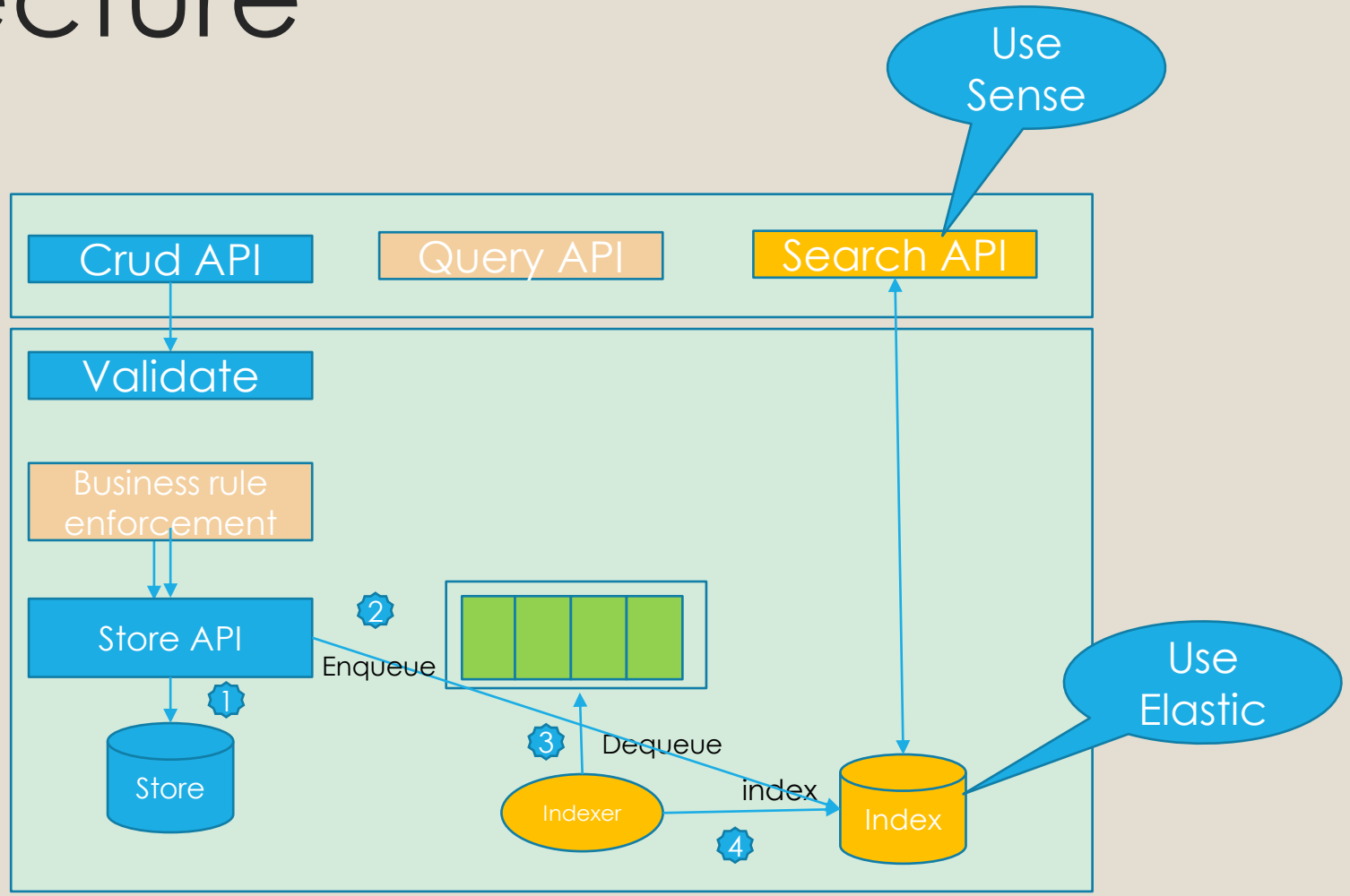
ADVANCED TOPIC IN BIG DATA

10/22/16

Architecture



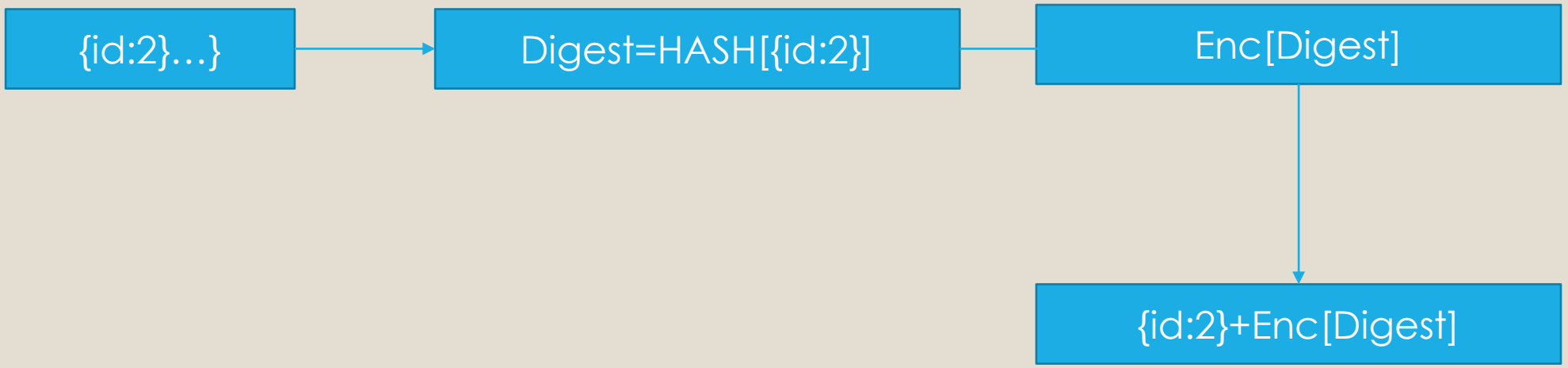
Prototype outline:

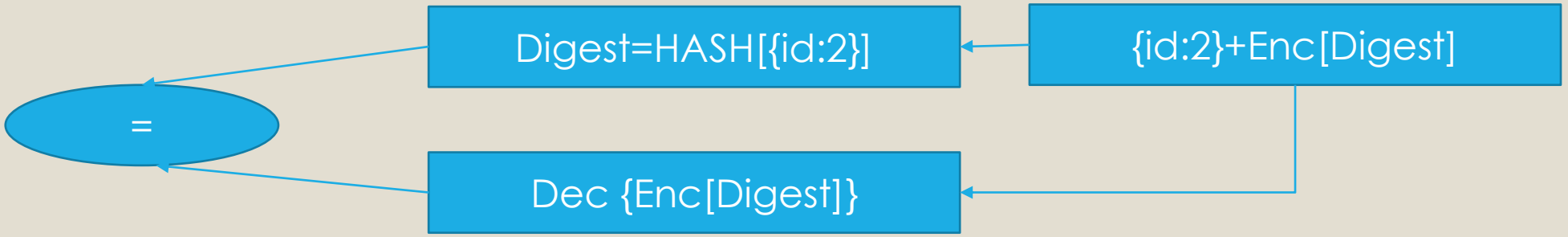
- Rest API that can handle any structured data in Json
- Rest API with support for crud operations, including merge support, cascaded delete
- Rest API with support for validation
- Json Schema describing the data model for the use case
- Advanced semantics with rest API operations such as update if not changed
- Storage of data in key/value store
- Search with join using Elastic
  - Parent-Child indexing
- Queueing
- Security

Catching up on old topics

- Implementing merge semantics
- Etag implementations
- Security implementation progress
  - Revisit security implementation using tokens

Asymmetric Crypto





Join and elastic search

- How to implement to join with elastic search?
  - Pros
  - Cons

- Demonstrate this approach using sense

http://fideloper.com/api-etag-conditional-get
They use MD5 having to calculate the Etag.
http://stackoverflow.com/questions/415953/how-can-i-generate-an-md5-hash

Fulltext search

- Basic concepts:
  - Indexing:
    - Is the process of creating an index.
    - An index is defined as a collection of fields. Each field can be either single value/multivalued, have a type, stored, indexed, required, can be associated with different tokenizer's/analyzers
    - Dynamic fields is a very useful feature
    - An index contains a collection of documents.
    - A document is a collection of property (field) / value pairs
  - Searching
    - Is the process of discovering a document in an index that meets certain criteria's
    - the criterias are specified using fields that are found in a document

Query samples

- find all documents containing name:jeff
- find all documents containing name:jeff and age:30 (Or any other logical relation, e.g. or, not, and)
- find all documents created after 9-16-2016
- find all documents of type plan
- find all documents of type pla*; E.g.; plans, planning, planner etc.
- Find all the Unique terms of the field "type" in the system
- Counts:
  - how many times a certain value occurs in the index

- Aggregates:
  - Max, Min, Average, Sum, percentiles, etc.

- How many cameras are on sale between 50 and $100?

Faceted queries

- Is the bucketing of search results into buckets based on terms in the index
- Useful for determining the unique terms for a field and returns a count for each of those terms.
- Makes it easy to explore search results
- Faceting example is found here:
  - https://lucidworks.com/post/faceted-search-with-solr/

Faceting..

- Field faceting – retrieve the counts for all terms, or just the top terms in any given field. The field must be indexed.
- Query faceting – return the number of documents in the current search results that also match the given query.
- Date faceting – return the number of documents that fall within certain date ranges.

Filter queries

- Used to filter the results of the previous query
  - Often used to implement drill down into search results
- When filter query is added to the previous query, its effect is to exclude results that do not match the filter

- Example:
  - Return all cameras by manufacturer and their count
    - /query?q=camera facet.field=manu
  - Return all cameras in this price range by manufacturer and their count
    - http://localhost:8983/solr/query?q=camera &facet.field=manu &**fq=price:[400 to 500] (fq is filter query)**

Elastic Search

- Getting started:
  - https://www.elastic.co/guide/en/elasticsearch/guide/current/getting-started.html

Homework

- Demonstrate an example of join queries using elastic search. This is due 10/22

References

- https://cwiki.apache.org/confluence/display/solr/About+This+Guide
- https://lucidworks.com/post/faceted-search-with-solr/https://www.elastic.co/guide/en/elasticsearch/guide/current/denormalization.html
- Getting started:
  - https://www.elastic.co/guide/en/elasticsearch/guide/current/getting-started.html