Check pointing:

Checkpointing is an essential part of maintaining and persisting filesystem metadata in HDFS. It is crucial for efficient NameNode recovery and restart,

and it is an important indicator of overall cluster health.

At a high level, the NameNode's primary responsibility is storing the HDFS namespace. This means things like the

directory tree, file permissions and the mapping of files to block ID's. Its important that irs metadata

are safely persisted to stable storage for fault tolerance.

This file system metadata is stored in two different constructs: the FSImage and Edit logs. The FSImage is a file that

represents a point in time snapshot of the filesystem's metadata. However while the FSImage file format is very efficient to read, it is

unsuitable for making small incremental updates like renaming a single file. Thusm rather than writing a new FSImage every time the namespace is

modifiedm the NameNode instead records the modifying operation in the edit log for durability. By this method, if the NameNode crashes, iit cna

restore its state by first loading the FSImage then replaying all the operations in the edit log to catch up to the most recent state of the namesystem.

The edit log comprises a series of files, called edit log segments, that together represent all the namesuystem

modifications made since the creation of the FSImage.

Why Checkpointing is important?

A typical edit ranges from 10s to 100s of bytes, but over time enough edits can accumulate to edits can

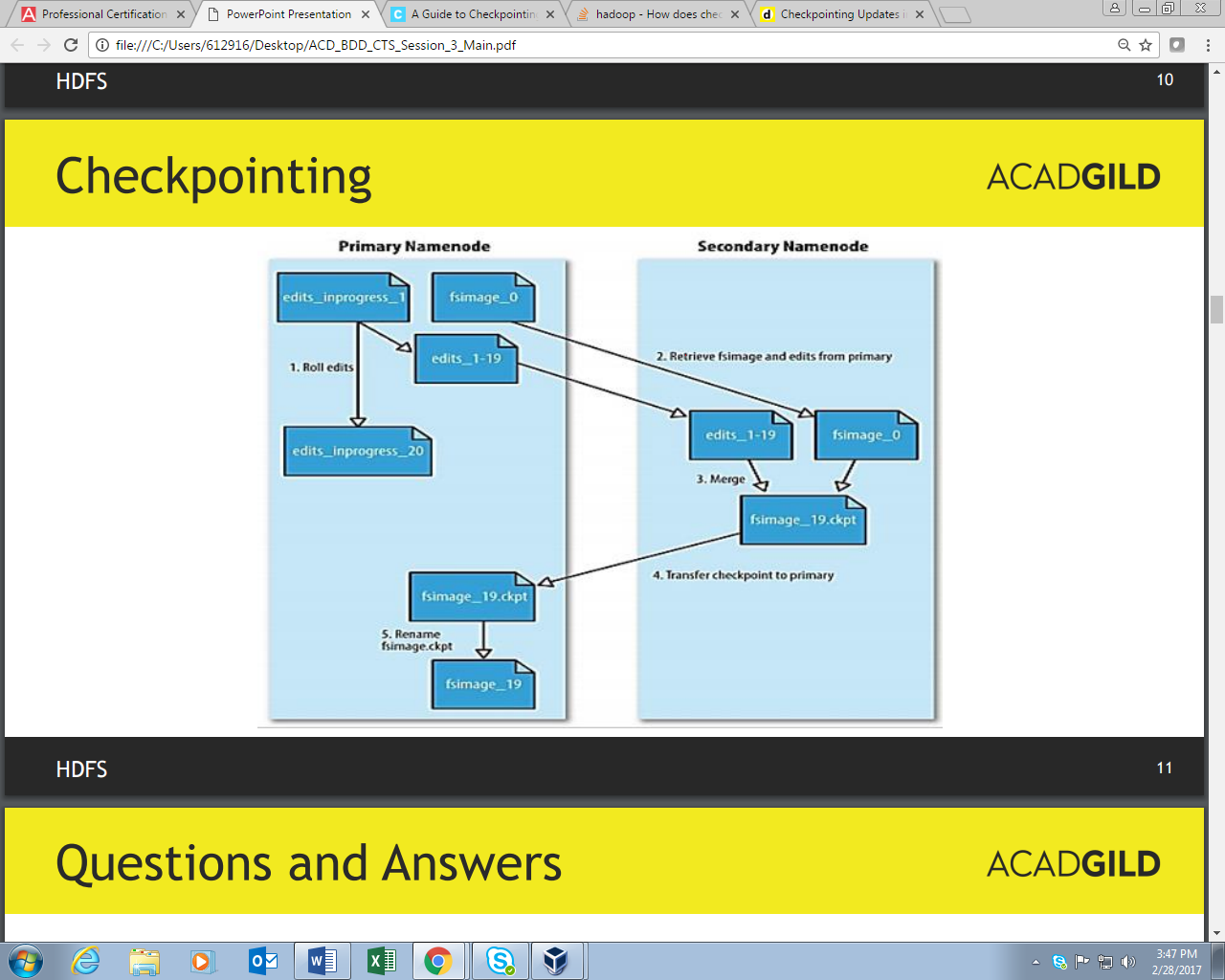
accumulate to become too large or difficult to modify. A couple of problems can arise from these large edit logs.

It can fill up all the available disk capacity on a node using indirect methods, a large edit log cna subbstantially delay

NameNode reapplies all the edits. This is where Checkpointing comes in use.

Checkpointing is a process that takes an FSImage and edit log and compacts them into a new FS Image. This method, instead of replaying a potentially unbounded edit log, the NameNode can load the final in memorys statae directly from the FSImage.

This is a far more efficient operation and reduces NameNode startup time.



However creating a new FSImage is and Input , output and CPU-intensive operation, sometimes taking minutes to perform.

Durinfg a checkpoint, the namesystem also needs to restrict concurrent access from other users.

So, rather than pausing the active NameNode to perform a checkpoint, HDFS defers it to either the

Secondary NameNode or Standby NameNode, depending on whether NameNode high-availability is configured. The mechanics of checkpointing differs

depending on if NameNode high-availability is configured. In either case though, checkpointing is triggered by one of two conditions.

1)If enough time has elapased since the last checkpoint or

2) If enough new edit log transactions have accumulated.

The checkpointing node periodically checks if either of these conditions are met.