# XML and NoSQL

## data from documents

What is XML

What is NoSQL

Tutorial 1: Handling XML with Python

Tutorial 2: BaseX: an XML database

# XML
## extensible markup language

**Extensible**
can be used to represent (almost) any type of data

**Markup**
Provides data about the data

**Language**
...uses words?

# Uses of XML

Word documents (.doc, .docx)

Powerpoints and Excel files

Images (.svg)

RSS and XHTML

Many proprietary uses

# When to use XML?

If human readability is important

If data takes “tree” shape naturally

If data can be described as a “document”

# XML and Data Science

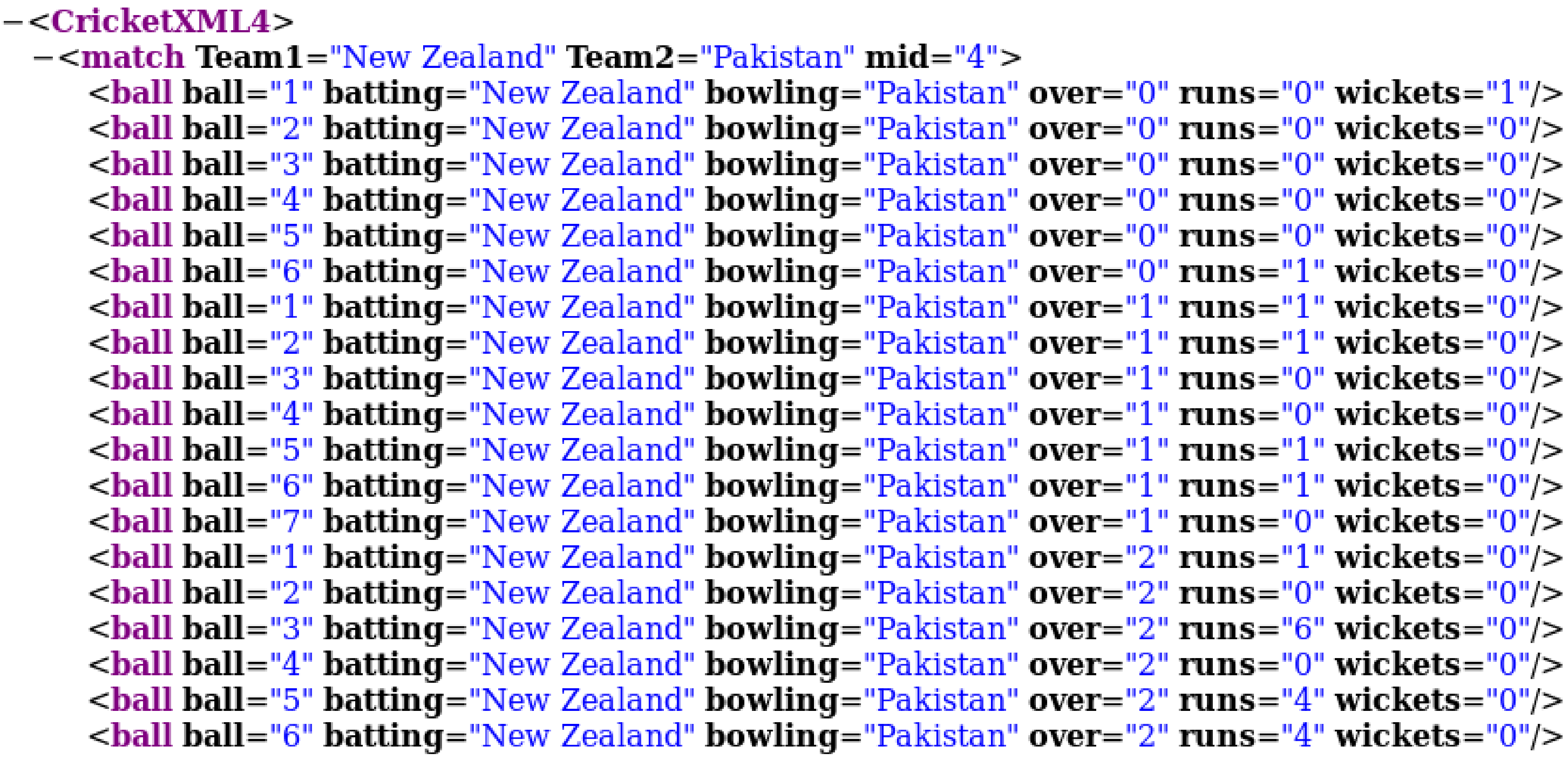
XML is unfriendly to data scientists

Format is unregulated

Trees are unpredictable

Data does not take “matrix” format

# What does XML look like?



```
–<CricketXML4>
  –<match Team1="New Zealand" Team2="Pakistan" mid="4">
      <ball ball="1" batting="New Zealand" bowling="Pakistan" over="0" runs="0" wickets="1"/>
      <ball ball="2" batting="New Zealand" bowling="Pakistan" over="0" runs="0" wickets="0"/>
      <ball ball="3" batting="New Zealand" bowling="Pakistan" over="0" runs="0" wickets="0"/>
      <ball ball="4" batting="New Zealand" bowling="Pakistan" over="0" runs="0" wickets="0"/>
      <ball ball="5" batting="New Zealand" bowling="Pakistan" over="0" runs="0" wickets="0"/>
      <ball ball="6" batting="New Zealand" bowling="Pakistan" over="0" runs="1" wickets="0"/>
      <ball ball="1" batting="New Zealand" bowling="Pakistan" over="1" runs="1" wickets="0"/>
      <ball ball="2" batting="New Zealand" bowling="Pakistan" over="1" runs="1" wickets="0"/>
      <ball ball="3" batting="New Zealand" bowling="Pakistan" over="1" runs="0" wickets="0"/>
      <ball ball="4" batting="New Zealand" bowling="Pakistan" over="1" runs="0" wickets="0"/>
      <ball ball="5" batting="New Zealand" bowling="Pakistan" over="1" runs="1" wickets="0"/>
      <ball ball="6" batting="New Zealand" bowling="Pakistan" over="1" runs="1" wickets="0"/>
      <ball ball="7" batting="New Zealand" bowling="Pakistan" over="1" runs="0" wickets="0"/>
      <ball ball="1" batting="New Zealand" bowling="Pakistan" over="2" runs="1" wickets="0"/>
      <ball ball="2" batting="New Zealand" bowling="Pakistan" over="2" runs="0" wickets="0"/>
      <ball ball="3" batting="New Zealand" bowling="Pakistan" over="2" runs="6" wickets="0"/>
      <ball ball="4" batting="New Zealand" bowling="Pakistan" over="2" runs="0" wickets="0"/>
      <ball ball="5" batting="New Zealand" bowling="Pakistan" over="2" runs="4" wickets="0"/>
      <ball ball="6" batting="New Zealand" bowling="Pakistan" over="2" runs="4" wickets="0"/>
```

# Parts of XML File

```
<element attribute=”value”>
    <subelement attribute=”value”> content </subelement>
    <subelement attribute=”value”> content </subelement>
        <subsubelement attribute=”value” attribute2=”value”/>
    <subelement attribute=”value”> content </subelement>
</element>
```

# Parts of XML File

```
<email sender=”jwolohan@indiana.edu” date=”June. 1, 2017”>
  <subject> Assignment 7 Question </subject>
  <body> Professor Luciano, I didn’t complete assignment 7 on XML – would
I still be able to get credit for turning it in late? I’m really enjoying the course
so far! And your module on Linked Data was great! </body>
</email>
```

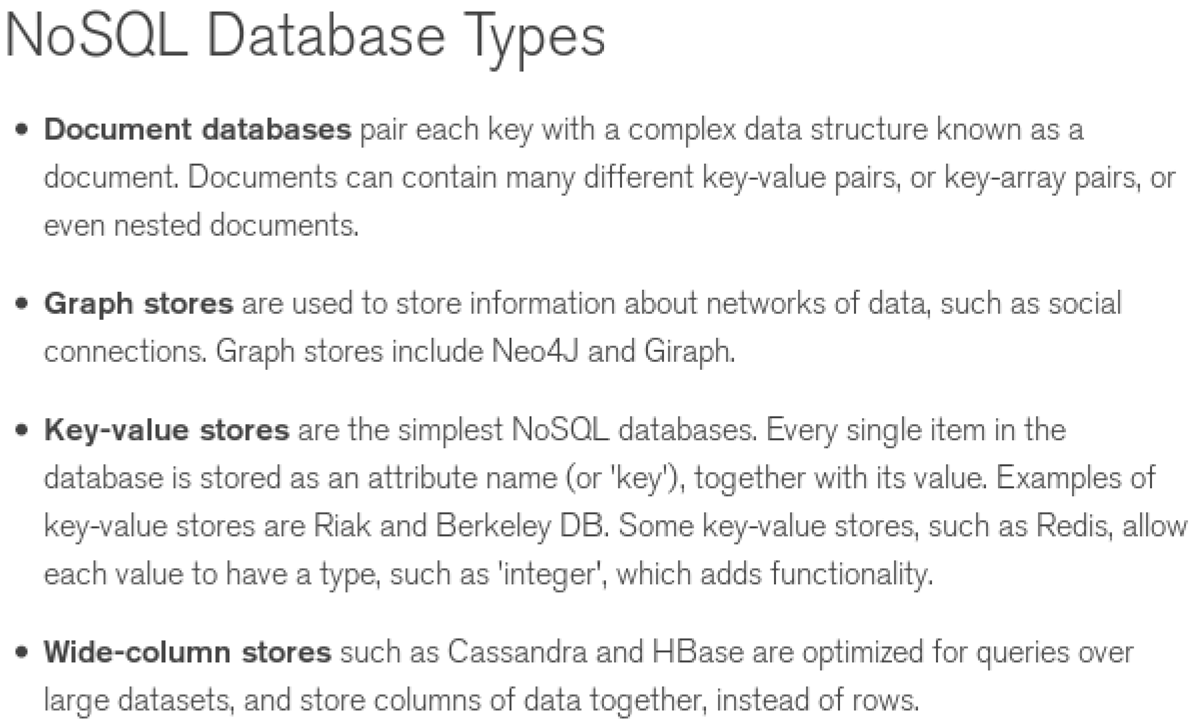
# Parts of XML File

```
<email>
  <sender> jwolohan@indiana.edu </sender>
  <date>June. 1, 2017</date>
  <subject> Assignment 7 Question </subject>
  <body> Professor Luciano, I didn’t complete assignment 7 on XML – would I still be able to get credit for turning it in late? I’m really enjoying the course so far! And your module on Linked Data was great! </body>
</email>
```

That seems messy.

# NoSQL

## Databases for “documents” (ish)



### NoSQL Database Types

- **Document databases** pair each key with a complex data structure known as a document. Documents can contain many different key-value pairs, or key-array pairs, or even nested documents.
- **Graph stores** are used to store information about networks of data, such as social connections. Graph stores include Neo4J and Giraph.
- **Key-value stores** are the simplest NoSQL databases. Every single item in the database is stored as an attribute name (or 'key'), together with its value. Examples of key-value stores are Riak and Berkeley DB. Some key-value stores, such as Redis, allow each value to have a type, such as 'integer', which adds functionality.
- **Wide-column stores** such as Cassandra and HBase are optimized for queries over large datasets, and store columns of data together, instead of rows.

**Manage data with different, changing, or loose structures**

# Popular NoSQL Databases

Cassandra

SAP HANA

Apache CouchDB

IBM Domino

MongoDB

Oracle NoSQL

Berkley DB

# XML Databases

Structured query language for unstructured data

**Technologies**: XPath and XQuery

**Implementations**: BaseX and BerkeleyDB

# Recap

Not all data comes in tables

XML is useful, popular way of representing “messy” data

NoSQL databases help us manage complexity