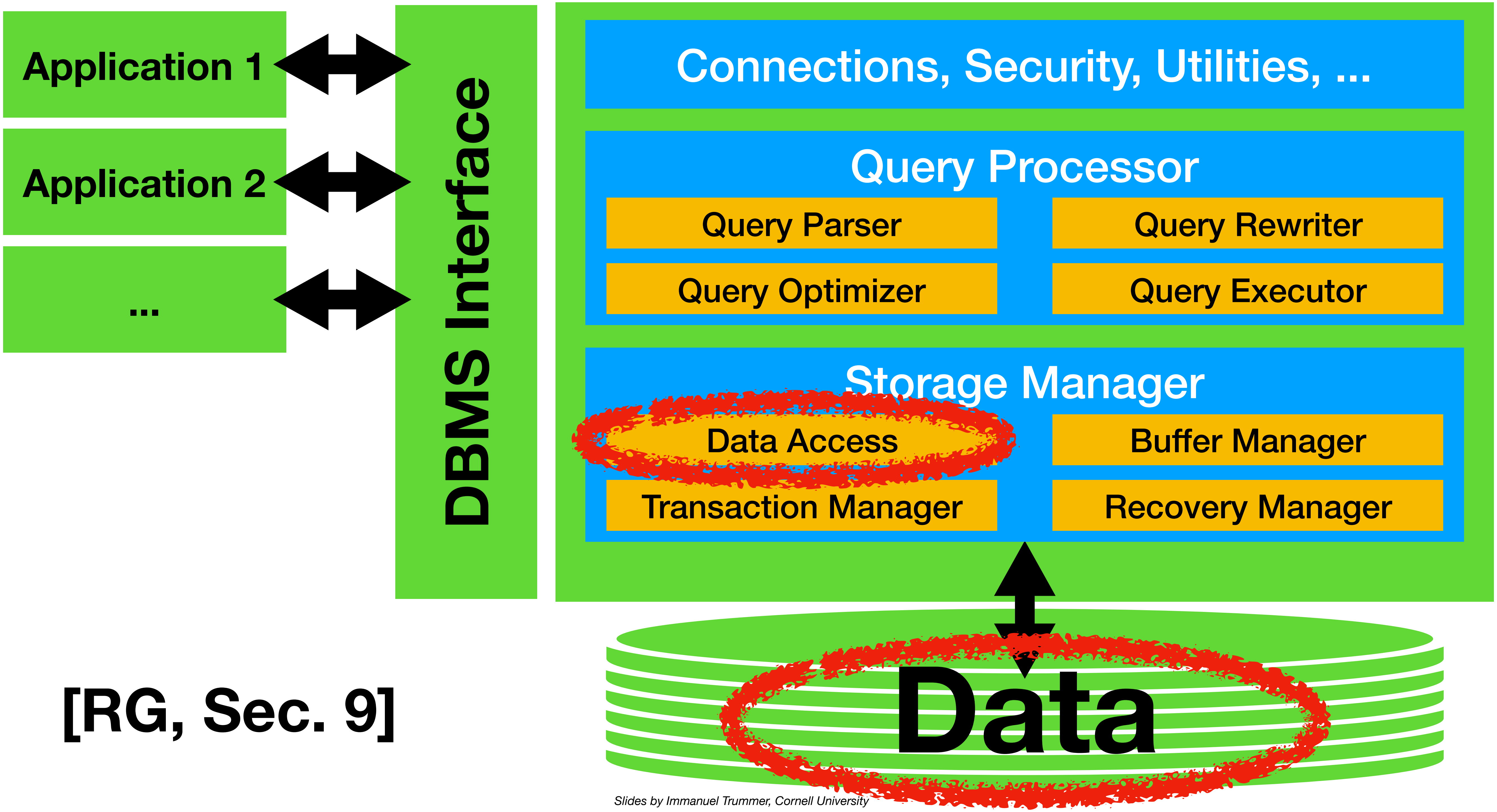
Tree Indexes

Immanuel Trummer
itrummer@cornell.edu
www.itrummer.org

Database Management Systems (DBMS)



Application 1

Application 2

...

DBMS Interface

Connections, Security, Utilities, ...

Query Processor

Query Parser

Query Rewriter

Query Optimizer

Query Executor

Storage Manager

Data Access

Buffer Manager

Transaction Manager

Recovery Manager

Data

[RG, Sec. 9]

Slides by Immanuel Trummer, Cornell University

Quickly Finding Data

- Table **Enrollment(sid, cid)** links students to courses
- E.g., search entries for **specific student** (e.g., sid=5)
- Data stored as unordered file - must **scan** all pages!
- Better: data sorted by student ID - apply **binary search**!
- **Problem**: sometimes search for **specific courses**!
- Only **one sort order**, cannot duplicate data ...

Slides by Immanuel Trummer, Cornell University

Solution: Indexes

- **Index**: auxiliary data structure for **finding data faster**
- Exactly same principle as for **books**!
- Can have **multiple indexes** for same table, e.g.
  - One index for finding info on specific **students**
  - One index for finding info on specific **courses**

How It Works

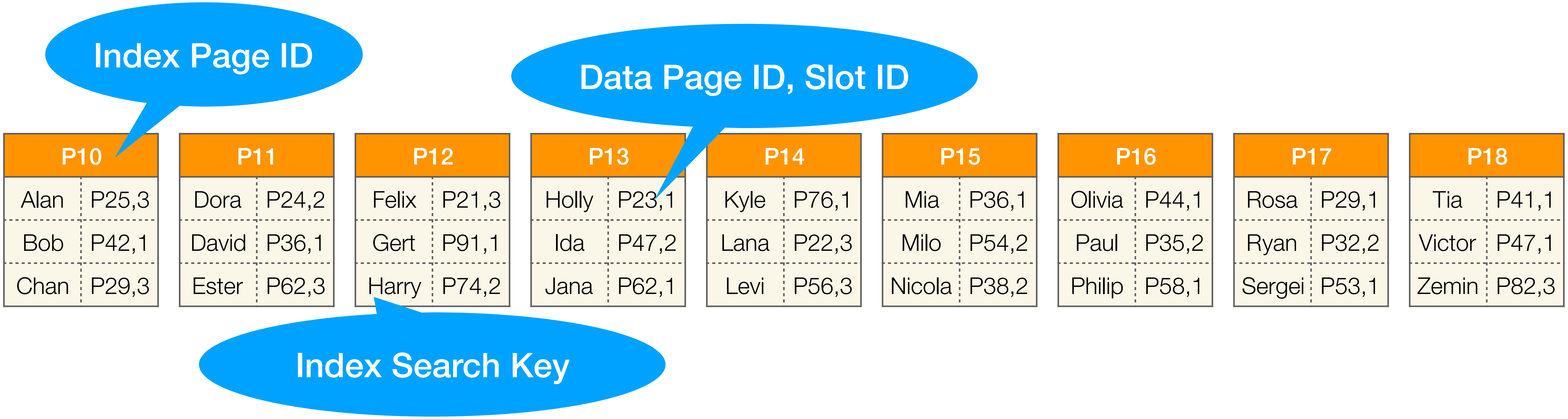
- Index stores **references** to data records
  - I.e., stores **page** IDs and **slot** IDs
- Index **groups records** by values in specific columns
- Those columns are called the **index search key**
- Index retrieves records for specific **search key values**

Example

P10		P11		P12		P13		P14		P15		P16		P17		P18	
Alan	P25,3	Dora	P24,2	Felix	P21,3	Holly	P23,1	Kyle	P76,1	Mia	P36,1	Olivia	P44,1	Rosa	P29,1	Tia	P41,1
Bob	P42,1	David	P36,1	Gert	P91,1	Ida	P47,2	Lana	P22,3	Milo	P54,2	Paul	P35,2	Ryan	P32,2	Victor	P47,1
Chan	P29,3	Ester	P62,3	Harry	P74,2	Jana	P62,1	Levi	P56,3	Nicola	P38,2	Philip	P58,1	Sergei	P53,1	Zemin	P82,3

Index by Student Name

Example



Index Page ID

Data Page ID, Slot ID

P10		P11		P12		P13		P14		P15		P16		P17		P18	
Alan	P25,3	Dora	P24,2	Felix	P21,3	Holly	P23,1	Kyle	P76,1	Mia	P36,1	Olivia	P44,1	Rosa	P29,1	Tia	P41,1
Bob	P42,1	David	P36,1	Gert	P91,1	Ida	P47,2	Lana	P22,3	Milo	P54,2	Paul	P35,2	Ryan	P32,2	Victor	P47,1
Chan	P29,3	Ester	P62,3	Harry	P74,2	Jana	P62,1	Levi	P56,3	Nicola	P38,2	Philip	P58,1	Sergei	P53,1	Zemin	P82,3

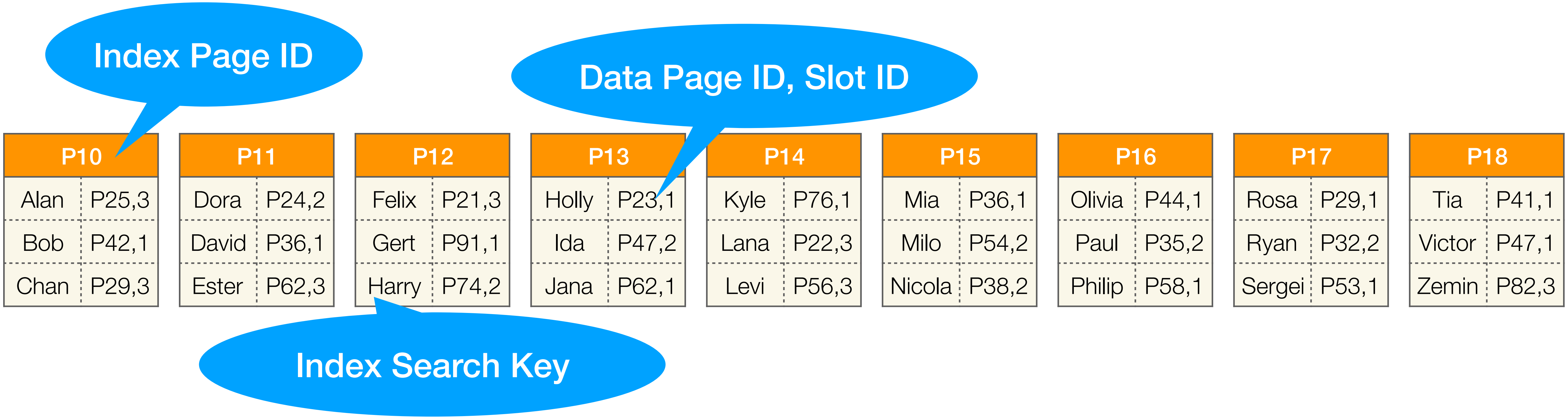
Index Search Key

Index by Student Name

Slides by Immanuel Trummer, Cornell University

Example

Searching for Student "Alan"



Index Page ID

Data Page ID, Slot ID

Index Search Key

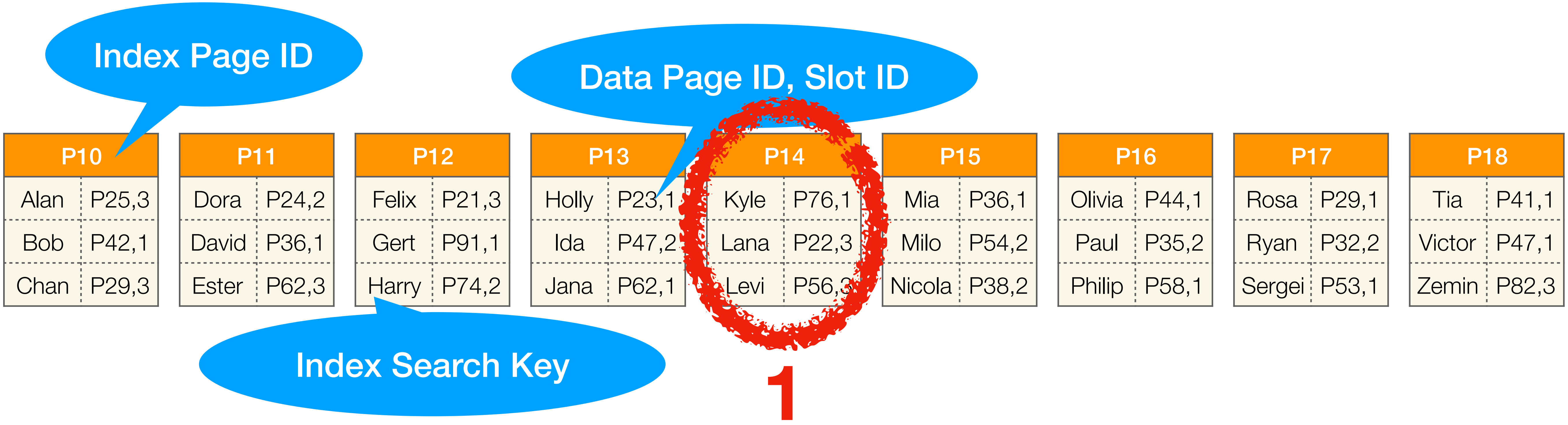
P10		P11		P12		P13		P14		P15		P16		P17		P18	
Alan	P25,3	Dora	P24,2	Felix	P21,3	Holly	P23,1	Kyle	P76,1	Mia	P36,1	Olivia	P44,1	Rosa	P29,1	Tia	P41,1
Bob	P42,1	David	P36,1	Gert	P91,1	Ida	P47,2	Lana	P22,3	Milo	P54,2	Paul	P35,2	Ryan	P32,2	Victor	P47,1
Chan	P29,3	Ester	P62,3	Harry	P74,2	Jana	P62,1	Levi	P56,3	Nicola	P38,2	Philip	P58,1	Sergei	P53,1	Zemin	P82,3

Index by Student Name

Slides by Immanuel Trummer, Cornell University

Example

Searching for Student "Alan"



Index Page ID

Data Page ID, Slot ID

Index Search Key

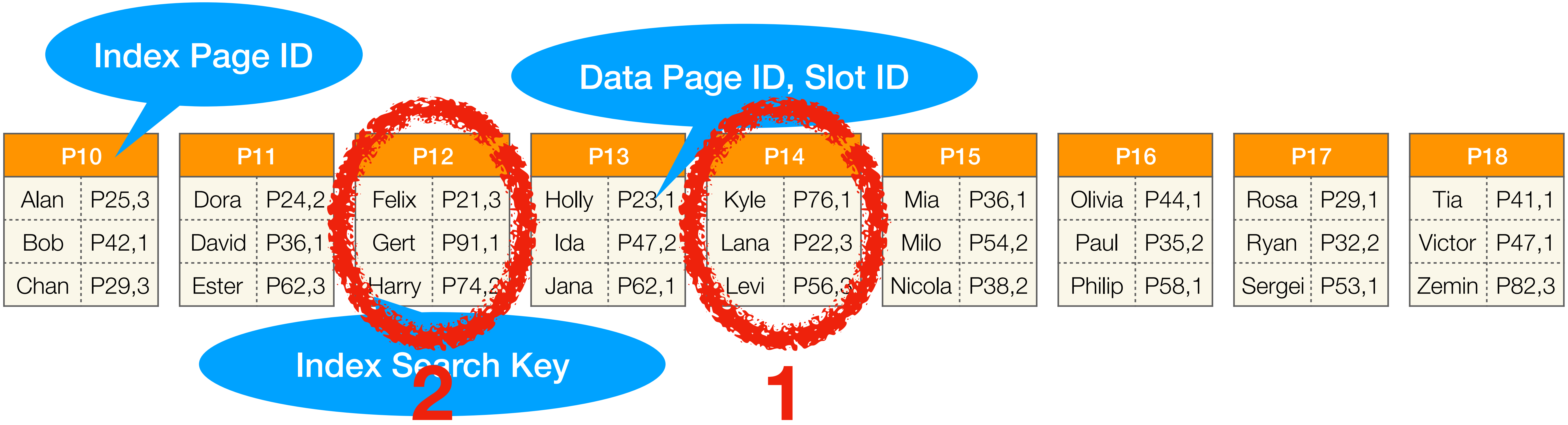
P10		P11		P12		P13		P14		P15		P16		P17		P18	
Alan	P25,3	Dora	P24,2	Felix	P21,3	Holly	P23,1	Kyle	P76,1	Mia	P36,1	Olivia	P44,1	Rosa	P29,1	Tia	P41,1
Bob	P42,1	David	P36,1	Gert	P91,1	Ida	P47,2	Lana	P22,3	Milo	P54,2	Paul	P35,2	Ryan	P32,2	Victor	P47,1
Chan	P29,3	Ester	P62,3	Harry	P74,2	Jana	P62,1	Levi	P56,3	Nicola	P38,2	Philip	P58,1	Sergei	P53,1	Zemin	P82,3

Index by Student Name

Slides by Immanuel Trummer, Cornell University

Example

Searching for Student "Alan"

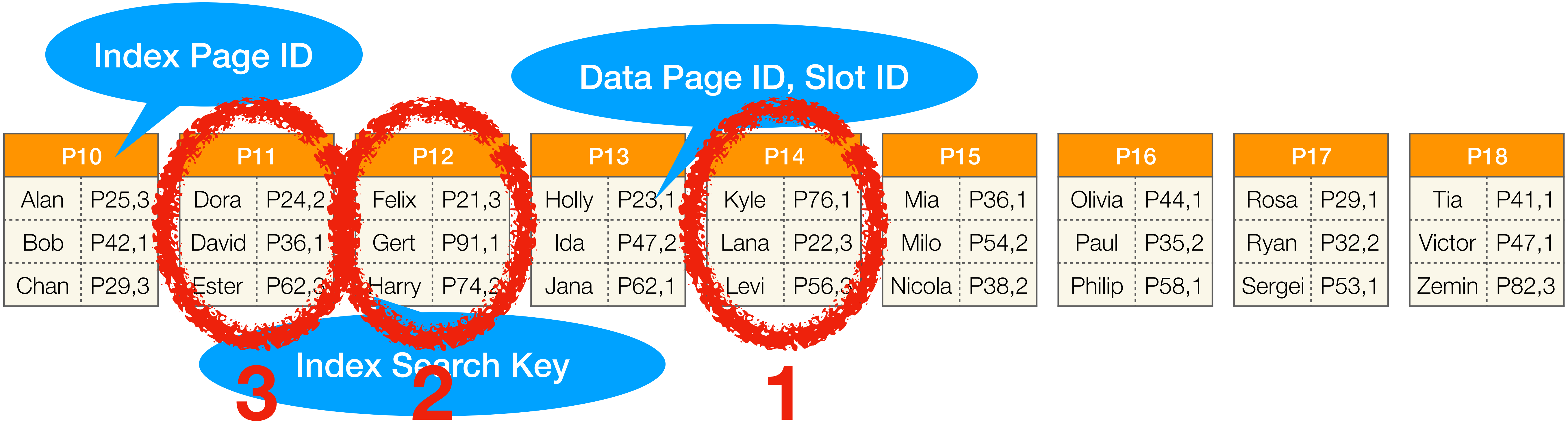


P10		P11		P12		P13		P14		P15		P16		P17		P18	
Alan	P25,3	Dora	P24,2	Felix	P21,3	Holly	P23,1	Kyle	P76,1	Mia	P36,1	Olivia	P44,1	Rosa	P29,1	Tia	P41,1
Bob	P42,1	David	P36,1	Gert	P91,1	Ida	P47,2	Lana	P22,3	Milo	P54,2	Paul	P35,2	Ryan	P32,2	Victor	P47,1
Chan	P29,3	Ester	P62,3	Harry	P74,2	Jana	P62,1	Levi	P56,3	Nicola	P38,2	Philip	P58,1	Sergei	P53,1	Zemin	P82,3

Index by Student Name

Example

Searching for Student "Alan"



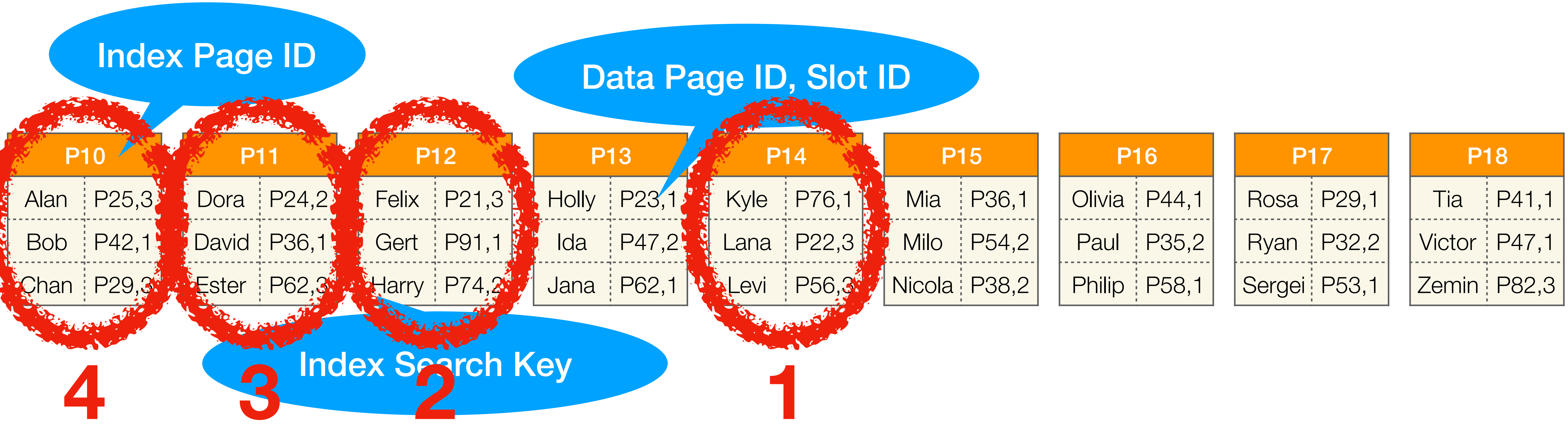
P10		P11		P12		P13		P14		P15		P16		P17		P18	
Alan	P25,3	Dora	P24,2	Felix	P21,3	Holly	P23,1	Kyle	P76,1	Mia	P36,1	Olivia	P44,1	Rosa	P29,1	Tia	P41,1
Bob	P42,1	David	P36,1	Gert	P91,1	Ida	P47,2	Lana	P22,3	Milo	P54,2	Paul	P35,2	Ryan	P32,2	Victor	P47,1
Chan	P29,3	Ester	P62,3	Harry	P74,2	Jana	P62,1	Levi	P56,3	Nicola	P38,2	Philip	P58,1	Sergei	P53,1	Zemin	P82,3

Index by Student Name

Slides by Immanuel Trummer, Cornell University

Example

Searching for Student "Alan"



P10		P11		P12		P13		P14		P15		P16		P17		P18	
Alan	P25,3	Dora	P24,2	Felix	P21,3	Holly	P23,1	Kyle	P76,1	Mia	P36,1	Olivia	P44,1	Rosa	P29,1	Tia	P41,1
Bob	P42,1	David	P36,1	Gert	P91,1	Ida	P47,2	Lana	P22,3	Milo	P54,2	Paul	P35,2	Ryan	P32,2	Victor	P47,1
Chan	P29,3	Ester	P62,3	Harry	P74,2	Jana	P62,1	Levi	P56,3	Nicola	P38,2	Philip	P58,1	Sergei	P53,1	Zemin	P82,3

Index by Student Name

Slides by Immanuel Trummer, Cornell University

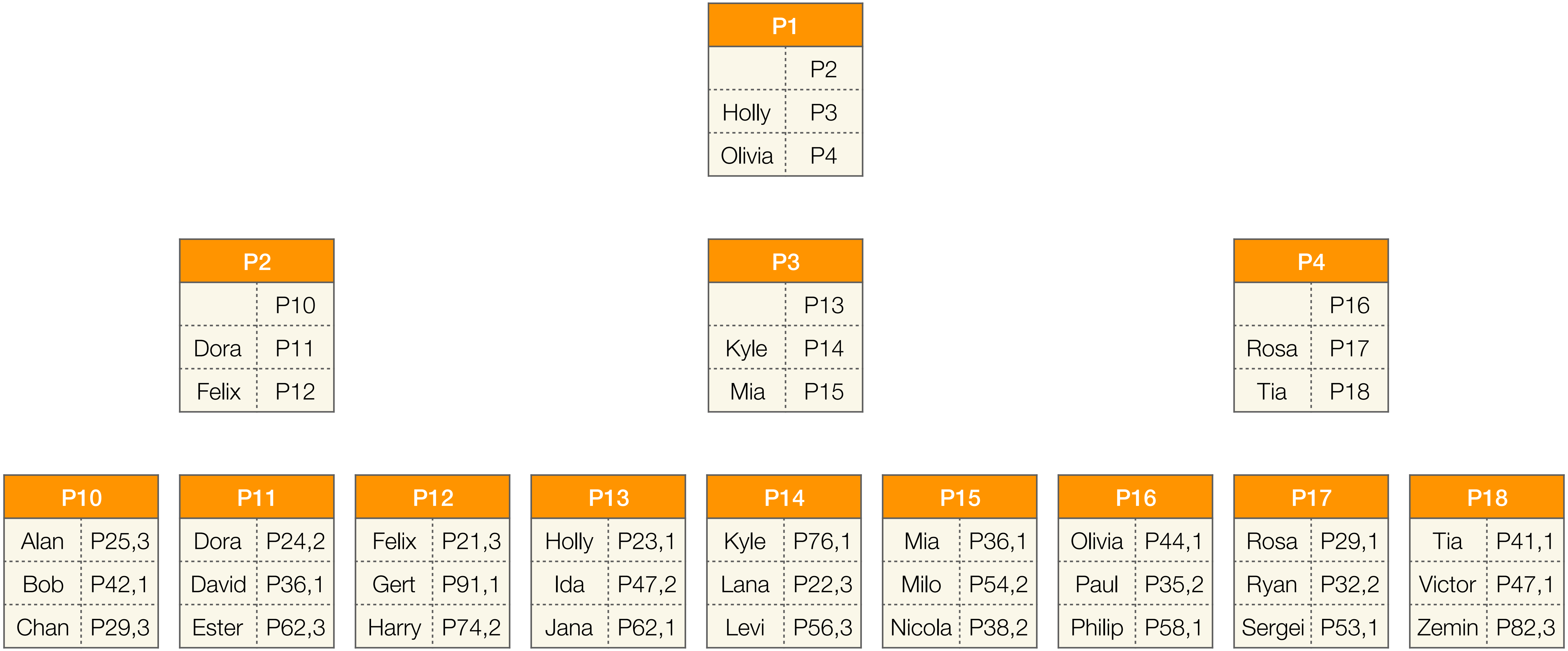
Ideas for Improvements

- Binary search narrows down "search space" by **factor 2**
- Can we get a **higher pruning factor** per page read?
- Idea: (non-binary) **search trees**!

Index Types

- Tree indexes **Now!**
- Hash indexes **Next Lecture**

Example



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

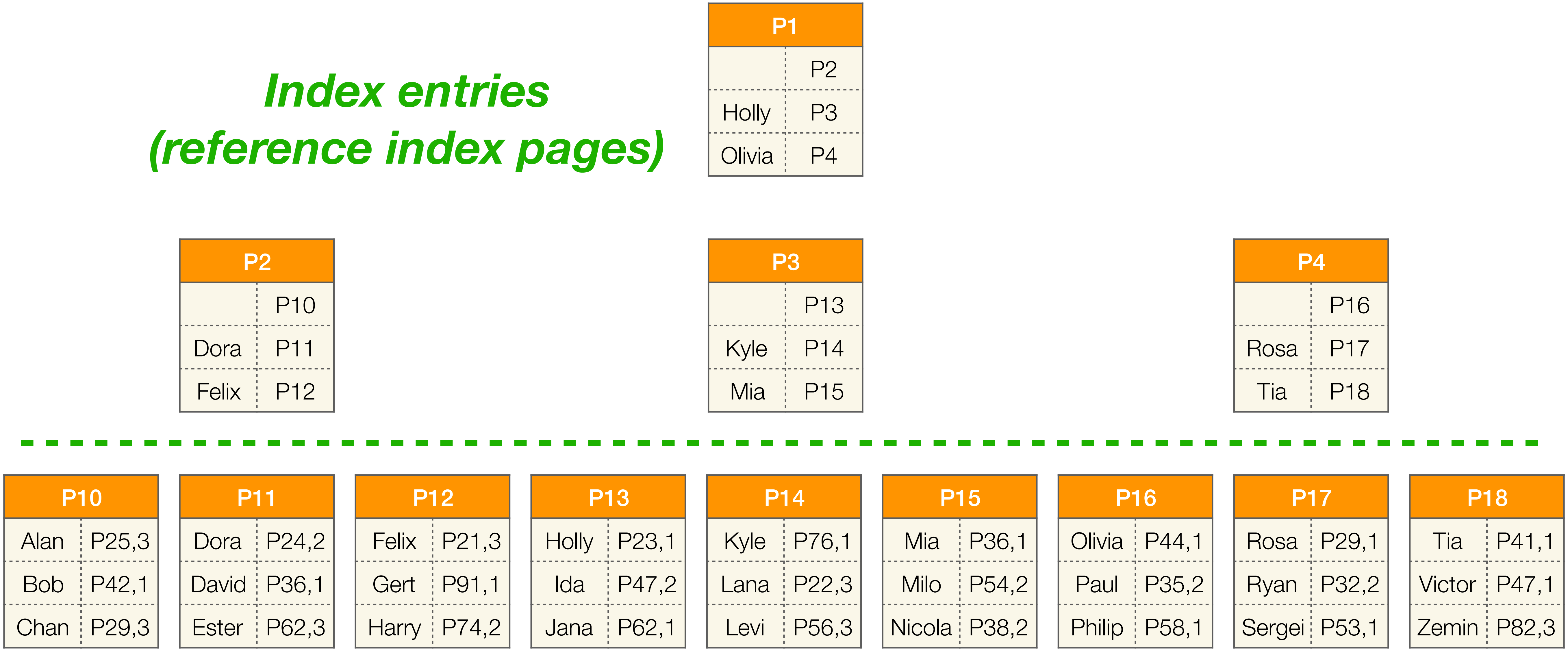
P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Index by Student Name

Example



Index entries
(reference index pages)

P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

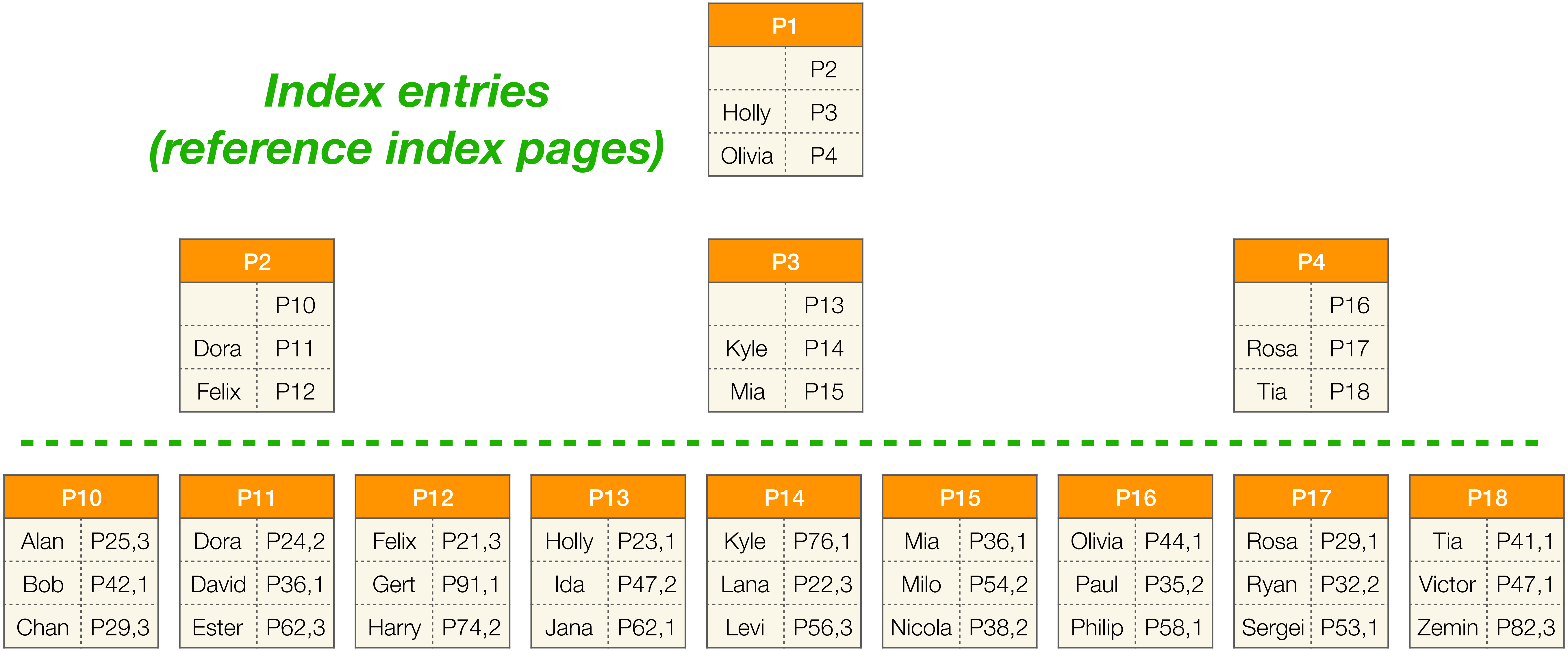
P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Data entries
(reference data pages)

Index by Student Name

Example



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Index entries (reference index pages)

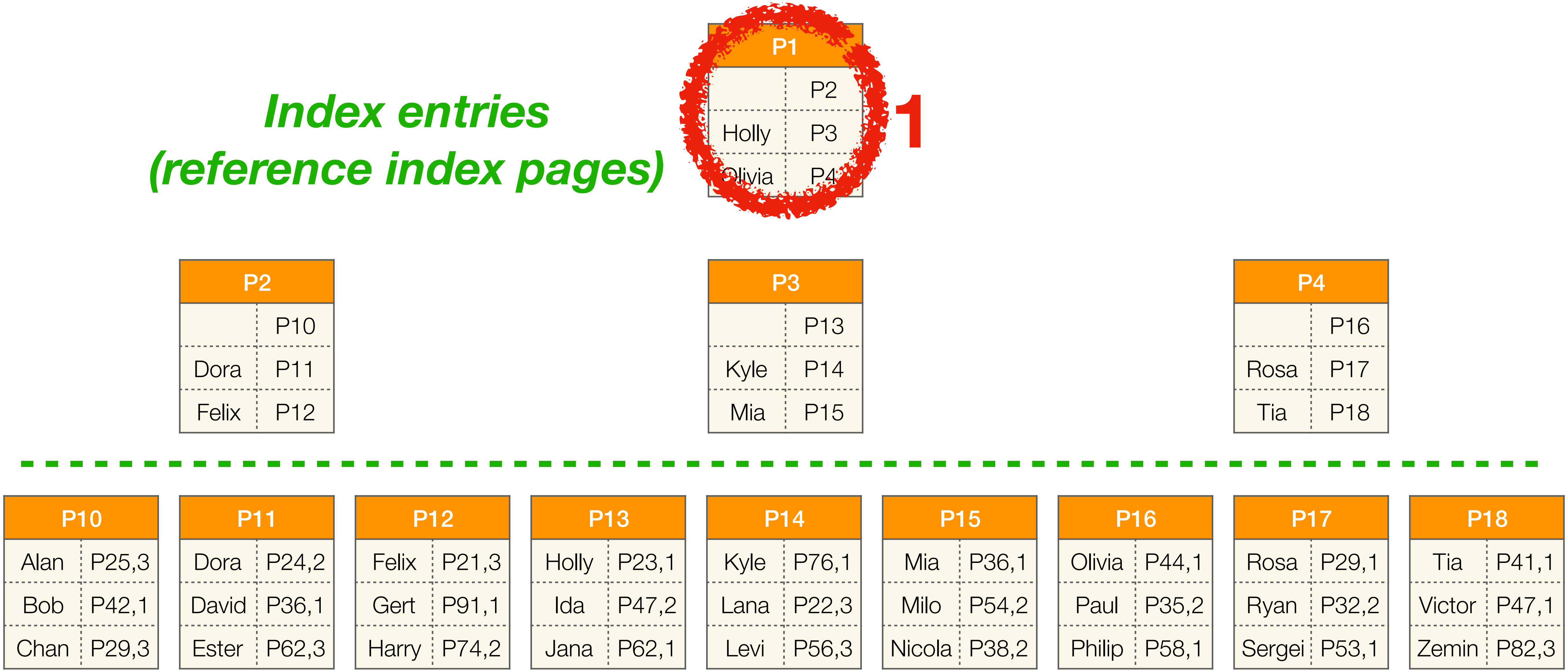
Data entries (reference data pages)

Searching for Student "Alan"

Index by Student Name

Example

Index entries
(reference index pages)



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

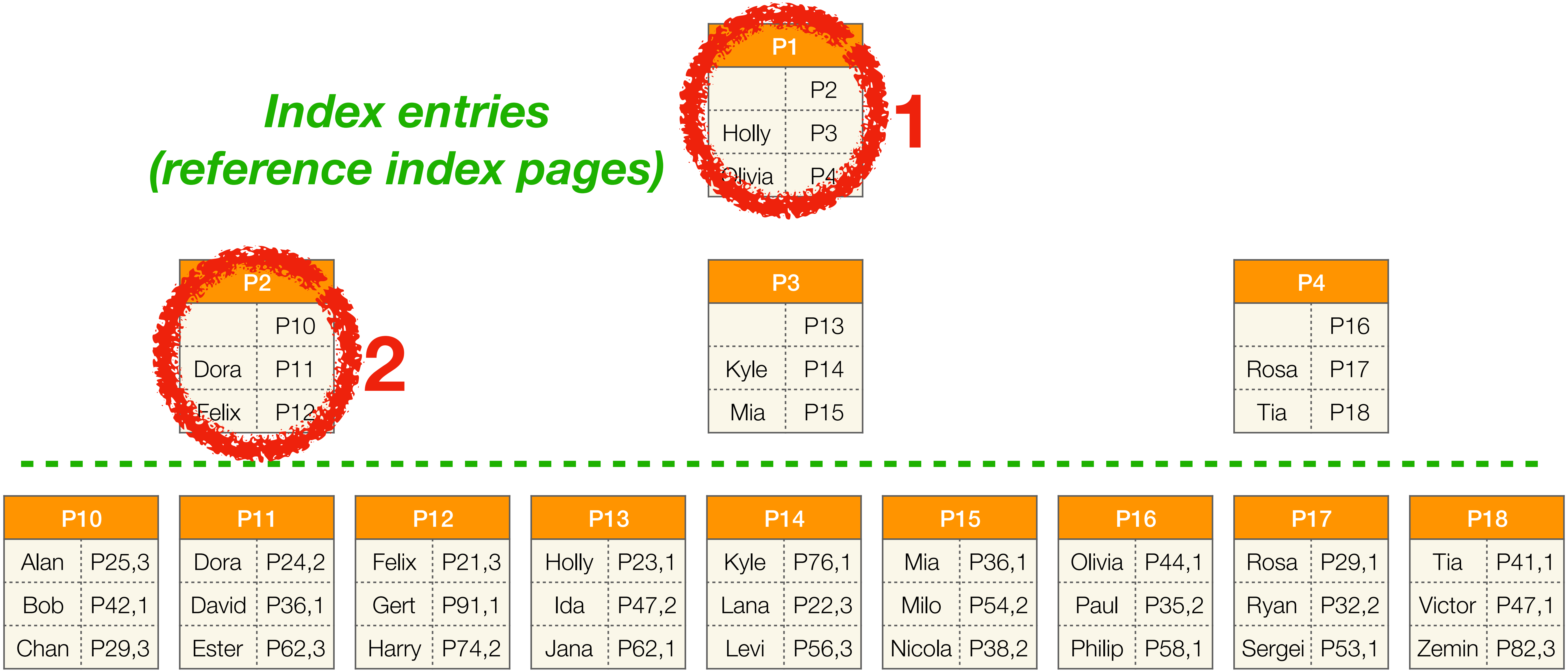
P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Data entries
(reference data pages)

Searching for Student "Alan"

Index by Student Name

Example



Index entries
(reference index pages)

P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

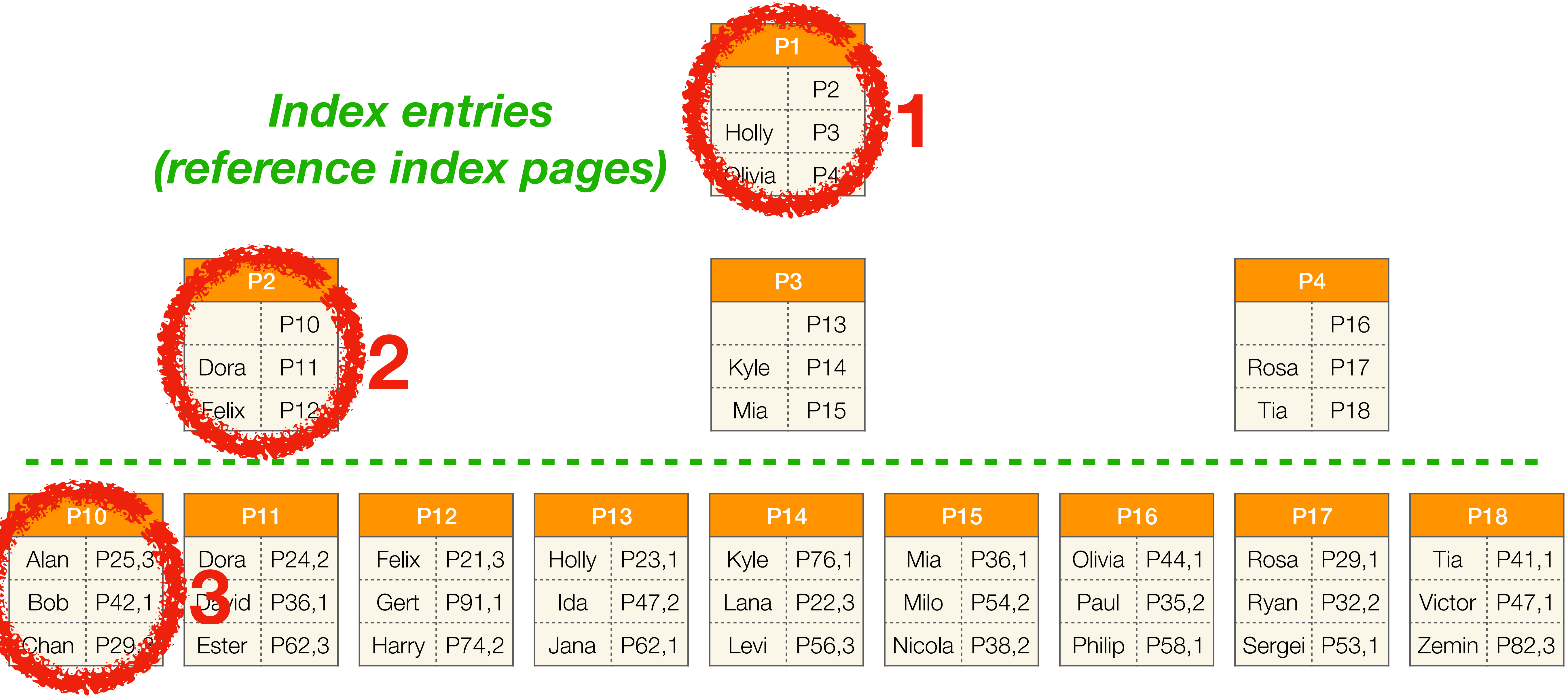
Data entries
(reference data pages)

Searching for Student "Alan"

Index by Student Name

Example

Index entries (reference index pages)



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,[illegible]

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Data entries (reference data pages)

Searching for Student "Alan"

Index by Student Name

Index Node Content

- Content of **inner nodes**:
    - R(0), K(1), R(1), K(2), R(2), ...
    - R(i) leads to entries (strictly) ordered **before** K(i+1)
    - R(i) references an **index** page
- Content of **leaf nodes**:
    - K(1), R(1), K(2), R(2), K(3), R(3), ...
    - R(i) leads to data entries with **key K(i)**
    - R(i) references a **data page** and a **slot** on that page

Where to Use Tree Indexes?

- Can use index for queries with **equality predicates**
  - E.g., ... **WHERE Sname = 'Alan'**
- Can use index for queries with **inequality predicates**
  - E.g., ... **WHERE gpa > 3**
- Both cases: works if predicate references **index key**

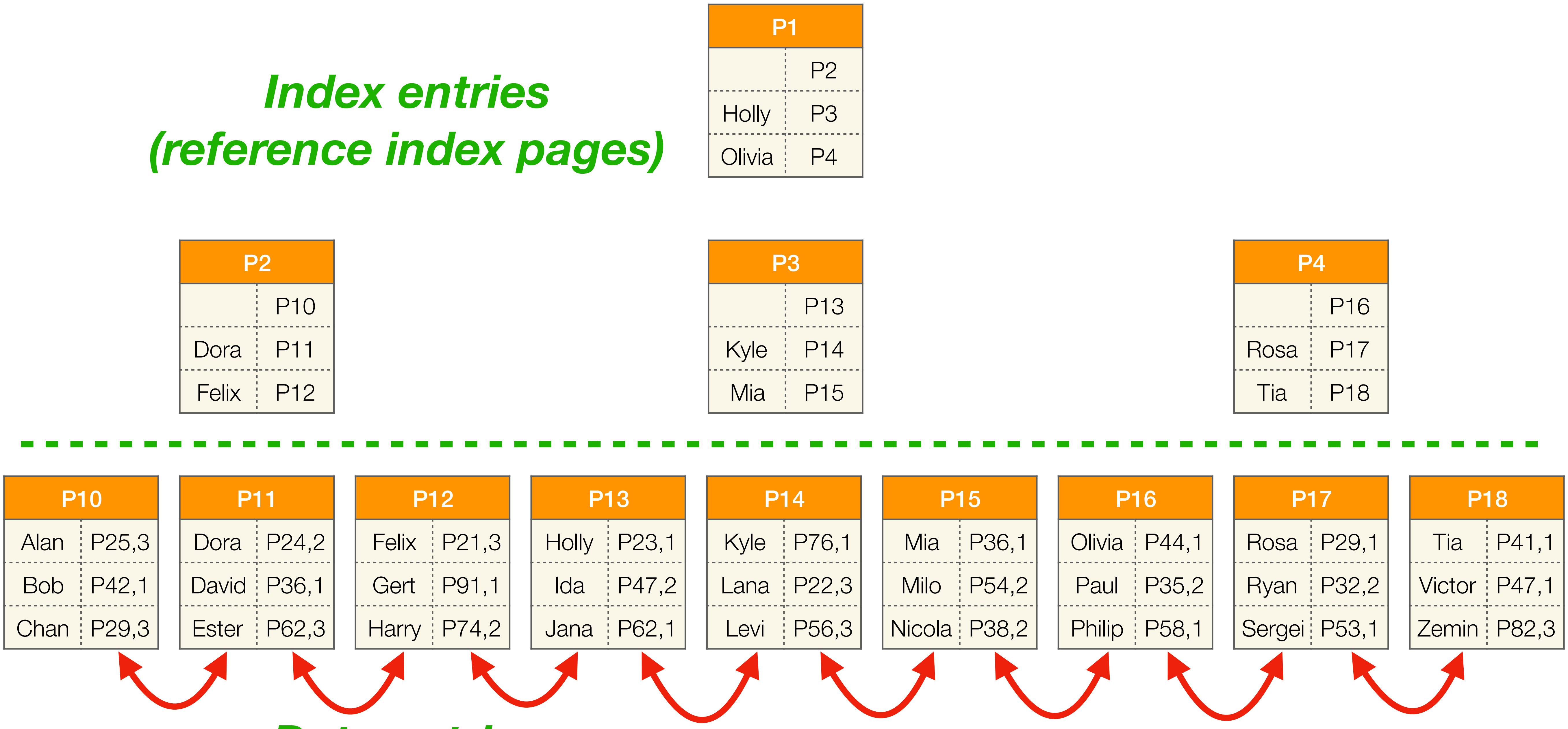
Using Index for Equality

- Searching for entries with **key value V** → Start at **root** node
- Until reaching a leaf node:
  - Search for i such that **$V \geq K(i)$, $V < K(i+1)$**
  - Follow associated **reference $R(i)$**
- At leaf node:
  - Search for i such that **$K(i) = V$**
  - **Retrieve data** from $R(i)$ if found, otherwise **return empty**

Linking Leaf Nodes

- Often want entries from **neighboring** leaf nodes
- Could get leaf node references from **parent** nodes
- Better: **store pointer** to next/previous neighbor in leaf
- Leaf pages essentially become **doubly linked list**

Example



Index entries (reference index pages)

P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Data entries (reference data pages)

Index by Student Name

Using Index for Inequalities

- Searching for index entries with **key value from [L,U]**
- Use equality search procedure to **find entry with value L**
- Follow links between leaf nodes until **reaching value U**
- Retrieve **referenced data** on the way

Composite Keys

- Index search key may consist of **multiple columns**
- Must decide **priority order** between key columns
- **Key comparisons** use that priority order
    - I.e., consider second column if same value in first etc.
- Can use index for (in)equalities on ***prefix*** of key columns

Explain Restriction to Key Prefix!

Indexes in Postgres

- CREATE INDEX **<index-name>** on **<table>** (**<columns>**)
  - Creates index for table using specified search key
  - Refer to index later via **<index-name>**
  - **<columns>** is comma-separated column list (key)
- DROP INDEX **<index-name>**
  - Delete index with given name

Which Indexes to Create?

- Depends a lot on your **typical queries**
- Analyze **predicates** of queries for index ideas
- Too many indexes can be **bad** for performance
- Active area of **research** in databases
  - Some tools are available, e.g.:
    - **Dexter**: https://ankane.org/introducing-dexter

Concise Data Entries

- Many references for **same search key** value?
  - Optimization: store search key value with **reference list**
  - Advantage: avoids storing key values **redundantly**
  - Disadvantage: creates **variable length** field (list)

Merging Index and Data

- Idea: **index stores data** instead of references to data
- This is called a **clustered index**
- Can have **at most one** clustered index per table (why?)
- More efficient as it saves chasing **one reference**
- More importantly: **collocates** data with same key

Slides by Immanuel Trummer, Cornell University

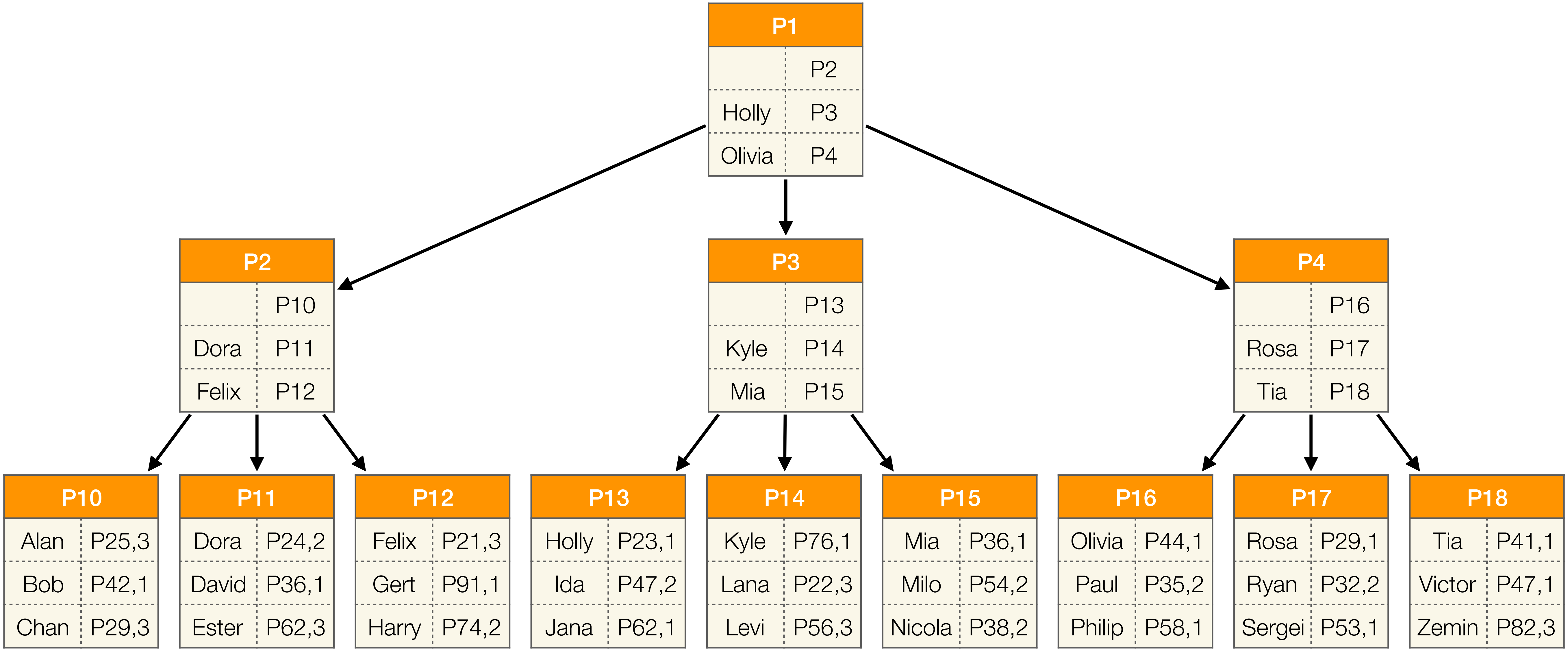
Tree Index Variants

- **B+ tree index** (focus so far) is very popular
- Some tree indexes put data references in **inner nodes**
- Some **omit the links** between leaf nodes
- ...
- Also: differences in how **updates** are handled (next)!

Handling Updates

- Index refers to database table
- If **data changes**, so must the index!
- Need to **change index** in case of inserts/deletes
- Ideally: want to keep index **balanced** during updates
  - Not required but can improve **efficiency**!

Updates without Balancing



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

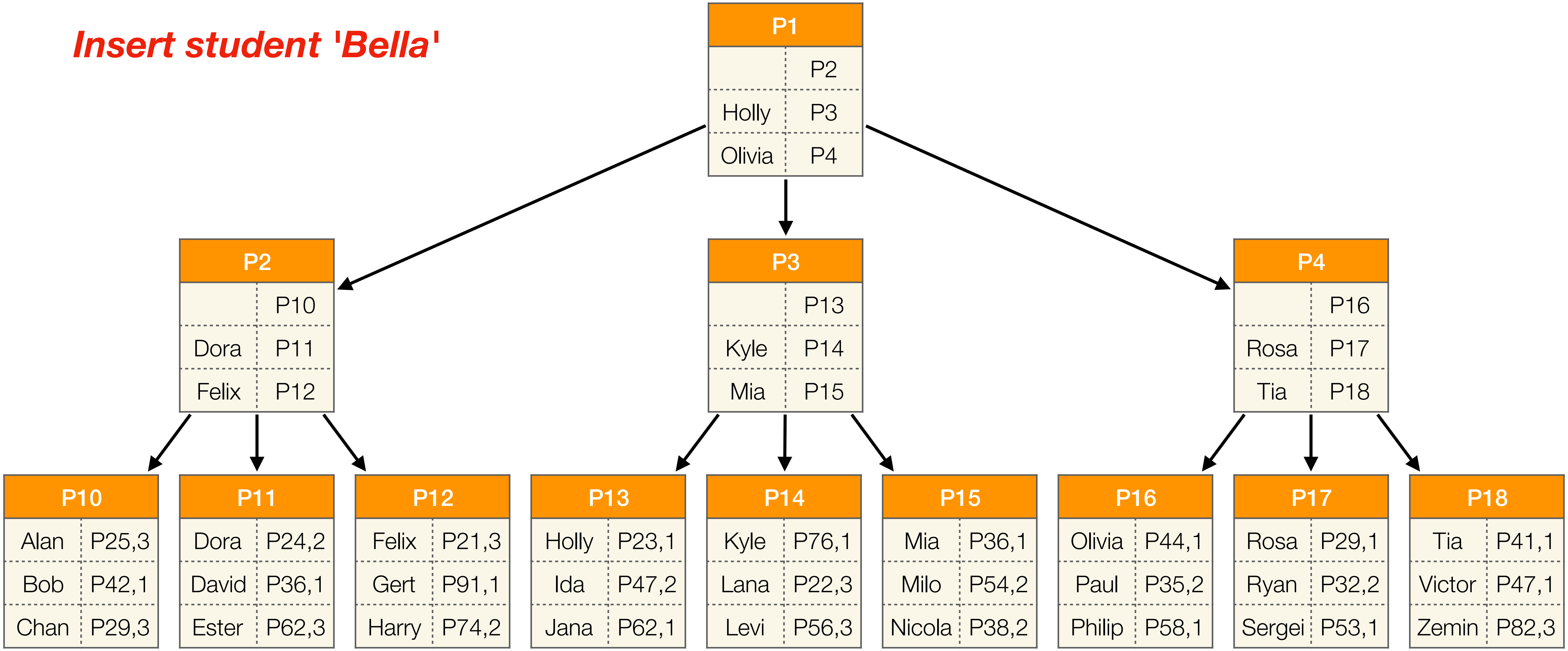
P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Slides by Immanuel Trummer, Cornell University

Updates without Balancing

Insert student 'Bella'



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

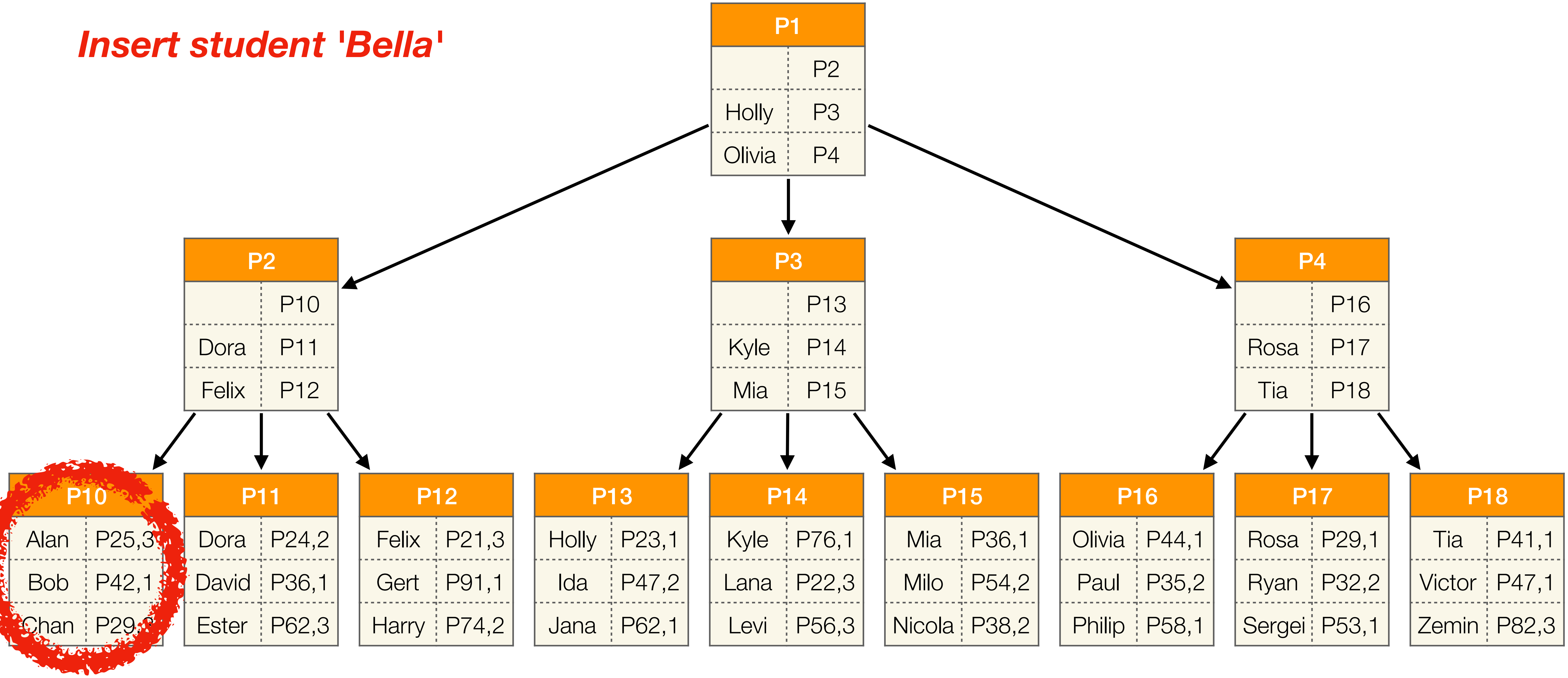
P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Updates without Balancing

Insert student 'Bella'



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,[illegible]

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

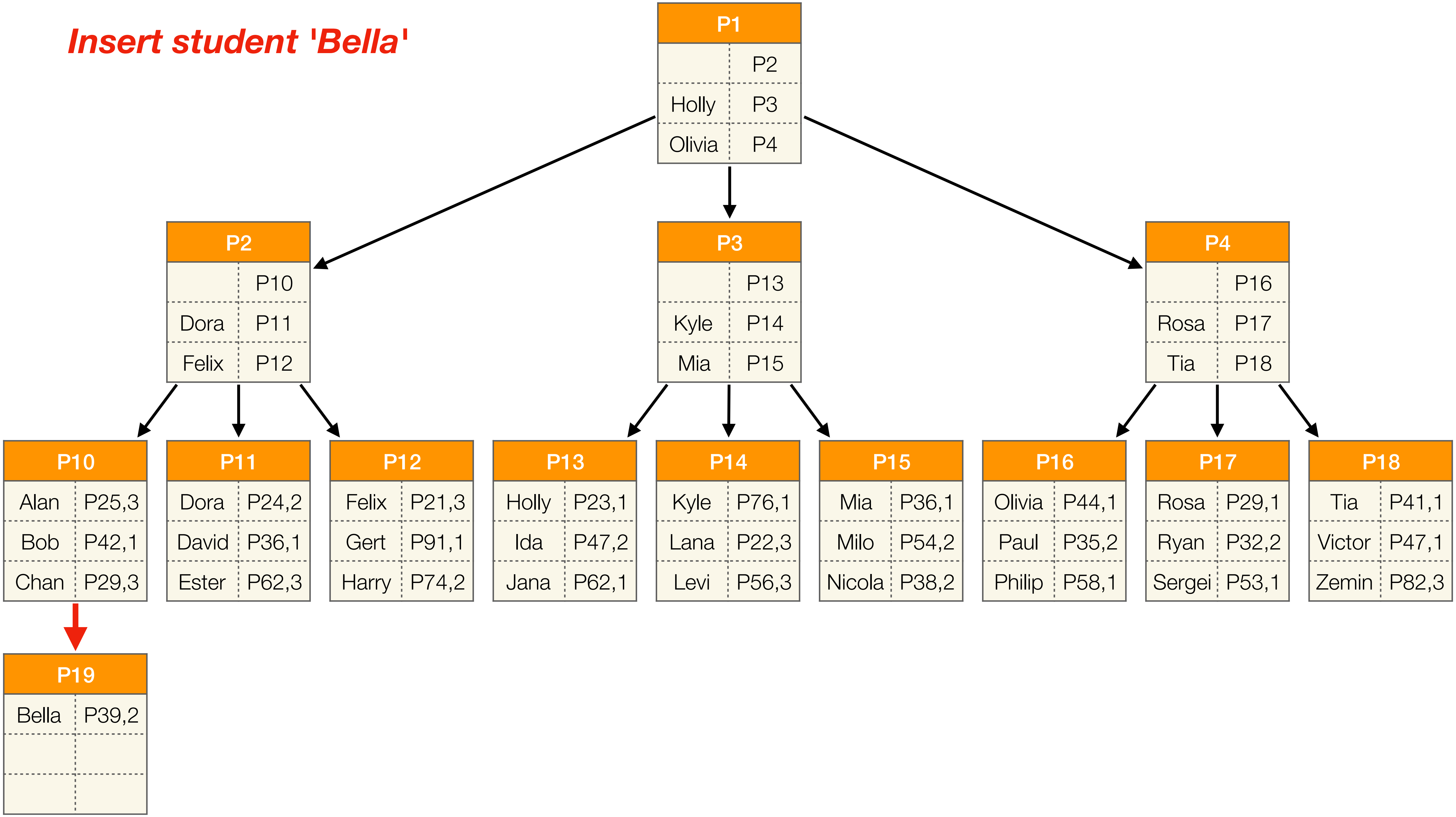
P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Slides by Immanuel Trummer, Cornell University

Updates without Balancing

Insert student 'Bella'



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

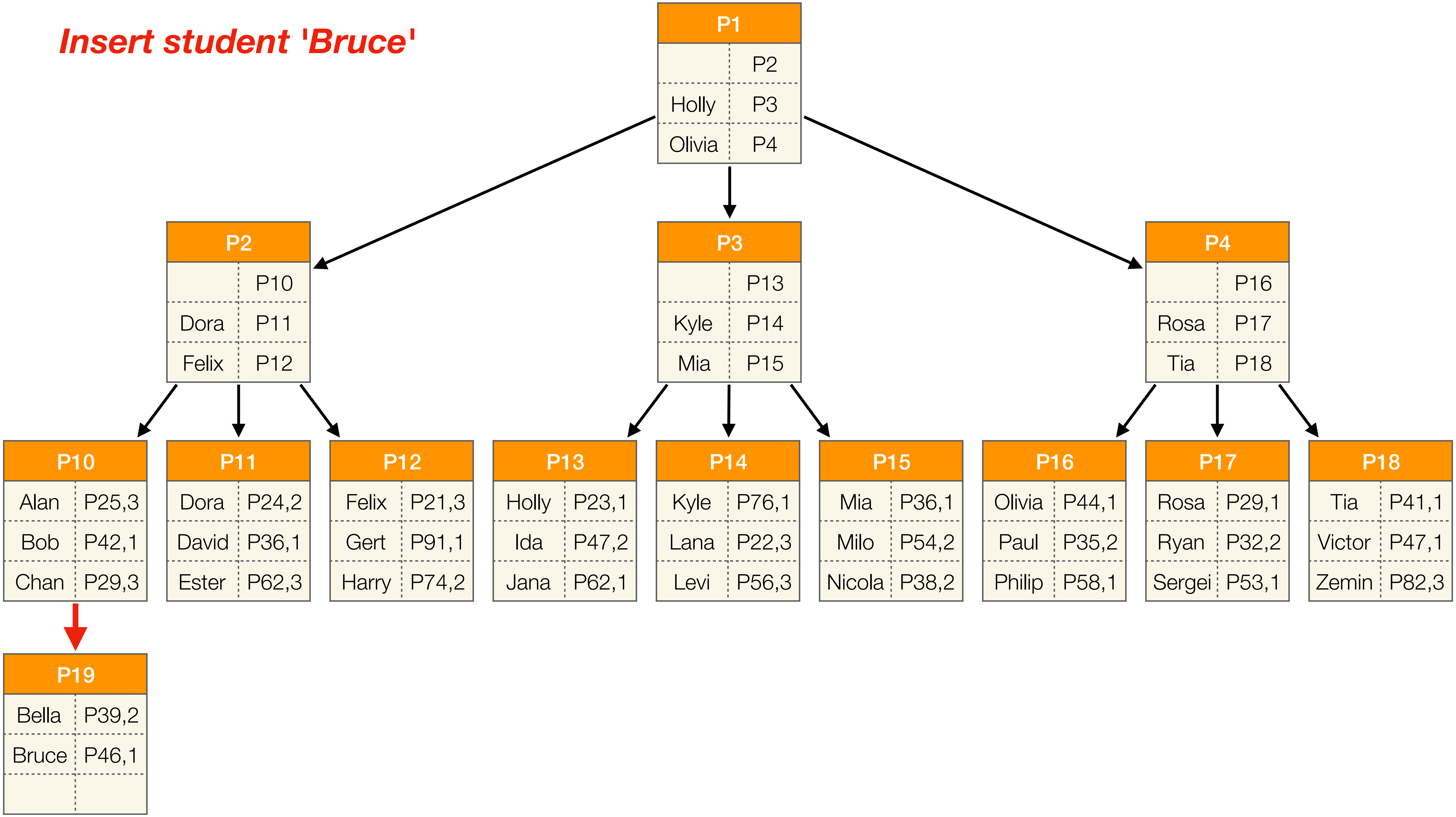
P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

P19	
Bella	P39,2

Slides by Immanuel Trummer, Cornell University

Updates without Balancing

Insert student 'Bruce'



P1

	P2
Holly	P3
Olivia	P4

P2

	P10
Dora	P11
Felix	P12

P3

	P13
Kyle	P14
Mia	P15

P4

	P16
Rosa	P17
Tia	P18

P10

Alan	P25,3
Bob	P42,1
Chan	P29,3

P11

Dora	P24,2
David	P36,1
Ester	P62,3

P12

Felix	P21,3
Gert	P91,1
Harry	P74,2

P13

Holly	P23,1
Ida	P47,2
Jana	P62,1

P14

Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15

Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16

Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17

Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18

Tia	P41,1
Victor	P47,1
Zemin	P82,3

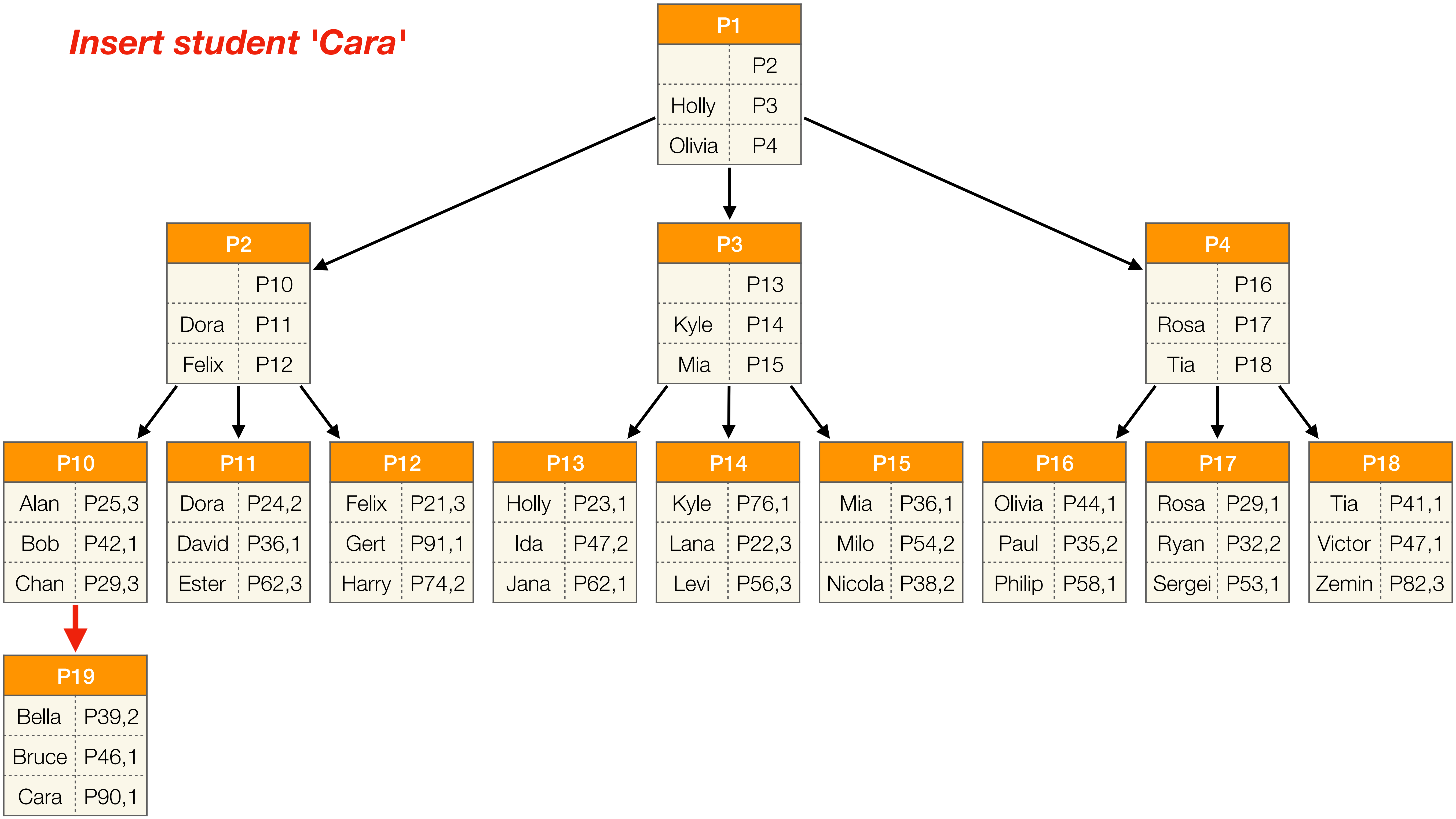
P19

Bella	P39,2
Bruce	P46,1

Slides by Immanuel Trummer, Cornell University

Updates without Balancing

Insert student 'Cara'



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

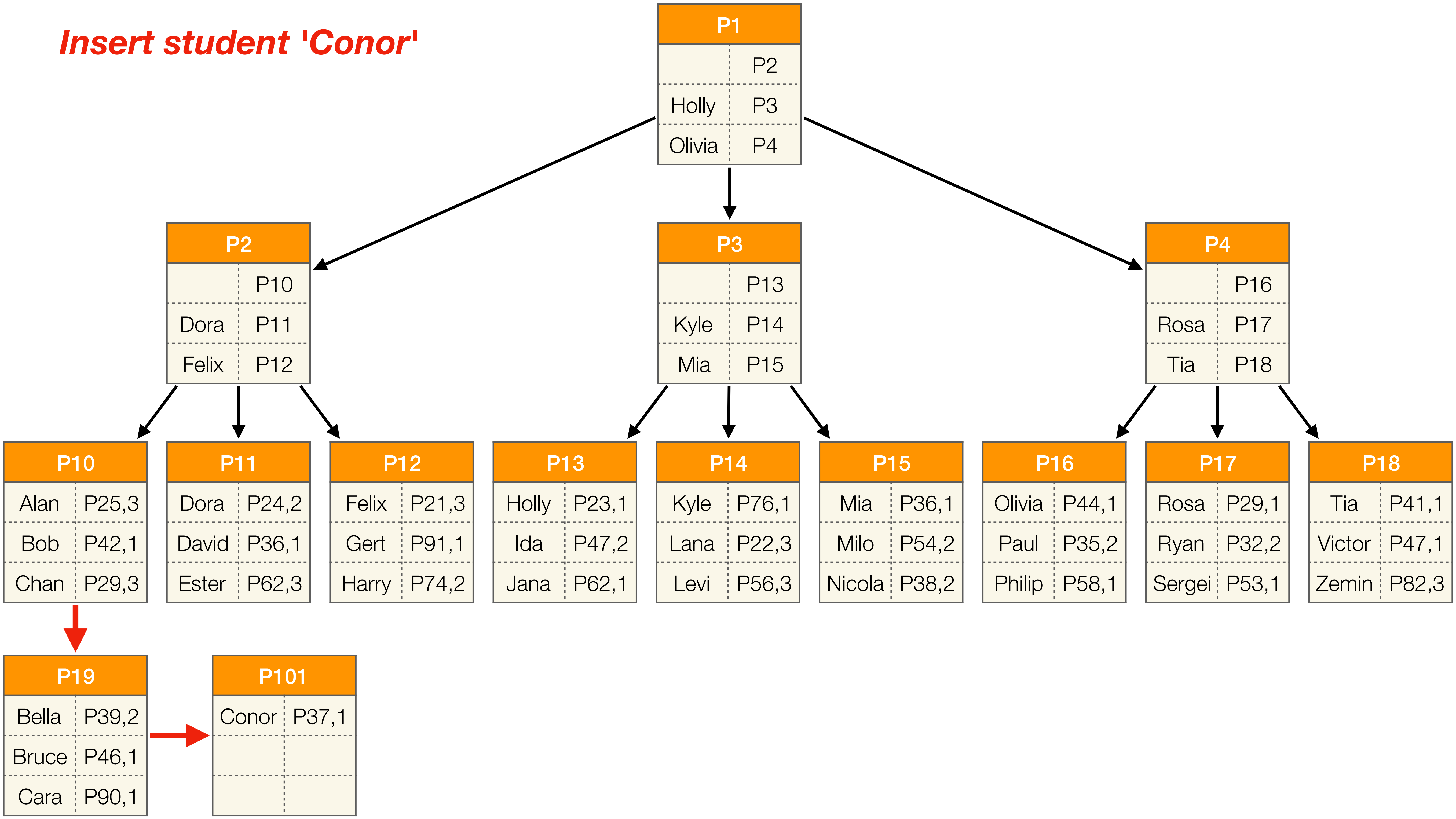
P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

P19	
Bella	P39,2
Bruce	P46,1
Cara	P90,1

Slides by Immanuel Trummer, Cornell University

Updates without Balancing

Insert student 'Conor'



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

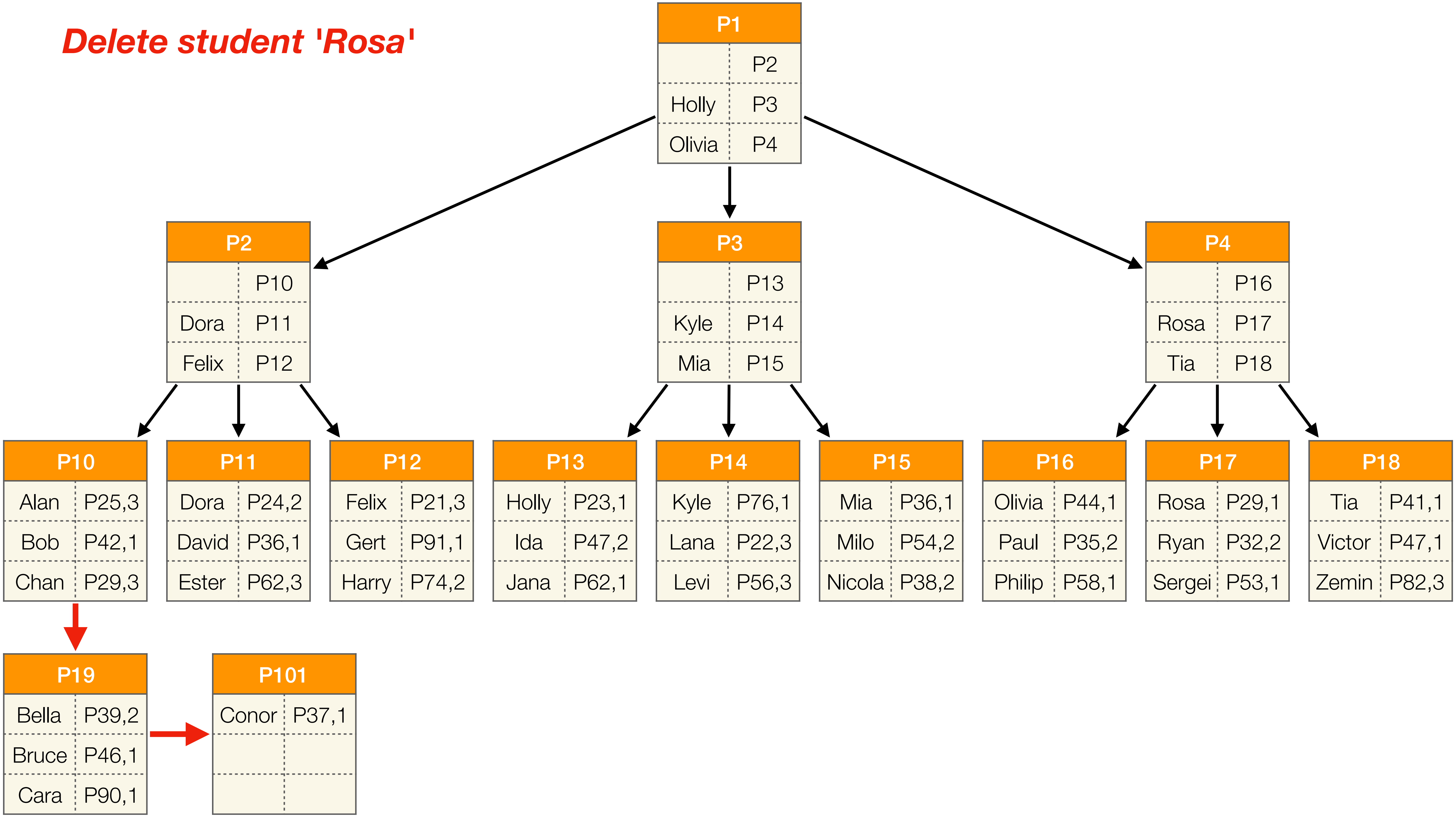
P19	
Bella	P39,2
Bruce	P46,1
Cara	P90,1

P101	
Conor	P37,1

Slides by Immanuel Trummer, Cornell University

Updates without Balancing

Delete student 'Rosa'



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

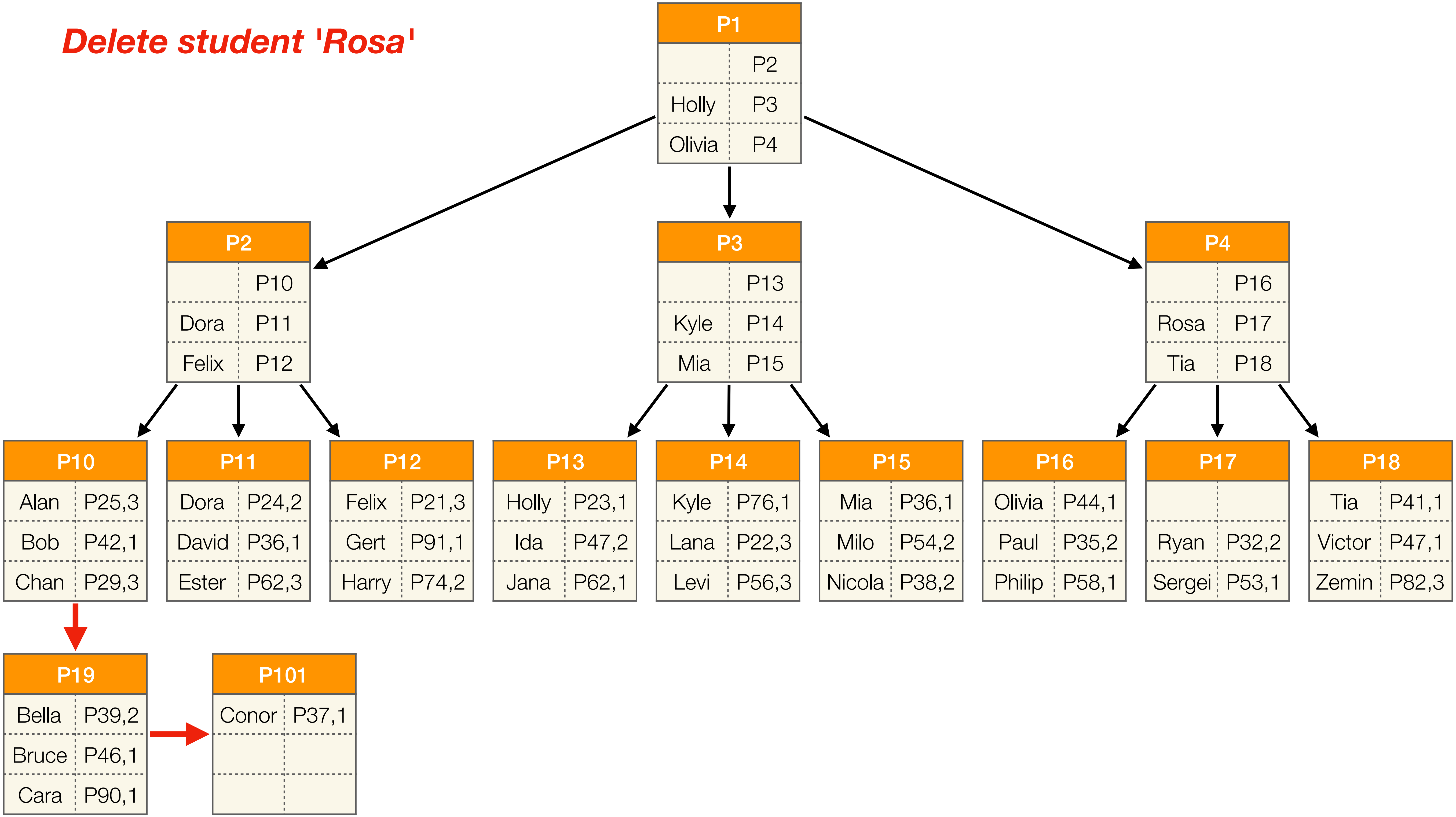
P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

P19	
Bella	P39,2
Bruce	P46,1
Cara	P90,1

P101	
Conor	P37,1

Updates without Balancing

Delete student 'Rosa'



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

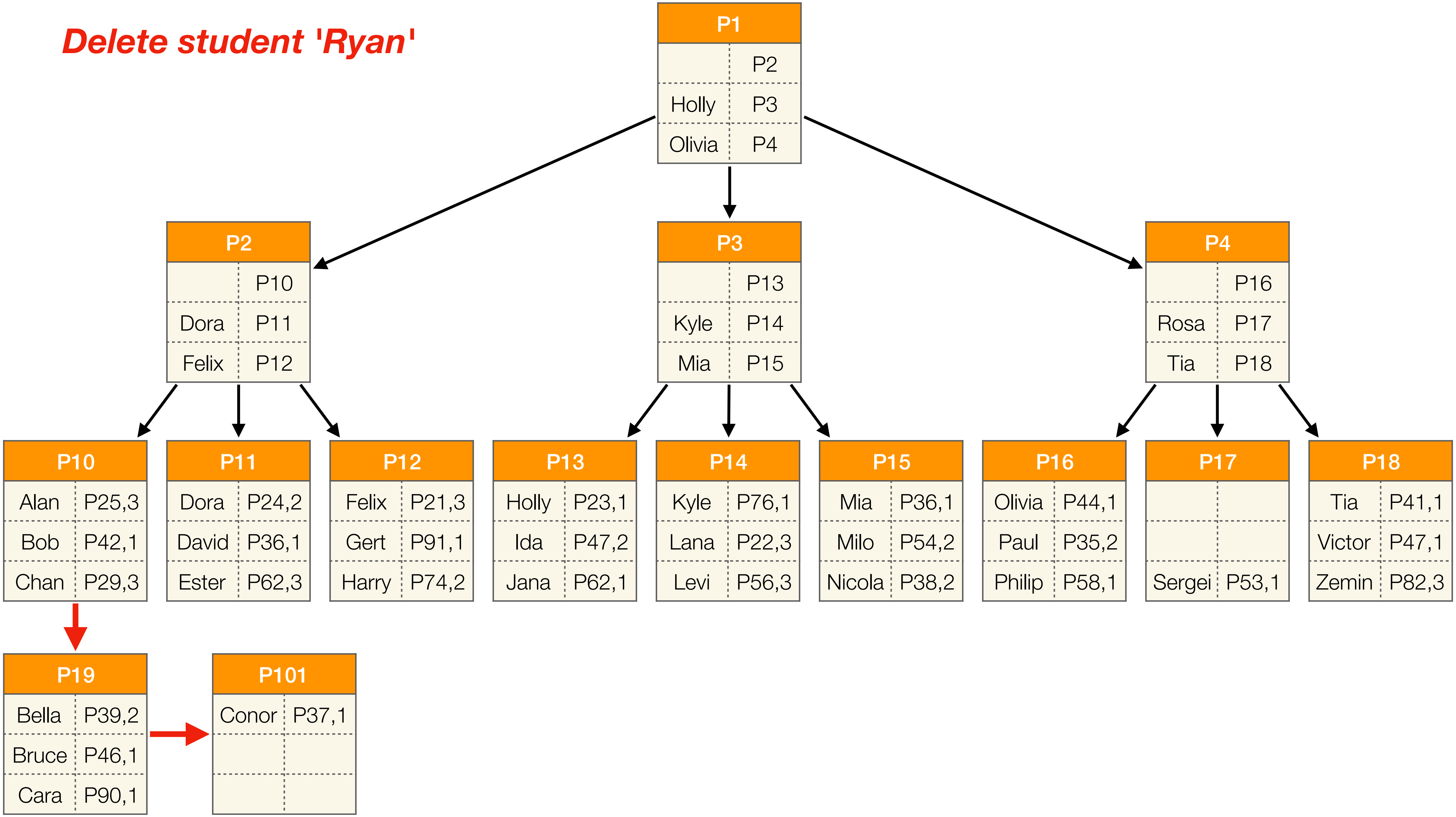
P19	
Bella	P39,2
Bruce	P46,1
Cara	P90,1

P101	
Conor	P37,1

Slides by Immanuel Trummer, Cornell University

Updates without Balancing

Delete student 'Ryan'



P1

	P2
Holly	P3
Olivia	P4

P2

	P10
Dora	P11
Felix	P12

P3

	P13
Kyle	P14
Mia	P15

P4

	P16
Rosa	P17
Tia	P18

P10

Alan	P25,3
Bob	P42,1
Chan	P29,3

P11

Dora	P24,2
David	P36,1
Ester	P62,3

P12

Felix	P21,3
Gert	P91,1
Harry	P74,2

P13

Holly	P23,1
Ida	P47,2
Jana	P62,1

P14

Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15

Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16

Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17

Sergei	P53,1

P18

Tia	P41,1
Victor	P47,1
Zemin	P82,3

P19

Bella	P39,2
Bruce	P46,1
Cara	P90,1

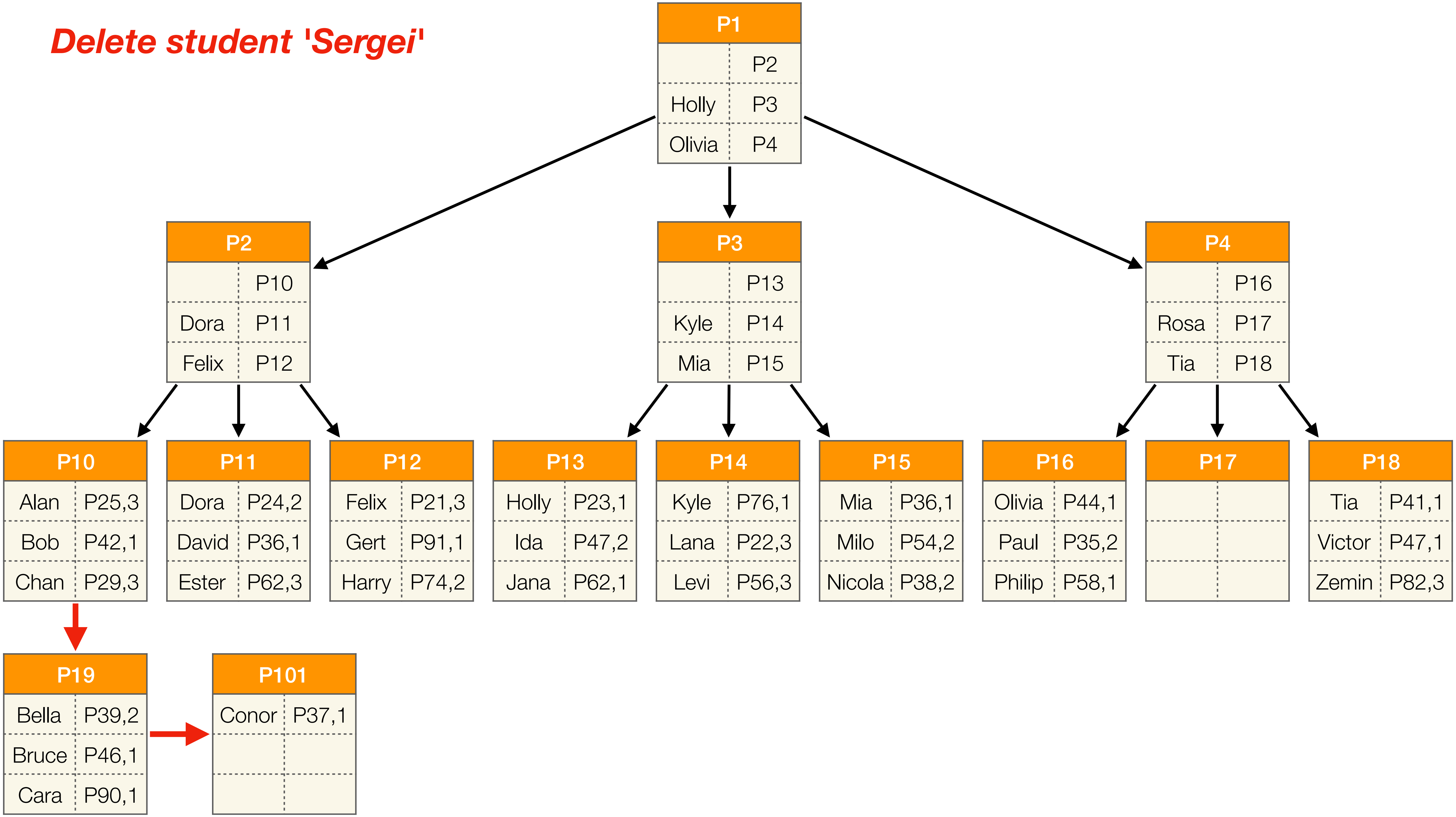
P101

Conor	P37,1

Slides by Immanuel Trummer, Cornell University

Updates without Balancing

Delete student 'Sergei'



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

P19	
Bella	P39,2
Bruce	P46,1
Cara	P90,1

P101	
Conor	P37,1

Slides by Immanuel Trummer, Cornell University

Problems

- Aforementioned approach used e.g. by **ISAM index**
- Ok for static data but **problematic if dynamic**
  - Lots of **overflow pages** reduce performance
  - **Empty pages** lead to space overheads

B+ Trees

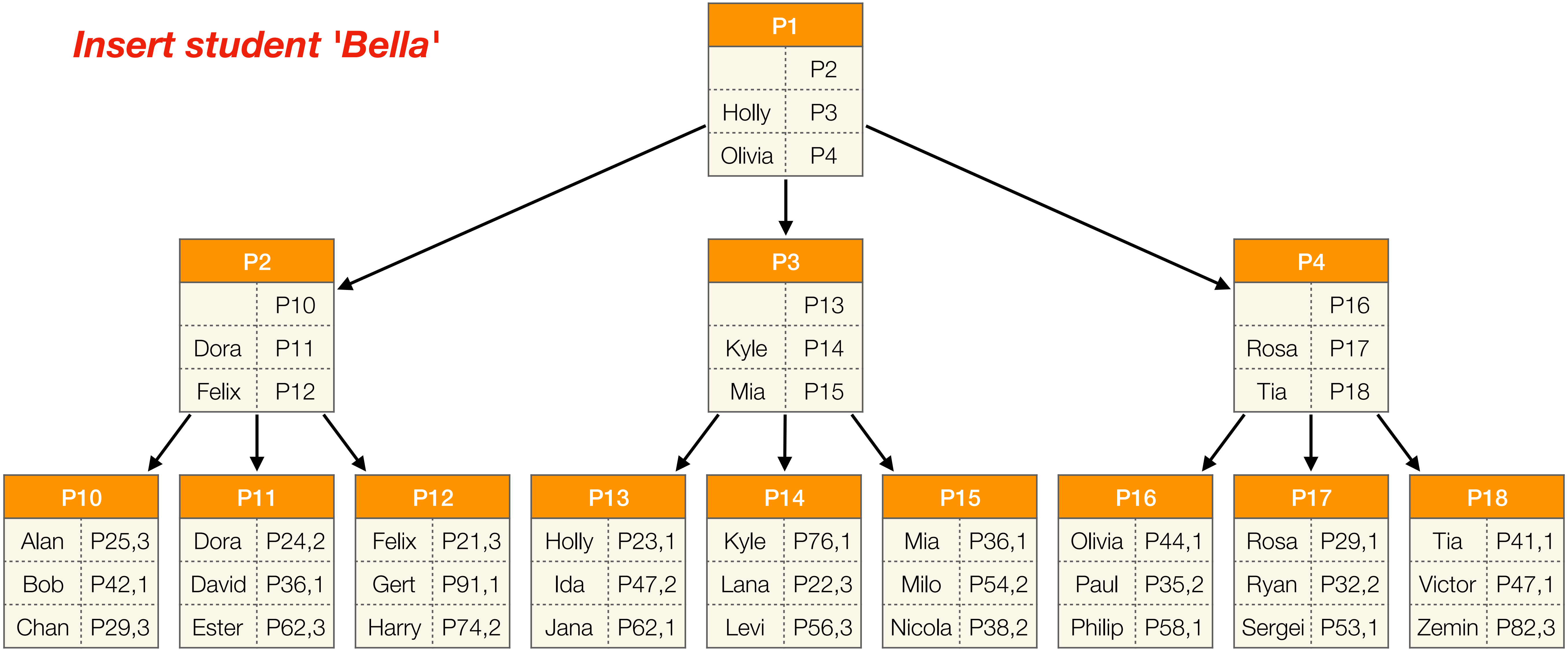
- One of the **most popular** index structure
  - E.g., the **default** index in Postgres
- **Balances** tree after insert/delete operations
- Keeps the tree **compact**
  - Each node (except root) is **at least half full**!
  - I.e., number entries between **d** and **2*d** (d is "**order**")

B+ Trees Are Shallow

- Typical order is 100, typical fill factor 67%
- → Average **fanout** (i.e., number of child nodes) is 133
- → Second level can have 133^2 = **17,689 nodes**
- → Third level can have 133^3 = **2,352,637 nodes**
- → Fourth level can have 133^4 = **312,900,721 nodes**
- ...

Updates with Balancing

Insert student 'Bella'



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

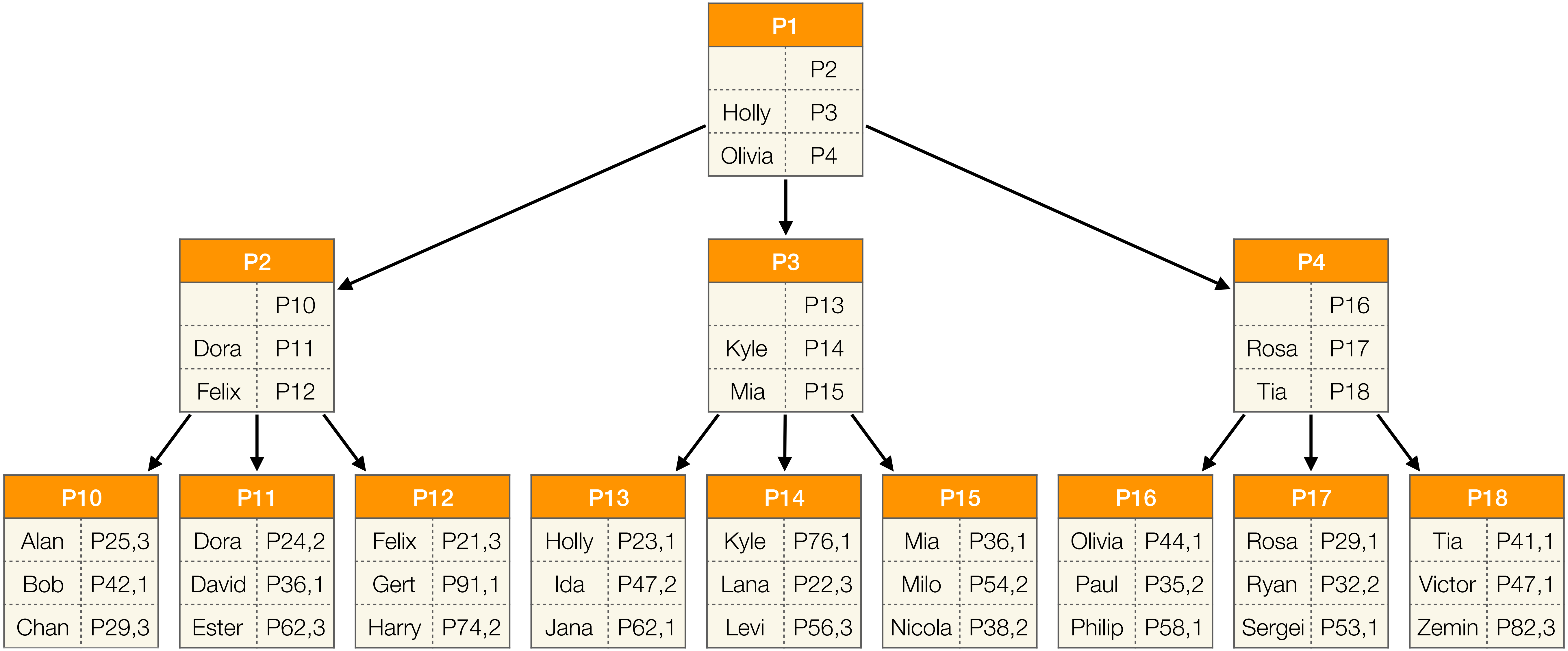
P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Slides by Immanuel Trummer, Cornell University

Updates with Balancing



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

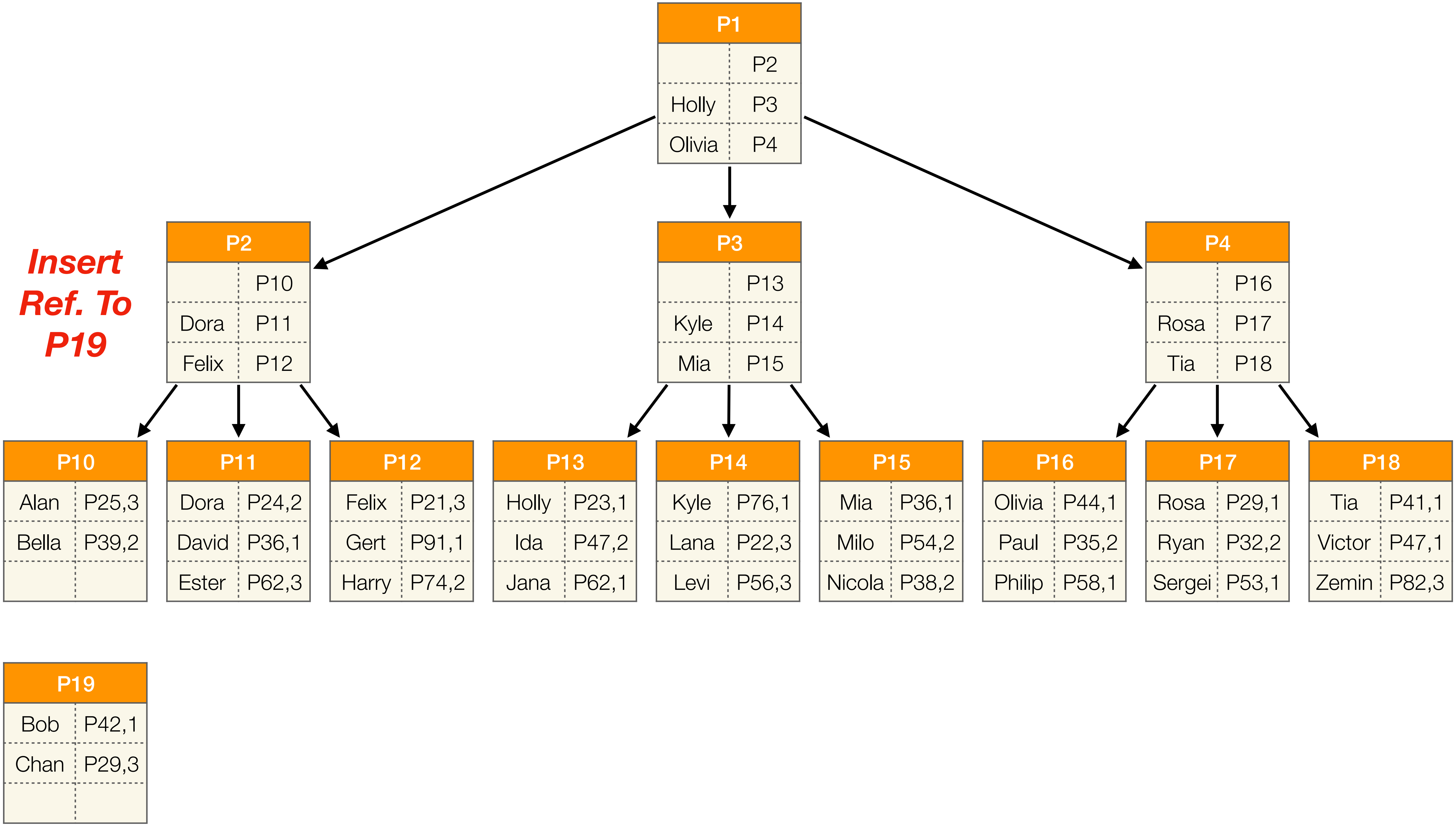
P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Insert student 'Bella'

Slides by Immanuel Trummer, Cornell University

Updates with Balancing



Insert Ref. To P19

P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bella	P39,2

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

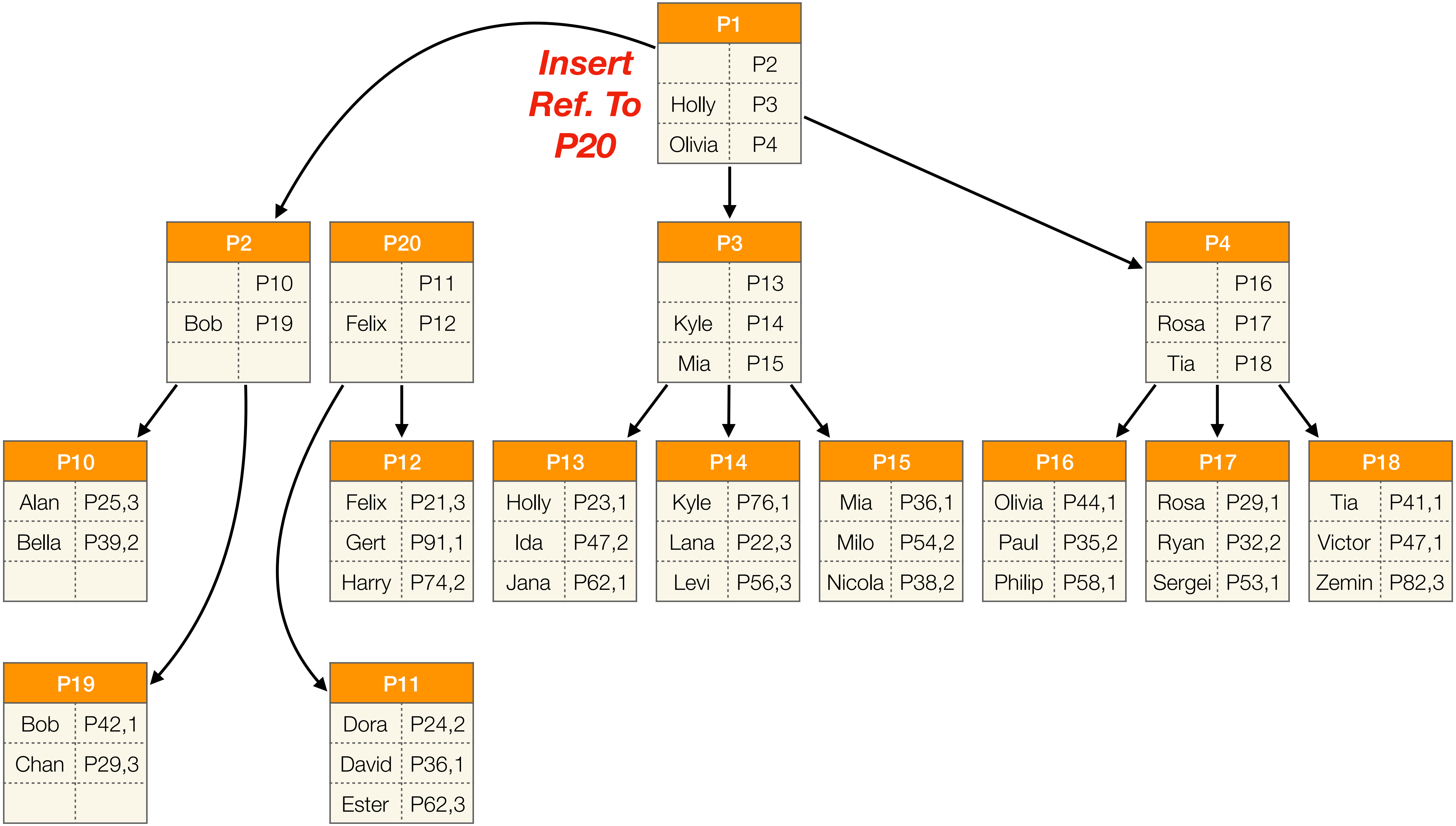
P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

P19	
Bob	P42,1
Chan	P29,3

Slides by Immanuel Trummer, Cornell University

Updates with Balancing



Insert Ref. To P20

P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Bob	P19

P20	
	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bella	P39,2

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

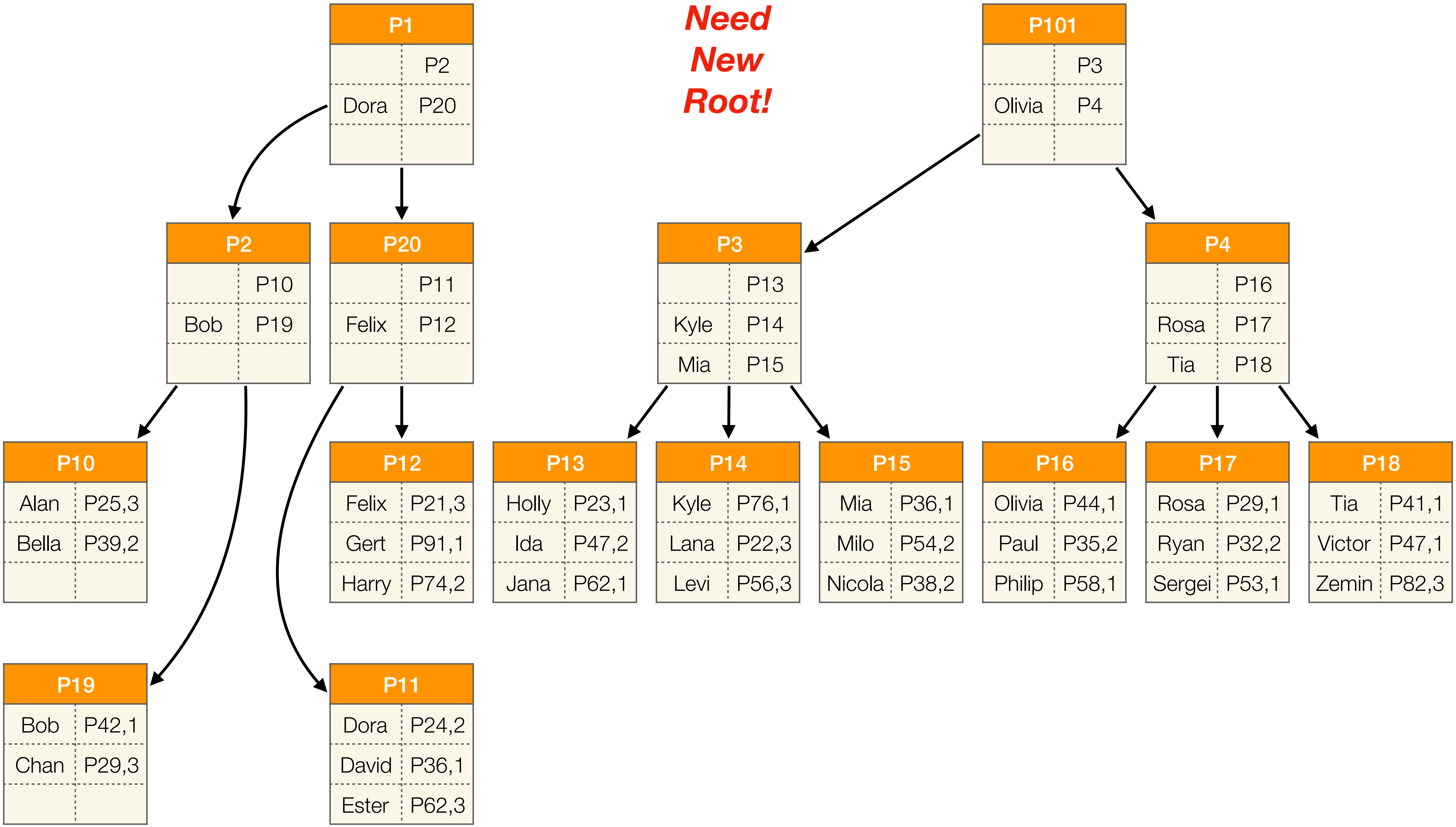
P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

P19	
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

Slides by Immanuel Trummer, Cornell University

Updates with Balancing



Need New Root!

P1	
	P2
Dora	P20

P101	
	P3
Olivia	P4

P2	
	P10
Bob	P19

P20	
	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bella	P39,2

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

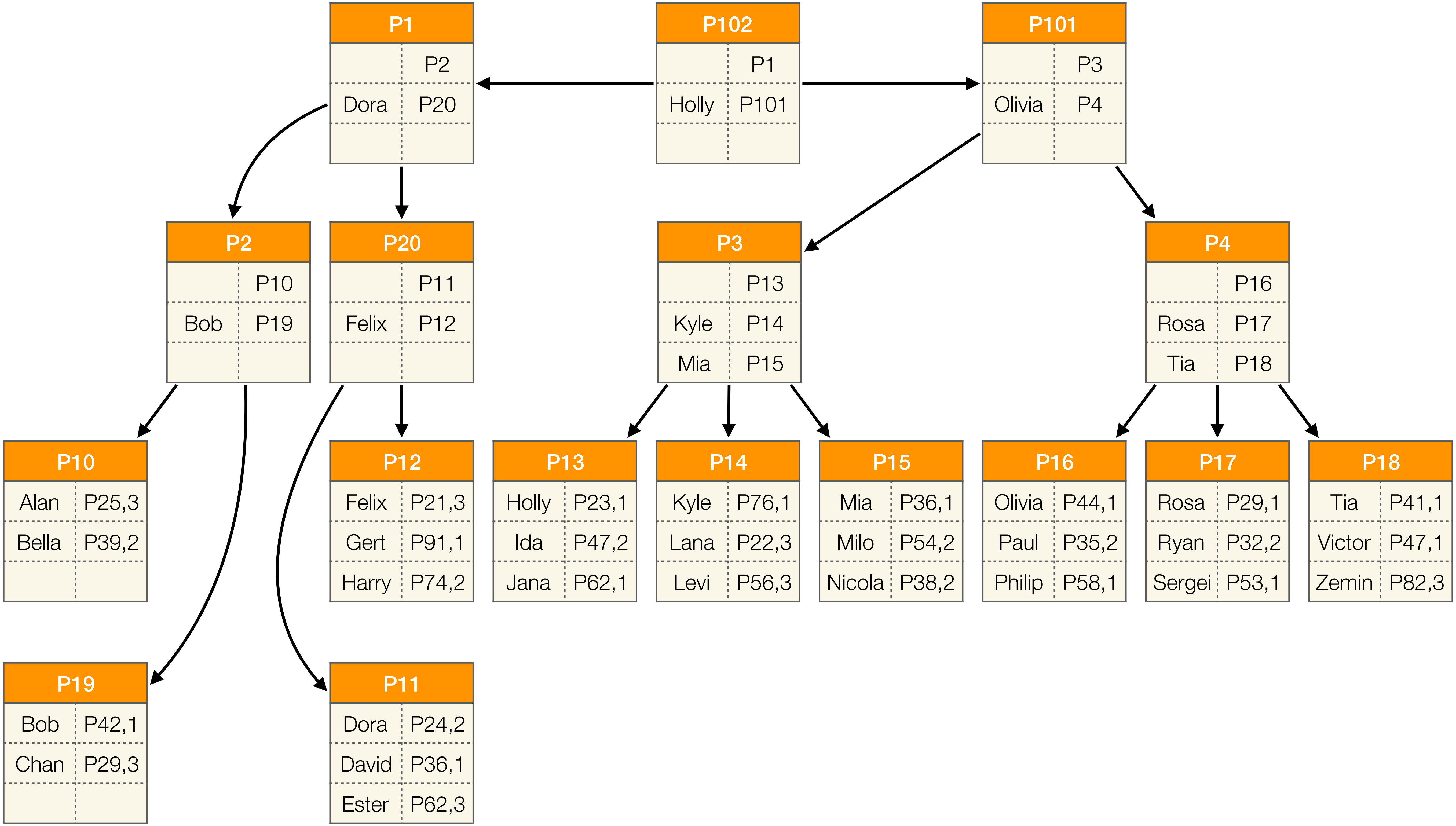
P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

P19	
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

Slides by Immanuel Trummer, Cornell University

Updates with Balancing



P1	
	P2
Dora	P20

P102	
	P1
Holly	P101

P101	
	P3
Olivia	P4

P2	
	P10
Bob	P19

P20	
	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bella	P39,2

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

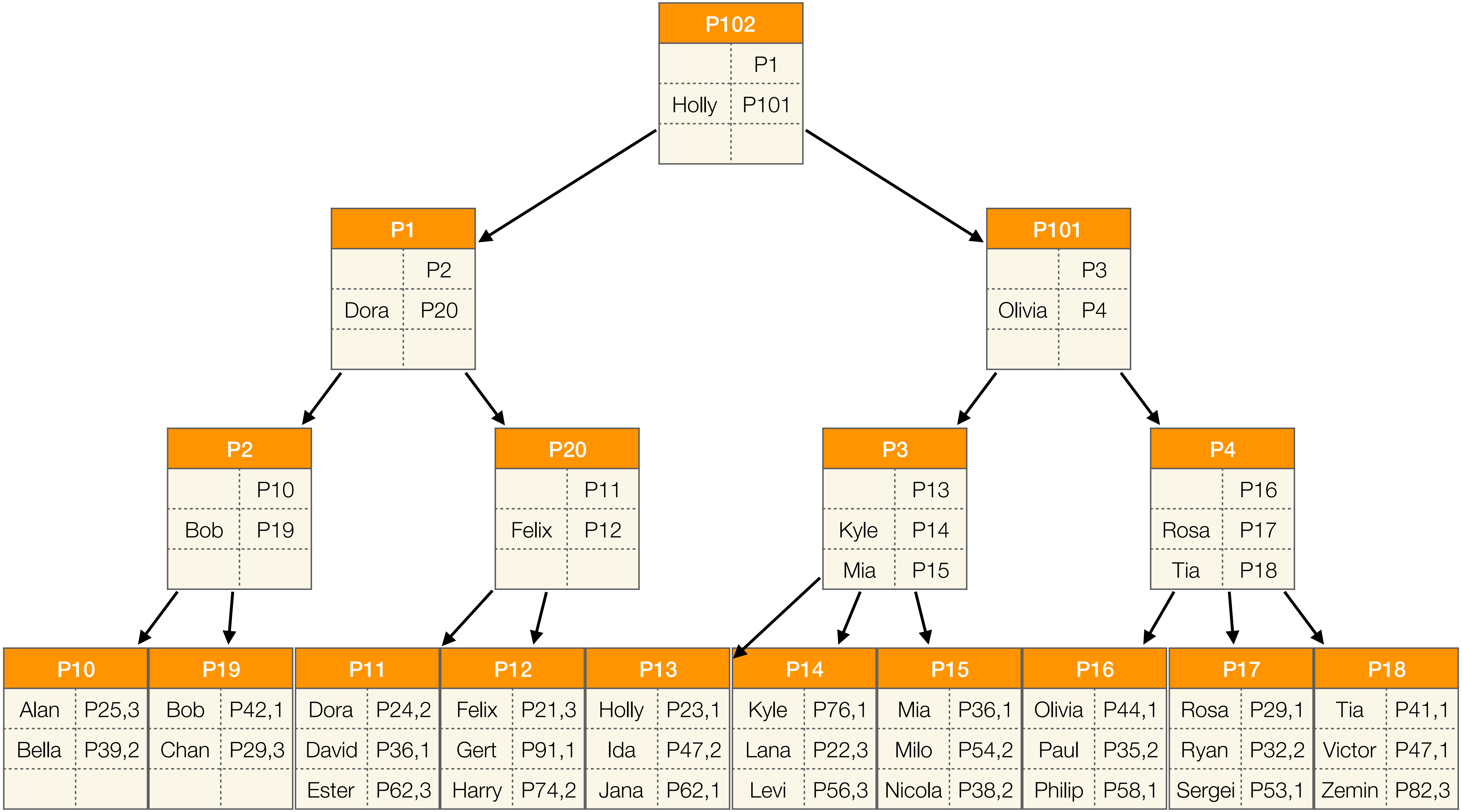
P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

P19	
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

Slides by Immanuel Trummer, Cornell University

Updates with Balancing



P102	
	P1
Holly	P101

P1	
	P2
Dora	P20

P101	
	P3
Olivia	P4

P2	
	P10
Bob	P19

P20	
	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bella	P39,2

P19	
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Ryan	P32,2
Sergei	P53,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

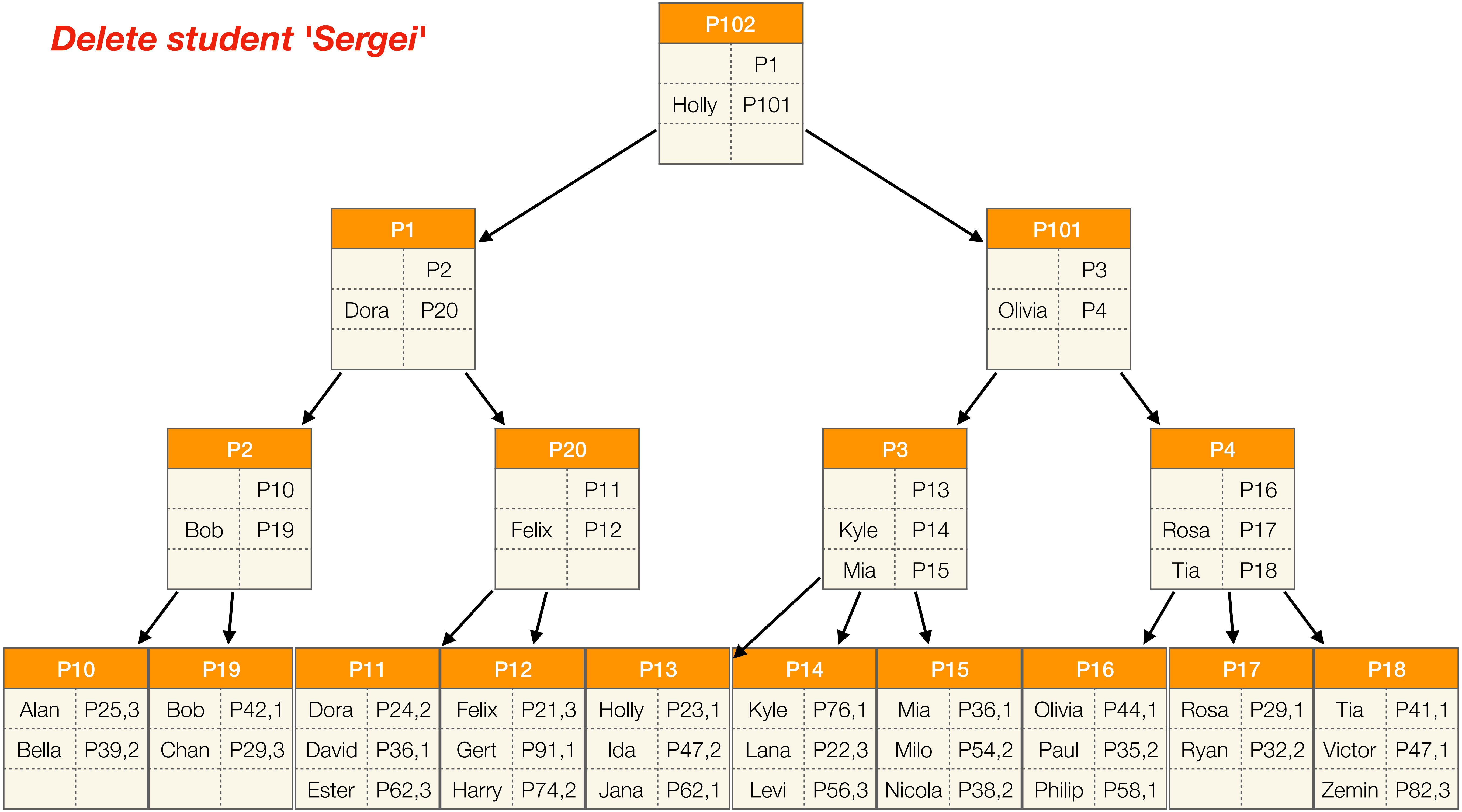
Slides by Immanuel Trummer, Cornell University

Remark on Example

- Typically, expect **even** number of maximal entries
  - I.e., maximal number of entries is **2 * [order]**
  - Nodes are "underfull" with less than **[order]** entries
- Have **up to three** entries per node in our example
- Will consider nodes with one entry as **"underfull"**

Deletes with Balancing

Delete student 'Sergei'



P102

	P1
Holly	P101

P1

	P2
Dora	P20

P101

	P3
Olivia	P4

P2

	P10
Bob	P19

P20

	P11
Felix	P12

P3

	P13
Kyle	P14
Mia	P15

P4

	P16
Rosa	P17
Tia	P18

P10

Alan	P25,3
Bella	P39,2

P19

Bob	P42,1
Chan	P29,3

P11

Dora	P24,2
David	P36,1
Ester	P62,3

P12

Felix	P21,3
Gert	P91,1
Harry	P74,2

P13

Holly	P23,1
Ida	P47,2
Jana	P62,1

P14

Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15

Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16

Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17

Rosa	P29,1
Ryan	P32,2

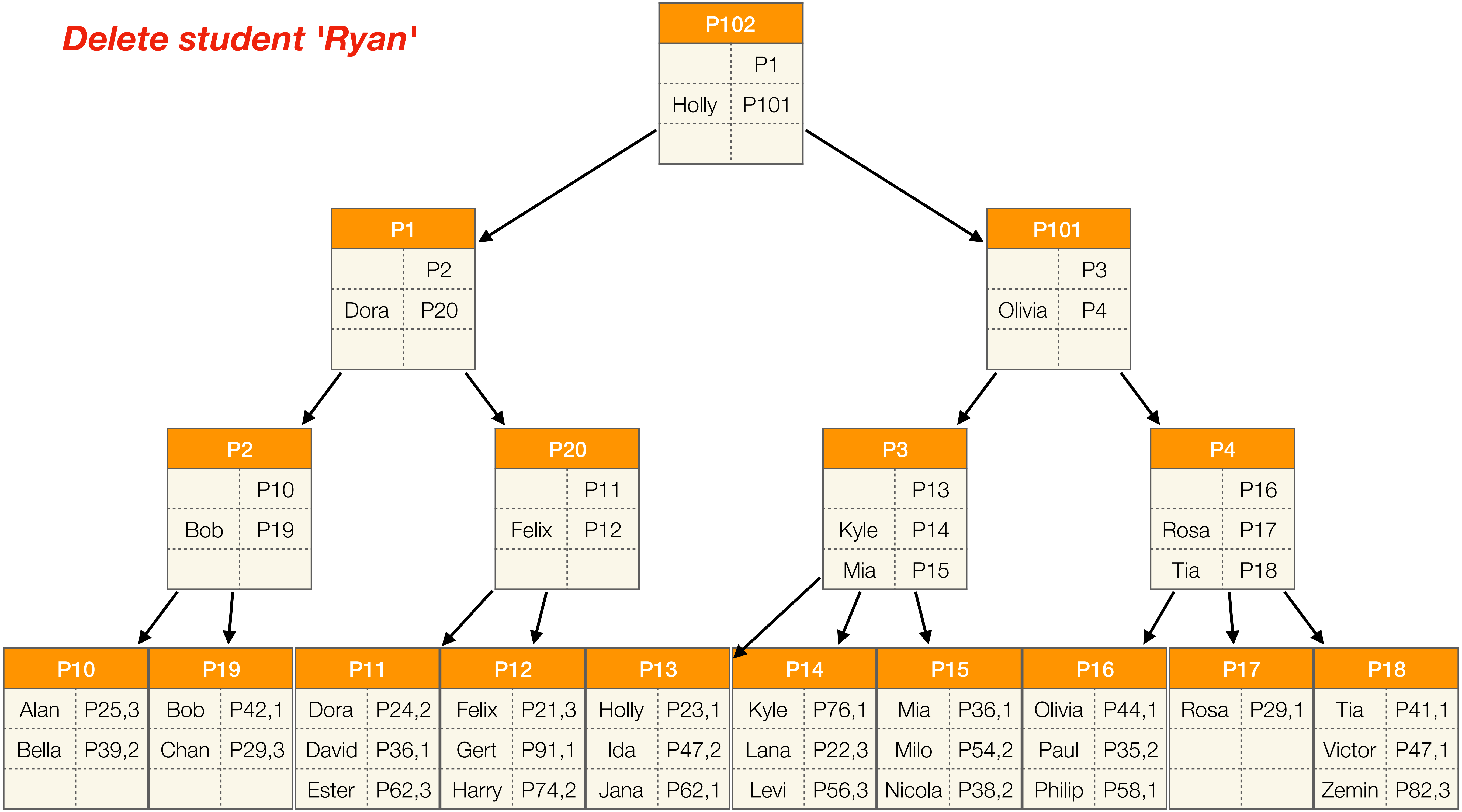
P18

Tia	P41,1
Victor	P47,1
Zemin	P82,3

Slides by Immanuel Trummer, Cornell University

Deletes with Balancing

Delete student 'Ryan'



P102
P1
Holly P101

P1
P2
Dora P20

P101
P3
Olivia P4

P2
P10
Bob P19

P20
P11
Felix P12

P3
P13
Kyle P14
Mia P15

P4
P16
Rosa P17
Tia P18

P10
Alan P25,3
Bella P39,2

P19
Bob P42,1
Chan P29,3

P11
Dora P24,2
David P36,1
Ester P62,3

P12
Felix P21,3
Gert P91,1
Harry P74,2

P13
Holly P23,1
Ida P47,2
Jana P62,1

P14
Kyle P76,1
Lana P22,3
Levi P56,3

P15
Mia P36,1
Milo P54,2
Nicola P38,2

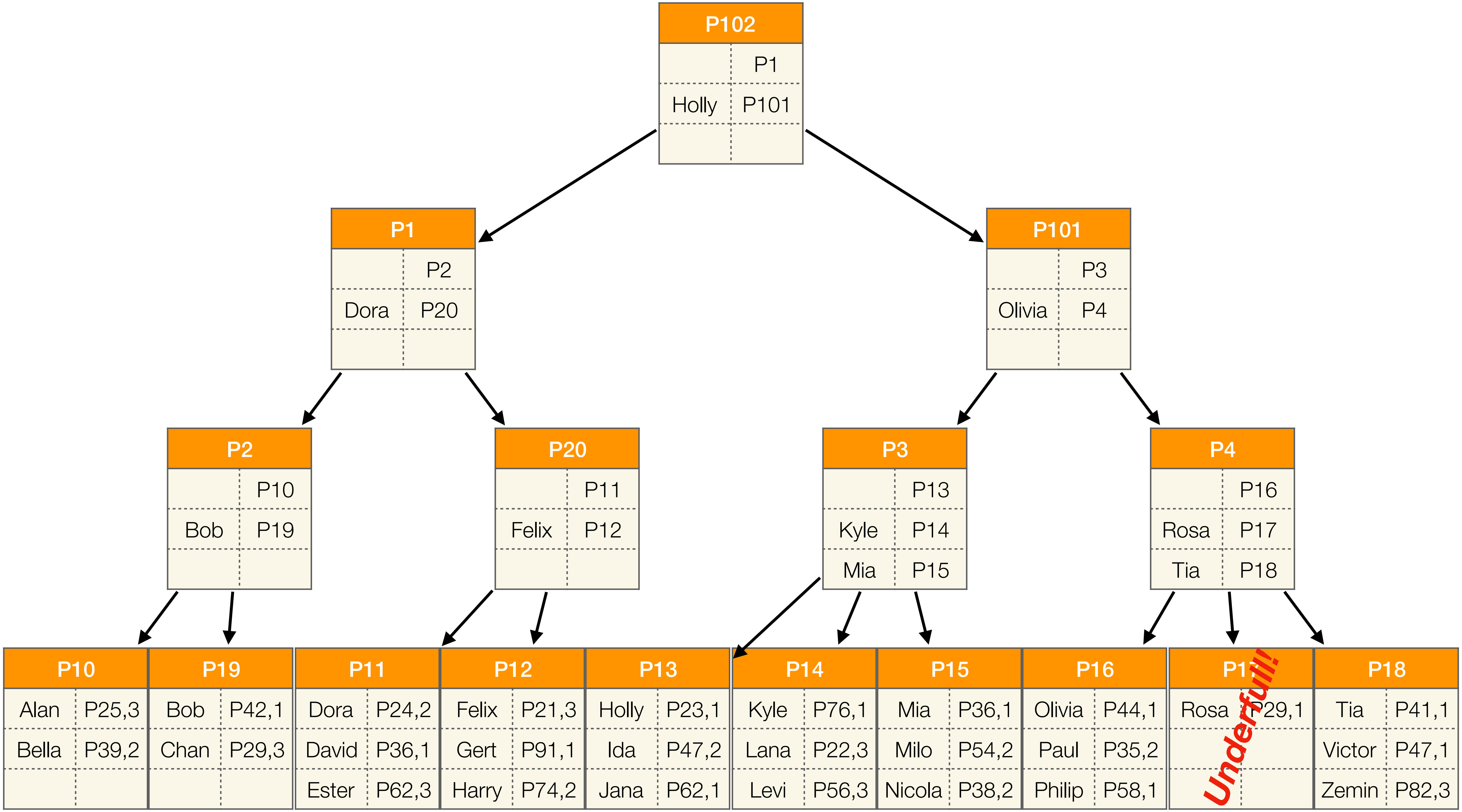
P16
Olivia P44,1
Paul P35,2
Philip P58,1

P17
Rosa P29,1

P18
Tia P41,1
Victor P47,1
Zemin P82,3

Slides by Immanuel Trummer, Cornell University

Deletes with Balancing



P102	
	P1
Holly	P101

P1	
	P2
Dora	P20

P101	
	P3
Olivia	P4

P2	
	P10
Bob	P19

P20	
	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Tia	P18

P10	
Alan	P25,3
Bella	P39,2

P19	
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

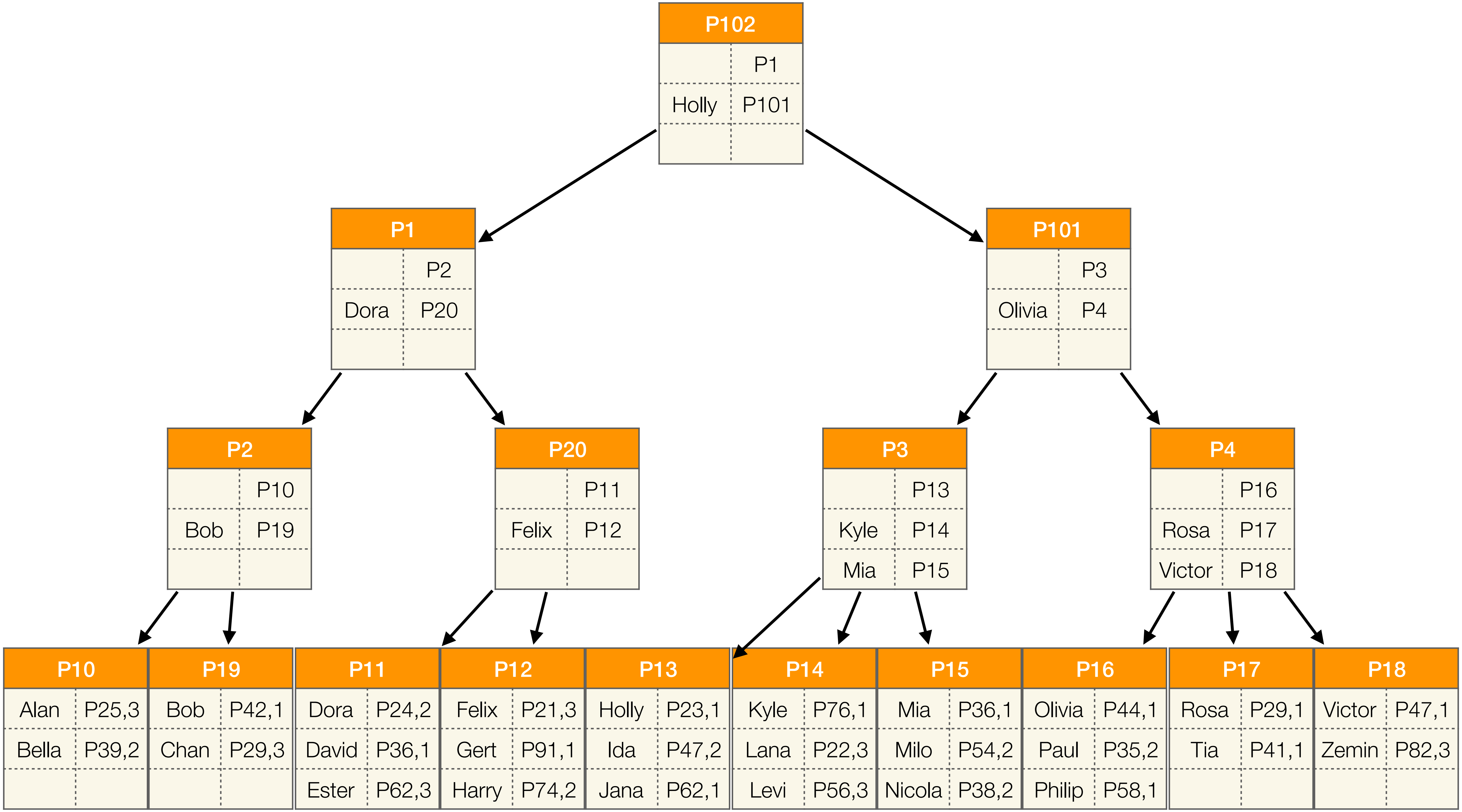
P17	
Rosa	P29,1

P18	
Tia	P41,1
Victor	P47,1
Zemin	P82,3

Underfull!

Slides by Immanuel Trummer, Cornell University

Deletes with Balancing



P102	
	P1
Holly	P101

P1	
	P2
Dora	P20

P101	
	P3
Olivia	P4

P2	
	P10
Bob	P19

P20	
	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Victor	P18

P10	
Alan	P25,3
Bella	P39,2

P19	
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Tia	P41,1

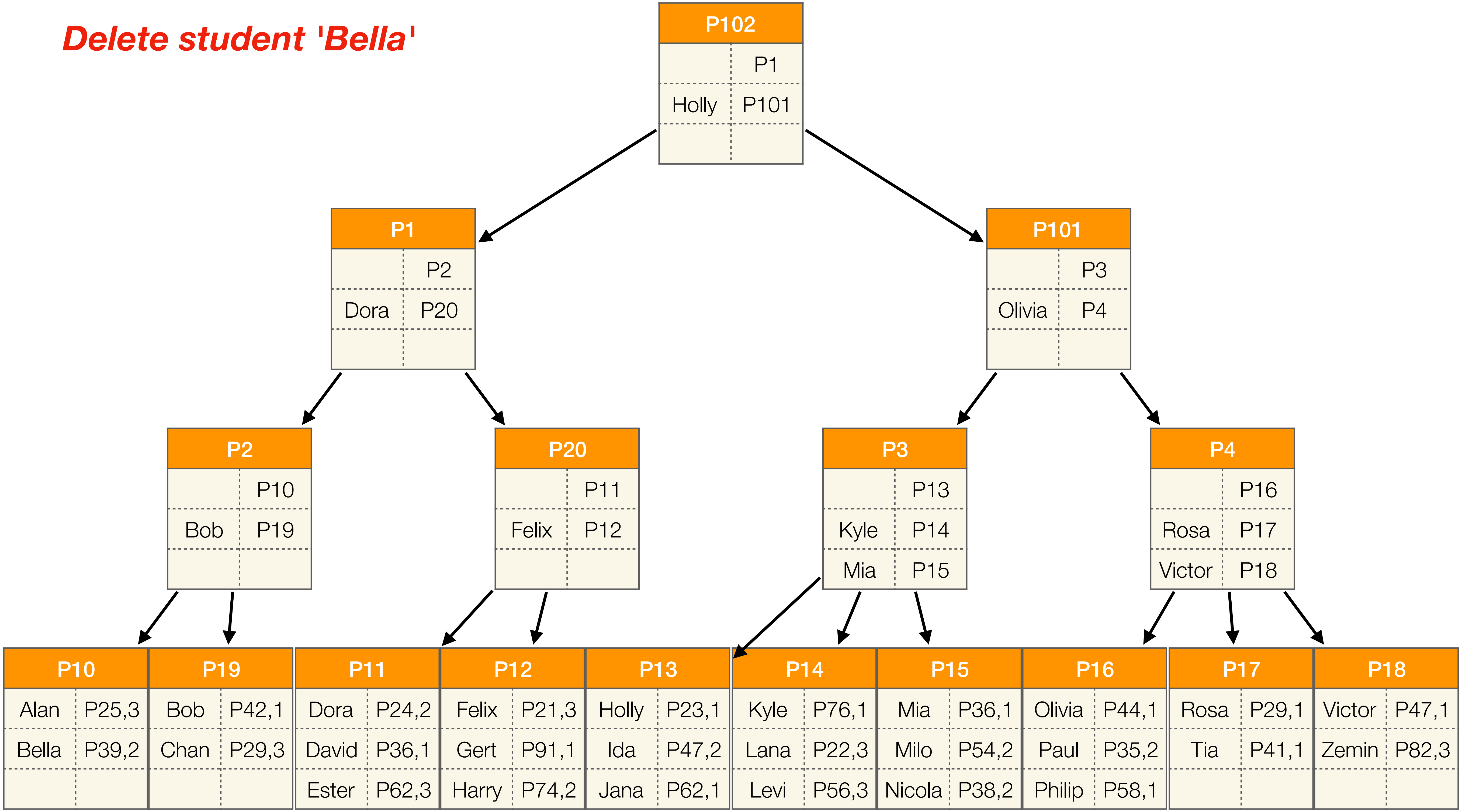
P18	
Victor	P47,1
Zemin	P82,3

More Options for Balancing

- After deletion, need to **fix nodes** that are underfull
- Here: have **redistributed** entries from sibling leafs
- Otherwise: may have to **merge** tree nodes together
- Merge operations may **propagate upwards** in tree
  - E.g., imagine student "Bella" is **deleted** again
  - Tree loses one level (**inverse** to insertion operation)

Deletes with Balancing

Delete student 'Bella'



P102
P1
Holly P101

P1
P2
Dora P20

P101
P3
Olivia P4

P2
P10
Bob P19

P20
P11
Felix P12

P3
P13
Kyle P14
Mia P15

P4
P16
Rosa P17
Victor P18

P10
Alan P25,3
Bella P39,2

P19
Bob P42,1
Chan P29,3

P11
Dora P24,2
David P36,1
Ester P62,3

P12
Felix P21,3
Gert P91,1
Harry P74,2

P13
Holly P23,1
Ida P47,2
Jana P62,1

P14
Kyle P76,1
Lana P22,3
Levi P56,3

P15
Mia P36,1
Milo P54,2
Nicola P38,2

P16
Olivia P44,1
Paul P35,2
Philip P58,1

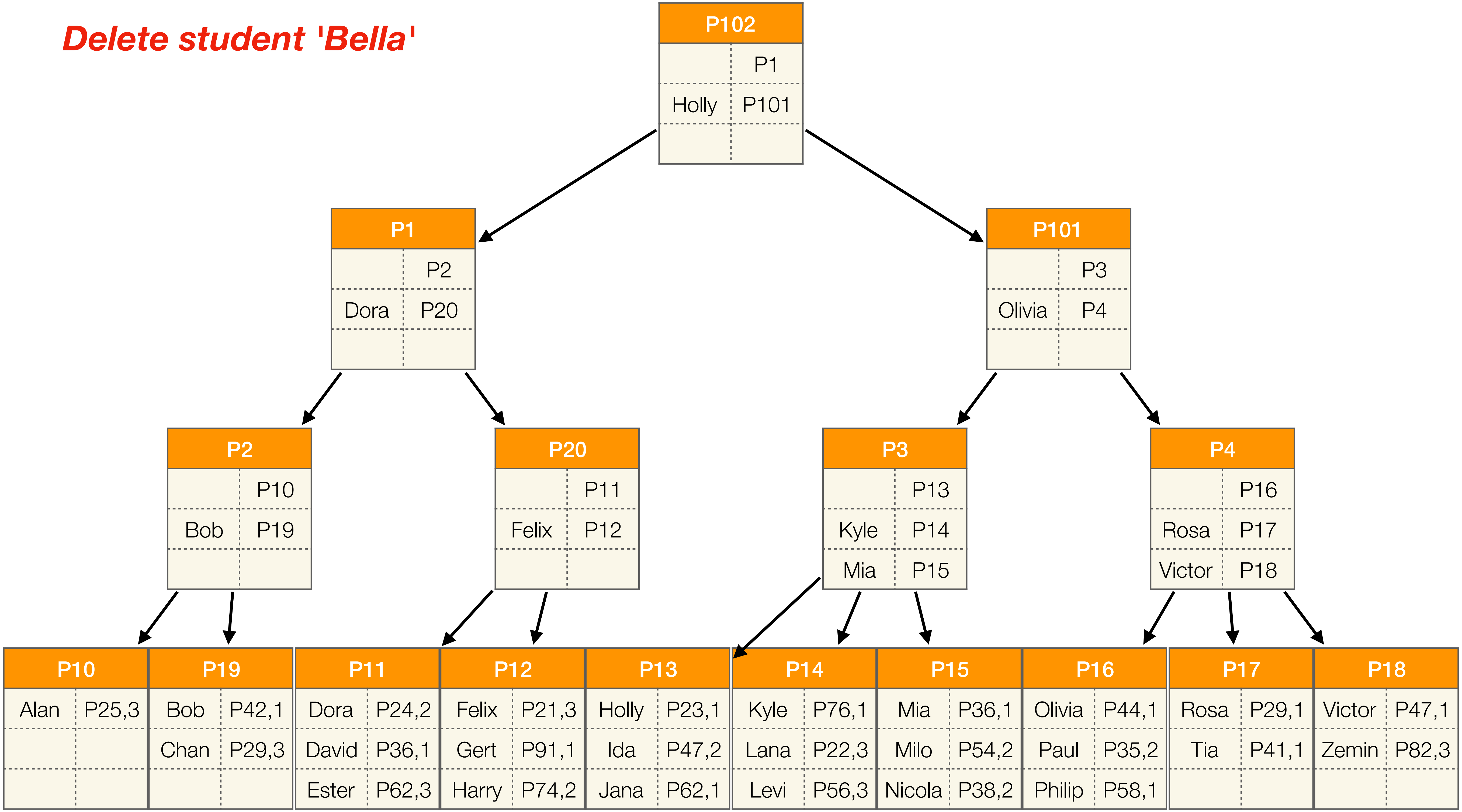
P17
Rosa P29,1
Tia P41,1

P18
Victor P47,1
Zemin P82,3

Slides by Immanuel Trummer, Cornell University

Deletes with Balancing

Delete student 'Bella'



P102
| | P1
Holly | P101

P1
| | P2
Dora | P20

P101
| | P3
Olivia | P4

P2
| | P10
Bob | P19

P20
| | P11
Felix | P12

P3
| | P13
Kyle | P14
Mia | P15

P4
| | P16
Rosa | P17
Victor | P18

P10
Alan | P25,3

P19
Bob | P42,1
Chan | P29,3

P11
Dora | P24,2
David | P36,1
Ester | P62,3

P12
Felix | P21,3
Gert | P91,1
Harry | P74,2

P13
Holly | P23,1
Ida | P47,2
Jana | P62,1

P14
Kyle | P76,1
Lana | P22,3
Levi | P56,3

P15
Mia | P36,1
Milo | P54,2
Nicola | P38,2

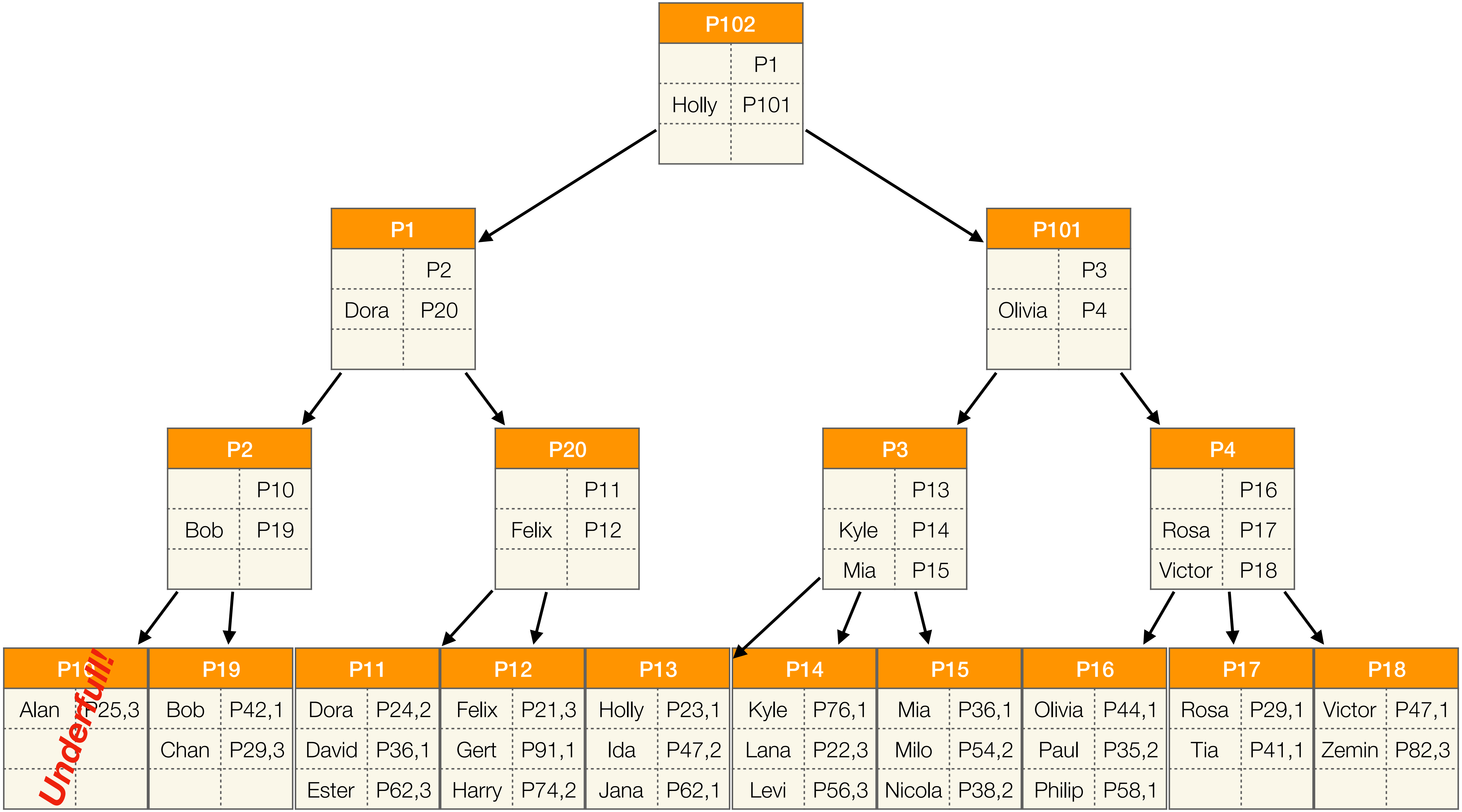
P16
Olivia | P44,1
Paul | P35,2
Philip | P58,1

P17
Rosa | P29,1
Tia | P41,1

P18
Victor | P47,1
Zemin | P82,3

Slides by Immanuel Trummer, Cornell University

Deletes with Balancing



P102	
	P1
Holly	P101

P1	
	P2
Dora	P20

P101	
	P3
Olivia	P4

P2	
	P10
Bob	P19

P20	
	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Victor	P18

P10	
Alan	P25,3

Underfull!

P19	
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

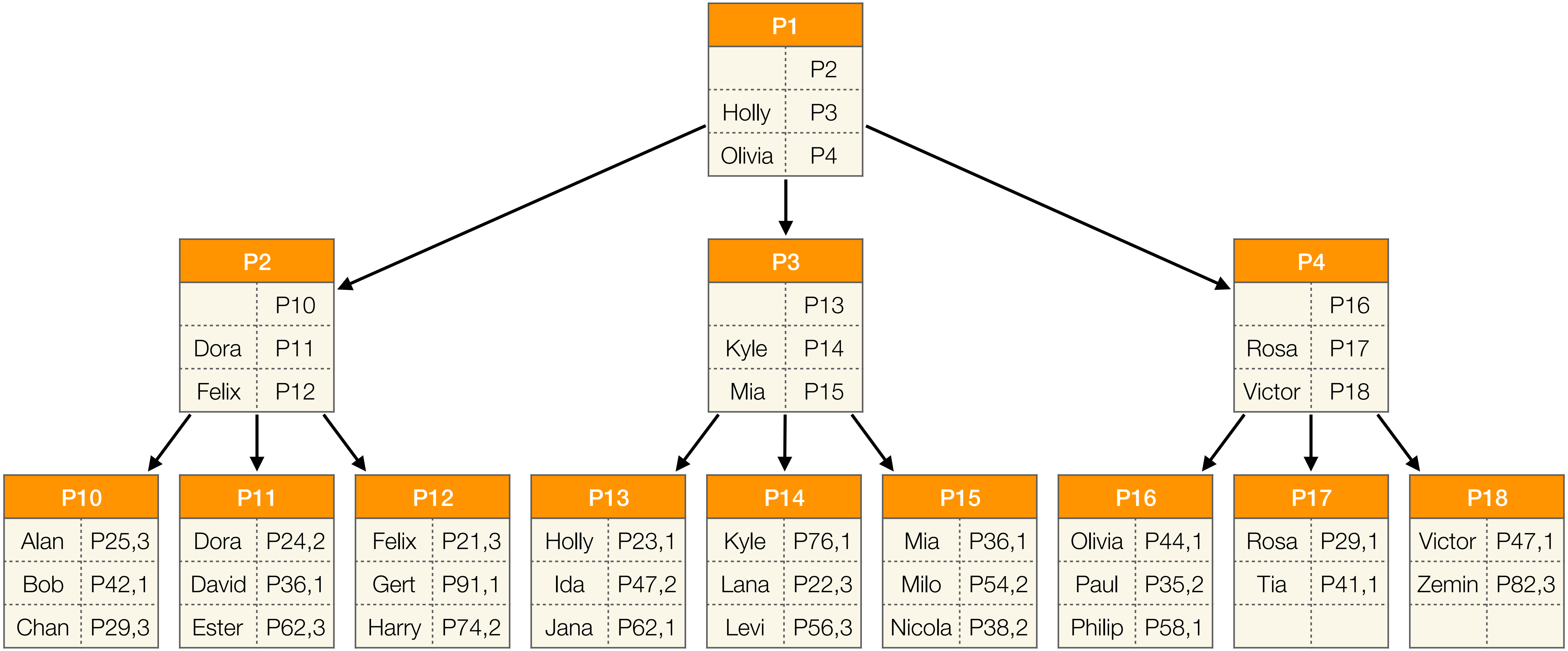
P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Tia	P41,1

P18	
Victor	P47,1
Zemin	P82,3

Slides by Immanuel Trummer, Cornell University

Deletes with Balancing



P1	
	P2
Holly	P3
Olivia	P4

P2	
	P10
Dora	P11
Felix	P12

P3	
	P13
Kyle	P14
Mia	P15

P4	
	P16
Rosa	P17
Victor	P18

P10	
Alan	P25,3
Bob	P42,1
Chan	P29,3

P11	
Dora	P24,2
David	P36,1
Ester	P62,3

P12	
Felix	P21,3
Gert	P91,1
Harry	P74,2

P13	
Holly	P23,1
Ida	P47,2
Jana	P62,1

P14	
Kyle	P76,1
Lana	P22,3
Levi	P56,3

P15	
Mia	P36,1
Milo	P54,2
Nicola	P38,2

P16	
Olivia	P44,1
Paul	P35,2
Philip	P58,1

P17	
Rosa	P29,1
Tia	P41,1

P18	
Victor	P47,1
Zemin	P82,3

Slides by Immanuel Trummer, Cornell University