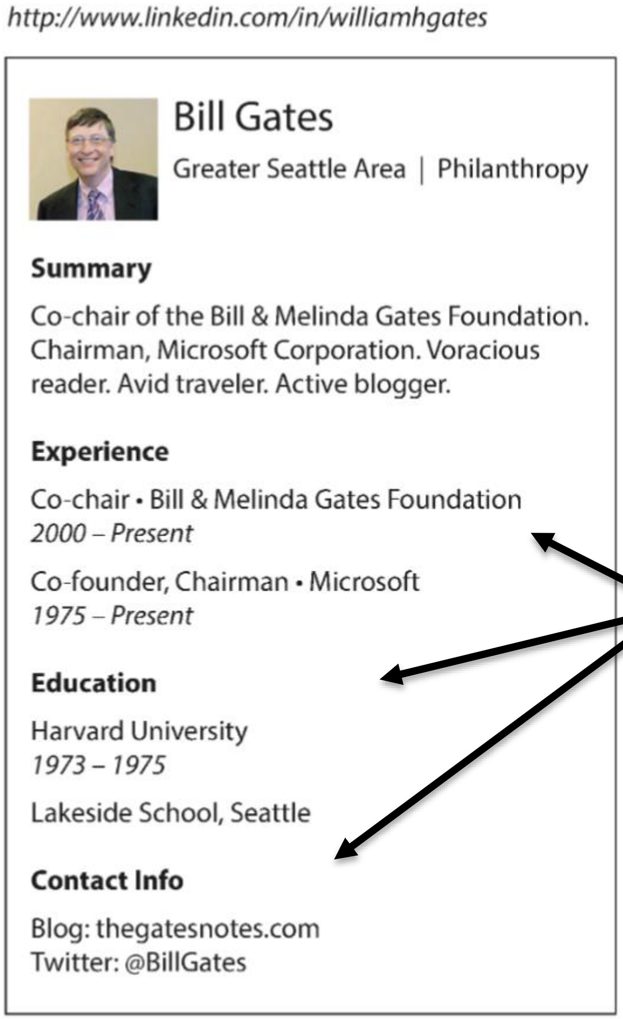
Data Model Design

DQDA-22

LinkedIn Profile Page



http://www.linkedin.com/in/williamhgates

Bill Gates

Greater Seattle Area | Philanthropy

Summary

Co-chair of the Bill & Melinda Gates Foundation. Chairman, Microsoft Corporation. Voracious reader. Avid traveler. Active blogger.

Experience

Co-chair • Bill & Melinda Gates Foundation
2000 – Present

Co-founder, Chairman • Microsoft
1975 – Present

Education

Harvard University
1973 – 1975

Lakeside School, Seattle

Contact Info

Blog: thegatesnotes.com
Twitter: @BillGates

<u>Once per user:</u>

- `user_id`
- `first_name, last_name,`
- …

⇒ Columns on a users table

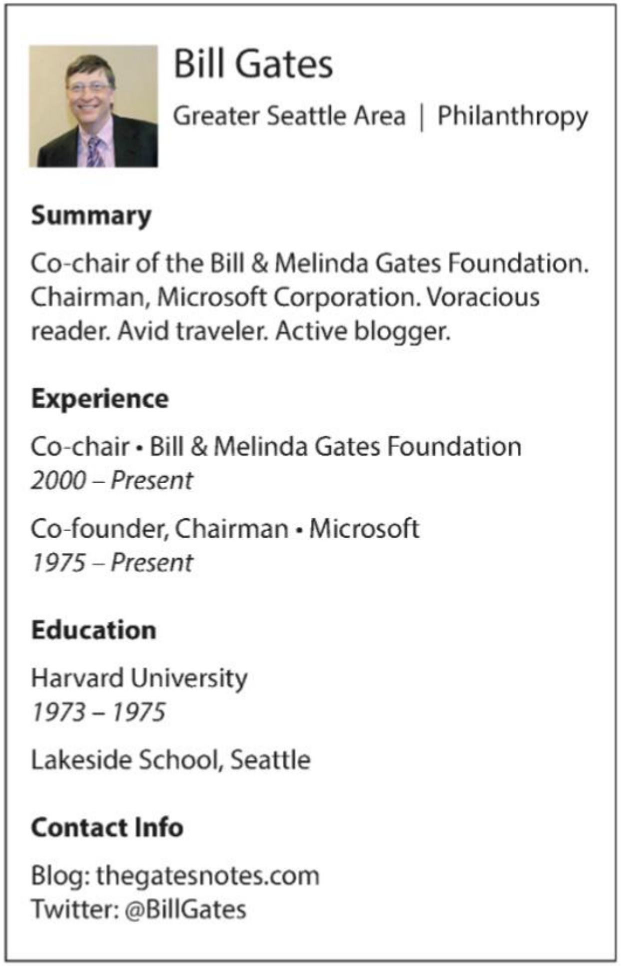
<u>Multiple entries per user:</u>

- More than one job in their career (positions)
- Varying numbers of periods of education
- Number of pieces of contact information

⇒ **one-to-many relationships**

Data Model and Relationships

http://www.linkedin.com/in/williamhgates



Bill Gates

Greater Seattle Area | Philanthropy

Summary

Co-chair of the Bill & Melinda Gates Foundation. Chairman, Microsoft Corporation. Voracious reader. Avid traveler. Active blogger.

Experience

Co-chair • Bill & Melinda Gates Foundation
2000 – Present

Co-founder, Chairman • Microsoft
1975 – Present

Education

Harvard University
1973 – 1975

Lakeside School, Seattle

Contact Info

Blog: thegatesnotes.com
Twitter: @BillGates

Traditional SQL:

- Separate tables, with a foreign key reference

Structured datatypes, JSON/XML[1]:

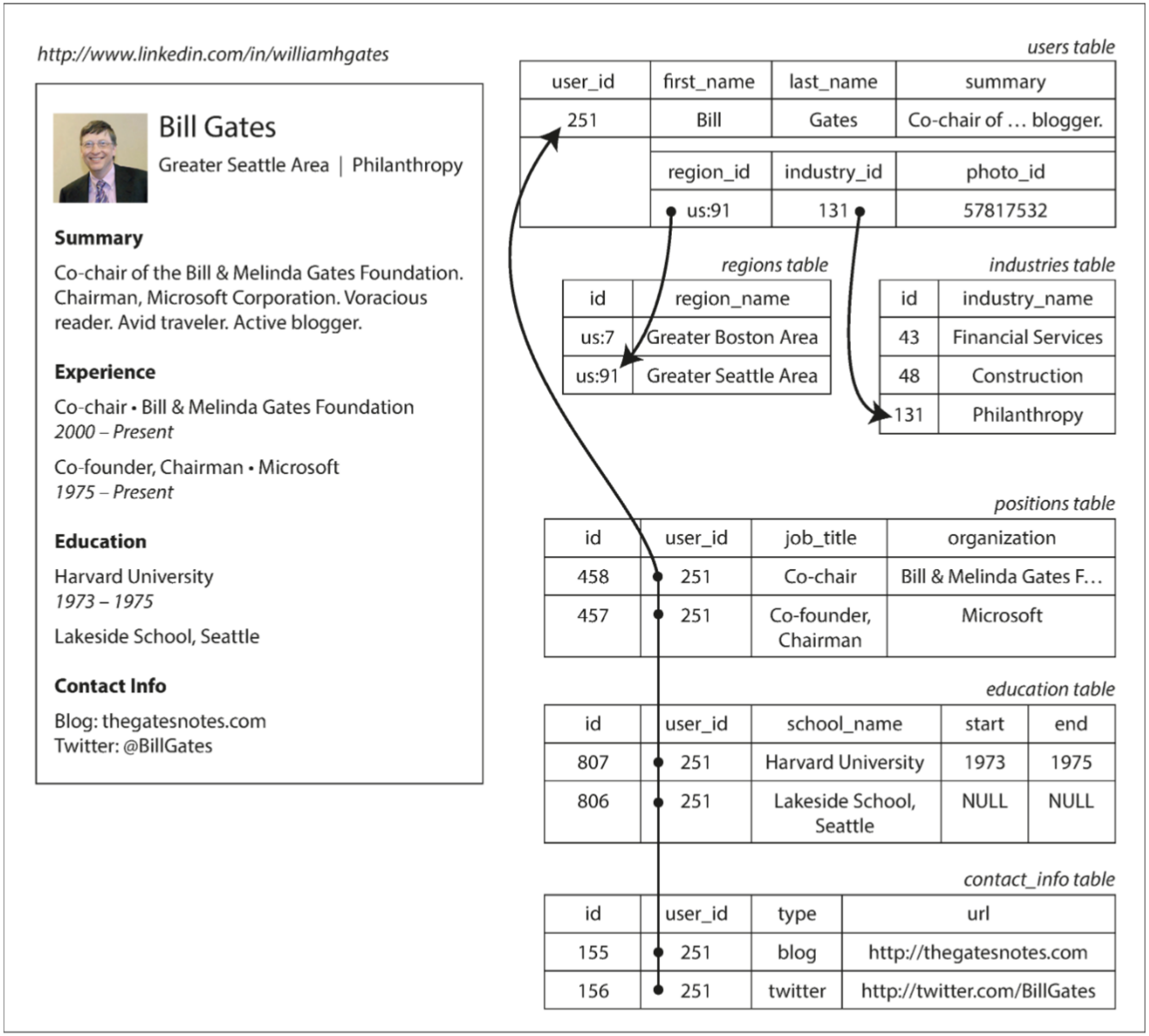
- Multi-Valued data in single row
- Querying, indexing inside those documents

JSON or XML document:

- Encode and store as a self-contained document
- Applications to interpret its structure and content
- better locality than the multi-table schema

[1] Supported by: Oracle, IBM DB2, MSSQL Server, PostgreSQL, MySQL

Traditional SQL



http://www.linkedin.com/in/williamhgates

Bill Gates

Greater Seattle Area | Philanthropy

Summary

Co-chair of the Bill & Melinda Gates Foundation. Chairman, Microsoft Corporation. Voracious reader. Avid traveler. Active blogger.

Experience

Co-chair • Bill & Melinda Gates Foundation
2000 – Present

Co-founder, Chairman • Microsoft
1975 – Present

Education

Harvard University
1973 – 1975

Lakeside School, Seattle

Contact Info

Blog: thegatesnotes.com
Twitter: @BillGates

users table

user_id	first_name	last_name	summary
251	Bill	Gates	Co-chair of … blogger.
	region_id	industry_id	photo_id
	us:91	131	57817532

regions table

id	region_name
us:7	Greater Boston Area
us:91	Greater Seattle Area

industries table

id	industry_name
43	Financial Services
48	Construction
131	Philanthropy

positions table

id	user_id	job_title	organization
458	251	Co-chair	Bill & Melinda Gates F…
457	251	Co-founder, Chairman	Microsoft

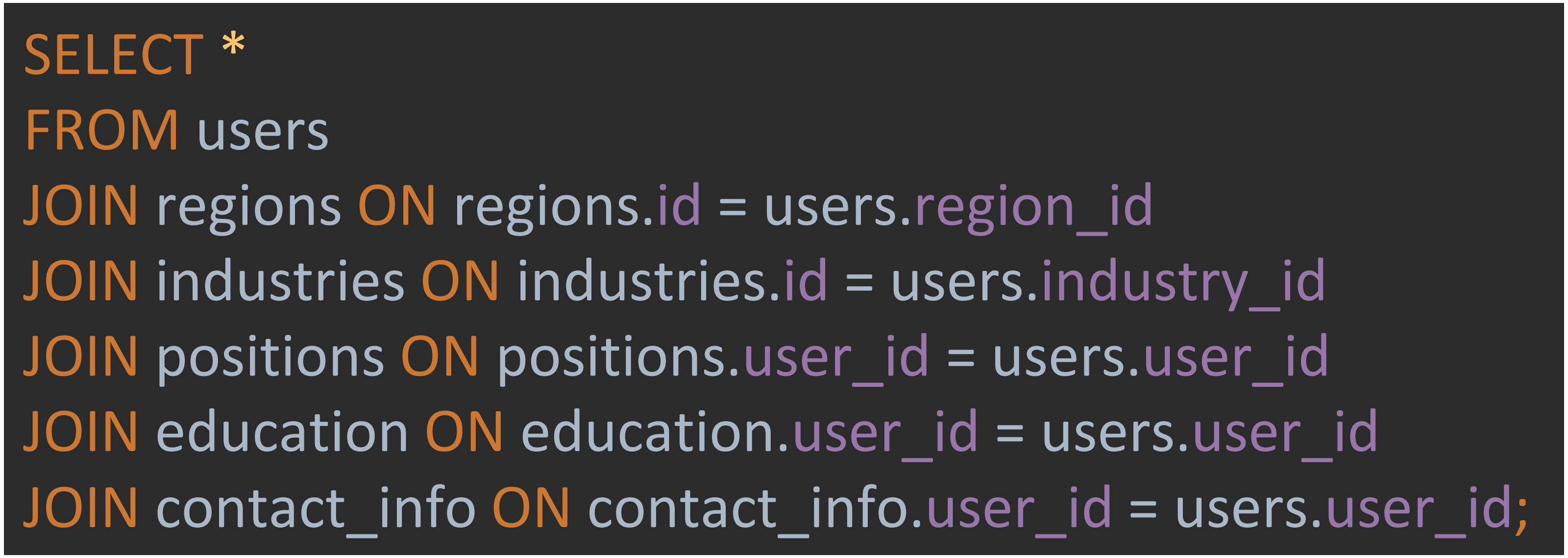
education table

id	user_id	school_name	start	end
807	251	Harvard University	1973	1975
806	251	Lakeside School, Seattle	NULL	NULL

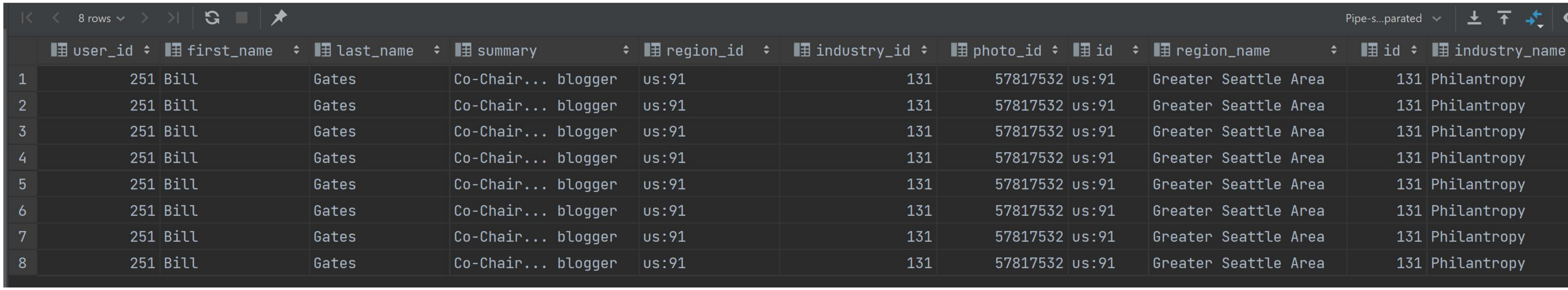
contact_info table

id	user_id	type	url
155	251	blog	http://thegatesnotes.com
156	251	twitter	http://twitter.com/BillGates

Traditional SQL Demo



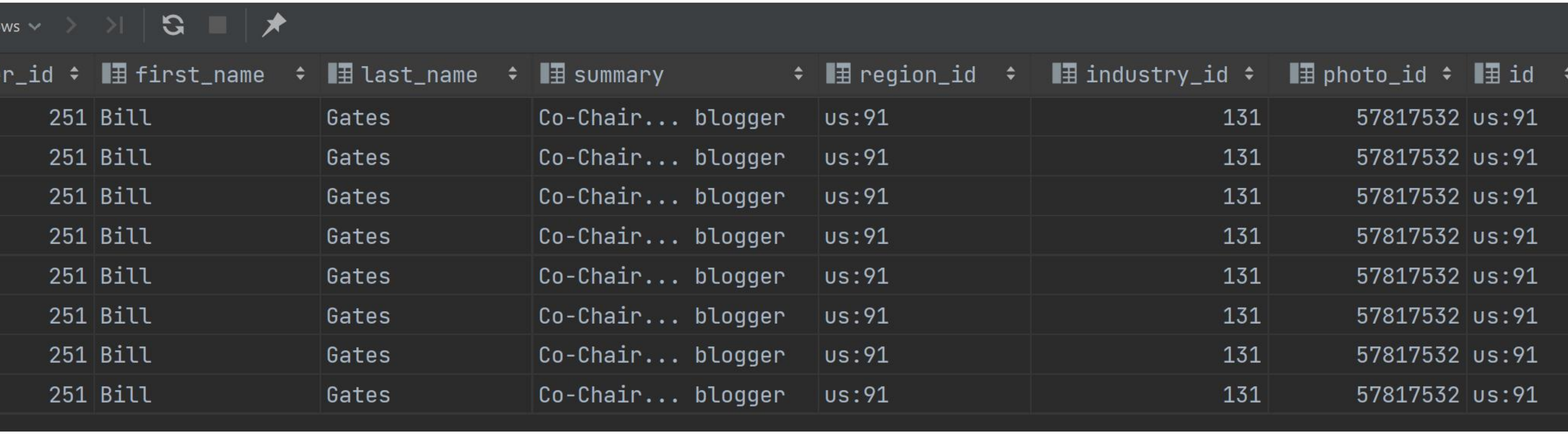
```
SELECT *
FROM users
JOIN regions ON regions.id = users.region_id
JOIN industries ON industries.id = users.industry_id
JOIN positions ON positions.user_id = users.user_id
JOIN education ON education.user_id = users.user_id
JOIN contact_info ON contact_info.user_id = users.user_id;
```



	user_id	first_name	last_name	summary	region_id	industry_id	photo_id	id	region_name	id	industry_name
1	251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91	Greater Seattle Area	131	Philantropy
2	251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91	Greater Seattle Area	131	Philantropy
3	251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91	Greater Seattle Area	131	Philantropy
4	251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91	Greater Seattle Area	131	Philantropy
5	251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91	Greater Seattle Area	131	Philantropy
6	251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91	Greater Seattle Area	131	Philantropy
7	251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91	Greater Seattle Area	131	Philantropy
8	251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91	Greater Seattle Area	131	Philantropy

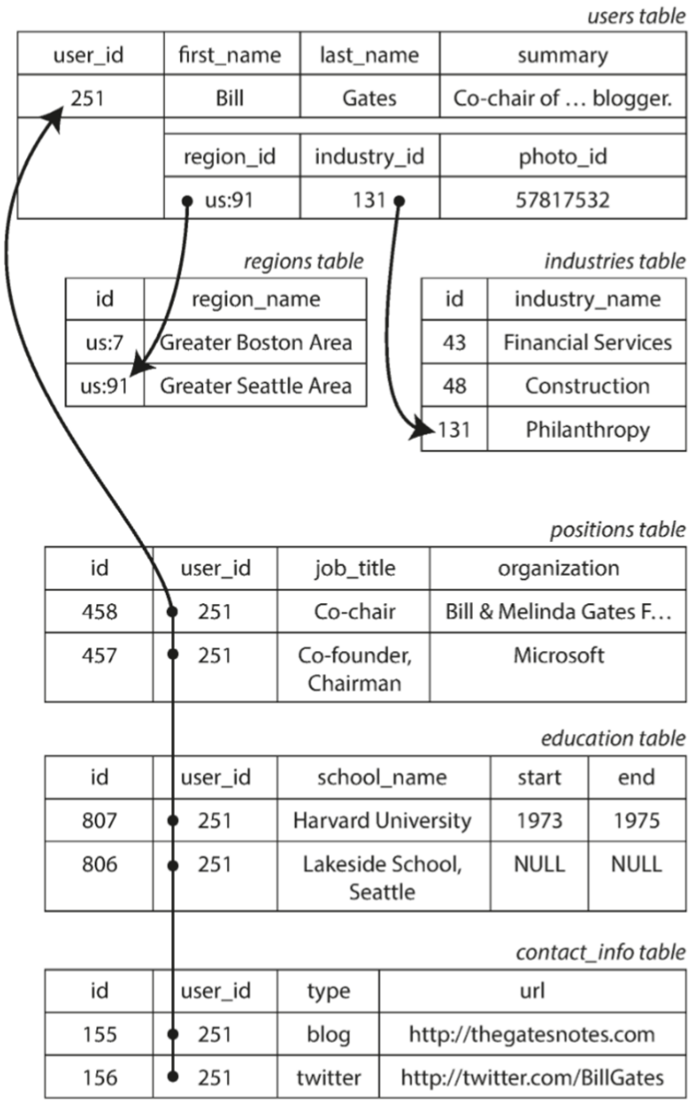
Combinatorial/Cartesian Explosion

- Amount of duplicated data may grow and adversely affect performance and ambiguity



r_id	first_name	last_name	summary	region_id	industry_id	photo_id	id
251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91
251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91
251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91
251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91
251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91
251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91
251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91
251	Bill	Gates	Co-Chair... blogger	us:91	131	57817532	us:91

Traditional Model



users table

user_id	first_name	last_name	summary
251	Bill	Gates	Co-chair of … blogger.

region_id	industry_id	photo_id
us:91	131	57817532

regions table

id	region_name
us:7	Greater Boston Area
us:91	Greater Seattle Area

industries table

id	industry_name
43	Financial Services
48	Construction
131	Philanthropy

positions table

id	user_id	job_title	organization
458	251	Co-chair	Bill & Melinda Gates F…
457	251	Co-founder, Chairman	Microsoft

education table

id	user_id	school_name	start	end
807	251	Harvard University	1973	1975
806	251	Lakeside School, Seattle	NULL	NULL

contact_info table

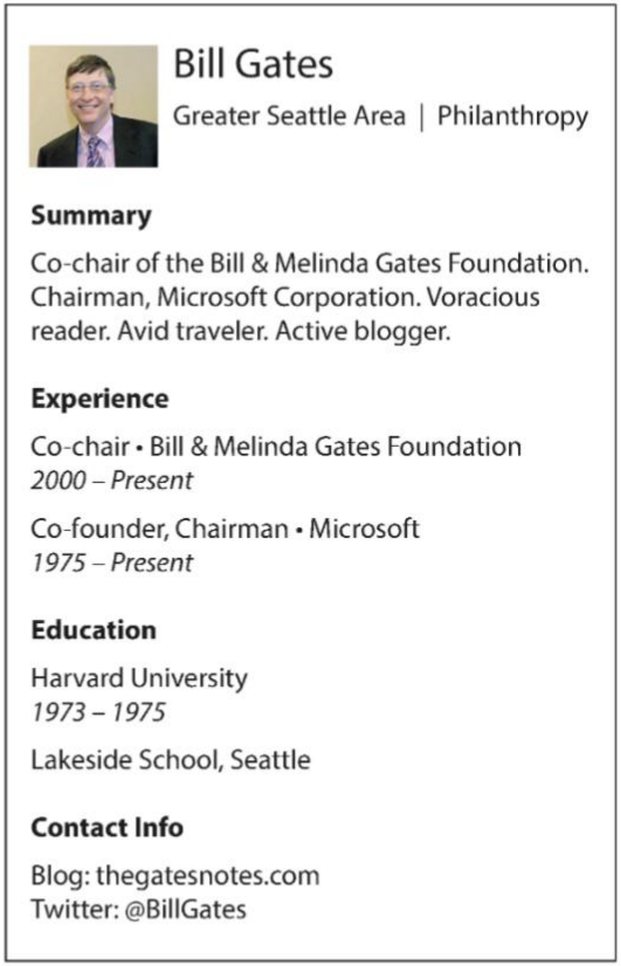
id	user_id	type	url
155	251	blog	http://thegatesnotes.com
156	251	twitter	http://twitter.com/BillGates

- Perform multiple queries (query each table by `user_id`)
- Multi-way join between the users table and subordinate tables
- Applications in OOP:
  Translation layer required for objects in the database
  Disconnect is called *impedance mismatch*[1]

[1] Term from electronics:
Every electric circuit has a certain impedance (resistance to alternating current) on its inputs and outputs. When you connect one circuit's output to another one's input, the power transfer across the connection is maximized if the output and input impedances of the two circuits match. An impedance mismatch can lead to signal reflections and other troubles.

Store as Document

http://www.linkedin.com/in/williamhgates



Bill Gates

Greater Seattle Area | Philanthropy

Summary

Co-chair of the Bill & Melinda Gates Foundation. Chairman, Microsoft Corporation. Voracious reader. Avid traveler. Active blogger.

Experience

Co-chair • Bill & Melinda Gates Foundation
2000 – Present

Co-founder, Chairman • Microsoft
1975 – Present

Education

Harvard University
1973 – 1975

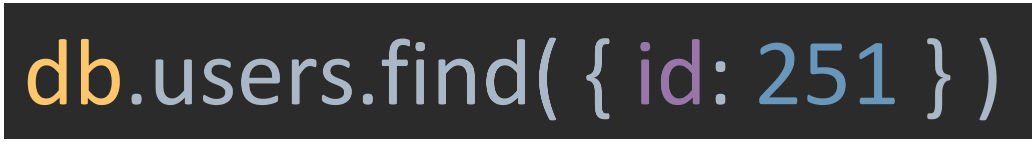
Lakeside School, Seattle

Contact Info

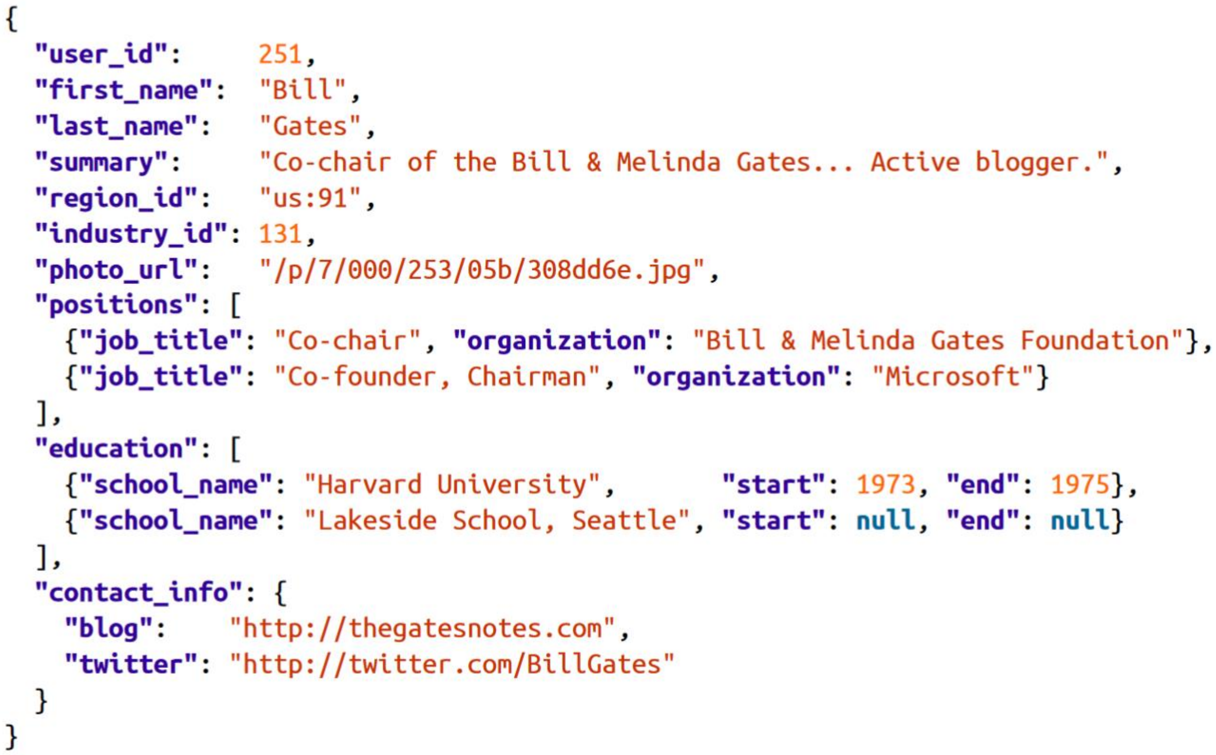
Blog: thegatesnotes.com
Twitter: @BillGates

```
{
  "user_id":     251,
  "first_name":  "Bill",
  "last_name":   "Gates",
  "summary":     "Co-chair of the Bill & Melinda Gates... Active blogger.",
  "region_id":   "us:91",
  "industry_id": 131,
  "photo_url":   "/p/7/000/253/05b/308dd6e.jpg",
  "positions": [
    {"job_title": "Co-chair", "organization": "Bill & Melinda Gates Foundation"},
    {"job_title": "Co-founder, Chairman", "organization": "Microsoft"}
  ],
  "education": [
    {"school_name": "Harvard University",       "start": 1973, "end": 1975},
    {"school_name": "Lakeside School, Seattle", "start": null, "end": null}
  ],
  "contact_info": {
    "blog":    "http://thegatesnotes.com",
    "twitter": "http://twitter.com/BillGates"
  }
}
```

That would be it

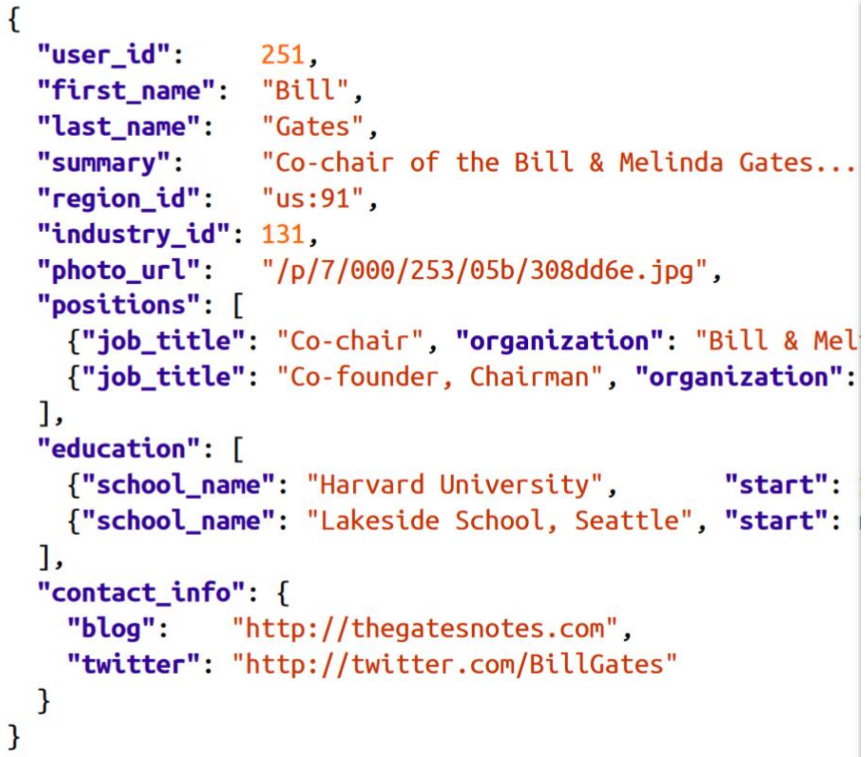


```
db.users.find( { id: 251 } )
```



```
{
  "user_id":     251,
  "first_name":  "Bill",
  "last_name":   "Gates",
  "summary":     "Co-chair of the Bill & Melinda Gates... Active blogger.",
  "region_id":   "us:91",
  "industry_id": 131,
  "photo_url":   "/p/7/000/253/05b/308dd6e.jpg",
  "positions": [
    {"job_title": "Co-chair", "organization": "Bill & Melinda Gates Foundation"},
    {"job_title": "Co-founder, Chairman", "organization": "Microsoft"}
  ],
  "education": [
    {"school_name": "Harvard University",       "start": 1973, "end": 1975},
    {"school_name": "Lakeside School, Seattle", "start": null, "end": null}
  ],
  "contact_info": {
    "blog":    "http://thegatesnotes.com",
    "twitter": "http://twitter.com/BillGates"
  }
}
```

Stored as Document



```
{
  "user_id":     251,
  "first_name":  "Bill",
  "last_name":   "Gates",
  "summary":     "Co-chair of the Bill & Melinda Gates...
  "region_id":   "us:91",
  "industry_id": 131,
  "photo_url":   "/p/7/000/253/05b/308dd6e.jpg",
  "positions": [
    {"job_title": "Co-chair", "organization": "Bill & Mel
    {"job_title": "Co-founder, Chairman", "organization":
  ],
  "education": [
    {"school_name": "Harvard University",       "start":
    {"school_name": "Lakeside School, Seattle", "start":
  ],
  "contact_info": {
    "blog":    "http://thegatesnotes.com",
    "twitter": "http://twitter.com/BillGates"
  }
}
```

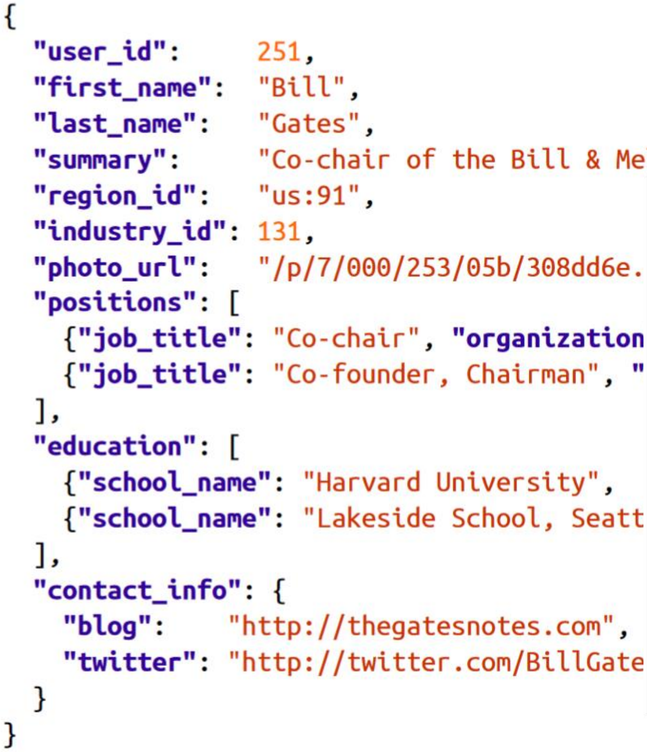
- JSON model reduces the impedance mismatch between
  the application code and the storage layer

- Better locality than the multi-table schema

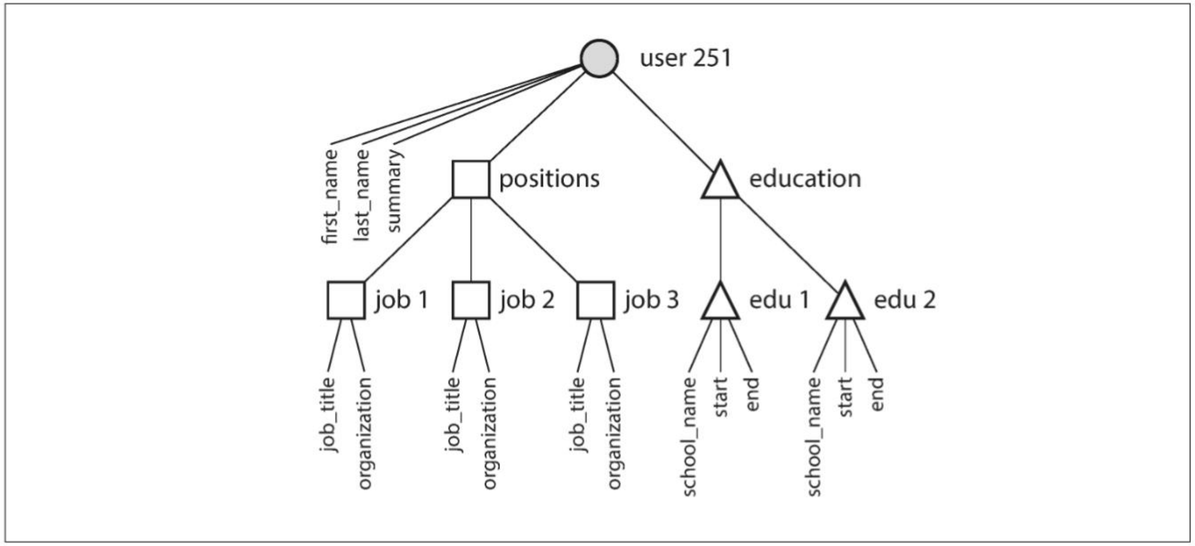
- All the relevant information is in one place and one query is sufficient (No Joins)

- Document Databases

Representation as Hierarchy



```
{
  "user_id":     251,
  "first_name":  "Bill",
  "last_name":   "Gates",
  "summary":     "Co-chair of the Bill & Melinda Gates... Active blogger.",
  "region_id":   "us:91",
  "industry_id": 131,
  "photo_url":   "/p/7/000/253/05b/308dd6e.
  "positions": [
    {"job_title": "Co-chair", "organization
    {"job_title": "Co-founder, Chairman", "
  ],
  "education": [
    {"school_name": "Harvard University",
    {"school_name": "Lakeside School, Seatt
  ],
  "contact_info": {
    "blog":    "http://thegatesnotes.com",
    "twitter": "http://twitter.com/BillGate
  }
}
```



Documents vs Relations

- Main arguments in favor of documents:
  - schema flexibility
  - better performance due to locality
  - for some applications it is closer to the data structures used by the application.
- The relational model counters by providing
  - better support for joins
  - many-to-one
  - many-to-many relationships

Challenges

- Consistency
  - Referential Integrity: Take a moment to think about updates, e.g. The city is stored in only one place, so it is easy to update across the board if it ever needs to be changed
  - The ID can remain the same, even if the information it identifies changes
- Extending the model
- Integrity due to “schema flexiblity”

That’s not all...

- Many more differences when comparing relational databases to non-relational
- Fault-tolerance properties, handling of concurrency....
- We focused only on model representation
- We will have a deeper look into DocumentDBs in the next session

Key take aways for now...

- Multiple ways to model and store data
  Different kind of data may be stored in different kind of ways

- There is no "right way". It all boils down to your business, use case, infrastructure, company, money, in-house knowledge,...

- RDBMS used alongside a broad variety of nonrelational datastores – "*polyglot persistence*"

Polyglot Persistence

MongoDB
Document Store

- ✓ Product Catalog & Details

Elasticsearch
Search Engine

- ✓ Product Search
- ✓ Faceting & Filtering

Redis
Key-Value Store

- ✓ Social sharing
- ✓ Ratings/Reviews
- ✓ Messaging

PostgreSQL
Relational

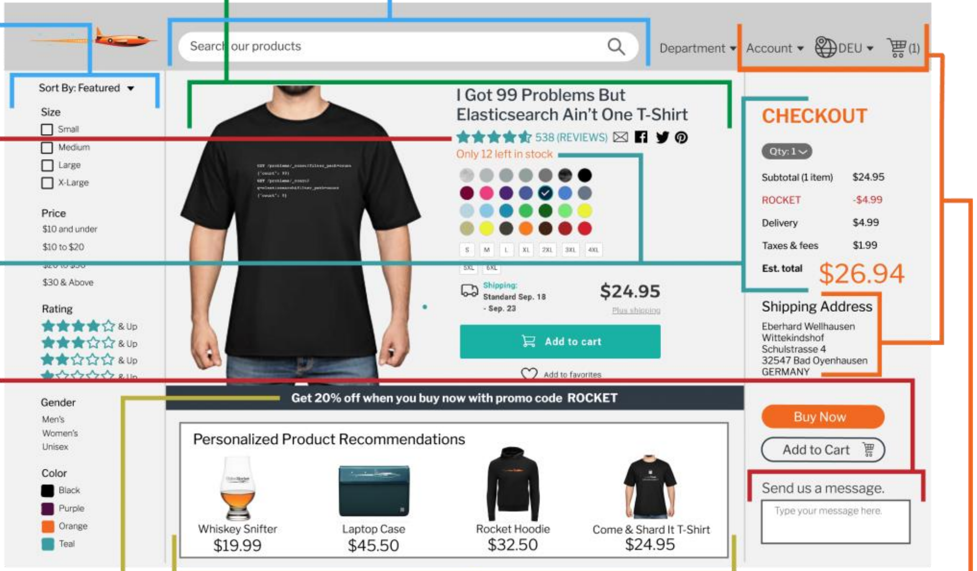
- ✓ Checkout
- ✓ Inventory
- ✓ Order Management

CockroachDB
Distributed SQL

- ✓ Geo-Partitioning for Low Latency & Compliance

Hadoop
Wide-Column Store

- ✓ Customized Product Recommendations
- ✓ Activity-Based Deals/Offers



* Courtesy of objectrocket

NoSQL-DBs - Examples

- BigTable
  - Google’s proprietary NoSQL system
  - Column-based or wide column store
- DynamoDB (Amazon)
  - Key-value data store
- Cassandra (Facebook)
  - Uses concepts from both key-value store and column-based systems
- MongoDB and CouchDB
  - Document stores
- Neo4J and GraphBase
  - Graph-based NoSQL systems
- OrientDB
  - Combines several concepts
- Database systems classified on the object model
  - Or native XML model

Datenformate

- Die Datenformate, auf die zugegriffen werden soll, sind äußerst vielfältig
- Beispiele:
  - **JSON:** Messages, REST, IoT-Events, Measurements, Document Dbs
  - **XML:** SOAP
  - **RDF:** RDF etabliert sich zunehmend als neues ‘Meta-Format’
  - **CSV/Tabellen:** jegliche Formen (Excel, CSV, …)
  - **Unstrukturierte Textdokumente:** Bilder, PDF, Rohtext, …
  - **Semistrukturierte Textdokumente:** Log-Dateien, JSON, XML, HTML, …
  - **Proprietäre Formate:** spezialisierte Anwendungen (z.B. SPSS für Statistik)
  - **Datenbanken, Datawarehouse-Systeme**

... in conclusion

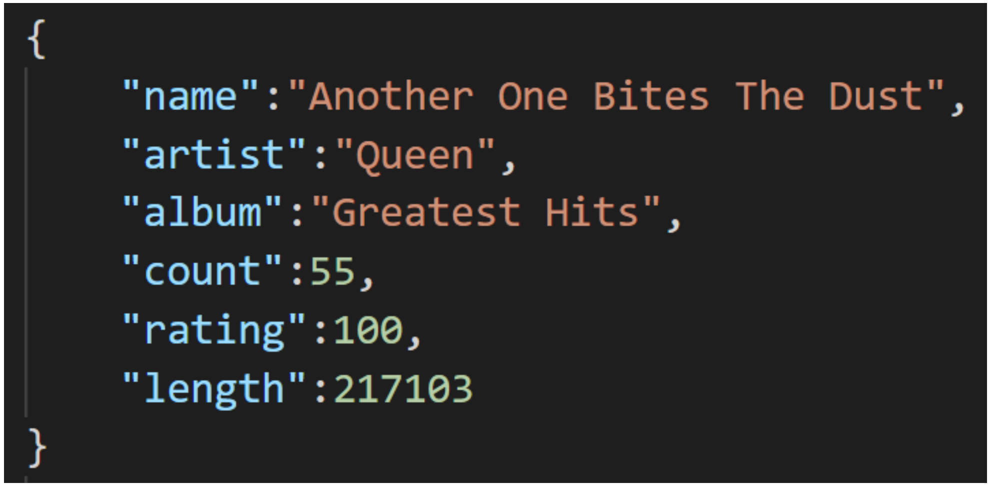
- Polyglot persistence: Multiple data sources and databases

- As Data Scientists we need to able to deal with that heterogeneity.
  We need to merge, join, aggregate data from different source in different formats. (data blending) – one of the biggest challenges today

- We’ll discover various approaches and technologies allowing us to deal with that challenge ☺

Music Store / Music Label

- 3rd party data from seller
  - JSON via api
- E.g. Apple Music, Spotify, YouTube Music,...
- Providing information:
  How often streamed? Ratings?
- Integrate Data in our Database



```json
{
    "name":"Another One Bites The Dust",
    "artist":"Queen",
    "album":"Greatest Hits",
    "count":55,
    "rating":100,
    "length":217103
}
```

Think for a moment

- Our goal is to combine our different data sources in order to run some evaluations:
  - How much to bill the 3rd party seller?
  - Who the best reviewed artists are?
  - There's data about other tracks and albums. We would like to figure out which artists we should sign in addition to existing ones.
  - .....
- What options do we have?

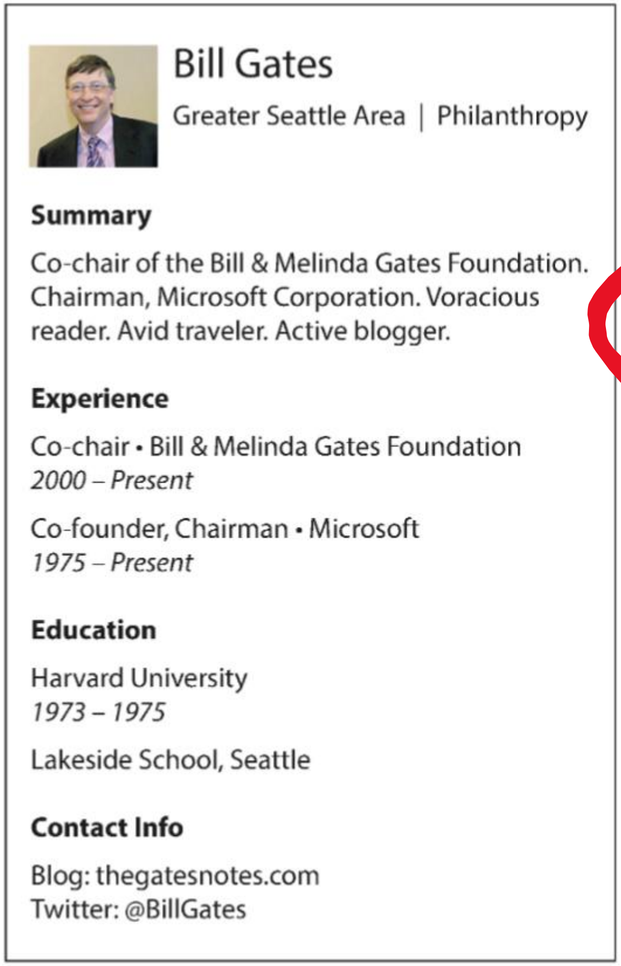
Options

- Write Data into a Kafka Stream
- Build an ETL Pipeline (e.g. with Talend)
- Write custom software and join + process data in memory and generate report. (e.g.: python + pandas,..)
- Transform JSON and store as table and process in SQL
- Export necessary data from SQL-DB and combine in a new db with JSON
- ...

- Many options available and we want to explore various options with you ☺

Data Model and Relationships

http://www.linkedin.com/in/williamhgates



Bill Gates

Greater Seattle Area | Philanthropy

Summary

Co-chair of the Bill & Melinda Gates Foundation. Chairman, Microsoft Corporation. Voracious reader. Avid traveler. Active blogger.

Experience

Co-chair • Bill & Melinda Gates Foundation
2000 – Present

Co-founder, Chairman • Microsoft
1975 – Present

Education

Harvard University
1973 – 1975

Lakeside School, Seattle

Contact Info

Blog: thegatesnotes.com
Twitter: @BillGates

Traditional SQL:

- Separate tables, with a foreign key reference

Structured datatypes, JSON/XML[1]:

- Multi-Valued data in single row
- Querying, indexing inside those documents

JSON or XML document:

- Encode and store as a self-contained document
- Applications to interpret its structure and content
- better locality than the multi-table schema

[1] Supported by: Oracle, IBM DB2, MSSQL Server, PostgreSQL, MySQL

Ingest JSON in PostgreSQL

- Various options available. Again, right choice does depend on your use case, company, infrastructure,...

- We want to like to proceed with the most practical solution and leverage existing functionality (= most often the best way to go)

- NOT limited to Postgres, option available in nearly every other RDBMS. In fact RDBMS are becoming more competitive towards DocumentDBs!

JSON in PostgreSQL

- JSONB type stores parsed JSON densely
  - saving space
  - making indexing more effective
  - efficient make query / retrieval

  (The "B" stands for "better“, historical alternatives JSON or HSTORE)

- It is largely what you’d expect from a JSON datatype
- Powerful operations

Still few situations where HSTORE or JSON has an advantage over JSONB
You can research that question online

JSON Operators

Operator	Right Operand Type	Description	Example
->	int	Get JSON array element	'[1,2,3]'::json->2
->	text	Get JSON object field	'{"a":1,"b":2}'::json->'b'
->>	int	Get JSON array element as text	'[1,2,3]'::json->>2
->>	text	Get JSON object field as text	'{"a":1,"b":2}'::json->>'b'
#>	array of text	Get JSON object at specified path	'{"a":[1,2,3],"b":[4,5,6]}'::json#>'{a,2}'
#>>	array of text	Get JSON object at specified path as text	'{"a":[1,2,3],"b":[4,5,6]}'::json#>>'{a,2}'

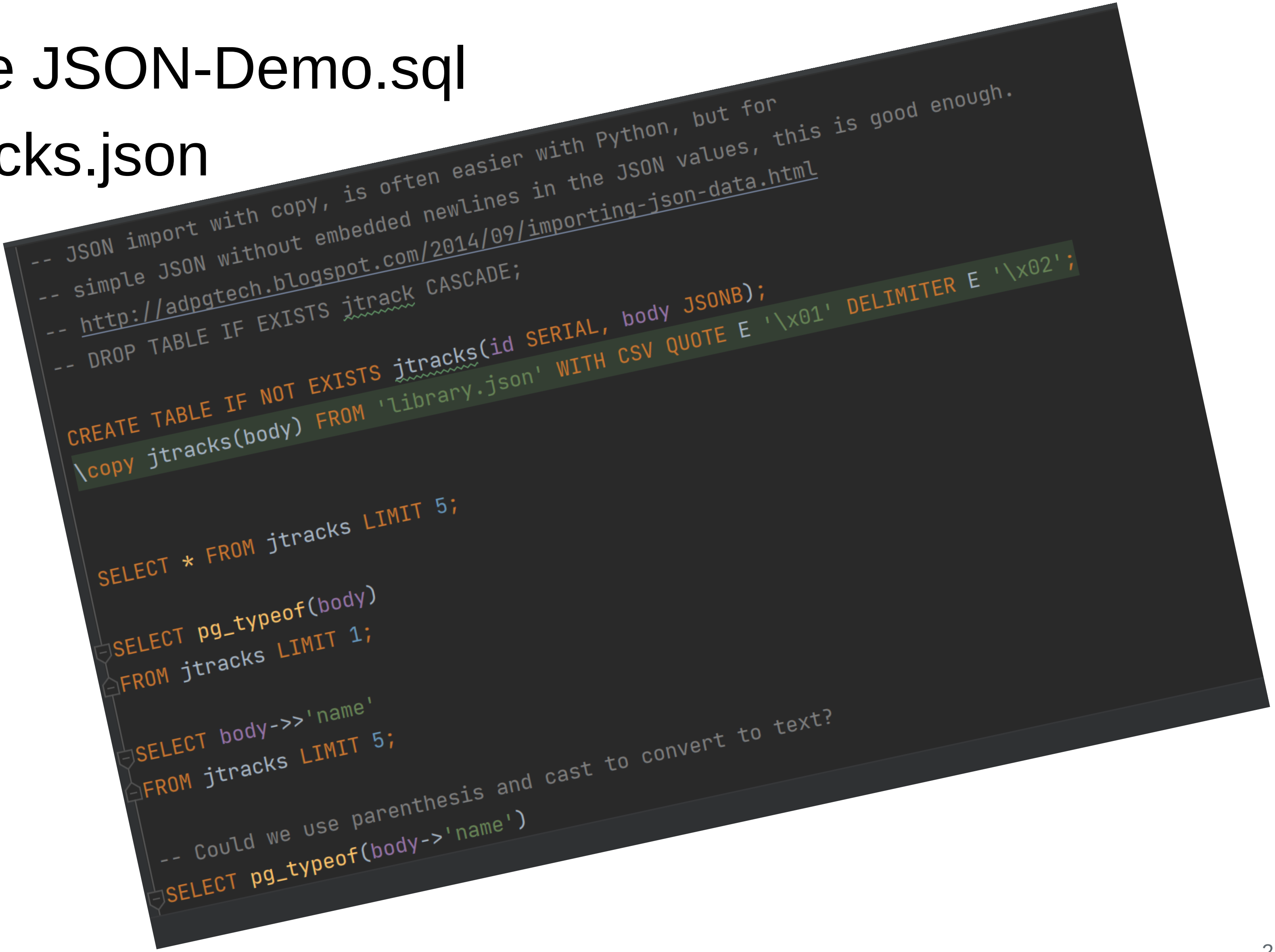
- https://www.postgresql.org/docs/9.3/functions-json.html

Additional operators

Operator	Right Type	Description	Example
@>	jsonb	Does the left JSON value contain within it the right value?	'{"a":1, "b":2}'::jsonb @> '{"b":2}'::jsonb
<@	jsonb	Is the left JSON value contained within the right value?	'{"b":2}'::jsonb <@ '{"a":1, "b":2}'::jsonb
?	text	Does the key/element string exist within the JSON value?	'{"a":1, "b":2}'::jsonb ? 'b'
?|	text[]	Do any of these key/element strings exist?	'{"a":1, "b":2, "c":3}'::jsonb ?| array['b', 'c']
?&	text[]	Do all of these key/element strings exist?	'["a", "b"]'::jsonb ?& array['a', 'b']

Demo

- See JSON-Demo.sql
- jtracks.json



```
-- JSON import with copy, is often easier with Python, but for
-- simple JSON without embedded newlines in the JSON values, this is good enough.
-- http://adpgtech.blogspot.com/2014/09/importing-json-data.html
-- DROP TABLE IF EXISTS jtrack CASCADE;

CREATE TABLE IF NOT EXISTS jtracks(id SERIAL, body JSONB);
\copy jtracks(body) FROM 'library.json' WITH CSV QUOTE E '\x01' DELIMITER E '\x02';


SELECT * FROM jtracks LIMIT 5;

SELECT pg_typeof(body)
FROM jtracks LIMIT 1;

SELECT body->>'name'
FROM jtracks LIMIT 5;

-- Could we use parenthesis and cast to convert to text?
SELECT pg_typeof(body->'name')
```

Fetch data from a RestAPI

- json: https://docs.python.org/3/library/json.html
- requests: https://docs.python-requests.org/en/latest/
- psycopg: https://www.psycopg.org/