



SNS COLLEGE OF TECHNOLOGY



(An Autonomous Institution)

Re-accredited by NAAC with A+ grade, Accredited by NBA(CSE, IT, ECE, EEE & Mechanical)
Approved by AICTE, New Delhi, Recognized by UGC, Affiliated to Anna University, Chennai

Department of MCA

Topic: Form Validation using HTML5 & JS

COURSE

19CAT901
Web Programming
Essentials

UNIT - III

Java Script

CLASS

I Semester /
I MCA



Form Validation using attributes



❑ Form validation using attributes

- **required** attribute

```
Enter Name: <input type="text" name="name" required>
```

- **type** attribute (use right control element for input value) like email, url, number, range, password

```
Age: <input type="number" size="6" name="age" min="18" max="99" value="21"><br>
```

```
Satisfaction: <input type="range" size="2" name="satisfaction" min="1" max="5"  
value="3">
```



Patterns for different data types



URL

```
pattern="https?:/.+"
```

Date

```
pattern="\d{1,2}/\d{1,2}/\d{4}"
```

Price

```
pattern="\d+(\.\d{2})?"
```

Price

```
pattern="\d+(\.\d{2})?"
```

Name

```
<input id="name" name="name" value="" aria-describedby="name-format" required  
aria-required="true" pattern="[A-Za-z-0-9]+\s[A-Za-z-'0-9]+>
```





NameNode data



- ❑ Checkpointing process (loading primary's in-memory filesystem metadata from secondary namenode) proceeds as follows
 1. *Secondary node asks the primary to roll its edits file, so new edits go to a new file.*
 1. *The secondary retrieves fsimage and edits from the primary (using HTTP GET).*
 2. *The secondary loads fsimage into memory, applies each operation from edits, then creates a new consolidated fsimage file.*
 3. *The secondary sends the new fsimage back to the primary (using HTTP POST)*



NameNode data



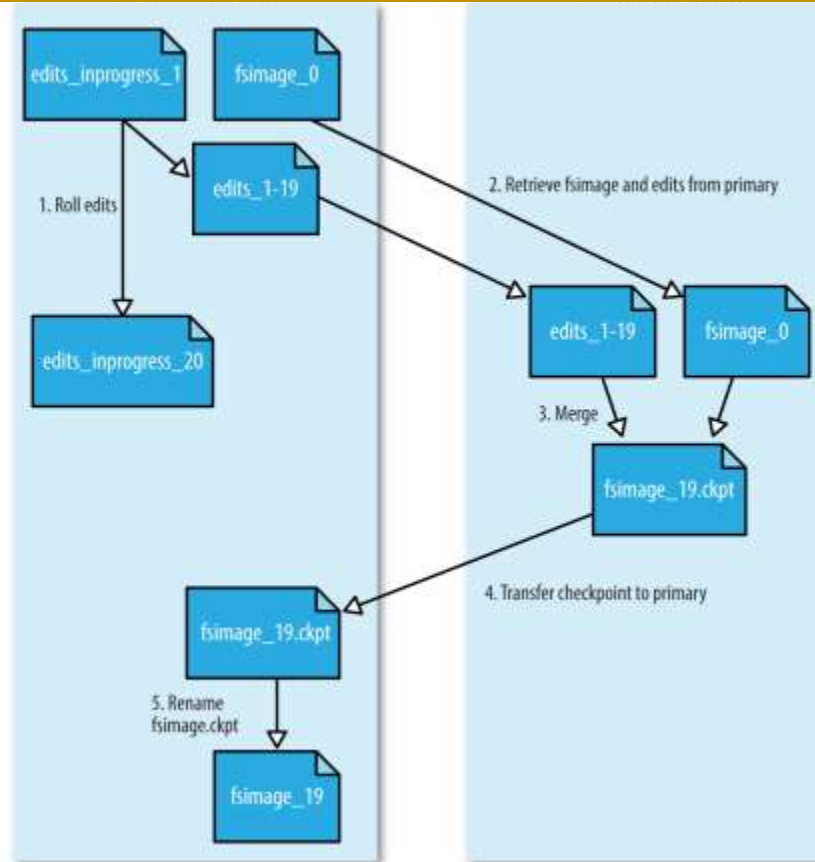
5. *The primary replaces the old fsimage with the new one from the secondary, and the old edits file with the new one it started in step 1.*

It also updates the fstime file to record the time that the checkpoint was taken





Check point Process





Secondary NameNode data



- The directory structure of secondary namenode is

`\${fs.checkpoint.dir}/current/VERSION

/edits

/fsimage

/fstime

/previous.checkpoint/VERSION

/edits

/fsimage

/fstime



DataNode data



- key files and directories of data node is as follows

```

${dfs.datanode.data.dir}/
├── current
│   ├── BP-526805057-127.0.0.1-1411980876842
│   │   ├── current
│   │   │   ├── VERSION
│   │   │   ├── finalized
│   │   │   ├── blk_1073741825
│   │   │   ├── blk_1073741825_1001.meta
│   │   │   ├── blk_1073741826
│   │   │   └── blk_1073741826_1002.meta
│   │   └── rbw
│   └── VERSION
└── in_use.lock
```




Dfsadmin

- ❑ Multipurpose tool for finding information about the state of HDFS, as well as performing administration operations on HDFS

```
-report  
-metasave  
-safemode  
-saveNamespace  
-fetchImage  
-refreshNodes  
-upgradeProgress  
-finalizeUpgrade  
-setQuota  
-refreshServiceAcl  
-allowSnapshot  
-disallowSnapshot
```



❑ **Filesystem check (fsck)** used to check the health of files in HDFS. It looks for blocks that are missing/under/ over-replicated blocks from datanodes

- *Over-replicated blocks*
- *Under-replicated blocks*
- *Misreplicated blocks*
- *Corrupt blocks*
- *Missing Replicas*

```
% hdfs fsck /
.....Status: HEALTHY
Total size: 511799225 B
Total dirs: 10
Total files: 22
Total blocks (validated): 22 (avg. block size 23263601 B)
Minimally replicated blocks: 22 (100.0 %)
Over-replicated blocks: 0 (0.0 %)
Under-replicated blocks: 0 (0.0 %)
Mis-replicated blocks: 0 (0.0 %)
Default replication factor: 3
Average block replication: 3.0
Corrupt blocks: 0
Missing replicas: 0 (0.0 %)
Number of data-nodes: 4
Number of racks: 1
```



Tools



Datanode block scanner - Every datanode runs a block scanner, which periodically verifies all the blocks stored on the datanode. This allows bad blocks to be detected and fixed before they are read by clients

Balancer- program is a Hadoop daemon that re-distributes blocks by moving them from over utilized datanodes to under-utilized datanodes

```
% start-balancer.sh
```

```
Time Stamp      Iteration#  Bytes Already Moved  ...Left To Move  ...Being Moved
Mar 18, 2009 5:23:42 PM  0                0 KB              219.21 MB        150.29 MB
Mar 18, 2009 5:27:14 PM  1                195.24 MB         22.45 MB         150.29 MB
The cluster is balanced. Exiting...
Balancing took 6.072933333333333 minutes
```



References



- ❑ Tom White, “ Hadoop: The Definitive Guide” Third Edition, O’reilly Media, 4th Edition, 2012

Web Resources

- ❑ https://www.tutorialspoint.com/map_reduce/map_reduce_hadoop_administration.htm
- ❑ <https://freevideolectures.com/course/3610/hadoop-administration/12>