Big Data Technologies

Trainer: Mr. Nilesh Ghule.

RDD creation and partitions

RDD creation

① rdd = sc.parallelize(collection)
- num of partitions can be given as arg.
- by default num of partitions = num of CPU cores (in spark local[*] mode)

② rdd = sc.textFile(filepath)
- if source file path is in HDFS, then num of parts = num of input splits ≃ num of HDFS blocks.
- if source file path is in local FS, then by default it creates 2 partitions for small files.

③ rdd = sc.wholeTextFiles("dirpath")
- num of partitions = num of files
- 1 file ⇒ 1 partition.

RDD caching/persistence

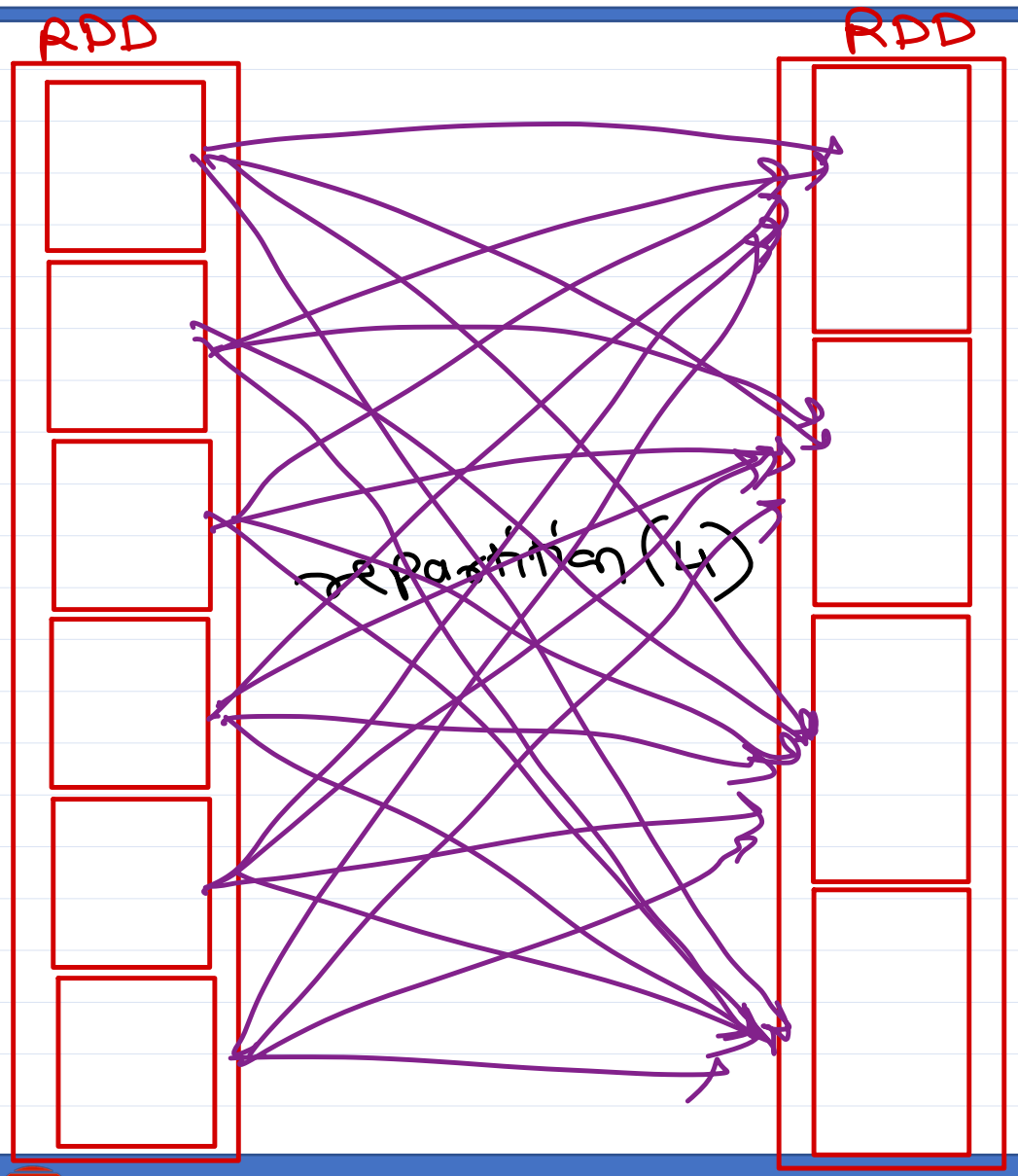
RDD can be cached/persisted if it is needed for multiple actions.

① rdd.cache()
- ↳ stored in mem when computed first time.

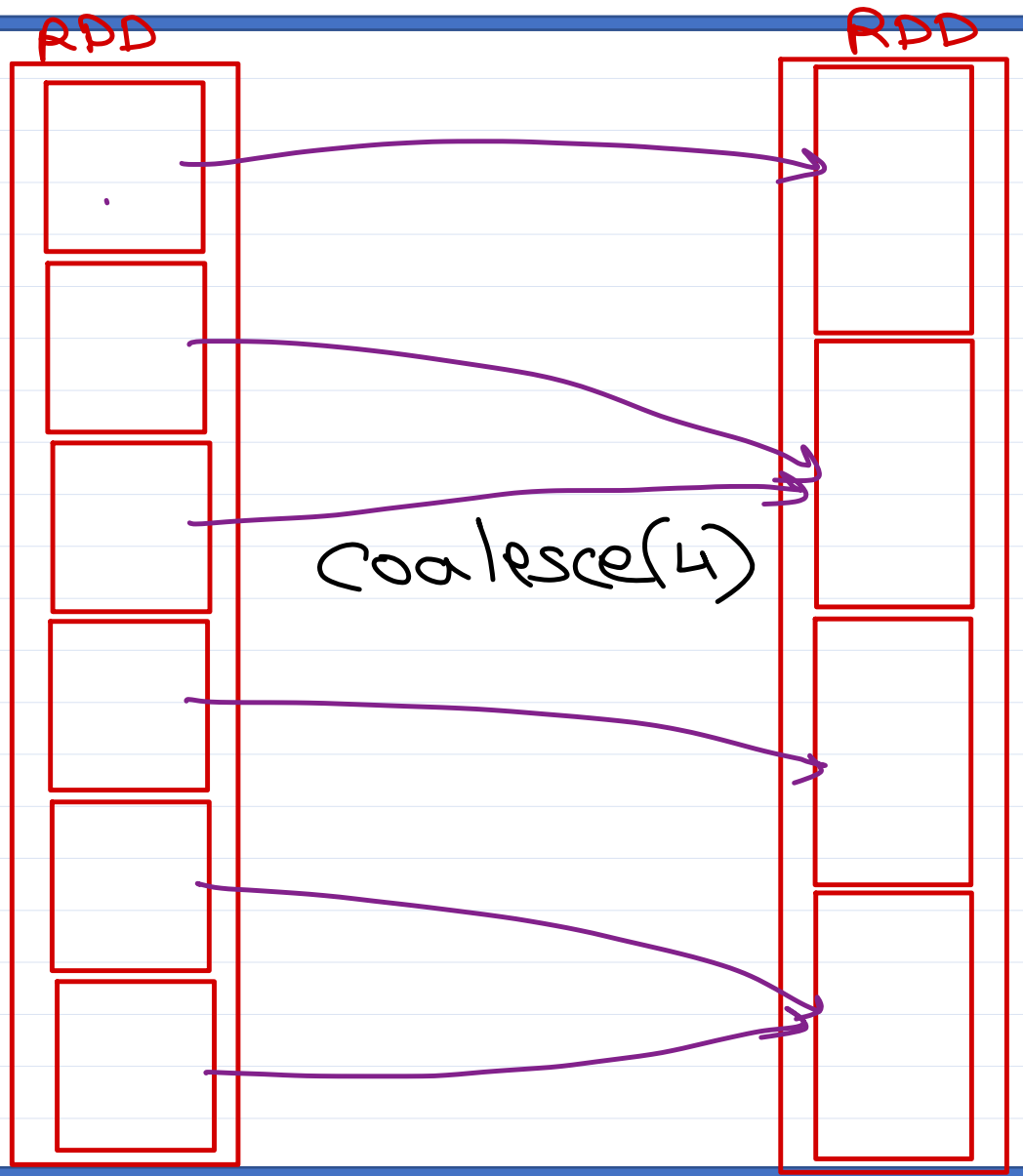
② rdd.persist(level)
- ↳ MEMORY_ONLY → same as cache()
- ↳ MEMORY_ONLY_SER → serialized. (compressed)
- ↳ MEMORY_AND_DISK → stored in mem & few partitions on disk (if not enough mem).
- ↳ MEMORY_AND_DISK_SER (compressed).
- ↳ DISK_ONLY

③ rdd.checkpoint(dir path)
- → for recovery from crash.

Repartition

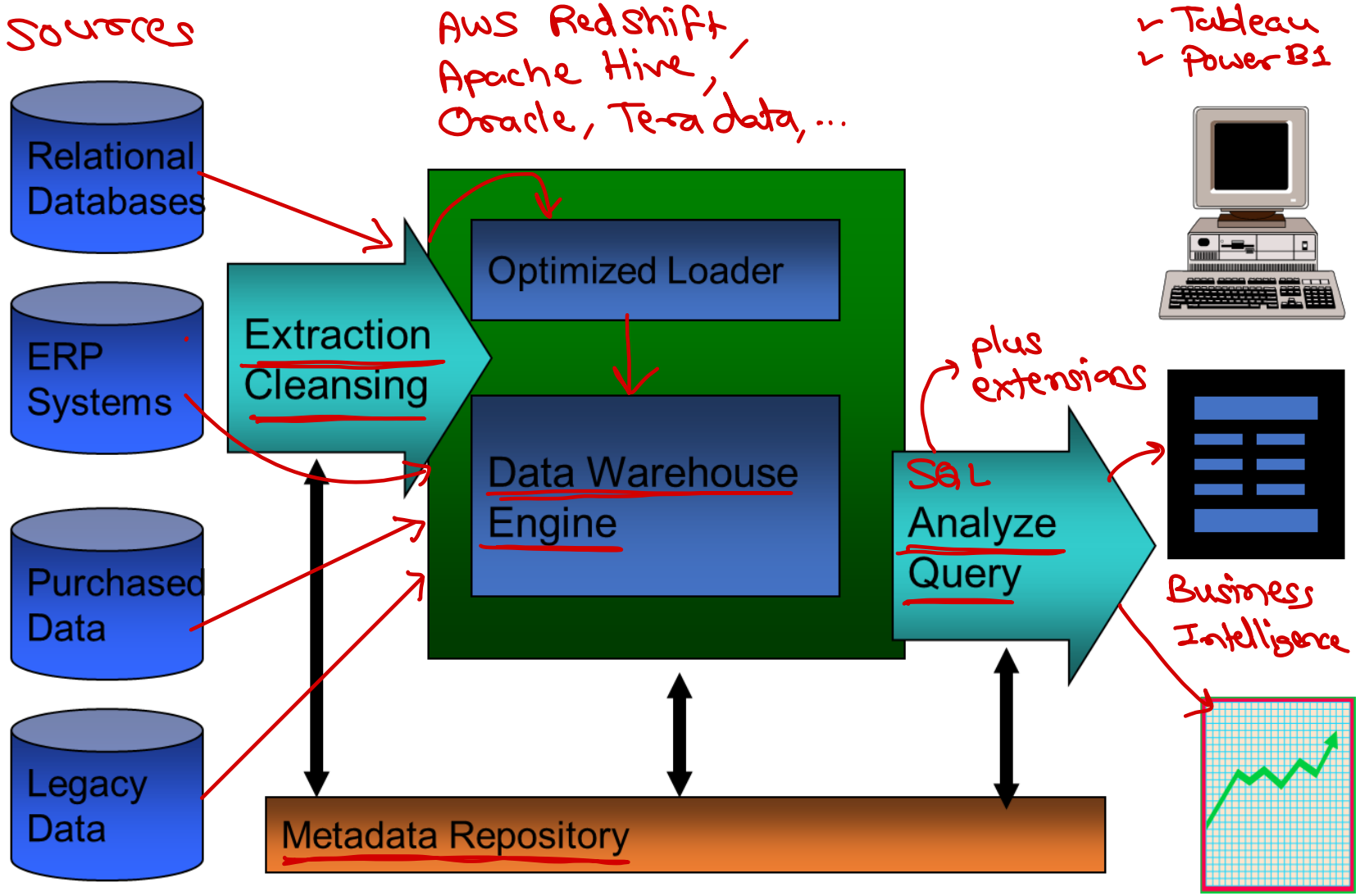


Coalesce

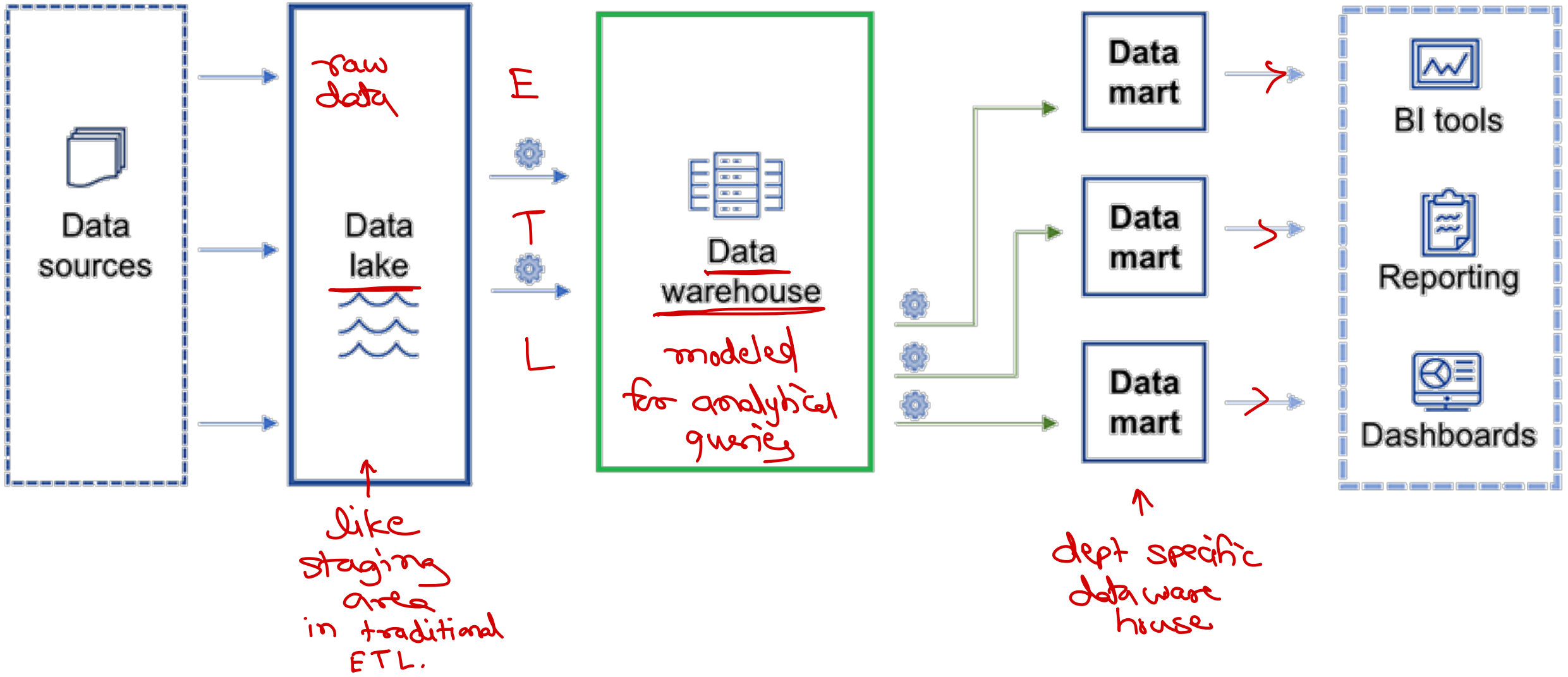


Data warehousing

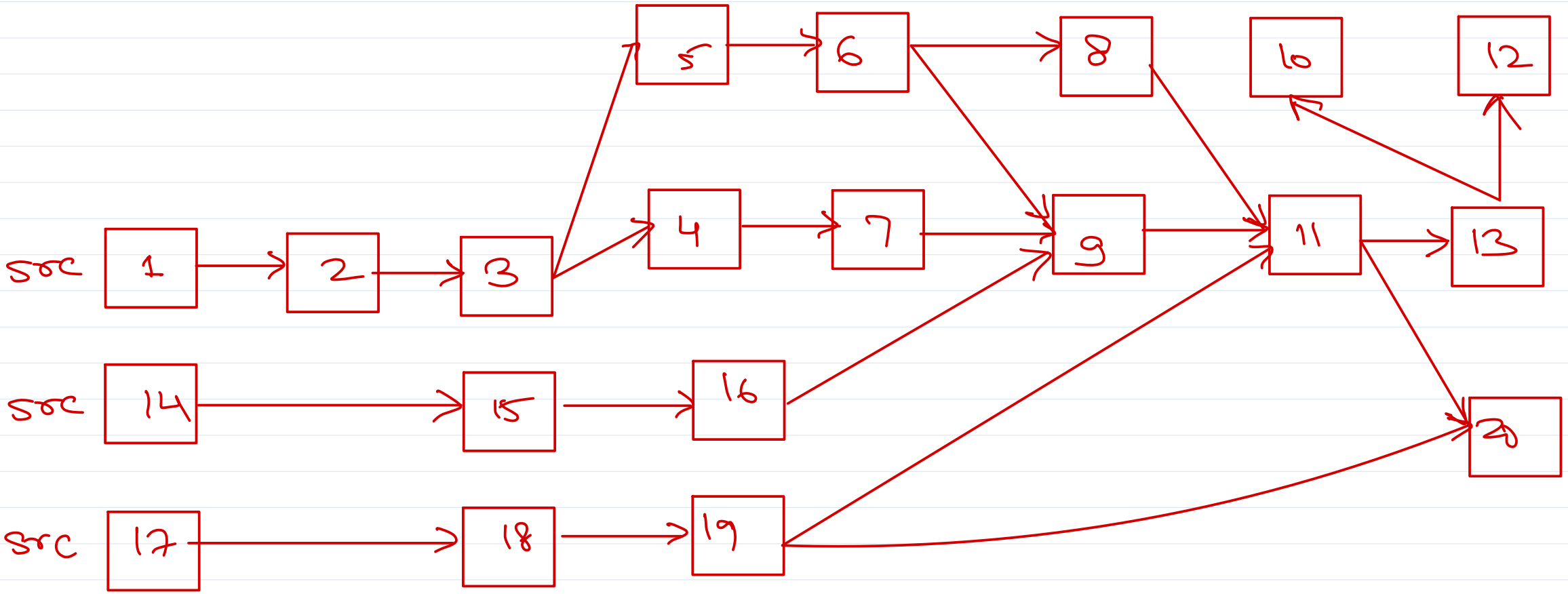
- Data warehouse is a single, complete and consistent store of data obtained from a variety of different sources made available to end users in a what they can understand and use in a business context.
- Data warehousing is a process of transforming data into information and making it available to users in a timely enough manner to make a difference.



Data lake vs Data warehouse vs Data mart



Complex DAGs



Thank you!

Nilesh Ghule <nilesh@sunbeaminfo.com>