Resource Oriented Architectures

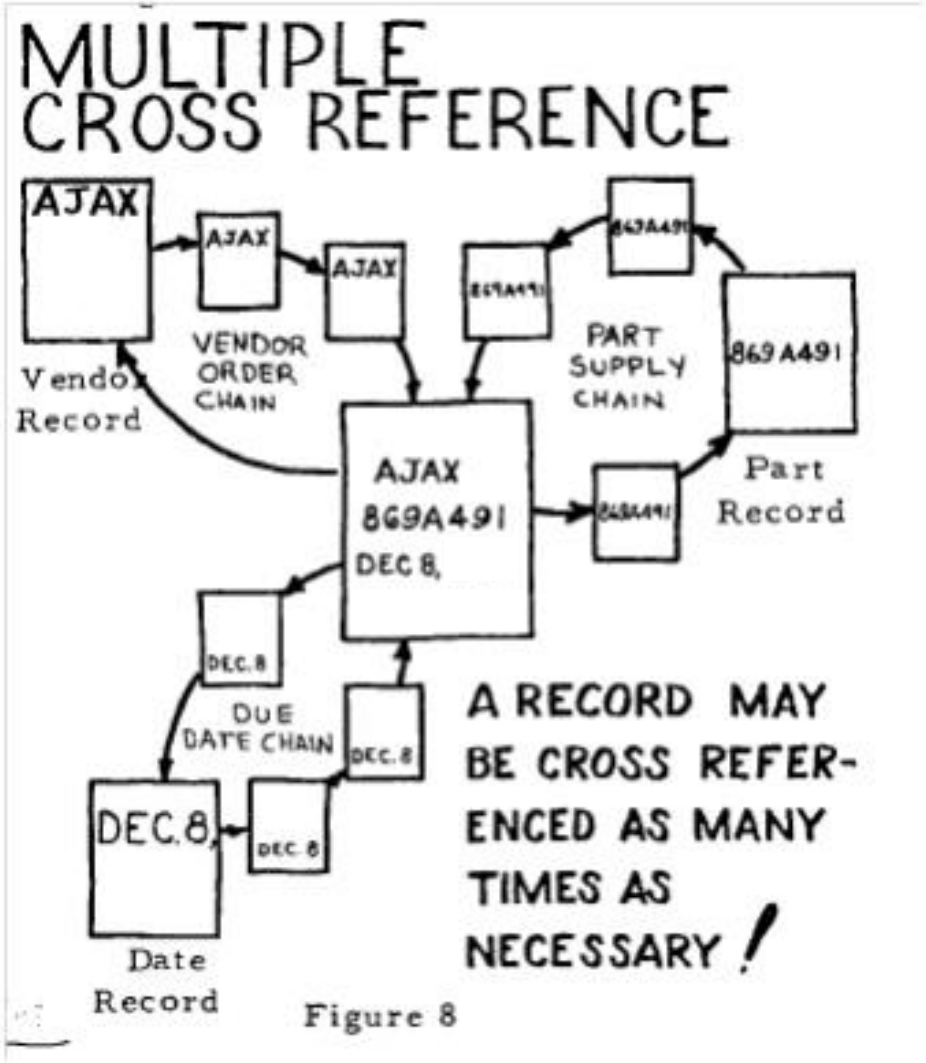
The Future of Data API?

@agilevic

How old do you think this is?

Drawing of an information processing system shows the use of chains to connect record.

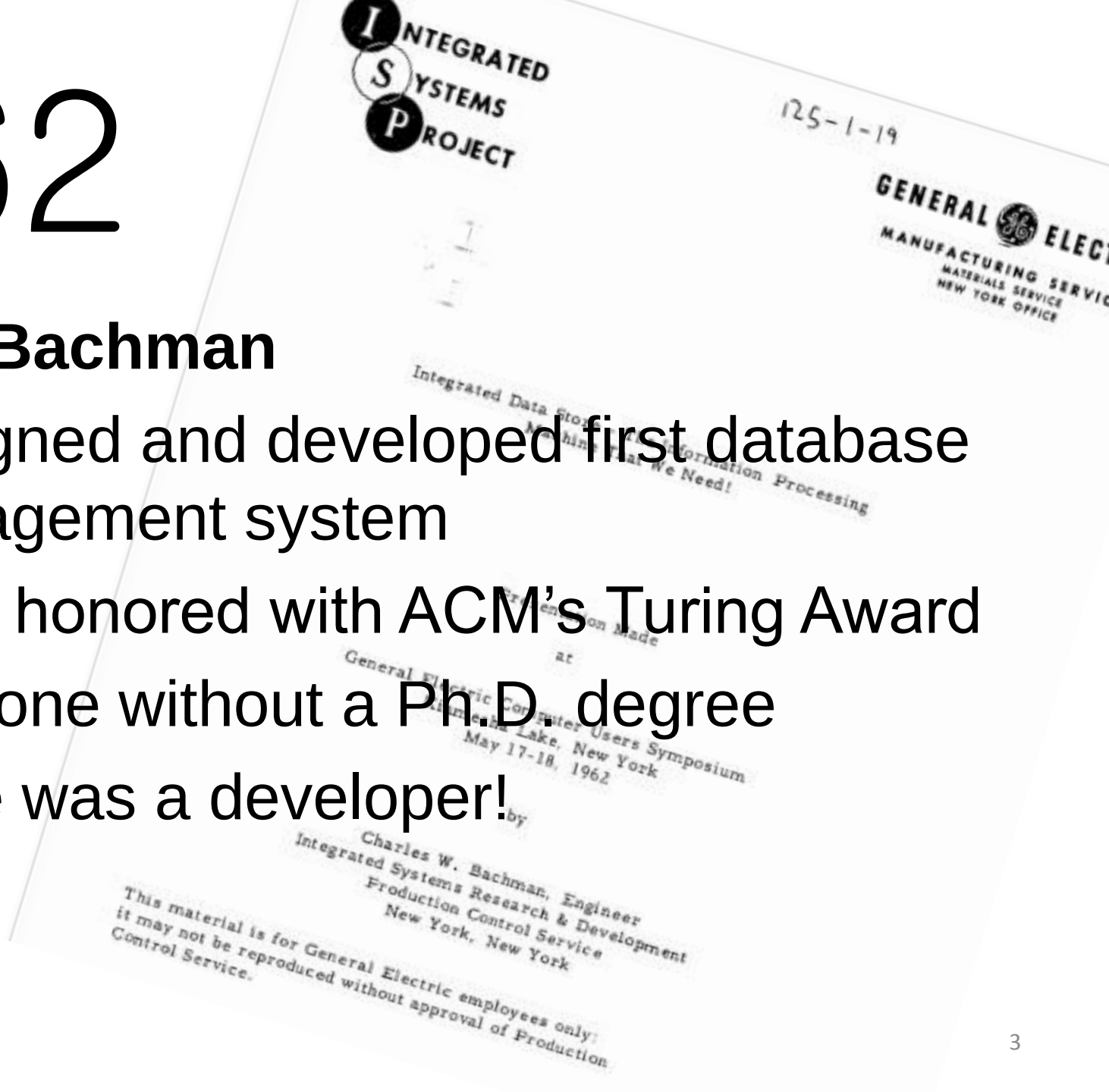
"The programmer used GET commands to navigate between related records."



1962

Charles Bachman

- Designed and developed first database management system
- Later honored with ACM's Turing Award
- First one without a Ph.D. degree
- Dude was a developer!



GE's Integrated Data System

- Introduced what would later be called the "network data model."
- Graph rather than hierarchy.
- "Programmer as Navigator," charting a path through the database from one record to another.

Navigational Data Systems

- In 1970s commercial systems based on this approach were some of the most successful products.
- Commercial relational systems have taken over since the late 1980s.
- In 1989 Sir Tim Berners-Lee writes a draft proposal for World Wide Web
  - Talks about access to existing databases too

Resource Oriented Architectures
Linked Data, Semantic Web
REST, HATEOS, RDF, SPARQL,
SQL, NoSQL

What is Resource Oriented Architecture

- “Style of software architecture and programming paradigm for designing and developing software in the form of resources with RESTful interfaces.”
  – *Wikipedia*
- Uniform data access layer to all data assets in their unobstructed form for reading and writing in various representations. – *my take*

What is Resource Oriented Architecture

Service Oriented

- Represents Action
- Transaction, Unit of Work
- Message
- API controlled by functional design
- Harder to adapt and scale beyond “enterprise”
- Harder to deprecate functionality

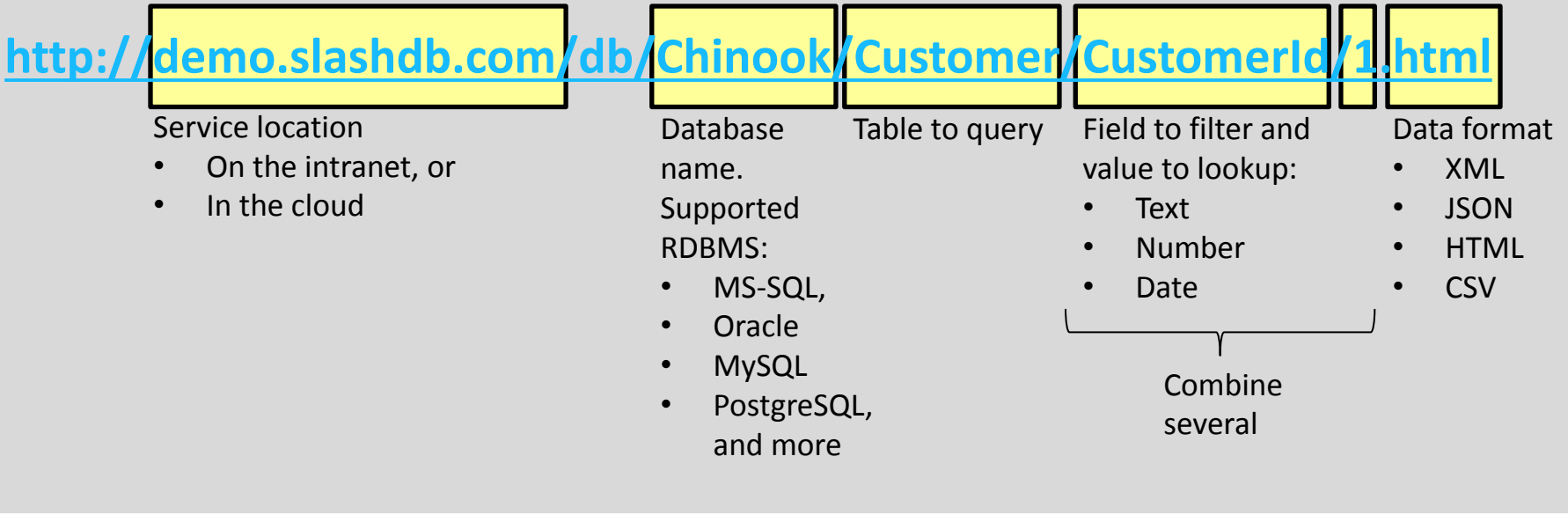
Resource Oriented

- Represents State
- Addressable Resource
- Update to Resource
- API automatically evolves with data
- Harder to model into complex transactions
- Clients must be resilient to change

Best Practices

- Don’t forget about “R” in REST
  - JSON isn’t the only data format
- URL should be easy to understand
  - Avoid inventing mini-query language
- Resources should be easy to discover
- Ideally every resource address should allow reading and writing
- Avoid query string to address data

Database Content as HTTP Resources



- **`/db`** automatically makes hyperlinks directly to data
- Related records are hyperlinked thus search engine ready
- Filtering, drill-down, slices are natural, URLs stay nice
- Custom queries also possible (SQL Pass-thru)

URLs?
Where we're going
We don't need URLs!

Linked Data, Semantic Web

- Resource Description Framework
- Web Ontology Language
- RDF/XML ~ 2004
- RDF/JSON 13 August 2013 (abandoned)
- JSON-LD 1.0 10 September 2013
  - W3C Recommendation
- Will it take off this time?

Get **Web Almanac 2013-2053!**

SPARQL Query Types

- **SELECT** - get raw values from a SPARQL endpoint in a table format.
- **CONSTRUCT** - extract information from the SPARQL endpoint and transform the results into valid RDF.
- **ASK** - a simple True/False result.
- **DESCRIBE** – get an RDF graph, the contents of which is left to the endpoint to decide based on what the maintainer deems as useful information.

SPARQL

```
# prefix declarations
PREFIX foo: <http://example.com/resources/>
...
# dataset definition
FROM ...
# result clause
SELECT ...
# query pattern
WHERE {
    ...
}
# query modifiers
ORDER BY ...
```


Thank You!
OUTATIME
Keep in Touch
@agilevic
victor@slashdb.com
SlashDB

References

- Linked Data the Story So Far
  http://tomheath.com/papers/bizer-heath-berners-lee-ijswis-linked-data.pdf
- Fifty Years of Databases
  http://wp.sigmod.org/?p=688
- SPARQL by Example
  http://www.cambridgesemantics.com/semantic-university/sparql-by-example
- Resource Oriented Architecture (Wikipedia)
  http://en.wikipedia.org/wiki/Resource_oriented_architecture
- Back to the Future (picture)
  http://www.redcarpetcrash.com/wp-content/uploads/2013/01/back.jpg
- Back to the Future II (picture)
  http://www.bttf.net/v/vspfiles/photos/ADVANCE-C-2.jpg
- Outatime (picture)
  http://2.bp.blogspot.com/-AnHNMztbj8o/TpXyXL2CmaI/AAAAAAAADGg/QZzORg_4l9o/s1600/outatime.jpg