates past join:

$$\varphi(a,b,d) = \underbrace{\Big(\sum_c \psi_1(a,b,c)\Big)}_{\psi_6(a,b)} \cdot \underbrace{\sum_e \Big(\psi_2(b,d,e) \cdot \underbrace{\sum_f \Big(\psi_3(d,e,f) \cdot \underbrace{\sum_h \psi_4(f,h)}_{\psi_7(f)}\Big)}_{\psi_9(d,e)} \cdot \underbrace{\sum_g \psi_5(e,g)}_{\psi_8(e)}\Big)}_{\psi_{10}(b,d)}$$

Covering the Results of Functional Aggregate Queries (FAQs)

Example

- φ counts for each tuple (a,b,d), the number of tuples in the result containing this tuple:

$$\varphi(a,b,d) = \sum_{c,e,f,g,h} \psi_1(a,b,c) \cdot \psi_2(b,d,e) \cdot \psi_3(d,e,f) \cdot \psi_4(f,h) \cdot \psi_5(e,g)$$

where

$\psi_1(a,b,c) = 1 \Leftrightarrow (a,b,c) \in R_1$ and so on …

- push aggregates past join:

$$\varphi(a,b,d) = \underbrace{\Big(\sum_c \psi_1(a,b,c)\Big)}_{\psi_6(a,b)} \cdot \underbrace{\sum_e \Big(\psi_2(b,d,e) \cdot \underbrace{\sum_f \Big(\psi_3(d,e,f) \cdot \underbrace{\sum_h \psi_4(f,h)}_{\psi_7(f)}\Big)}_{\psi_9(d,e)} \cdot \underbrace{\sum_g \psi_5(e,g)}_{\psi_8(e)}\Big)}_{\psi_{10}(b,d)}$$

- Compute a cover of the join of the relational encodings of $\psi_6(a,b)$ and $\psi_{10}(b,d)$

Functional Aggregate Queries [Khamis et al., PODS 2016]

FAQs encompass many problems in

- Constraint Satisfaction,
- Logic,
- Databases,
- Matrix Operations,
- Probabilistic Graphical Models,
- Machine Learning,
- …

Computation Time of an FAQ result

$$\widetilde{\mathcal{O}}(|\mathbf{D}|^{\mathrm{faqw}(\varphi)} + \text{output size})$$

Computation Time of an FAQ-Cover

$$\widetilde{\mathcal{O}}(|\mathbf{D}|^{\mathrm{faqw}(\varphi)})$$

Summary

Motivation

- Query results entail redundancies in representation and computation
- Design of a representation system for query results which reduces such redundancies

Covers

- One-relational, lossless, succinct representation of query results
- Allow for constant-delay enumeration
- Can be applied to a wide range of computational problems

On-going work

- Usage of covers in distributed databases to reduce communication complexity