



Advanced Search Engines with SQL

Lukas Smith - lukas@liip.ch
CodeWorks 2009 - PHP on the ROAD



Disclaimer

This talk is about very specific requirements and how we solved them using a very specific data set on a very specific architecture

The idea is to illustrate the various tricks that worked in this specific case

This is not meant as general advice to always solve these requirements in exactly the same way

But hopefully it will inspire the audience to find solutions when they are faced with similar challenges



Case Study Example

Telephone directory for the entire Raiffeisen Switzerland including:

- HQ in St. Gallen
- Bank Locations
- Meeting rooms
- Service Numbers

Total about 10k internals and external employees

Growing list of meeting rooms and service numbers



Various search interfaces

Livesearch for quick access by name

Blackberry optimized alternative

Profile pages list direct links to related profiles:
superiors, substitutes, assistants, office mates, office
neighbors, group members

iTunes like “browsing” of the hierarchy of the HQ (RCH)
and local bank branches (RB)



Various search interfaces

Full text multi word search



Im Telefonbuch suchen

Organisation Raiffeisen Schweiz

Organisation Raiffeisenbanken

Jakob Mann

Projekt

Suchen

Vordefinierte Suchabfragen









-- Abfrage auswählen --

Alle Felder einbeziehen Ähnliche Namen suchen

Anzeigen: Alles (8) RCH Mitarbeiter (6) RB Mitarbeiter (2)

Liste Drucken

Liste Exportieren

Name, Vorname, Funktion	Telefon, E-Mail	Org. Einheit, Büro/Standort	Trefferbegründung
 Amann, Andreas Mitarbeiter Projektleitung 1		Projektleitung 1 9000 St. Gallen, Raiffeisenplatz M.S.23.P	Basis Informationen
 Hartmann-Grimmer, Augustino Mitarbeiterin Projektportfolio- Management		Bellinzona 9000 Wil, Dorfstrasse Sekretariat	Basis Informationen
 Jaymond, Nadja Mitarbeiter Projektcontrolling	nadja.jaymond@raiffeisen.ch	Projektcontrolling 9000 St. Gallen, Raiffeisenplatz PM422	Basis Informationen
 Lu, Reto Bereichsleiter Intranet-RAIweb	071 923 55 50 reto.lu@raiffeisen.ch	Intranet-RAIweb 9000 Wil, Dorfstrasse Finanzberatung	Basis Informationen
 Mann, Jakob Gruppenleiter Kommunikations- Services	044 111 84 38 jakob.mann@raiffeisen.ch	Kommunikations-Services 9001 St. Gallen, Gartenstrasse LF112	Kenntnisse
 Mayer, Robert Bereichsleiter Marken-Services	044 111 84 10 robert.mayer@raiffeisen.ch	Marken-Services 8953 Dietikon, Lerzenstrasse 16 PM277	Basis Informationen
 Peters, Regina Projektmanager - Senior		St.Gallen 8953 Dietikon, Lerzenstrasse 16 XD123	Basis Informationen
 Stein, Gebhard Mitarbeiter Projektleitung 1		Projektleitung 1 9000 St. Gallen, Raiffeisenplatz 4 XD199	Basis Informationen



DEMO



Examples

This talk will feature several code examples

None of them fit on a screen

Its pseudo code that somewhat resembles the production code

They contain a fair amount of detail, but are stripped down a bit to hopefully be comprehensible

In order to keep the examples shorter, each example inherits the previous examples methods

[Search Base Class](#) | [Output Script](#) | [Highlight](#)



I - Search and sort

Requirements:

Search all relevant tables data in the database

Do not search or show data in fields that are not available due to ACL's to the current user

ACL's will operate on the employer (HQ or a bank) and depending on the data field a user definable flag

Display the first 50 results, but indicate the total number of matches (by RCH/RB)

Search and sort

Query Structure



20+ table join, fetching 30+ columns



II - Use indexes

Try FULLTEXT searching, but in my experience MySQL 's and SQL Server's solution sucks

Replace searches where possible with equality condition or at least "LIKE 'foo%'"

If necessary denormalize reversed strings with a trigger or a materialized view to be able to use an index

Use indexes

Query Structure



20+ table join, fetching 30+ columns



III - Add reason column

Requirements:

For each row show the user the reason why its included in the result

Similar fields are grouped (Name, Phone, Hobbies ...)

Add reason column

Query Structure



20+ table join, fetching 30+ columns



IV - Multiple search terms

Requirements:

Support entering of multiple words

Find a match for each word in any of the tables in order to include an entity in the result

Also include matches on the entire search entry

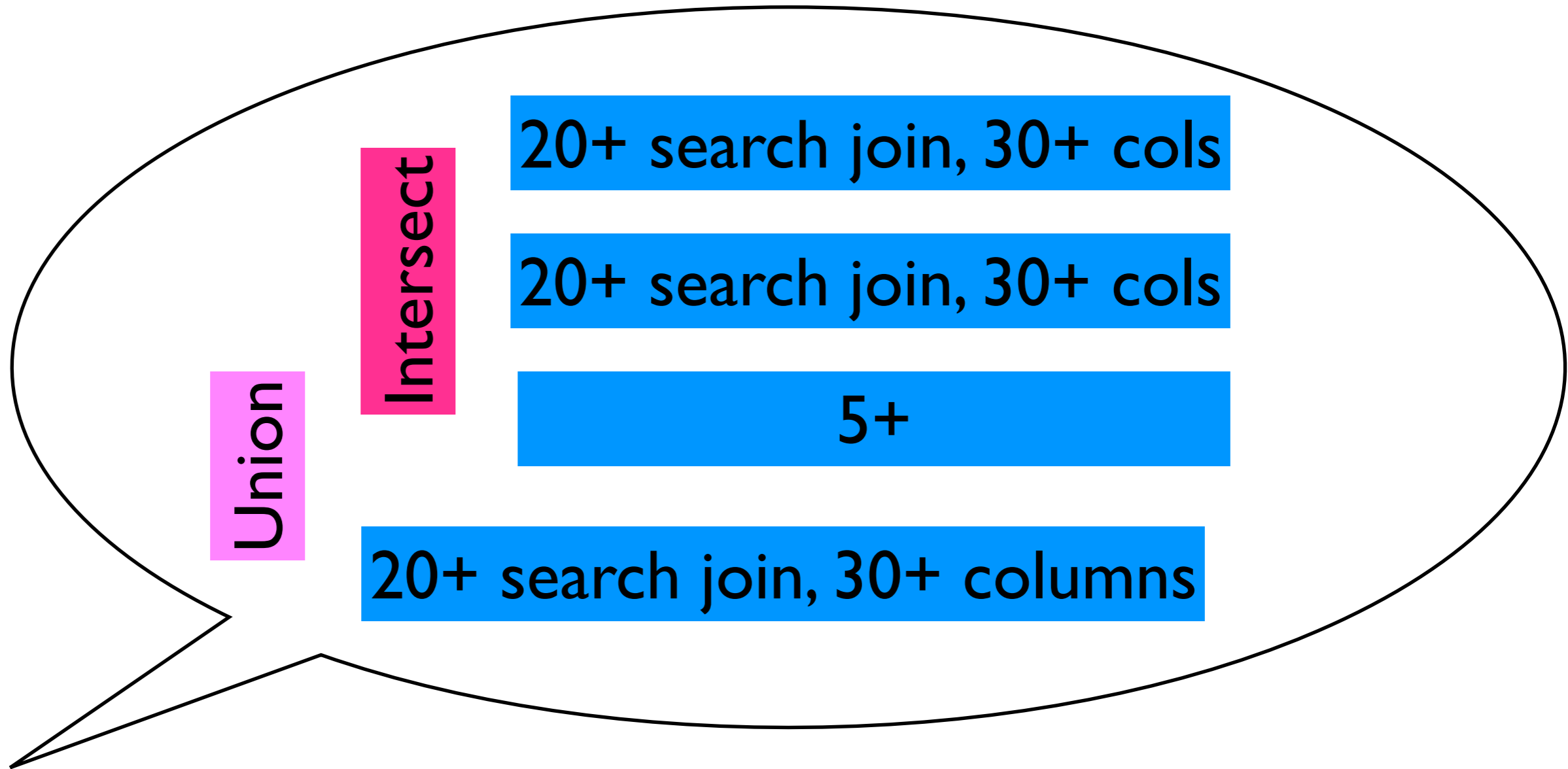
Multiple search terms

Query Structure



20+ table join, fetching 30+ columns

Query Structure





V - Fetch ID's first

Reduce disk I/O by first fetching the ID's and then fetching the data

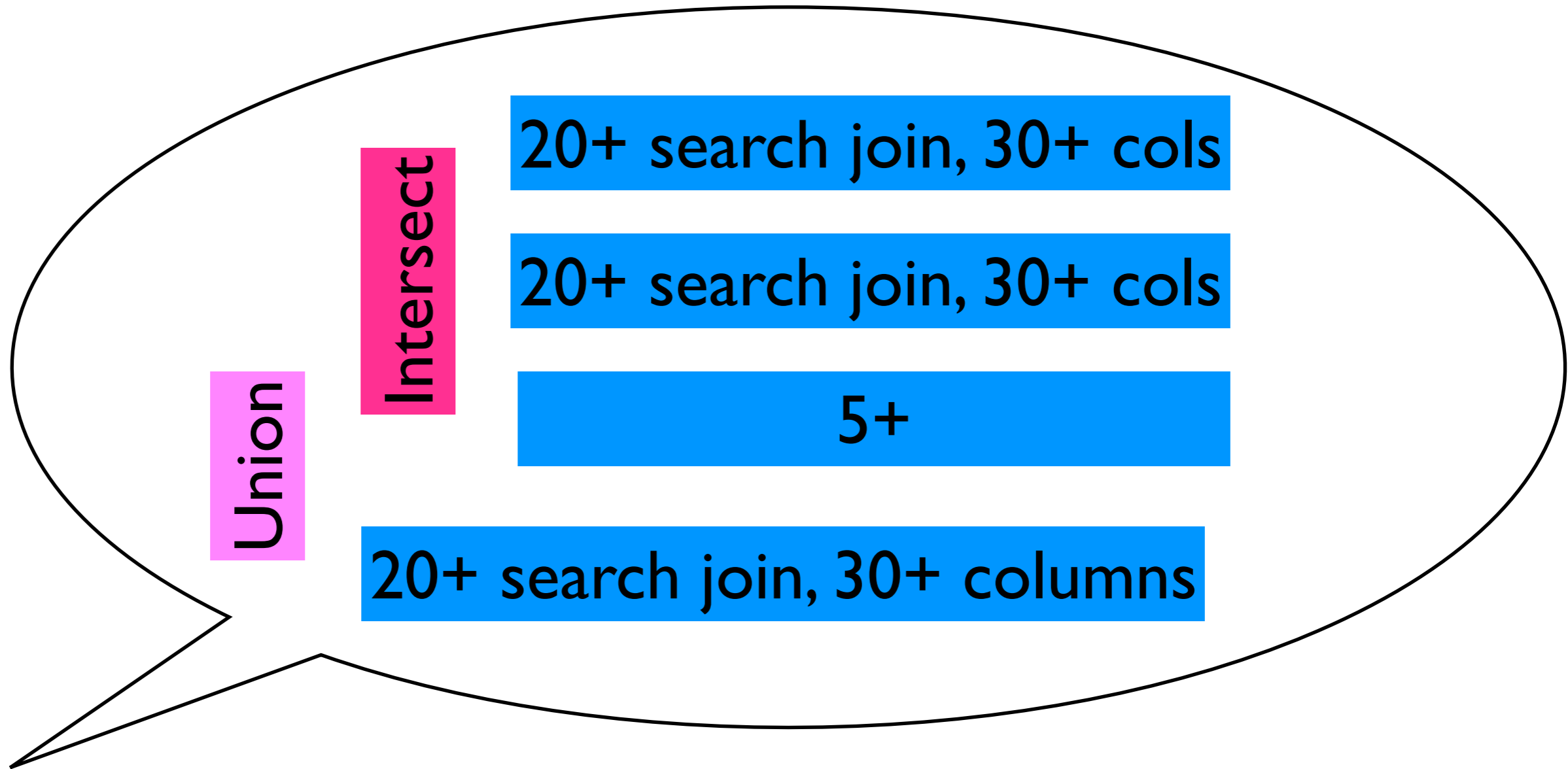
Move the sort in the data fetch query

Set the maximum result set size to a reasonable number

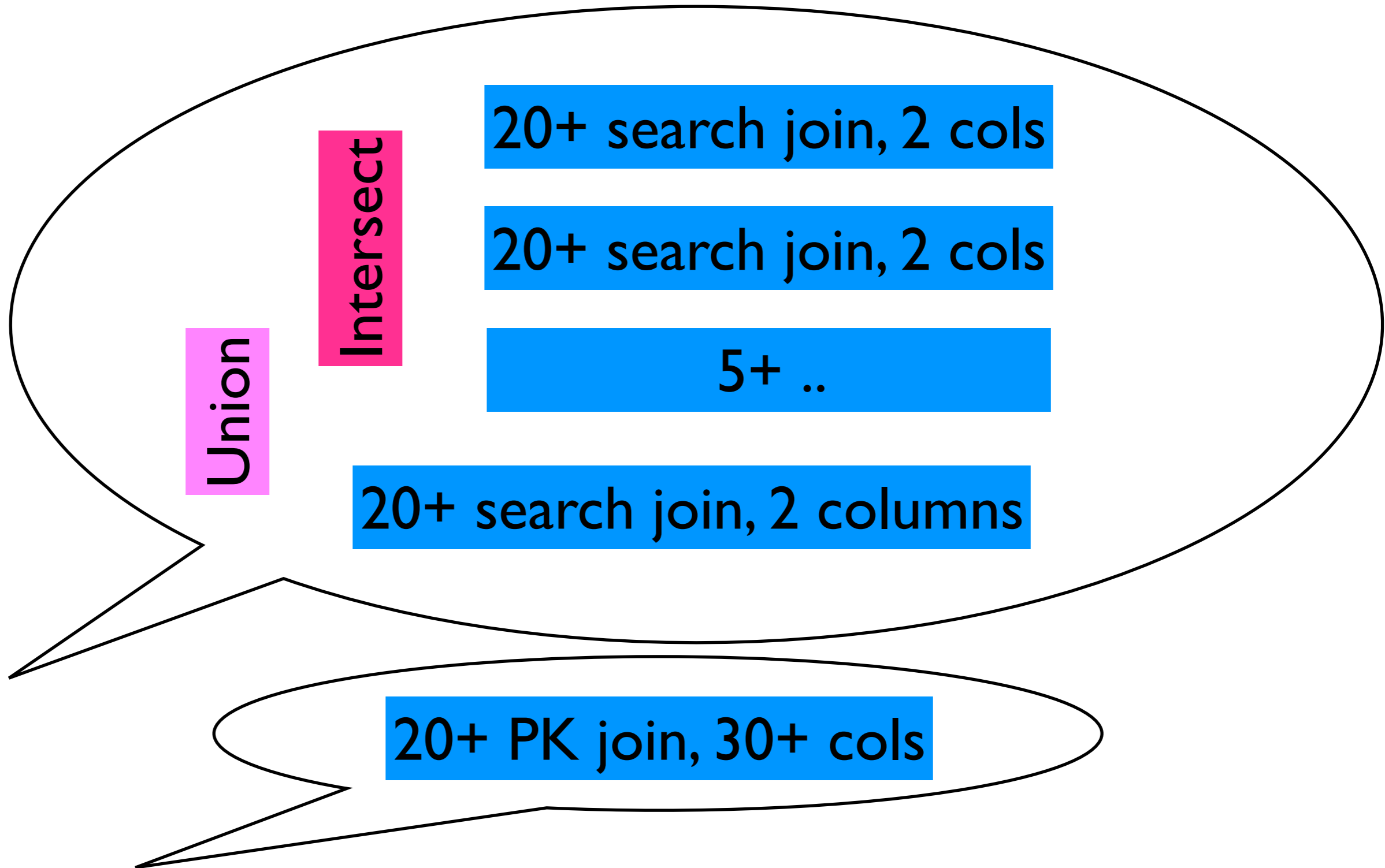
Ensure that the totals are still properly computed

Fetch ID's first

Query Structure



Query Structure





VI Add Fuzzy matches

Requirements:

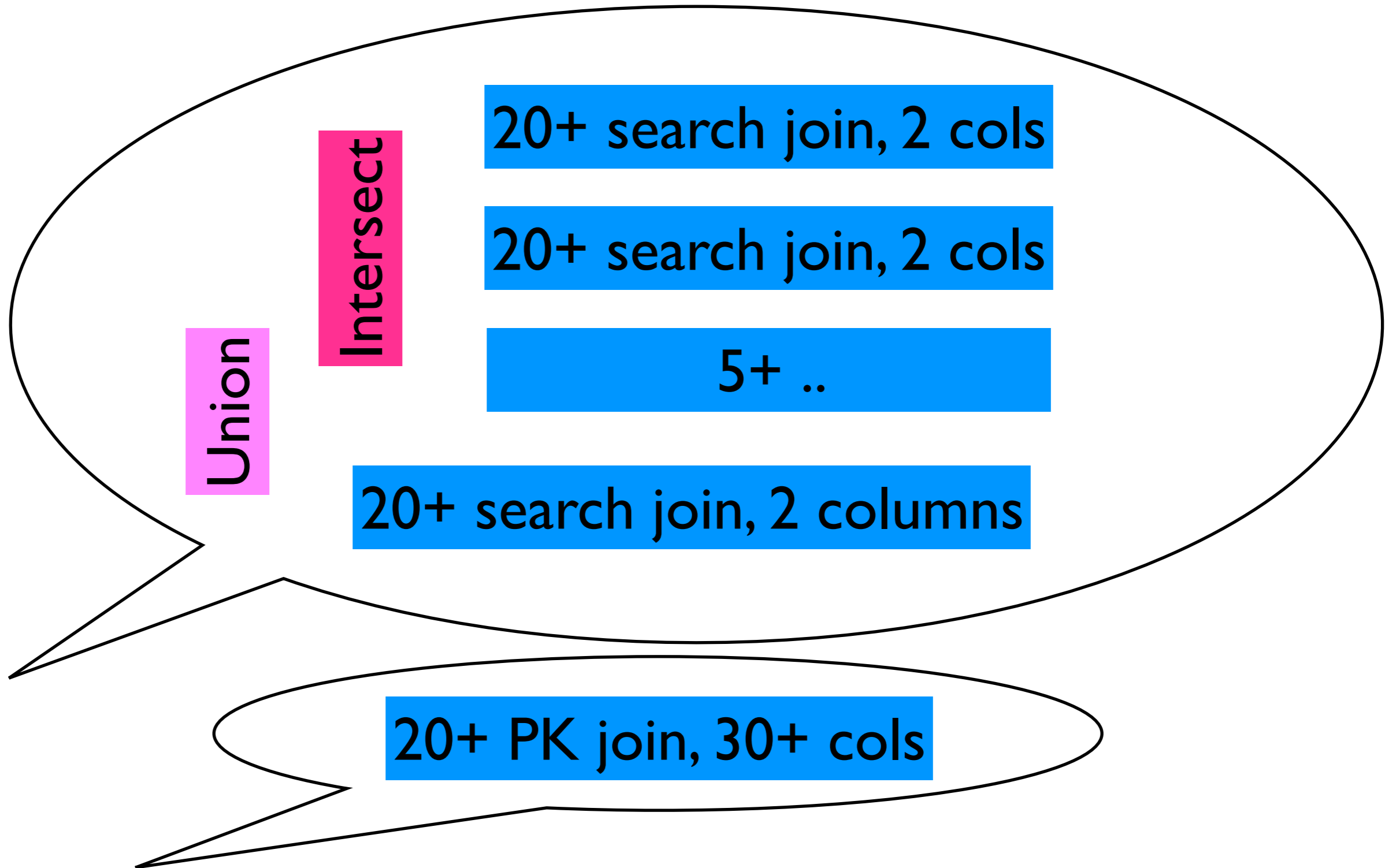
Similar spelling for names should match (Mayer, Meier)

Algorithm to find similarities should support names from all over the world

Umlauts and other special characters should be ignored (Boehmer, Böhmer)

Add fuzzy matches

Query Structure





VII - Switch to UNION

Joining 20+ tables is hard for any RDBMS

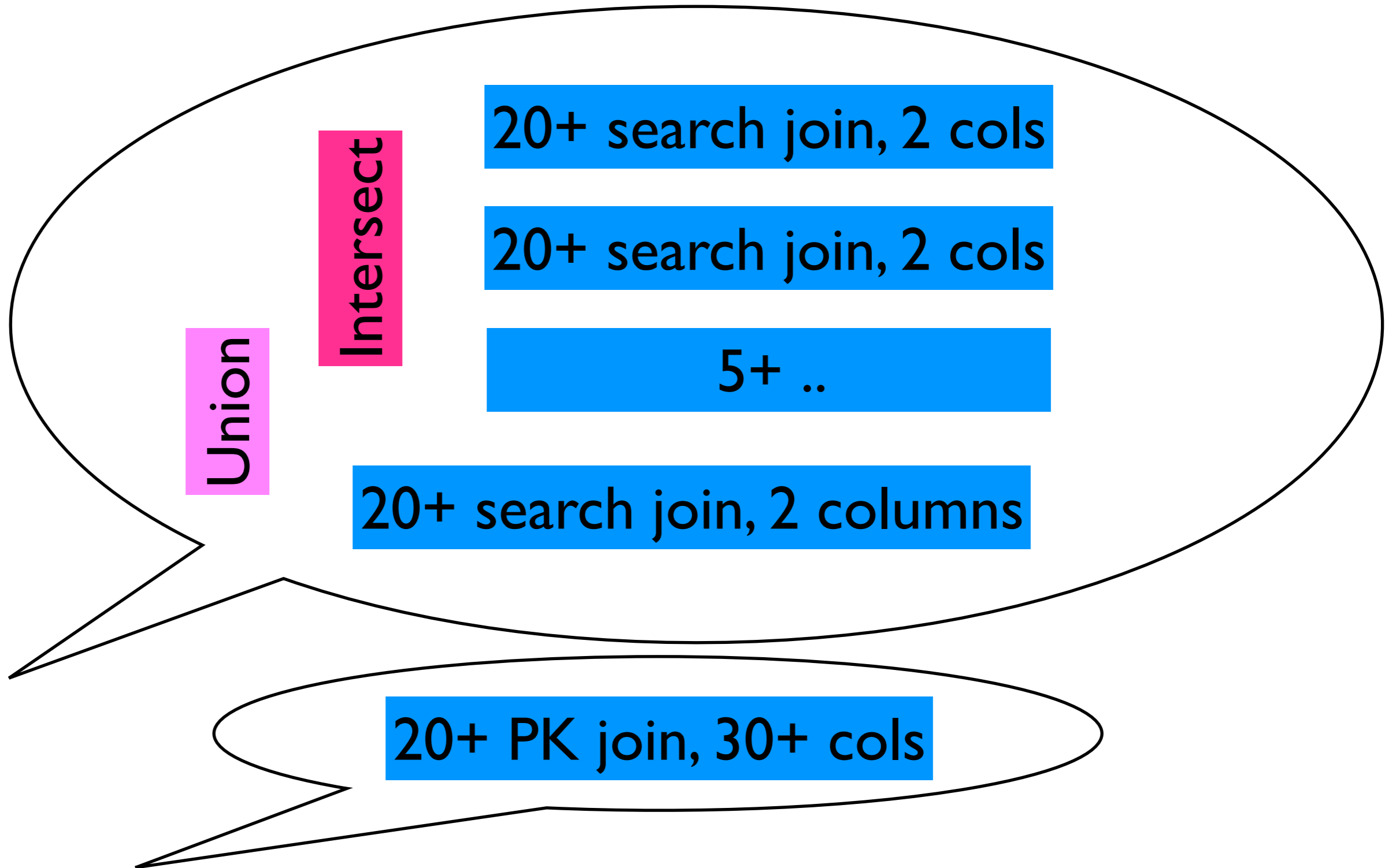
UNION uses less disc I/O, lower memory buffer usage

More stable query plans than with large joins

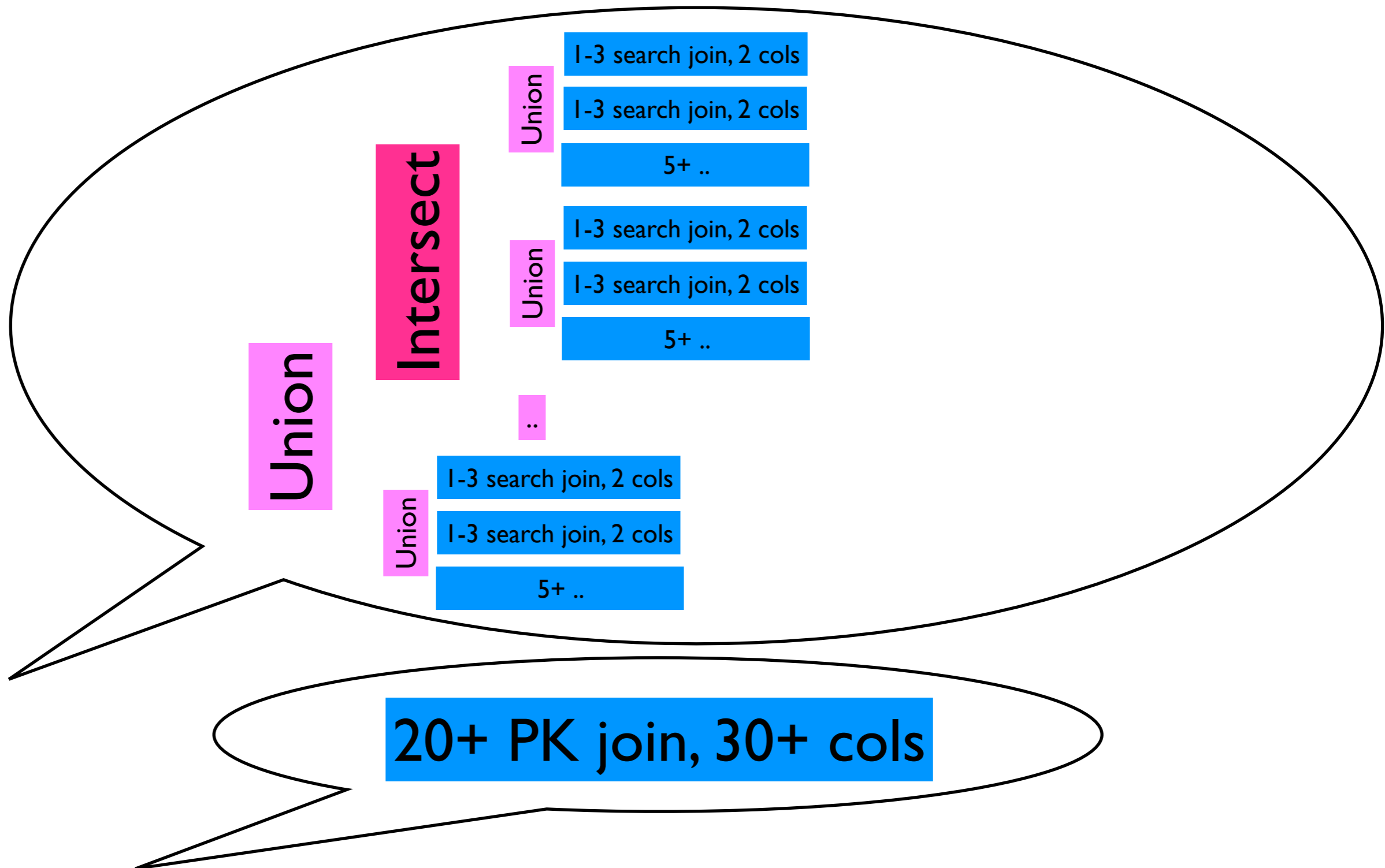
Easier to maintain, debug and optimize

Switch to UNION

Query Structure



php Query Structure





VIII -Dynamic filters

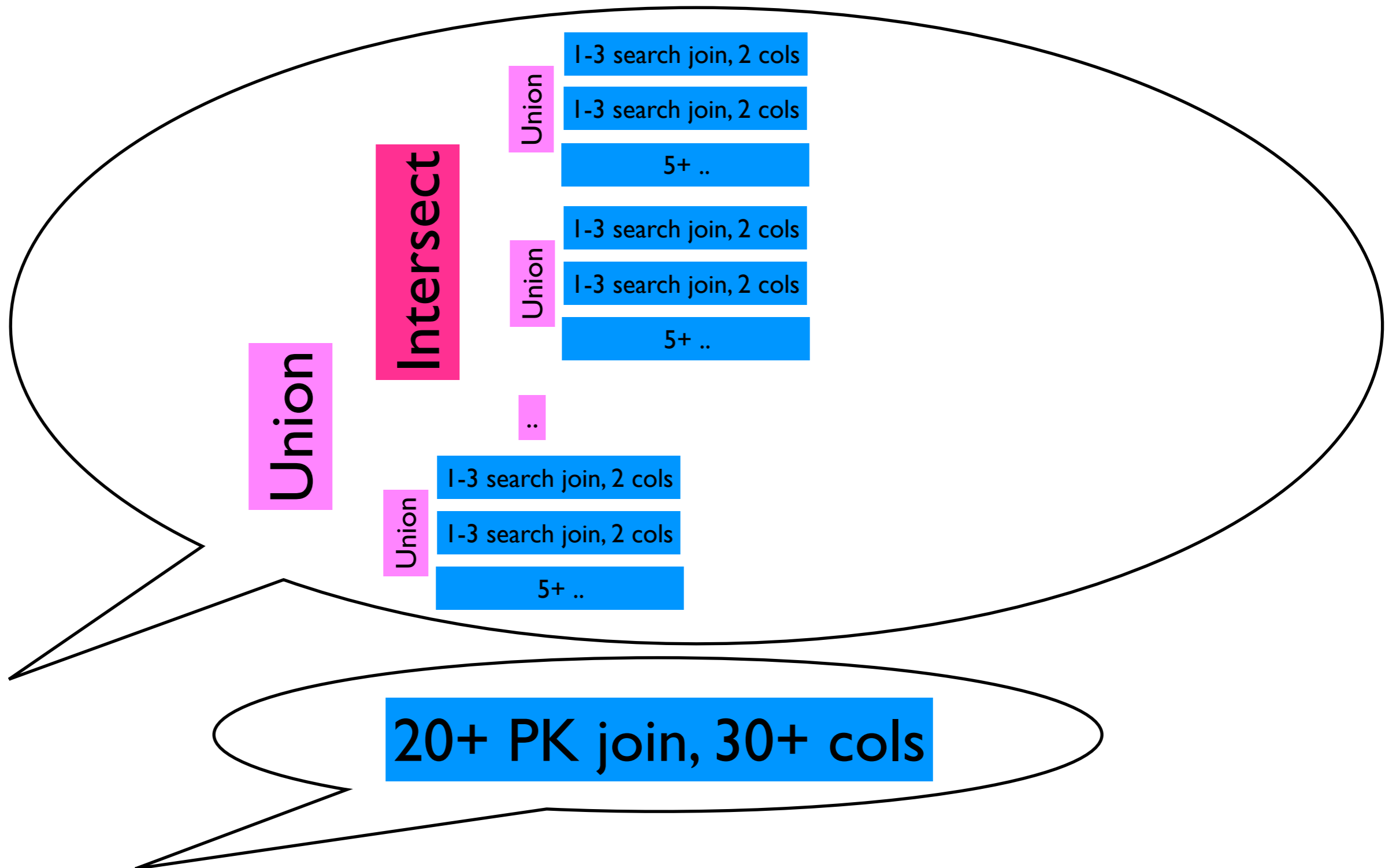
Only query tables and column that can contain the given search term

Numbers can be a phone number or the employee's user id

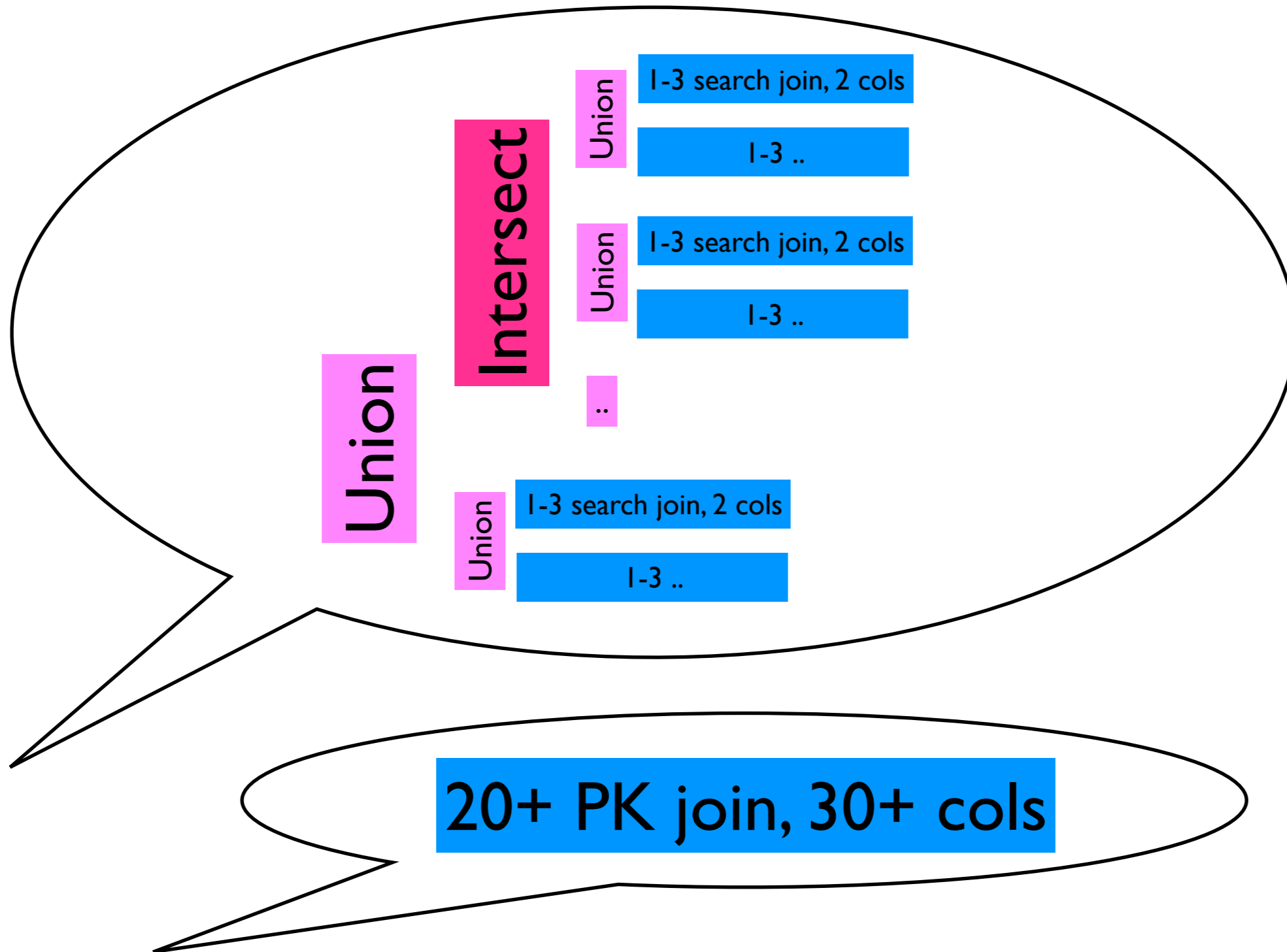
Letters cannot be contained in a phone number

Dynamic filters

php Query Structure



php Query Structure





IX - Sort by relevance

Requirements:

Ensure that more relevant results are presented first

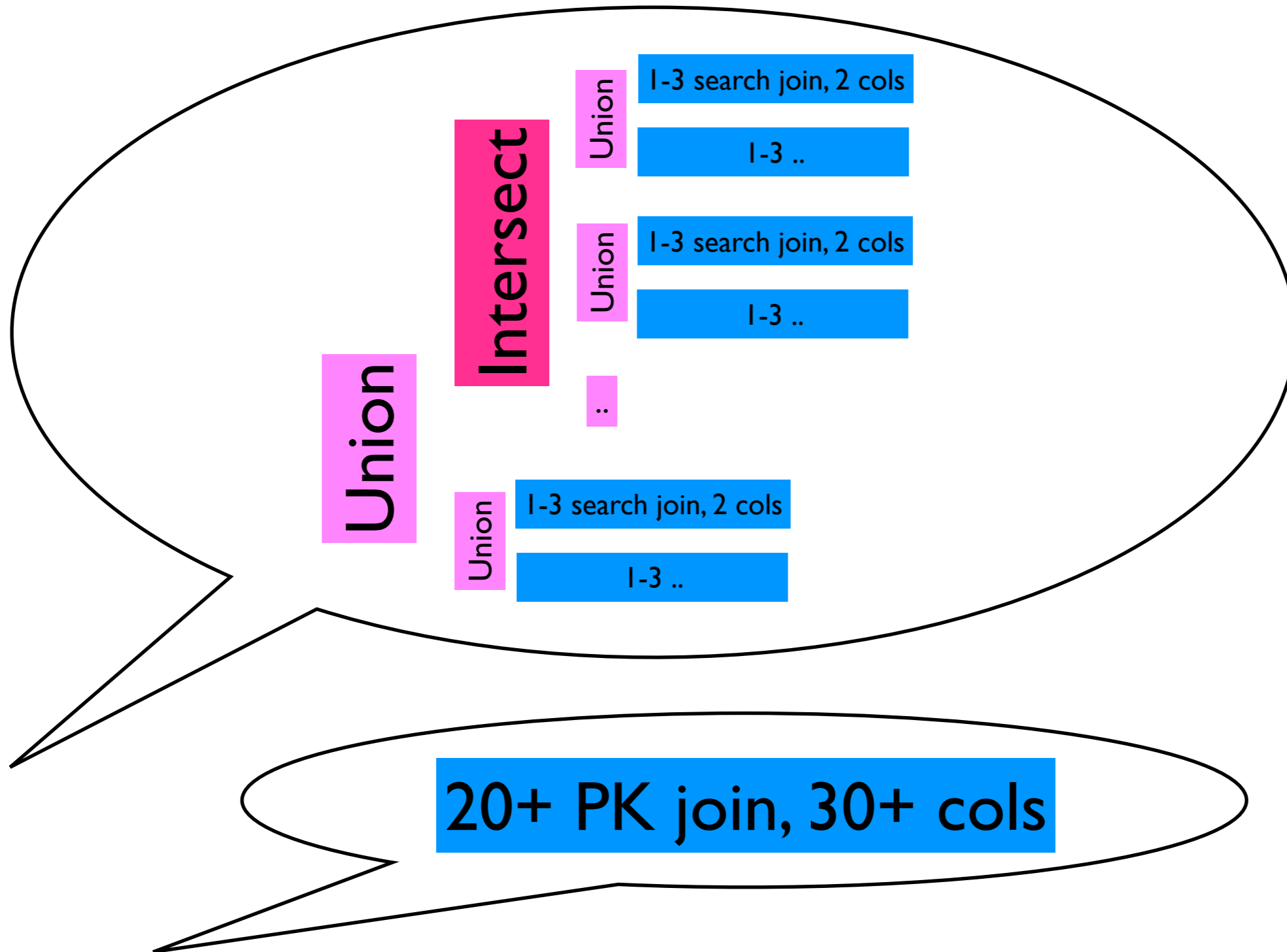
Prioritize the different result reasons

More matches for an entity rate it higher

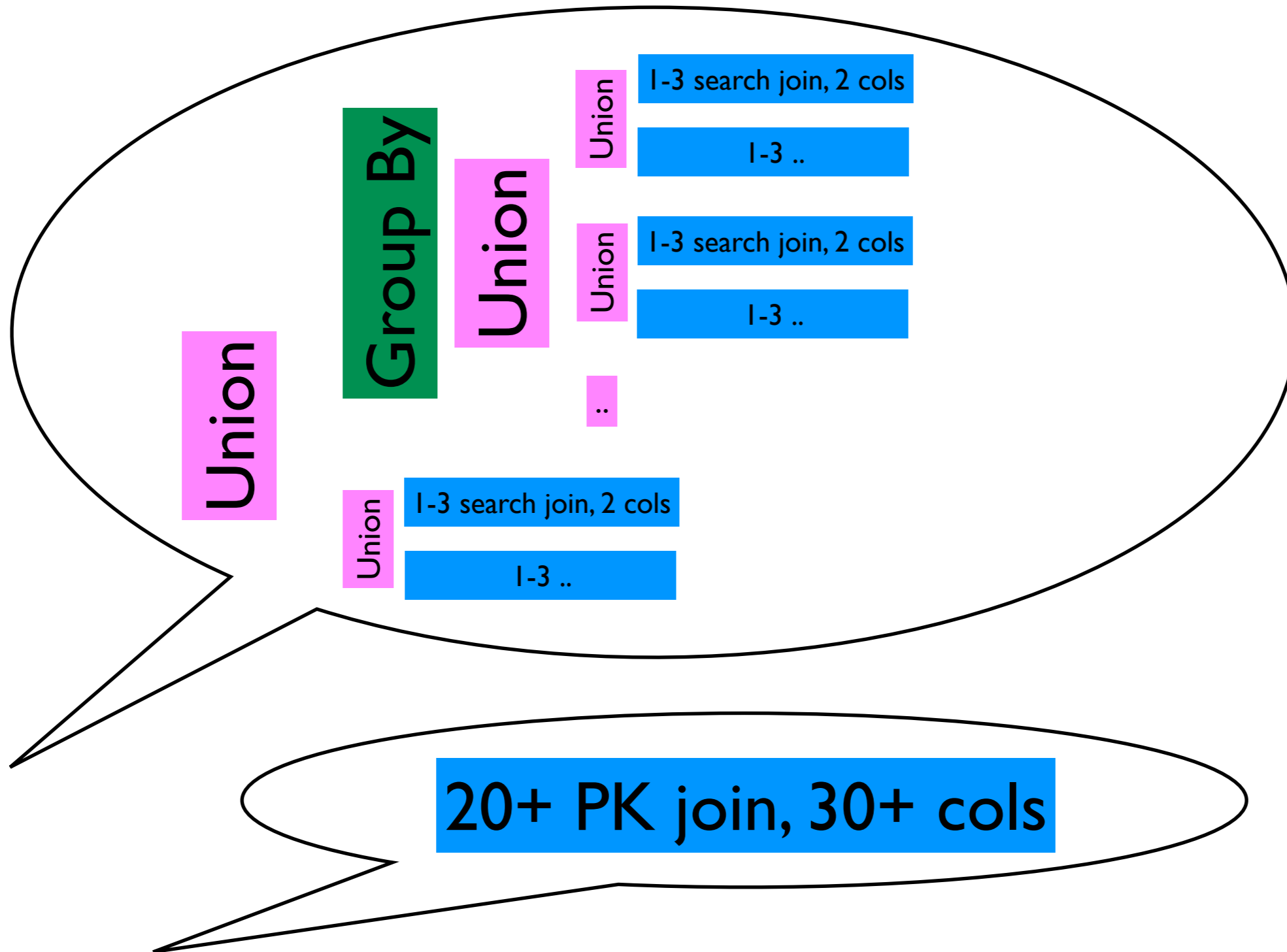
Exact matches should come before wild card searches

Sort by relevance

php Query Structure



Query Structure





XDebug DEMO



Thank you for listening!

Questions? Comments?

Lukas Smith - lukas@liip.ch

CodeWorks 2009 - PHP on the ROAD

Code examples: http://liip.to/advanced_sql_code_09