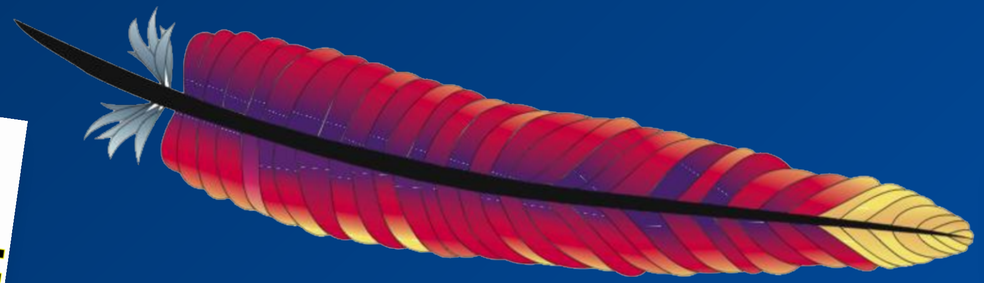
BRAND > code

Shane Curcuru, VP Brand Management, ASF

APACHE



SpamAssassin

Apache Solr

Apache OpenOffice™

APACHE felix

APACHE CORDOVA™

Cassandra

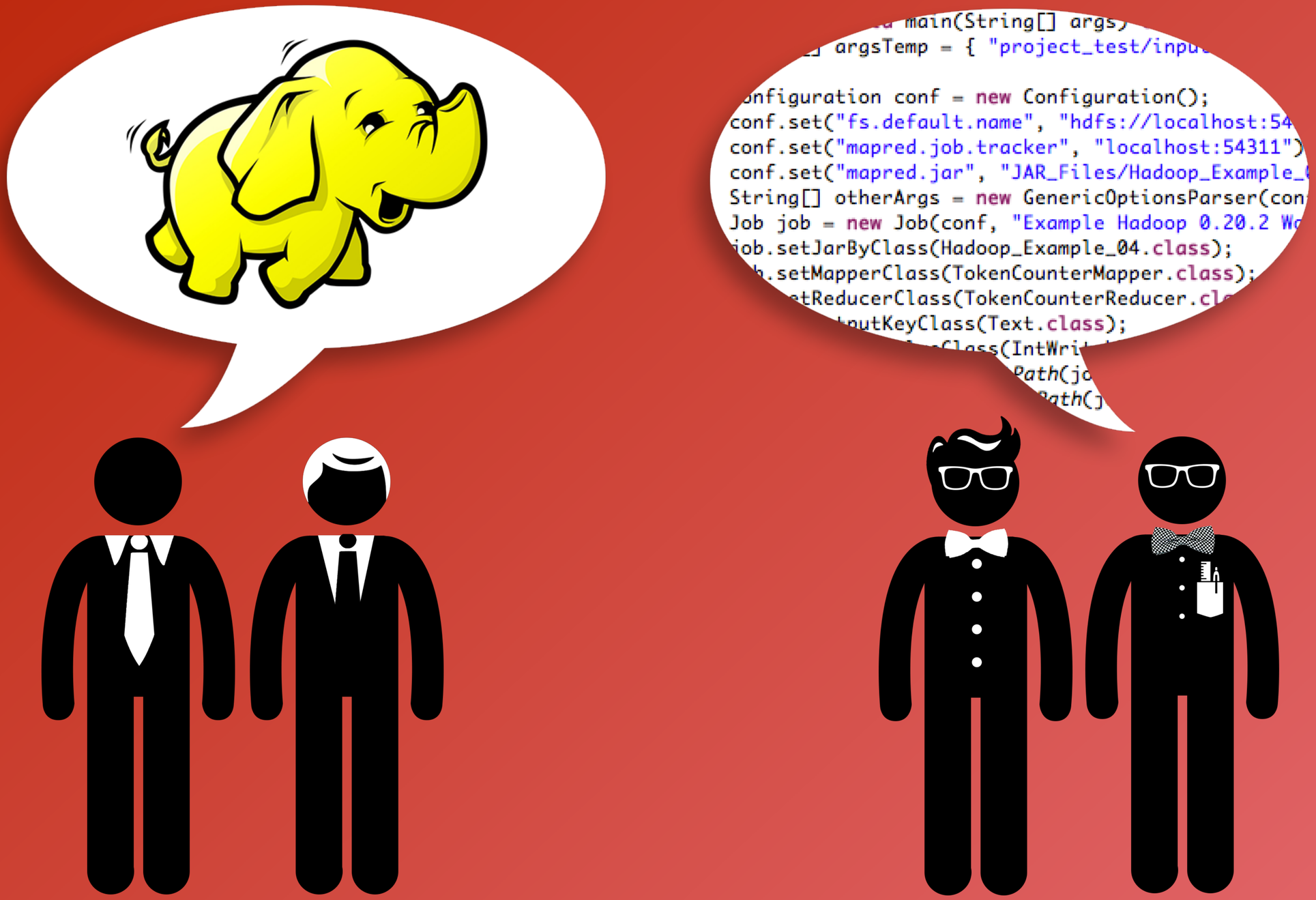
Struts

Apache etch

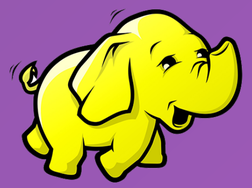
MESOS

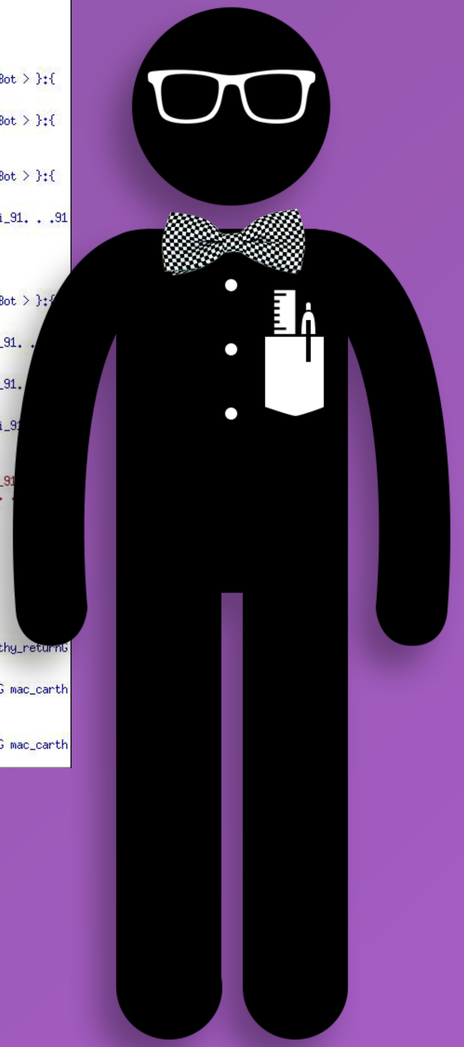
Spark


APACHE®
TM
SpamAssassin
hIVE
nutch
SUBVERSION
Apache Solr
AXIS2
Apache OpenOffice™
APACHE felix
APACHE CORDOVA™
Cassandra
buildr
APACHE HBASE
Struts
Apache etch
Apache Marmotta
mahout
apache Flex™
COCOON
MESOS
hadoop
cloudstack
open source cloud computing
chukwa
CouchDB
Spark
apache stanbo

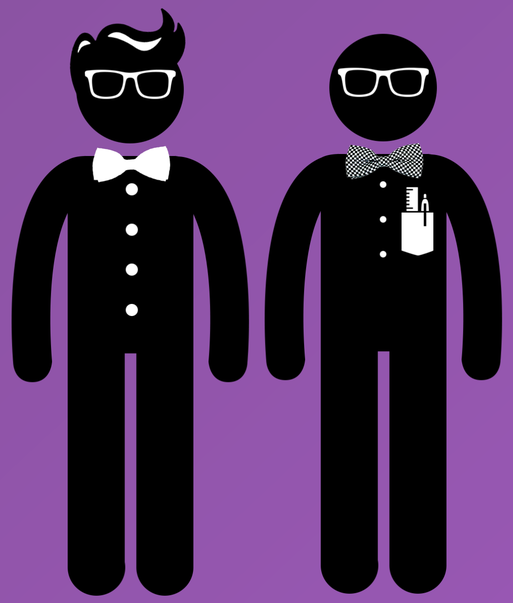

main(String[] args)
argsTemp = { "project_test/inpu
onfiguration conf = new Configuration();
conf.set("fs.default.name", "hdfs://localhost:54
conf.set("mapred.job.tracker", "localhost:54311")
conf.set("mapred.jar", "JAR_Files/Hadoop_Example_
String[] otherArgs = new GenericOptionsParser(con
Job job = new Job(conf, "Example Hadoop 0.20.2 Wo
iob.setJarByClass(Hadoop_Example_04.class);
.setMapperClass(TokenCounterMapper.class);
etReducerClass(TokenCounterReducer.cl
utKeyClass(Text.class);
Path(jo
ath(j

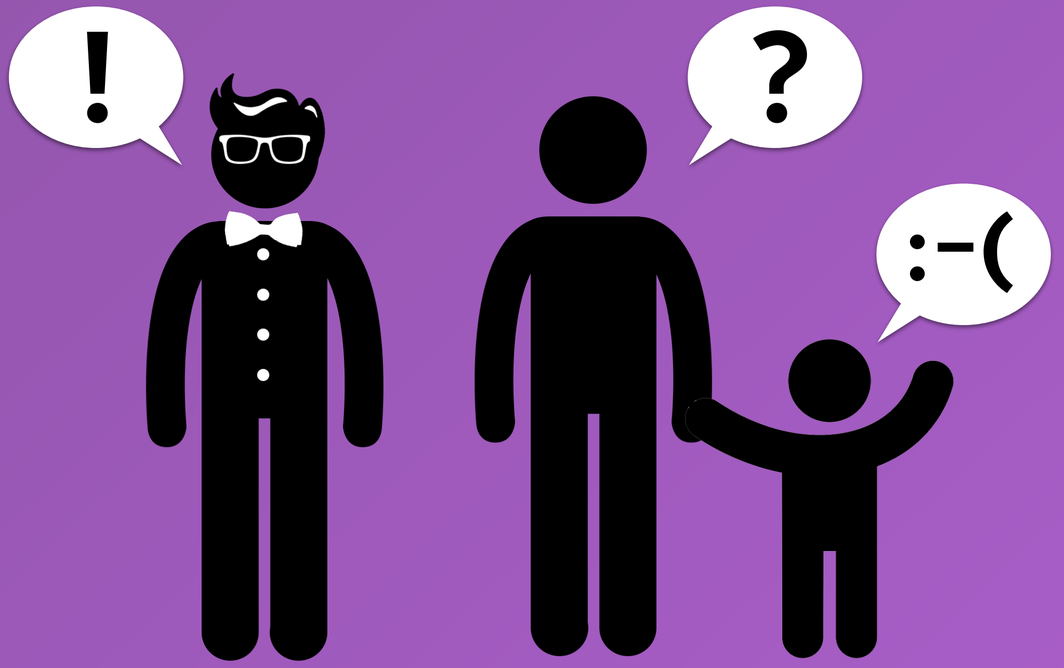
```
<7>
<8> int mac_carthy (int a)
<9> {
<10>   int b;
<11>

  <Union 223 envs> [mac_carthy:1:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .101.> }::{ <a,(Local),i_-10. . .111.>, <b,(Local), Bot > }:{
}:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
<12>   if (a > 100)
  <Union 223 envs> [mac_carthy:4:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .101.> }::{ <a,(Local),i_-10. . .111.>, <b,(Local), Bot > }:{
}:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
<13>     {

  <Union 223 envs> [mac_carthy:5:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .100.> }::{ <a,(Local),i_102. . .111.>, <b,(Local), Bot > }:{
}:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
<14>       return (a-10);
  <Union 223 envs> [mac_carthy:15:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .101.> }::{ <a,(Local),i_-10. . .111.>, <b,(Local),i_91. . .91
.> }:{  }:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
<15>     }
<16>   else
<17>     {

  <Union 223 envs> [mac_carthy:9:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .100.> }::{ <a,(Local),i_-10. . .100.>, <b,(Local), Bot > }:{
}:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
<18>       b = mac_carthy (mac_carthy (a+11));
  <Union 223 envs> [mac_carthy:14:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .91.> }::{ <a,(Local),i_-10. . .100.>, <b,(Local),i_91. .
}:{  }:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []

  <Union 223 envs> [mac_carthy:14:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .91.> }::{ <a,(Local),i_-10. . .100.>, <b,(Local),i_91.
:{  }:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
       return (b);
  nion 223 envs> [mac_carthy:15:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .101.> }::{ <a,(Local),i_-10. . .111.>, <b,(Local),i_9
:{  }:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []

 of function
 on 223 envs> [mac_carthy:15:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .101.> }::{ <a,(Local),i_-10. . .111.>, <b,(Local),i_91
{ <.x1,(Local),i_0. . .1.>, <.x2,(Local),i_100. . .100.>, <.x3,(Local),i_-10. . .-10.>, <.x4,(Local),i_10. . .10.>, <.x5,(Local),i_91. . .101.>, <.x6,(Local),i_1. . .
(Local),i_11. . .11.> }:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] [.x2L .x1L .x4L .x3L .x7L .x6L .x5L ]
    }
}


int main (int argc, char * argv[])
{
  int x;


  <Union 1 envs> [main:1:mac_carthy.c]::{ <mac_carthy_return,(Global), Bot >, <main,(Global), Bot > }::{ <x,(Local), Bot > }:{  }:{ No signal }:{ To do }[mainG mac_carthy_returnG
 ] [xL ] []
<28>   x = -10;
  <Union 1 envs> [main:3:mac_carthy.c]::{ <mac_carthy_return,(Global), Bot >, <main,(Global), Bot > }::{ <x,(Local),i_-10. . .-10.> }:{  }:{ No signal }:{ To do }[mainG mac_carth
y_returnG ] [xL ] []
<29>

  <Union 1 envs> [main:3:mac_carthy.c]::{ <mac_carthy_return,(Global), Bot >, <main,(Global), Bot > }::{ <x,(Local),i_-10. . .-10.> }:{  }:{ No signal }:{ To do }[mainG mac_carth
y_returnG ] [xL ] []
```






main(String[] args)
argsTemp = { "project_test/inpu
onfiguration conf = new Configuration();
conf.set("fs.default.name", "hdfs://localhost:54
conf.set("mapred.job.tracker", "localhost:54311")
conf.set("mapred.jar", "JAR_Files/Hadoop_Example_
String[] otherArgs = new GenericOptionsParser(con
Job job = new Job(conf, "Example Hadoop 0.20.2 Wo
job.setJarByClass(Hadoop_Example_04.class);
.setMapperClass(TokenCounterMapper.class);
etReducerClass(TokenCounterReducer.cla
utKeyClass(Text.class);
Path(jo
Path(j




!
?
:-(

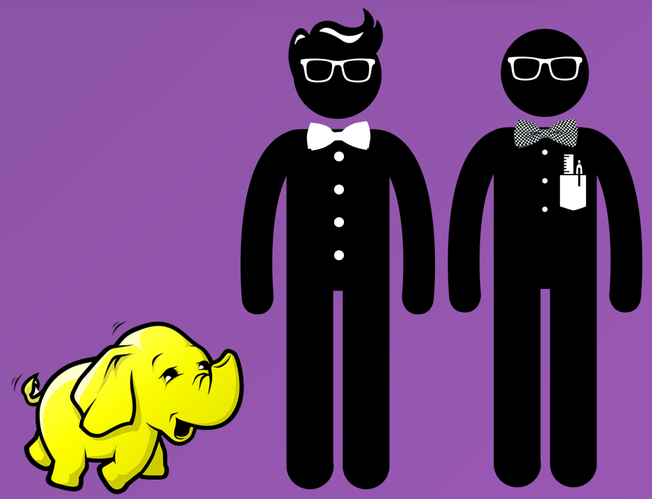
```
<7>
<8> int mac_carthy (int a)
<9> {
<10>   int b;
<11>
```

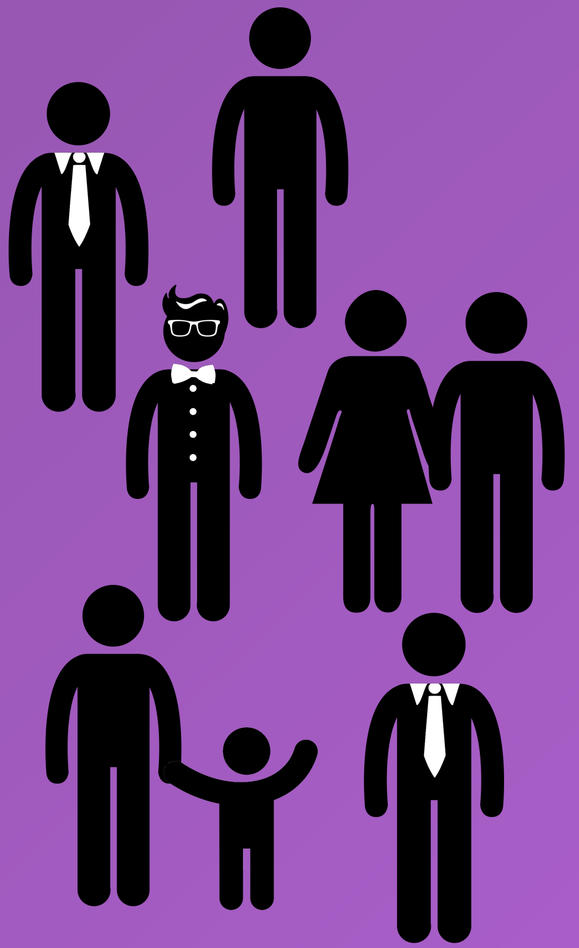
```
Start_encoding_picture:      // Begin encoding a video picture
      input desired D;       // Get the desired Distortion D value
      find D_Q nearest to D;  // Find the D value closes to the desired D
      Qnorm = h(D_Q);        // Determine normal Q with no masking
      lambda = f(Qnorm);     // Determine the Lagrange multiplier lambda
start_encoding_pixelblock:   // Begin encoding a pixelblock from the picture
      Q = Qnorm;             // Set Q to the normal Q with no masking
      calculate visual mask M;  // Determine the visual masking amount
      while(M-lambda > Lambda_max(Q) ){ // if strong masking, increase Q
            Q = Q+deltaQ;          // Raise the Quantizer Q size
      }
      code pixelblock( M-lambda, Q ); // Encode using M-lambda and Q
      if (encoder_buffer > Thulf(      // If buffer threatens to fill overflow
            lambda = lambda+deltalambda;        // Increase lambda
            if ( lambda > Lambda_max(Qnorm) ){   // Test lambda
                  Qnorm=Qnorm+deltaQ; //Increase Q if lambda too big
                  lambda = f(Qnorm);        // Calculate new lambda
            }
      }
      if (encoder_buffer < Tempty){    // If buffer threatens to fill underflow
            lambda = lambda-deltalambda;        // Decrease lambda
            if ( lambda < Lambda_min(Qnorm) ){   // Test lambda
                  Qnorm = Qnorm-deltaQ;  // Decr Q if lambda too small
                  lambda = f(Qnorm);        // Calculate new lambda
            }
      }
      if ( not last pixelblock) then goto start_encoding_pixelblock    //Next
// Done with picture.
```

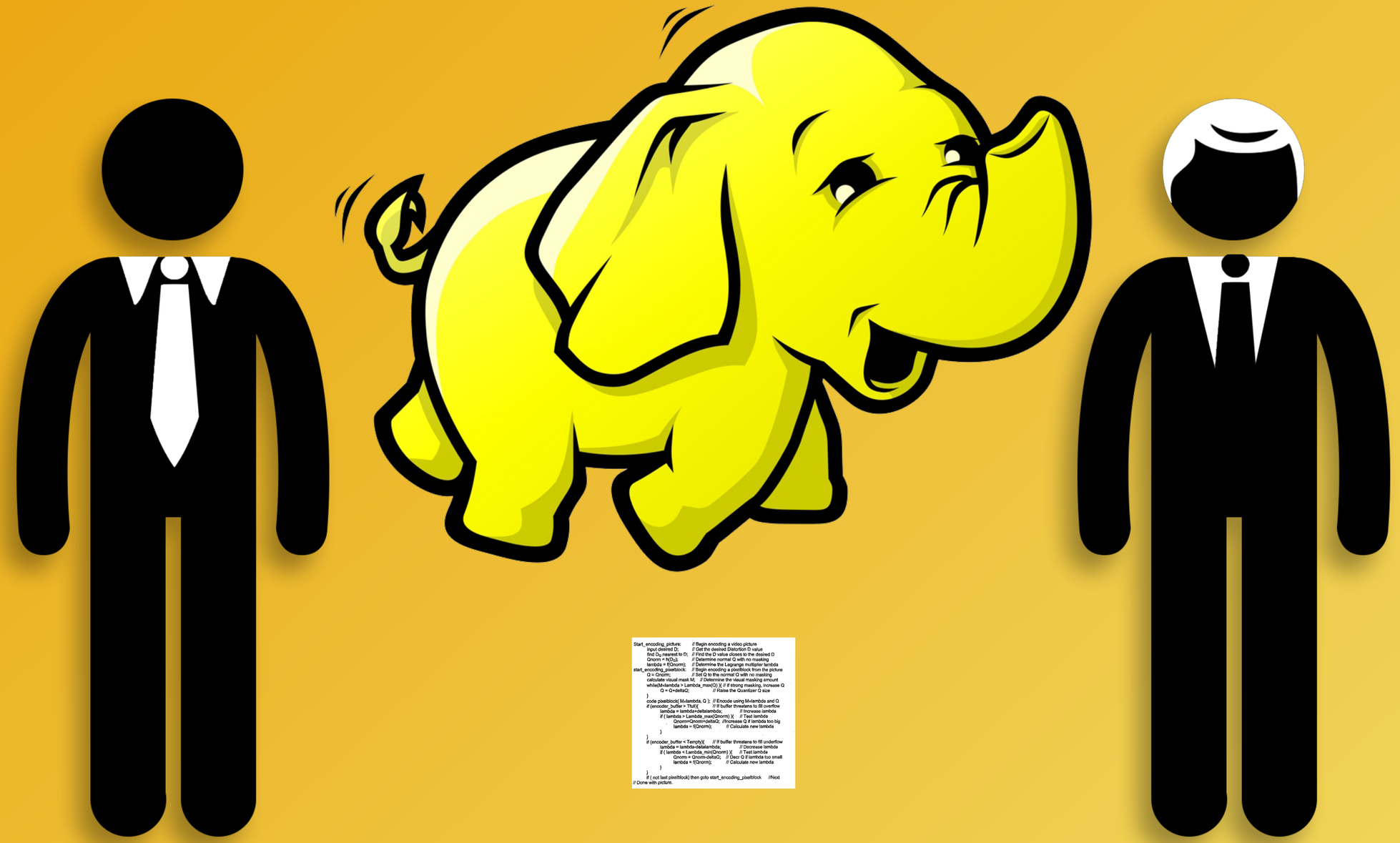
```
223 envs> [mac_carthy:1:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .101.> }::{ <a,(Local),i_-10. . .111.>, <b,(Loca
gnal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
 (a > 100)
223 envs> [mac_carthy:4:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .101.> }::{ <a,(Local),i_-10. . .111.>, <b,(Loca
gnal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
223 envs> [mac_carthy:5:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .100.> }::{ <a,(Local),i_102. . .111.>, <b,(Loca
gnal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
 return (a-10);
223 envs> [mac_carthy:15:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .101.> }::{ <a,(Local),i_-10. . .111.>, <b,(Loc
:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
.
pe
.
```

```
<Union 223 envs> [mac_carthy:9:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .100.> }::{ <a,(Local),i_-10. . .100.>, <b,(Local), Bot > }:{
}:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
<18>       b = mac_carthy (mac_carthy (a+11));
<Union 223 envs> [mac_carthy:14:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .91.> }::{ <a,(Local),i_-10. . .100.>, <b,(Local),i_91. . .91.
> }:{  }:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []

<Union 223 envs> [mac_carthy:14:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .91.> }::{ <a,(Local),i_-10. . .100.>, <b,(Local),i_91. . .91.
> }:{  }:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []
<19>       return (b);
<Union 223 envs> [mac_carthy:15:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .101.> }::{ <a,(Local),i_-10. . .111.>, <b,(Local),i_91. . .91
.> }:{  }:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] []

End of function
```

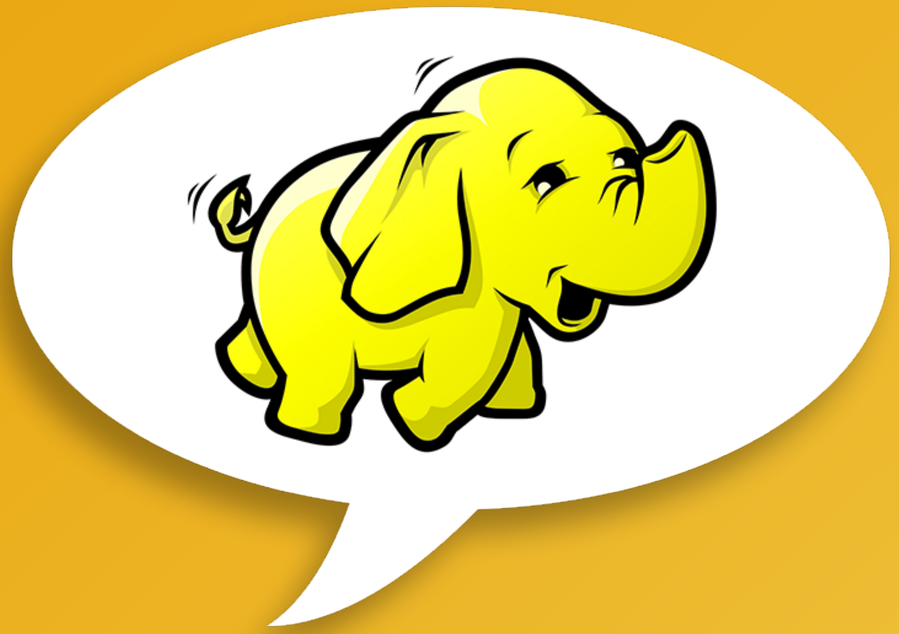
```
223 envs> [mac_carthy:15:mac_carthy.c]::{ <mac_carthy,(Global), Bot >, <mac_carthy_return,(Global),i_91. . .101.> }::{ <a,(Local),i_-10. . .111.>, <b,(Loca
,(Local),i_0. . .1.>, <.x2,(Local),i_100. . .100.>, <.x3,(Local),i_-10. . .-10.>, <.x4,(Local),i_10. . .10.>, <.x5,(Local),i_91. . .101.>, <.x6,(Local),
),i_11. . .11.> }:{ No signal }:{ To do }[mac_carthyG mac_carthy_returnG ] [aL bL ] [.x2L .x1L .x4L .x3L .x7L .x6L .x5L ]
.


main (int argc, char * argv[])

: x;


. envs> [main:1:mac_carthy.c]::{ <mac_carthy_return,(Global), Bot >, <main,(Global), Bot > }::{ <x,(Local), Bot > }:{  }:{ No signal }:{ To do }[mainG mac
.]
: -10;
. envs> [main:3:mac_carthy.c]::{ <mac_carthy_return,(Global), Bot >, <main,(Global), Bot > }::{ <x,(Local),i_-10. . .-10.> }:{  }:{ No signal }:{ To do }[
y_returnG ] [xL ] []
<23>

<Union 1 envs> [main:3:mac_carthy.c]::{ <mac_carthy_return,(Global), Bot >, <main,(Global), Bot > }::{ <x,(Local),i_-10. . .-10.> }:{  }:{ No signal }:{ To do }[mainG mac_carth
y_returnG ] [xL ] []
```

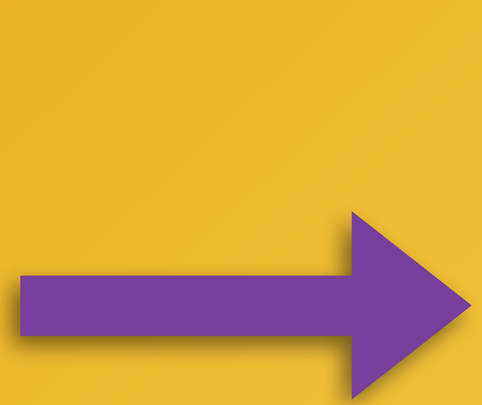




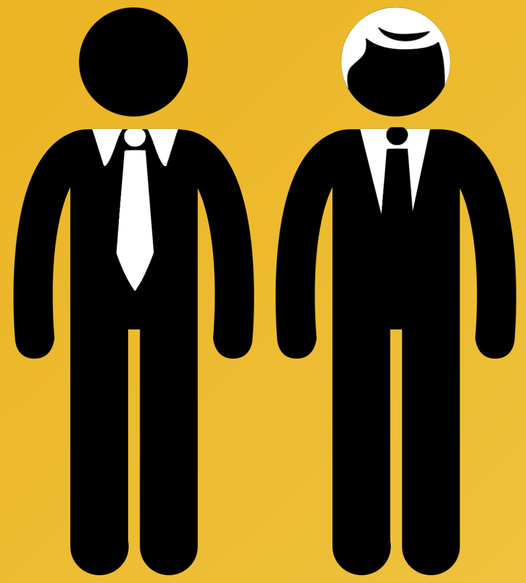


```
Start_encoding_picture;             // Begin encoding a video picture
      input desired D;              // Get the desired Distortion D value
      find DQ nearest to D;         // Find the D value closes to the desired D
      Qnorm = NQ-1();               // Determine normal Q with no masking
      lambda = f(Qnorm);            // Determine the Lagrange multiplier lambda
start_encoding_pixelblock;          // Begin encoding a pixelblock from the picture
      Q = Qnorm;                    // Set Q to the normal Q with no masking
      calculate visual mask Mi;     // Determine the visual masking amount
      while(M•lambda > Lambda_max(Q) ){ // If strong masking, increase Q
            Q = Q+deltaQ;                 // Raise the Quantizer Q size
      }
      code pixelblock( M•lambda, Q );  // Encode using M•lambda and Q
      if (encoder_buffer > TIull){          // If buffer threatens to fill overflow
            lambda = lambda+deltalambda;          // Increase lambda
            if ( lambda > Lambda_max(Qnorm) ){    // Test lambda
                  Qnorm=Qnorm+deltaQ;  //Increase Q if lambda too big
                  lambda = f(Qnorm);          // Calculate new lambda
            }
      }
      if (encoder_buffer < Tempty){       // If buffer threatens to fill underflow
            lambda = lambda-deltalambda;          // Decrease lambda
            if ( lambda < Lambda_min(Qnorm) ){    // Test lambda
                  Qnorm = Qnorm-deltaQ;   // Decr Q if lambda too small
                  lambda = f(Qnorm);          // Calculate new lambda
            }
      }
      if ( not last pixelblock) then goto start_encoding_pixelblock     //Next
// Done with picture.
```

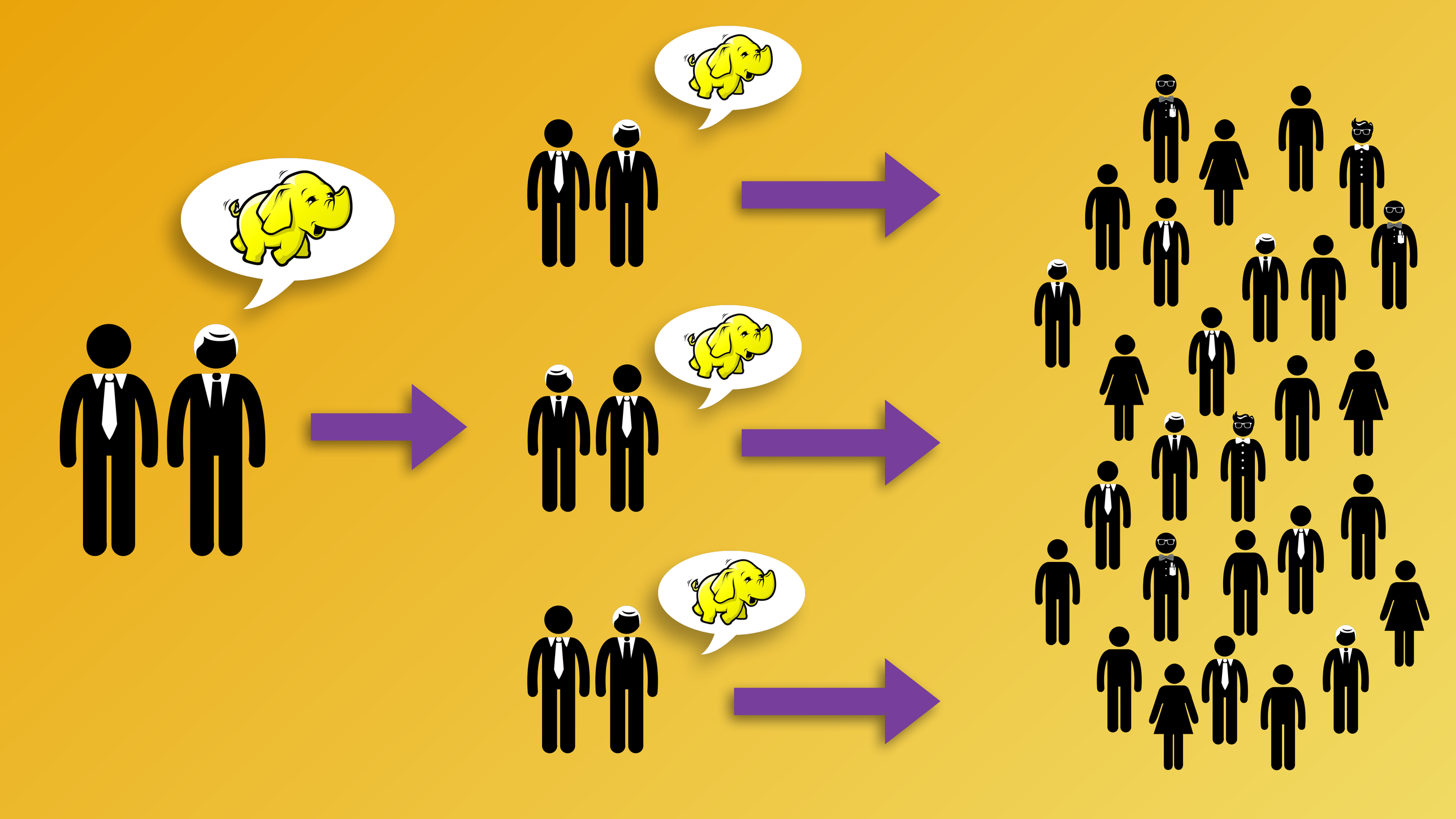


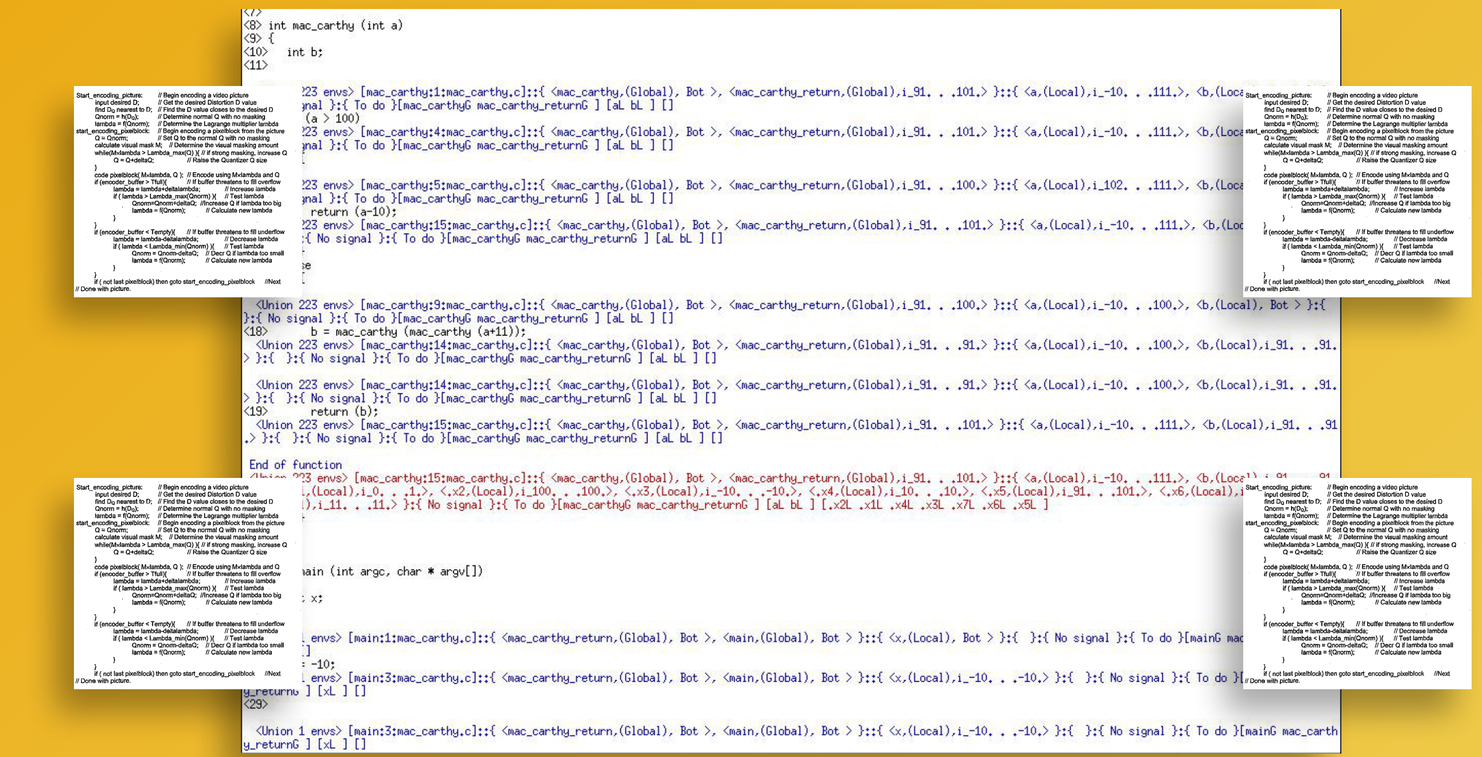



in
f




!
!
!
:-)

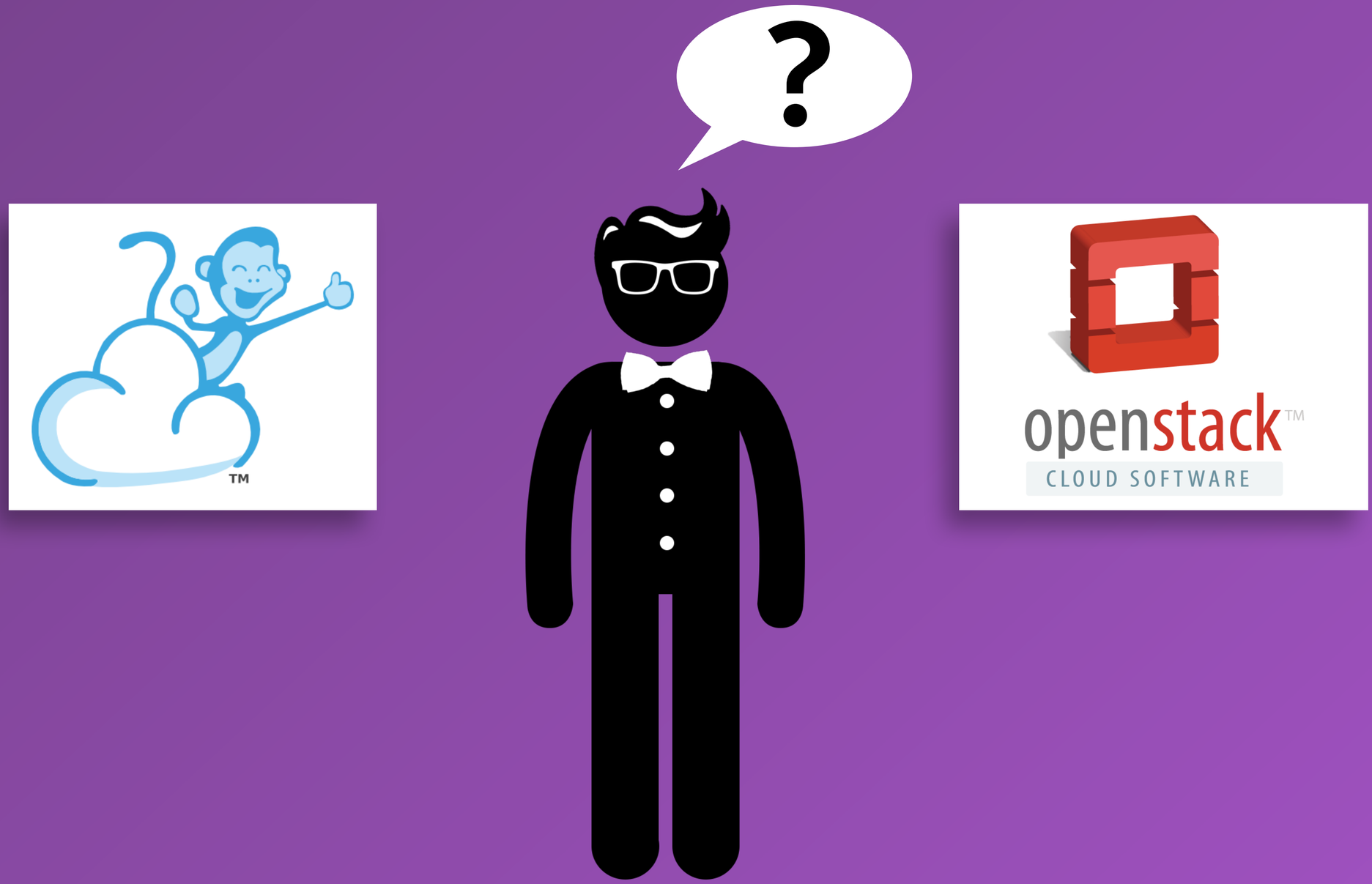


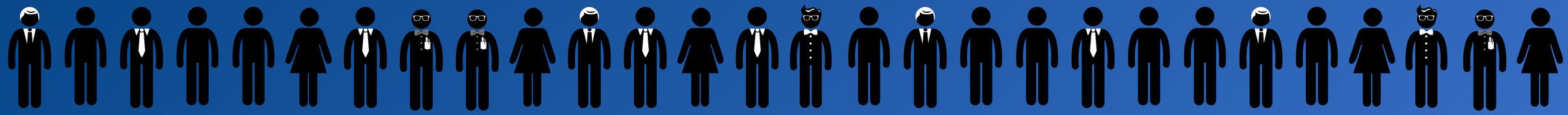

int mac_carthy (int a)
{
int b;
b = mac_carthy (mac_carthy (a+11));
return (b);
End of function


(*)
(8) int mac_carthy (int a)
(9) {
(10) int b;
(11)
(18) b = mac_carthy (mac_carthy (a+11));
(19) return (b);
(29)
End of function

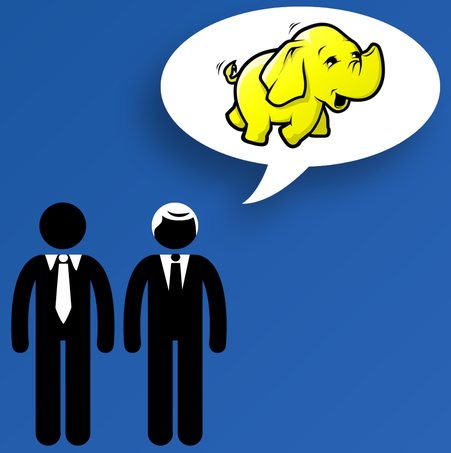

BACKBONE.JS
?
node JS
ANGULARJS
by Google
ember
jQuery
write less, do more.
eescript™
dojo
toolkit

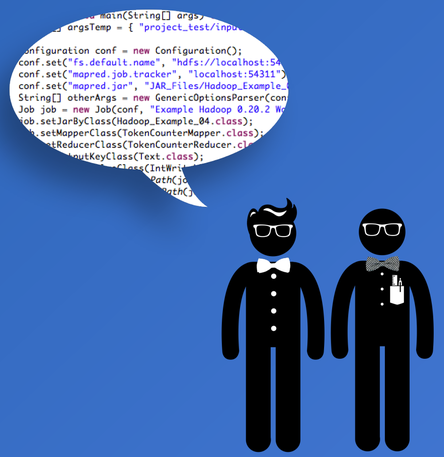

BACKBONE.JS
node JS
ANGULARJS
by Google
jQuery
write less, do more.
ember
eescript
dojo
toolkit


?
openstack™
CLOUD SOFTWARE



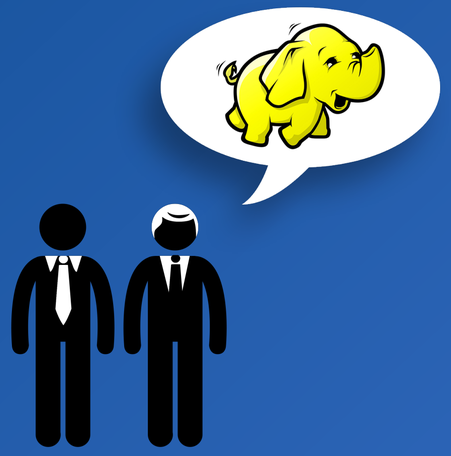
BRAND > code

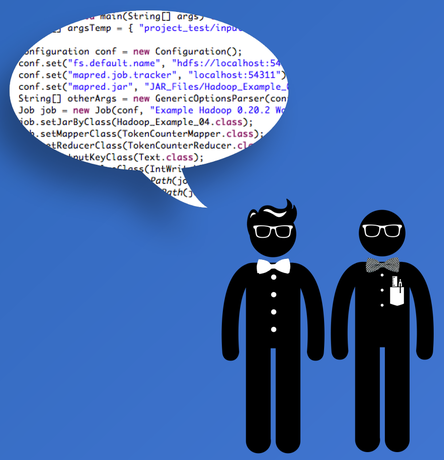


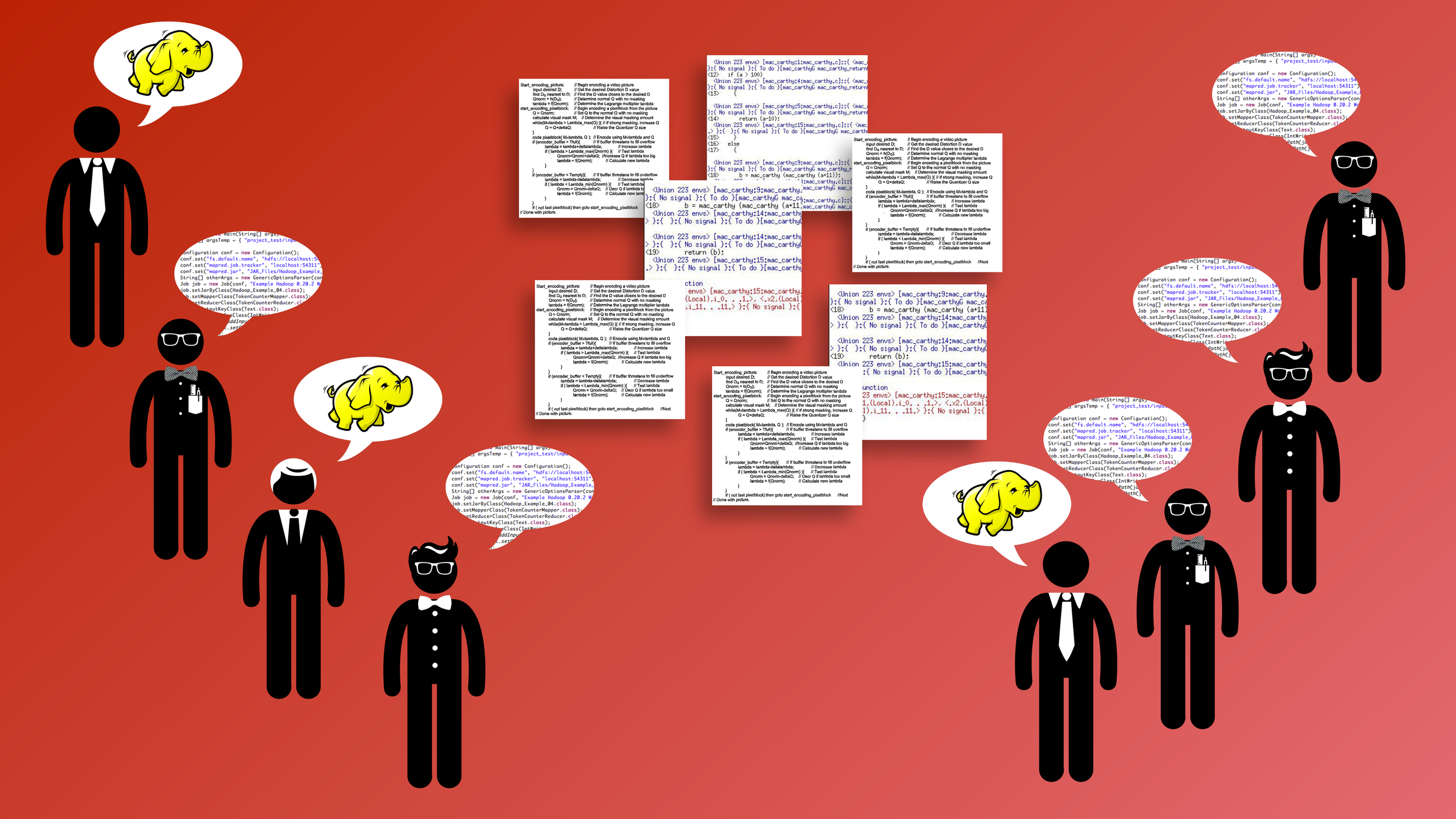


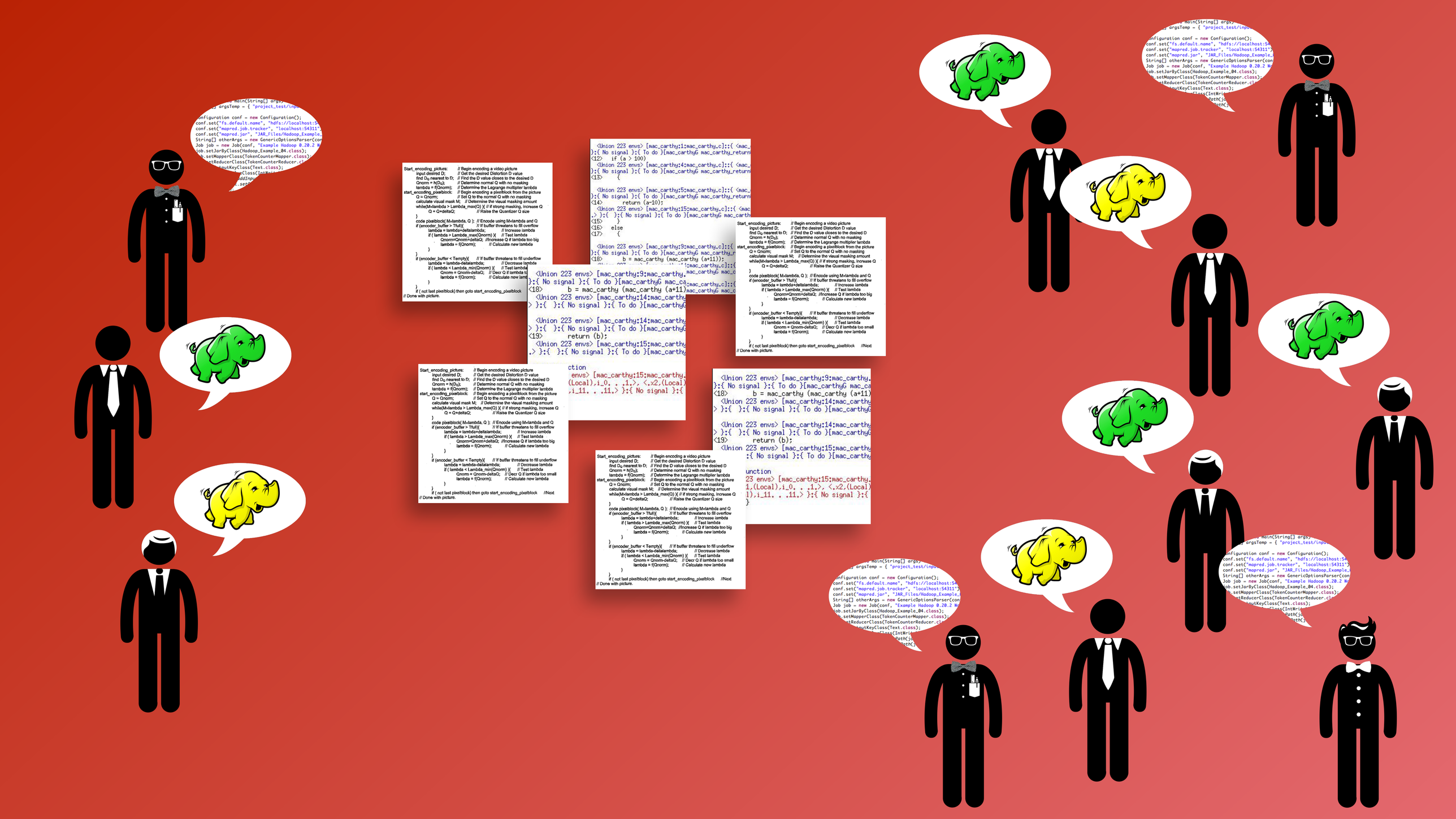
Community > code

BRAND > code

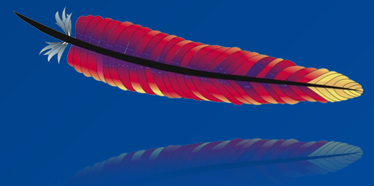








@shanecurcuru
http://CommunityOverCode.com
http://punderthings.com
http://www.apache.org



</presentation>