Hadoop Components Overview

Contents

- Hadoop Components
- Hadoop Common
- HDFS
- MapReduce
- Hadoop Tools

Hadoop Components

Components of Hadoop:

- Hadoop Common
- HDFS
- Yarn
- Map Reduce

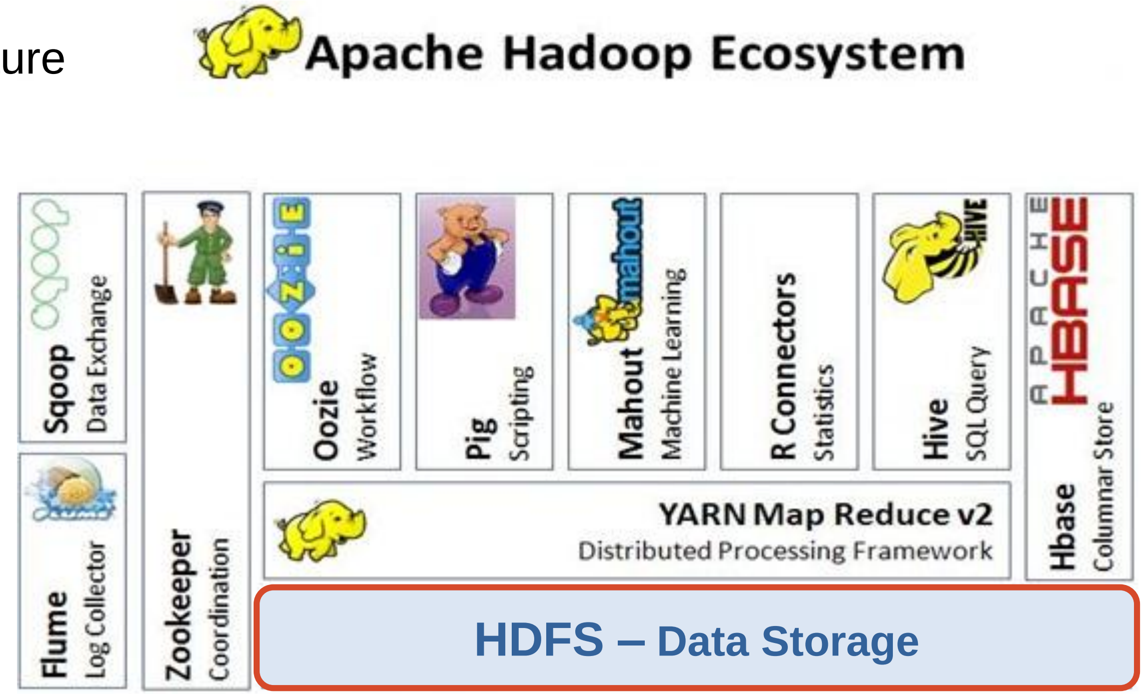
Hadoop Common

- ➢ A set of utilities that supports the Hadoop subprojects
- ➢ Provides access to the file systems supported by Hadoop
- ➢ Hadoop Common package contains:
  - ✓ The JAR files and scripts necessary to start Hadoop
  - ✓ Source code, documentation and a contribution section with projects from the Hadoop Community

HDFS

HDFS - **H**adoop **D**istributed **F**ile **S**ystem

- Primary storage system for Hadoop
- Distributed, portable, scalable file system written in Java
- Files are broken down and stored in multiple machines
- Designed for large scale distributed data processing
- Follows master/slave architecture



HDFS

HDFS is best suited for:

- Highly fault-tolerant
- Designed to deploy on low-cost hardware
- Suitable for applications that have large datasets(Giga to Terabytes)
- Enables streaming access to file system data

HDFS is not good for:

- Low Latency Data Access
- Lot of Small Files
- Multiple Writers, Arbitrary File Modifications

MapReduce (MR)

- Programming framework (library and runtime) for analyzing data sets stored in HDFS
- MapReduce jobs are composed of two functions:

map() → reduce()

sub-divide & conquer — *combine & reduce cardinality*

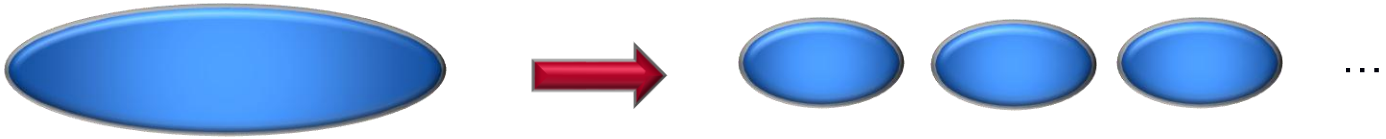
- User only writes the Map and Reduce functions
- MR framework provides all the “glue” and coordinates the execution of the Map and Reduce jobs on the cluster

MapReduce (contd..)

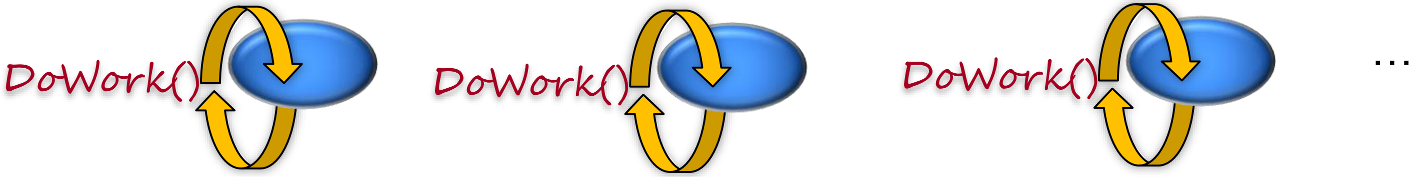
MAP

Essentially, it's…

1. Take a large problem and divide it into sub-problems

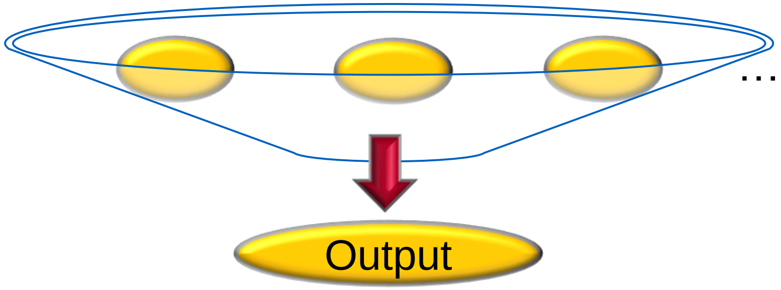


2. Perform the same function on all sub-problems

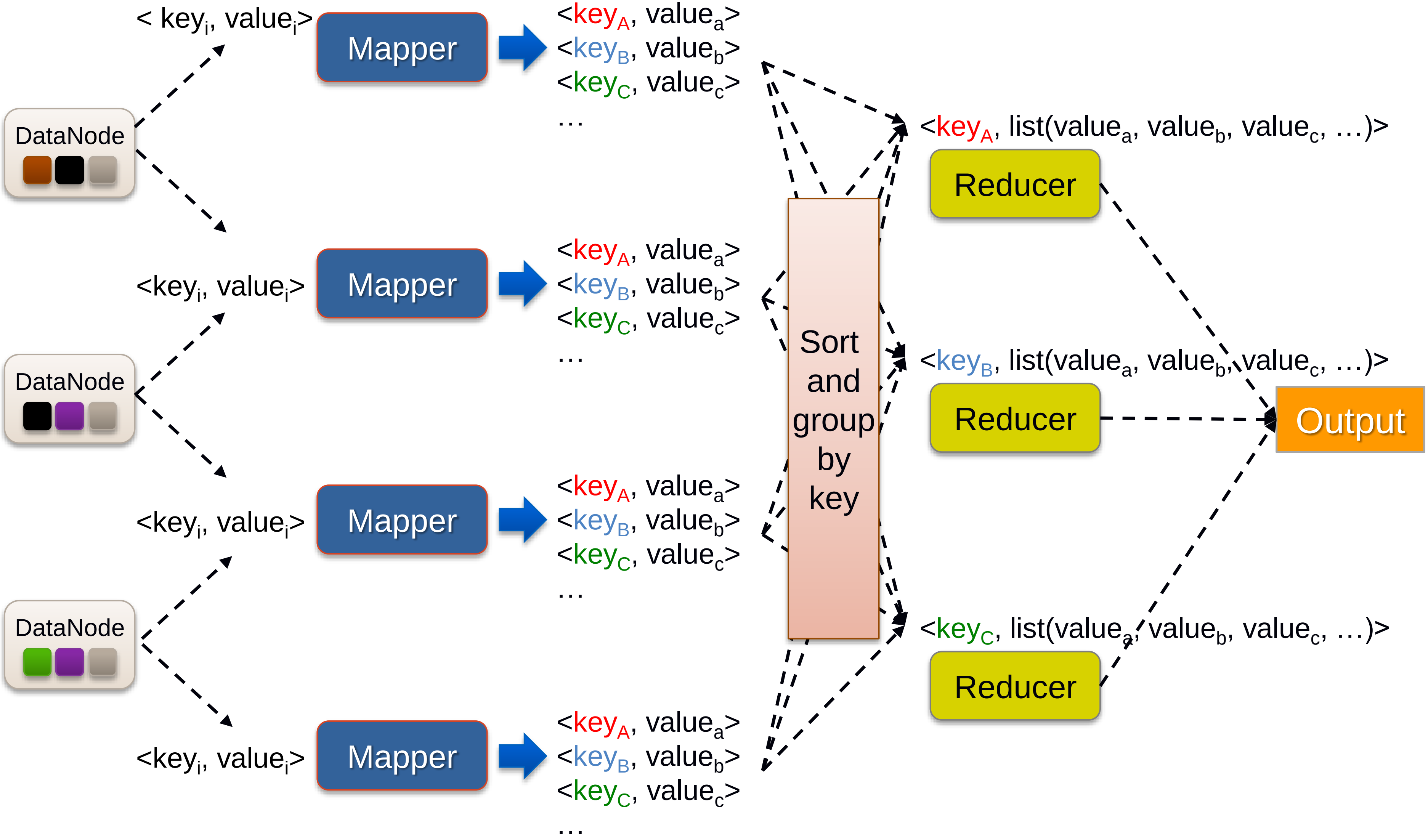


REDUCE

3. Combine the output from all sub-problems



MapReduce (contd..)



< key$_i$, value$_i$>
Mapper
<key$_A$, value$_a$>
<key$_B$, value$_b$>
<key$_C$, value$_c$>
…

DataNode

<key$_i$, value$_i$>
Mapper
<key$_A$, value$_a$>
<key$_B$, value$_b$>
<key$_C$, value$_c$>
…

DataNode

<key$_i$, value$_i$>
Mapper
<key$_A$, value$_a$>
<key$_B$, value$_b$>
<key$_C$, value$_c$>
…

DataNode

<key$_i$, value$_i$>
Mapper
<key$_A$, value$_a$>
<key$_B$, value$_b$>
<key$_C$, value$_c$>
…

Sort and group by key

<key$_A$, list(value$_a$, value$_b$, value$_c$, …)>
Reducer

<key$_B$, list(value$_a$, value$_b$, value$_c$, …)>
Reducer

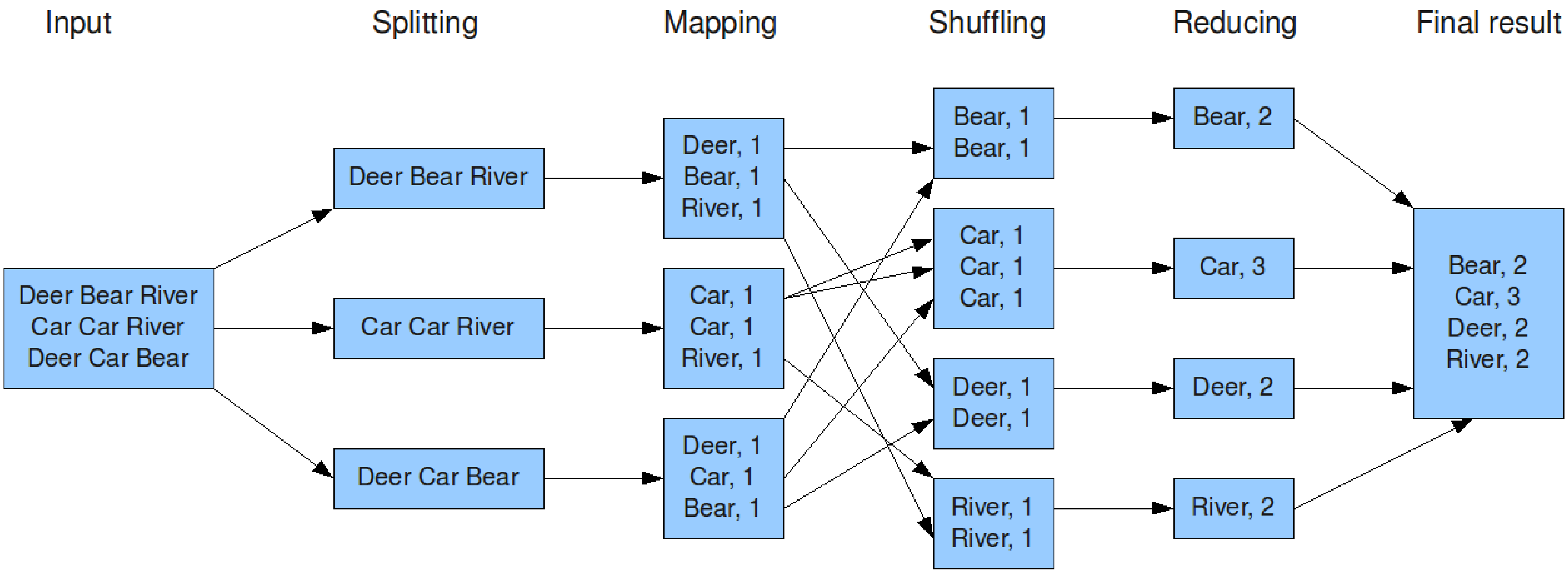
<key$_C$, list(value$_a$, value$_b$, value$_c$, …)>
Reducer

Output

A key-value pair (KVP) is a set of two linked data items:
Key - A unique identifier for data item
Value – Either the data that is identified or a pointer to the location of that data

MapReduce (contd..)

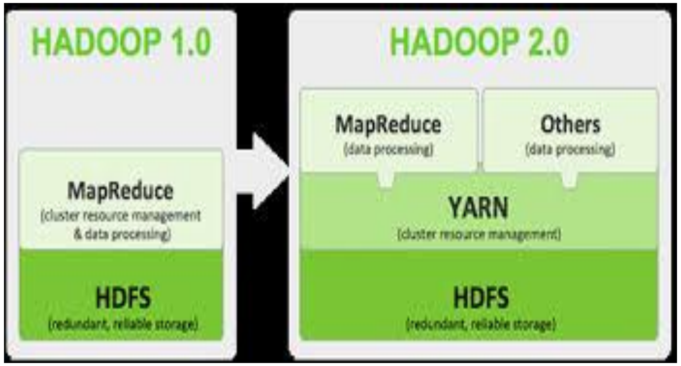
The overall MapReduce word count process



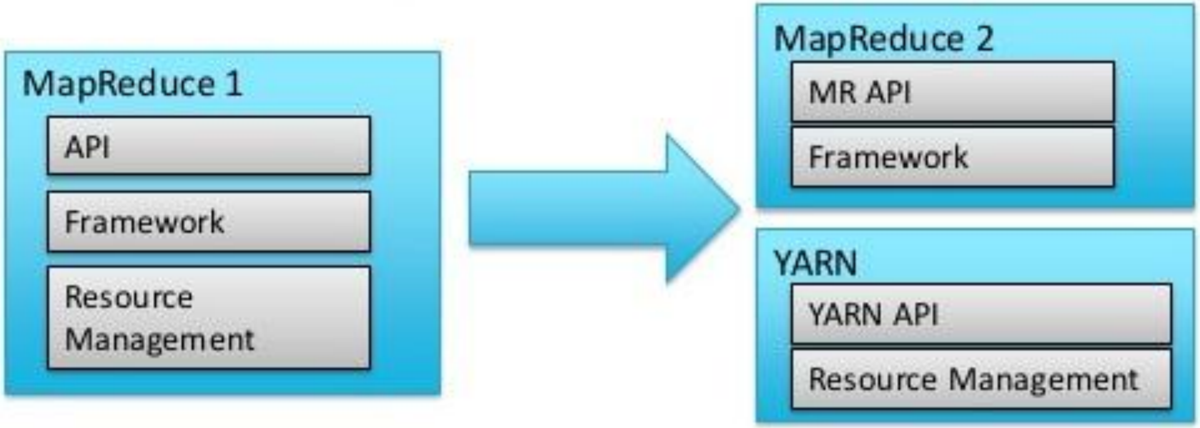
Hadoop YARN & MR2

Next generation MapReduce 2

- **MapReduce 1 ("Classic") has three main components**
  - API – for user-level programming of MR applications
  - Framework – runtime services for running Map and Reduce processes, shuffling and sorting, etc.
  - Resource management – infrastructure to monitor nodes, allocate resources, and schedule jobs


HADOOP 1.0
MapReduce
(cluster resource management & data processing)
HDFS
(redundant, reliable storage)
HADOOP 2.0
MapReduce
(data processing)
Others
(data processing)
YARN
(cluster resource management)
HDFS
(redundant, reliable storage)

- **MapReduce 2 ("NextGen") moves Resource Management into YARN**


MapReduce 1

API

Framework

Resource Management

MapReduce 2

MR API

Framework

YARN

YARN API

Resource Management

Current MapReduce vs YARN

Hadoop MapReduce

Job Tracker (Master)

- Resource management
- Job lifecycle management
  - Scheduling, progress monitoring, fault tolerance

Task Tracker (per node)

- Launch tasks
- Report status to Job tracker

Hadoop YARN

Resource Manager (Master)

- Resource management
- Scheduling

Application Master (per app)

- Job lifecycle management

Node Manager (per node)

- Launch Containers
- Monitor resource usage
- Report to RM

MapReduce itself is an Application on YARN

Other Hadoop Tools

Pig

- Pig is a platform for analysing large data sets that consists of a high-level language for expressing data analysis programs

- Consists of :
  - ✓ PigLatin : The high-level language
  - ✓ A Run-time environment where PigLatin programs are executed

HBase

- Column-oriented database management system on top of HDFS
- HBase system comprises a set of tables
- Allows Attributes (Columns) to be grouped together into “column families”
- All the elements of a column family are all stored together

Other Hadoop Tools (contd..)

Hive

- A data warehouse infrastructure built on top of Hadoop
- Provides tools to enable :
  - ✓ Easy data summarization
  - ✓ Ad hoc querying
  - ✓ Analysis of large datasets stored in Hadoop files

Sqoop - SQL To Hadoop

- Sqoop is a command-line tool with the following capabilities:
  - ✓ Imports tables or entire databases to files in HDFS
  - ✓ Generates Java classes to allow you to interact with imported data
  - ✓ Imports from SQL databases straight to Hive data warehouse

Other Hadoop Tools (contd..)

ZooKeeper

- A high-performance coordination service for distributed applications
- An open source Apache project
- Provides a centralized infrastructure and services that enable synchronization across a cluster
- Maintains common objects needed in large cluster environments
- Each client machine communicates with one of the many ZooKeeper servers in a cluster to retrieve, update its synchronization information

Other Hadoop Tools (contd..)

Oozie

- Is a workflow/coordination service to manage data processing jobs for Apache Hadoop
- Supports all types of Hadoop jobs and is integrated with the Hadoop stack
- Users can specify execution frequency and can wait for data arrival to trigger an action in the workflow

Other Hadoop Tools (contd..)

Flume

- Service for efficiently moving large amounts of data soon after the data is produced
- It is centrally managed & allows for intelligent dynamic management
- Its main goal is to deliver data from applications to Hadoop's HDFS

Mahout

- A scalable machine learning and data mining library
- An Apache project to produce free implementations of machine learning algorithms on the Hadoop platform

References

- *Understanding Big Data*- Chris Eaton, Dirk Deroos, Tom Deutsch, George Lapis, Paul Zikopaulos
- *Hadoop: The Definitive Guide*-O'Reilly
- *Hadoop In Action*-by Chuck Lam
- archive.cloudera.com/cdh3/hue
- github.com/nathanmarz/cascalog
- archive.cloudera.com/cdh3/hue
- http://en.wikipedia.org/wiki/Pentaho
- http://www.pentaho.com/hadoop/
- http://katta.sourceforge.net/
- http://developer.yahoo.com/hadoop/tutorial/module4.html

Thank You

Critical situations. Ruthless competition. Unforgiving customers. Thankfully you can be absolutely sure of your IT solutions with Tata Consultancy Services (TCS). As one of the world's fastest growing technology and business solutions providers, TCS has built a reputation of delivery excellence based on world-class IT solutions that are on time, within budget and consistently deliver superior quality. So, it comes as no surprise that we pioneered the concept of the Global Network Delivery Model. Developed Innovation Labs and Solution Accelerators. Achieving a level of delivery excellence that provides greater value to our customers and is the industry benchmark. Enabling our clients to experience certainty.

TATA CONSULTANCY SERVICES

Experience certainty.

IT Services • Business Solutions • Outsourcing

To learn how your business can experience certainty, visit www.tcs.com