**Other:**

Python

Shell scripting, Curl

PHP

AWS:

EC2, Parameter Store, Elastic Cache,

S3, Lambda Service, Cloud Watch, IAM,

Elastic Beanstalk, Sumo Logic, Dynamo

DB

Big Data:

Program Language:

Scala

Big data Analysis:

Storm, Apache Spark, Apache Hadoop MapReduce

Messaging system:

Apache Kafka

DB:

Apache Cassandra, Mongodb, Vertica, Apache HBase

Apache HDFS

**BNYM:**

1. **Sec Mod:**

Angular JS, Html, Js, Css, Node Js, Grunt, Bower

Java, Spring boot, MQ, File Processing, Micro Services, FTP, NAS Storage,

Cloud deployments(App Engine), Docker, Unix

Oracle db, Sql, Stored Procedures

1. **Caching Service(Trade Dashboard):**

Angular JS, Html, Js, Go Js,

Java, Derby Cache, Hazelcast, Kafka, MQ,

Cloud deployments(BXP2), Unix

1. **Compare Tool:**

Angular JS, Html, Js, Java, Cloud deployments(BXP2), Unix

1. **BTM**:

Web:  
 Ajax, jsonp, Websockets, Web Security, Cross Domain, Code profiling tools  
 Highcharts, D3  
 Written custom components like arrows  
 Cancelling Ajax requests  
 Logout functionality using jsp  
 Java:  
 Running the unix script from java.  
 Running the stats from java.  
 Cronjob documentation: http://quartz-scheduler.org/documentation/faq  
 Pool exhaust exception  
 Disk is full in tomcat vm  
   
 Using Maven to build the war file.  
 Used winscp to deploy to Dev for dev and hyperic to QA and BXP tool to deploy

into production.  
 Used spring web MVC features  
 Used Spring JDBC to connect to the oracle database.   
Database:

(Scripts from svn, sql quries and Performance changes)  
 Statistics Gathering Procedures in the DBMS\_STATS Package.  
 Creating indexes.  
 Worked with performance team to enhance the db speed  
 Stored procs to insert the new btm messages into oracle  
 Written various sql queries   
Environment:  
 Ext JS 4.2.1, Java/J2EE, Oracle, Spring, Spring-Jdbc, MVC, Sql, Pl/Sql, Stored

procs, Siteminder, Tomcat, WinScp, BxpCloud, Sql Developer, Eclipse, Putty,

Soap UI, Rest Client, Maven, Unix, Cloud deployments(BXP1),

Oracle

1. **Pentaho:**

Java:  
 Creating plugins for Spoon tool.  
 Creating Restful webservice. Using jersey client to consume the restful service.  
 XML parsing(Dom Parser)

**CME:**

Filtering tree store

Applying the styles to application using SASS.

**Refs**: [Mansoor.Ahmed@cmegroup.com](mailto:Mansoor.Ahmed@cmegroup.com), [Sumit.Tiwari@cmegroup.com](mailto:Sumit.Tiwari@cmegroup.com), [Neeraj.Mathur@cmegroup.com](mailto:Neeraj.Mathur@cmegroup.com), [Neha.Agarwal@cmegroup.com](mailto:Neha.Agarwal@cmegroup.com), [Manish.Pampatwar@cmegroup.com](mailto:Manish.Pampatwar@cmegroup.com), [Murali.Paladi@cmegroup.com](mailto:Murali.Paladi@cmegroup.com), [Matthew.Hartman@cmegroup.com](mailto:Matthew.Hartman@cmegroup.com), [Raj.Dandolu@cmegroup.com](mailto:Raj.Dandolu@cmegroup.com), [Sumeet.Singh@cmegroup.com](mailto:Sumeet.Singh@cmegroup.com)

**Nielsen:**

Tech Points:

1. Image files reading from server which is in different(NAS) location

2. Javascript windows.open security problem in IE8

3. Comet calls are asynchronous. Servlet 3 supports. Need Tomcat 7.

4. Our application can be accessed form Portal. Same application deployed into 2 tomcat servers with the same context like admin. Portal reroute the calls to servers depends on contexts admin1,admin2. But flex messagebroker calls are failing. Because the context applied to remoteObject endpoint in compile time. Because the iframe loaded in different contexts. So I take the url with context by document.URL.split('?')[0];.So I give the endpoint to remoteObject dynamically.

5. Different wsdl endpoints generating different jsessionIds.