Advanced Database Design
Intro to Map-Reduce

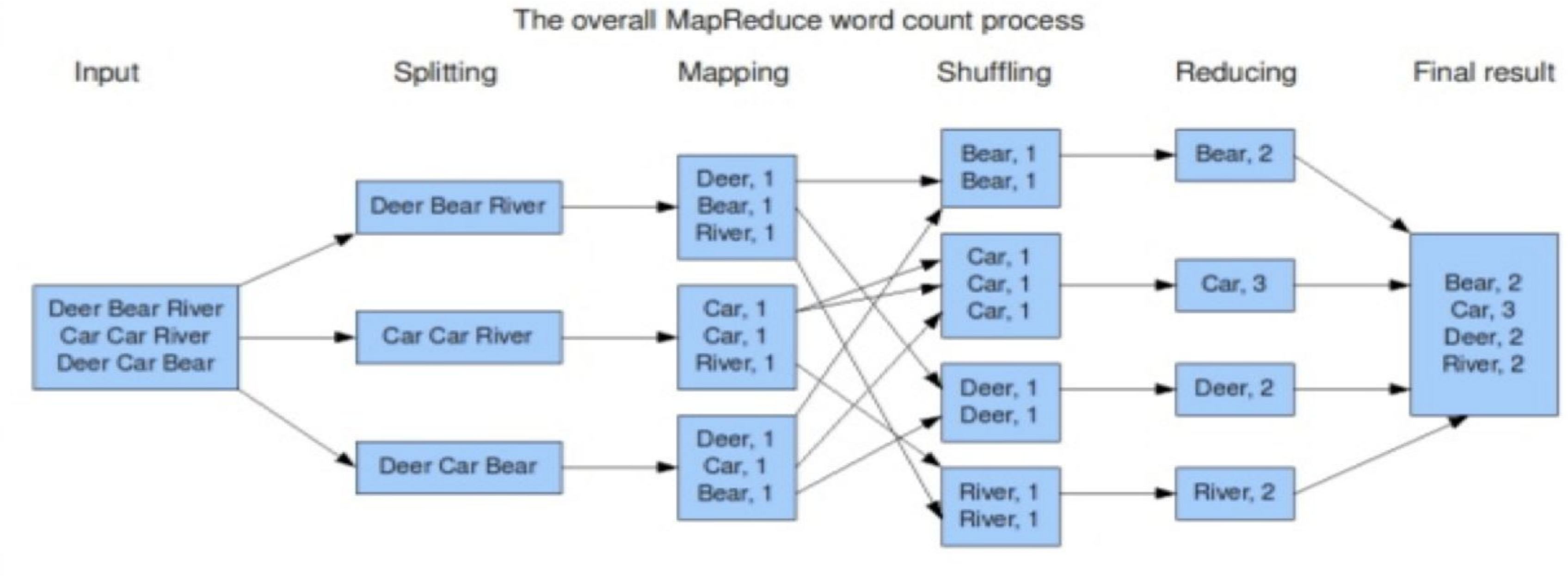
Gregory S DeLozier, Ph.D.

gdelozie@kent.edu

MapReduce

- Distributed query system
- Splits the whole search space into parts and distributes the parts
- *Maps* those parts onto partial solutions with keys
- Shuffles the elements of the solutions by key values
- *Reduces* all of the partial solutions with the same key into a key result
- Combines all of the key results into a final result

A MapReduce Example



Elements of Map/Reduce

Mapper

  Gets elements of a solution (lines, regions, groups, whatever)

  Emits Key/Value pairs

Reducer

  Gets Key,[Value,Value,Value] sets

  Emits Key/KeyResult pairs

Example Mapper

```
def mapper(self, _, line):
    for word in line.split():
        yield (word,1)
```

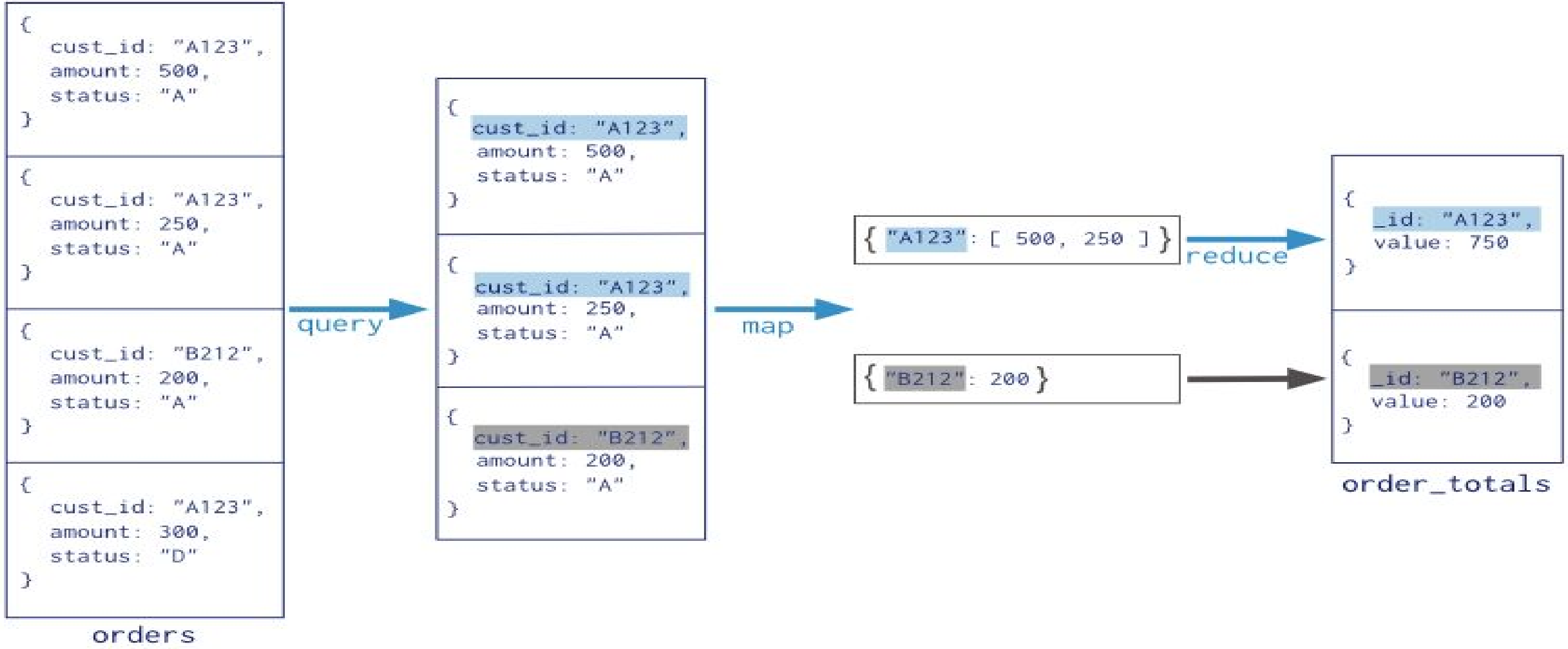
Example Reducer

```
def reducer(self, key, values):
    yield key, sum(values)
```

Python Demonstration

Demo Time!

MapReduce in Mongo



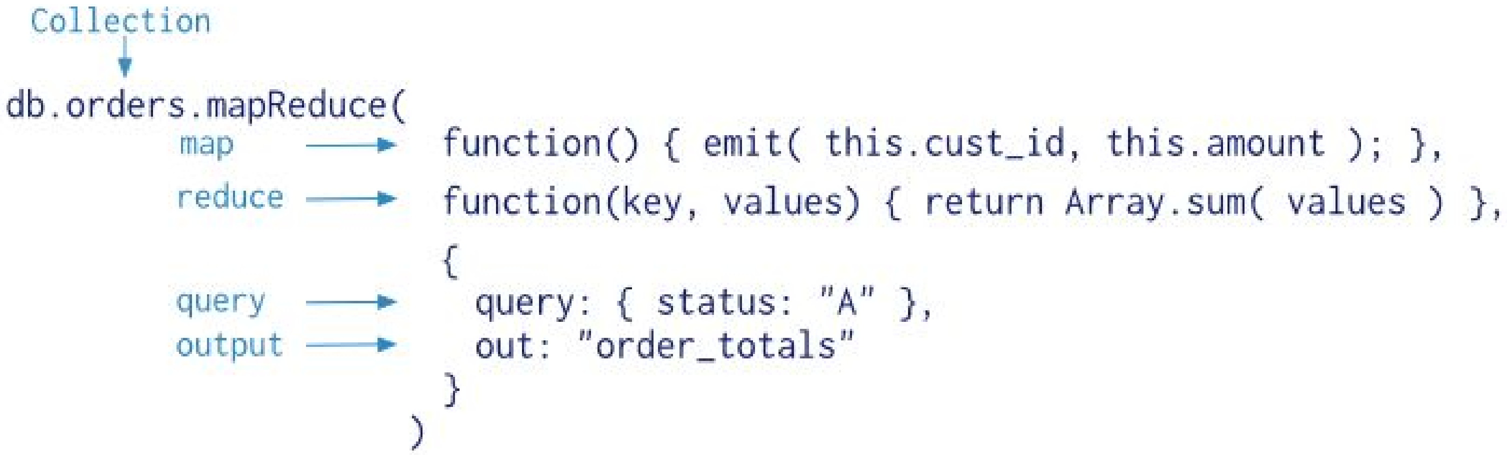
```
{
  cust_id: "A123",
  amount: 500,
  status: "A"
}
{
  cust_id: "A123",
  amount: 250,
  status: "A"
}
{
  cust_id: "B212",
  amount: 200,
  status: "A"
}
{
  cust_id: "A123",
  amount: 300,
  status: "D"
}
```

```
{
 cust_id: "A123",
 amount: 500,
 status: "A"
}
{
 cust_id: "A123",
 amount: 250,
 status: "A"
}
{
 cust_id: "B212",
 amount: 200,
 status: "A"
}
```

```
{ "A123": [ 500, 250 ] }
{ "B212": 200 }
```

```
{
  _id: "A123",
  value: 750
}
{
  _id: "B212",
  value: 200
}
```

Mongo MapReduce Command



```
db.orders.mapReduce(
                     function() { emit( this.cust_id, this.amount ); },
                     function(key, values) { return Array.sum( values ) },
                     {
                       query: { status: "A" },
                       out: "order_totals"
                     }
                   )
```

Mongo MapReduce in Python

```
>>> from bson.code import Code
>>> mapper = Code("""
...               function () {
...                 this.tags.forEach(function(z) {
...                   emit(z, 1);
...                 });
...               }
...               """)
```

Reducer Function in Python

```
>>> reducer = Code("""
...             function (key, values) {
...               var total = 0;
...               for (var i = 0; i < values.length; i++) {
...                 total += values[i];
...               }
...               return total;
...             }
...             """)
```

Running MapReduce

```
>>> result = db.things.map_reduce(mapper, reducer, "myresults")
>>> for doc in result.find():
...  print doc
```