
SCTE®
CABLE-TEC EXPO®
FAST FORWARD 2021
ATLANTA, GA
OCTOBER 11-14
SCTE
a subsidiary of CableLabs®
UNLEASH THE POWER OF LIMITLESS CONNECTIVITY
2021 Fall Technical Forum
SCTE® • NCTA • CABLELABS®

SCTE
a subsidiary of CableLabs®

Cloud & Virtualization

Designing a Cloud-Based DOCSIS Time Protocol Calibration Database

Roy Sun

Ph.D., Lead Architect
CableLabs

2021 Fall Technical Forum
SCTE • NCTA • CABLELABS

SCTE CABLE-TEC EXPO FAST FORWARD 2021
VIRTUAL EXPERIENCE
OCTOBER 11-14

Agenda

Introduction

- DOCSIS Time Protocol (DTP) history, status and gap
- Cloud database fills the DTP gap

DTP Calibration Method

- Why DTP needs calibration?
- How to do the DTP calibration

Cloud database design

Database application programming interface (API) design: message flow and data structure

Conclusion

What is DTP?

- DTP provides high-accuracy synchronization signal on HFC network
- Exchange DTP messages between CMTS and CM that correct DOCSIS 3.1 timestamp time error (TE) due to ranging, topology, equipment impairments and configurations
- Enable HFC network to meet 3GPP requirements as mobile backhaul

DTP history and status

- Invented in 2011 by John Chapman, et al., from Cisco
- Adopted by the DOCSIS Synchronization Specification in April 2020
- Prototyping CMTS and CM solutions available in 2020
- Proof-of-concept (PoC) testing started from Q3 2020
  - CableLabs, Charter Communications, Cisco and Hitron

Gap of DTP

- DTP requires calibration in the lab
- *Needs a cloud database distributes the calibration data*

How DTP Works?

DOCSIS 3.1 timestamp (CMTS sends to CM)

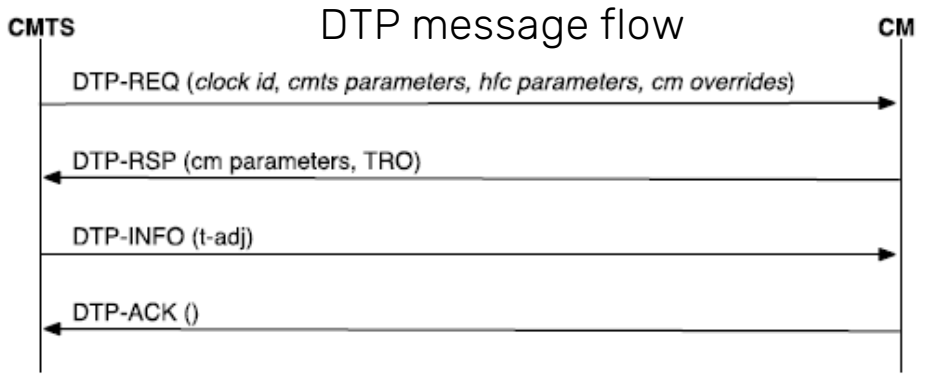
- The DOCSIS 3.1 timestamp is delayed in the HFC network downstream (DS) when arrived the CM *Inaccurate anymore!*

DTP corrects the DOCSIS 3.1 timestamp

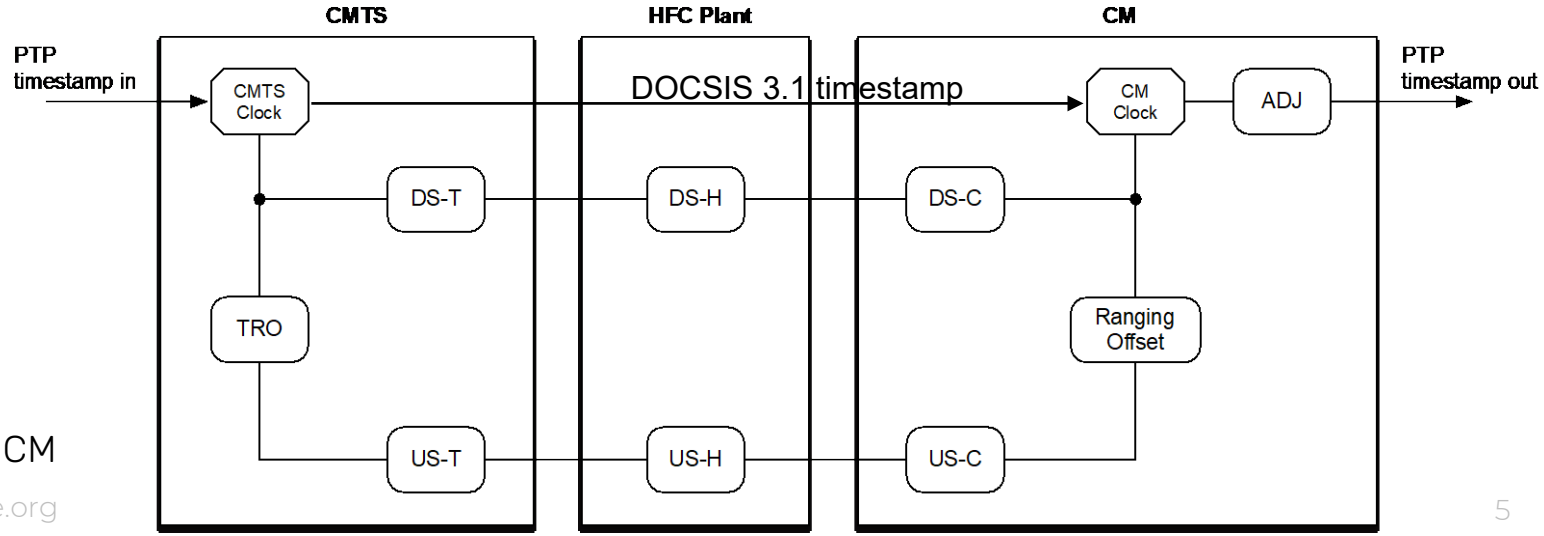
- Uses the true ranging offset (TRO): round-trip delay
- TRO cannot derive the DS delay *DS/US asymmetry*

Asymmetry calibration

- Ideally, timing parameter of CM and CMTS should be measured separately
- However, no such timing device exist
- *DTP calibration needs to be considered*



DS-T: downstream (DS) delays inside the CMTS
US-T: upstream (US) delays inside the CMTS
DS-H: DS delays in the HFC plant
US-H: US delays in the HFC plant
DS-C: DS delays inside the CM
US-C: US delays inside the CM
TRO: true ranging offset (measured by the CM)
t-adj: time adjustment that the CMTS sends to the CM



DTP Calibration Methods

DTP timing measurement method

- Again, no timing measurement device exist that supports DTP
- DTP timing measurement is based on Precision Time Protocol (PTP)
  - TE between the PTP timestamps as input of the CMTS and output of the CM

DTP calibration method

1. Collects DTP calibration data in the lab
2. Distribute the calibration data using a cloud database
3. CMTS applies the calibration data in the field

DTP calibration data collection

- Required for each pair of CMTS and CM, depends on hardware & firmware versions and key HFC network configurations
- Two options to collect the calibration data
  1. By CMTS and CM manufacturers or MSOs
  2. By a third-party lab, e.g., CableLabs/Kyrio established a Network Timing Lab

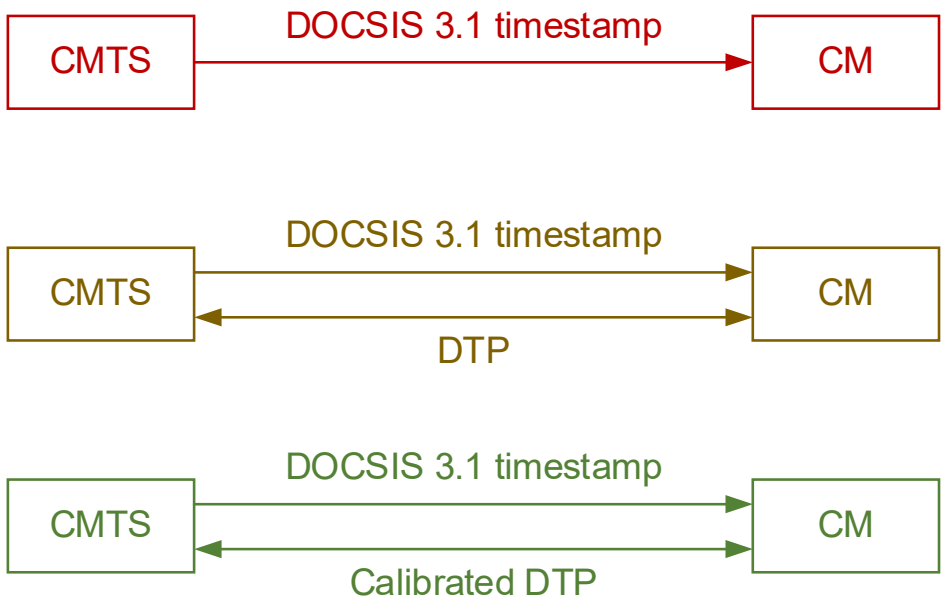
DTP calibration in the lab

- Find the asymmetrical delay: *additional time adjustment*

DTP in the field

- Use TRO to correct symmetrical delay
- Use *additional time adjustment* to correct asymmetrical delay

DTP Calibration



DOCSIS 3.1 timestamp includes the downstream time delay due to HFC plant length and equipment impairment.

DTP improves DOCSIS 3.1 timestamp accuracy:

- Using TRO that corrects the symmetrical time error;
- Still has asymmetrical time error.

Calibrated DTP further improves DOCSIS 3.1 timestamp accuracy:

- Using TRO that corrects the symmetrical time error;
- DTP calibration corrects the asymmetrical time error.

For example, DTP calibration reduced the time error from 3,223,800 ns to 13-31 ns

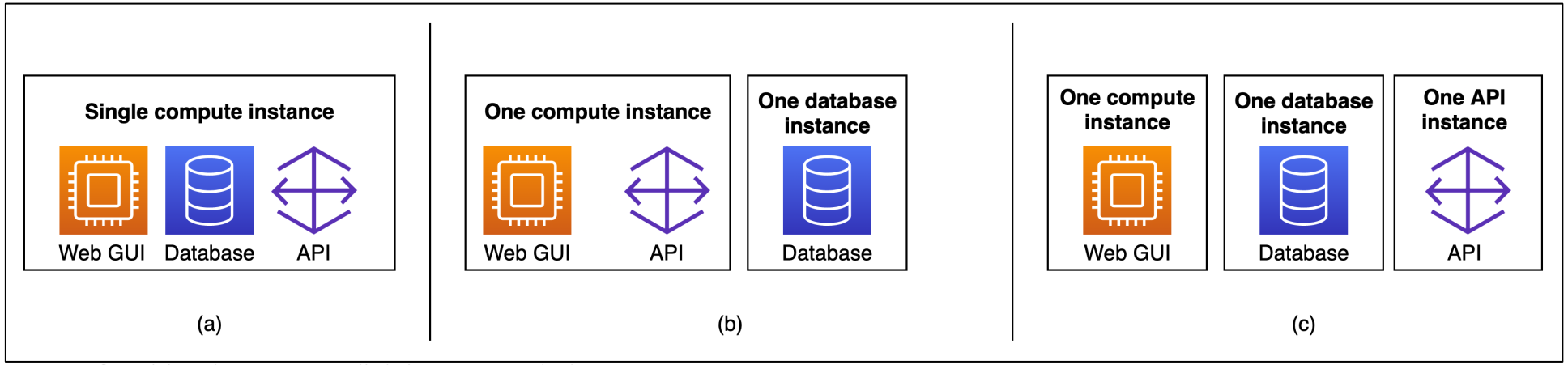
DTP Calibration Cloud Database Design

Choosing cloud database platform

- The application can be implemented on multiple cloud platforms
- We chose to prototype our application on Amazon Web Services (AWS)
- The architecture design will follow the AWS Well-Architected Framework

Primary components of the DTP calibration cloud database:

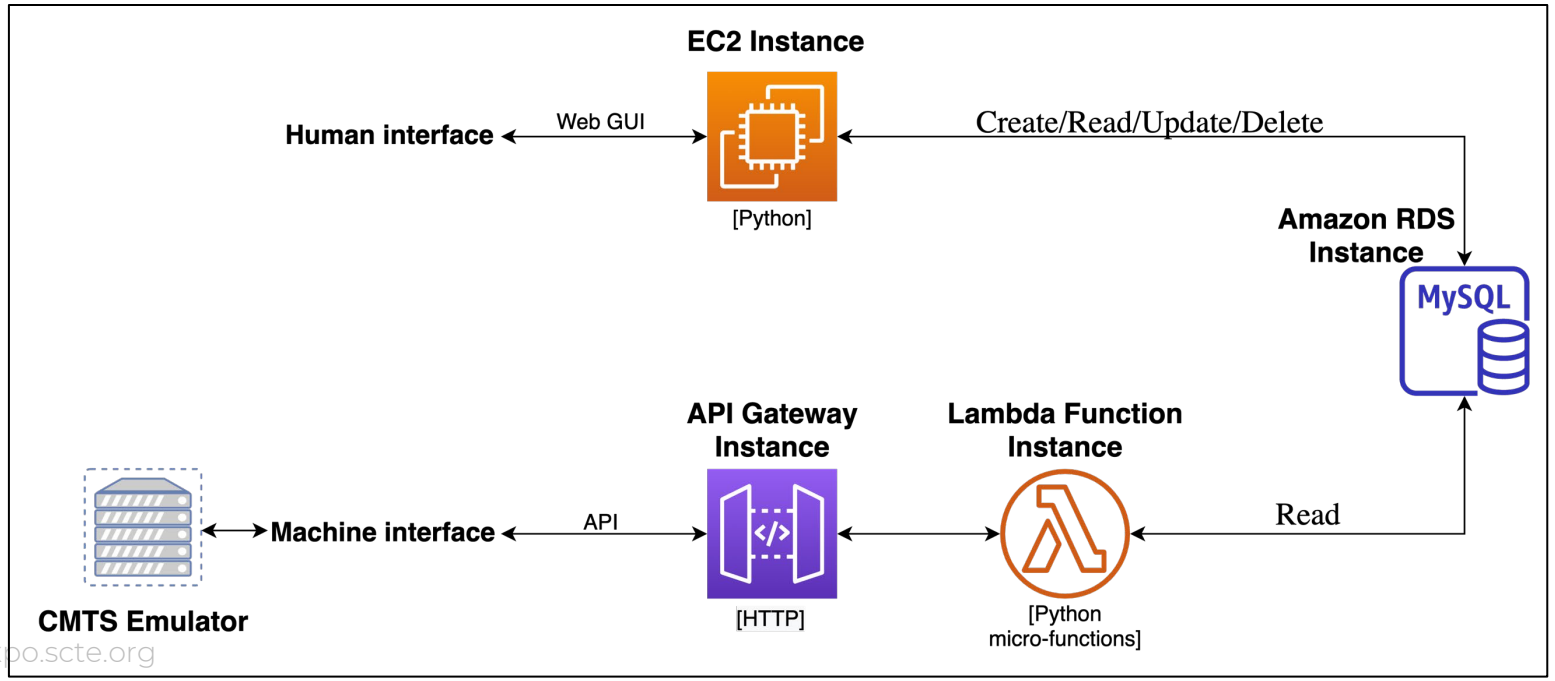
- a database
- a web-based graphical user interface (GUI) to provide a human interface for the lab engineer to add, read, and delete the DTP calibration entries
- an API framework to provide a machine interface for the CMTS to fetch the DTP calibration entries



(c) is used as all three components are implements in their own module

DTP Calibration Cloud Database Design

- The database is hosted using Amazon Relational Database Service (RDS)
- The web GUI is running on an Elastic Compute Cloud (EC2) instance
- The API framework includes Amazon API Gateway as the frontend and AWS Lambda service as the backend
- To validate the end-to-end functionality of the cloud application, a CMTS emulator was developed to test sending requests and receiving responses from the cloud application.



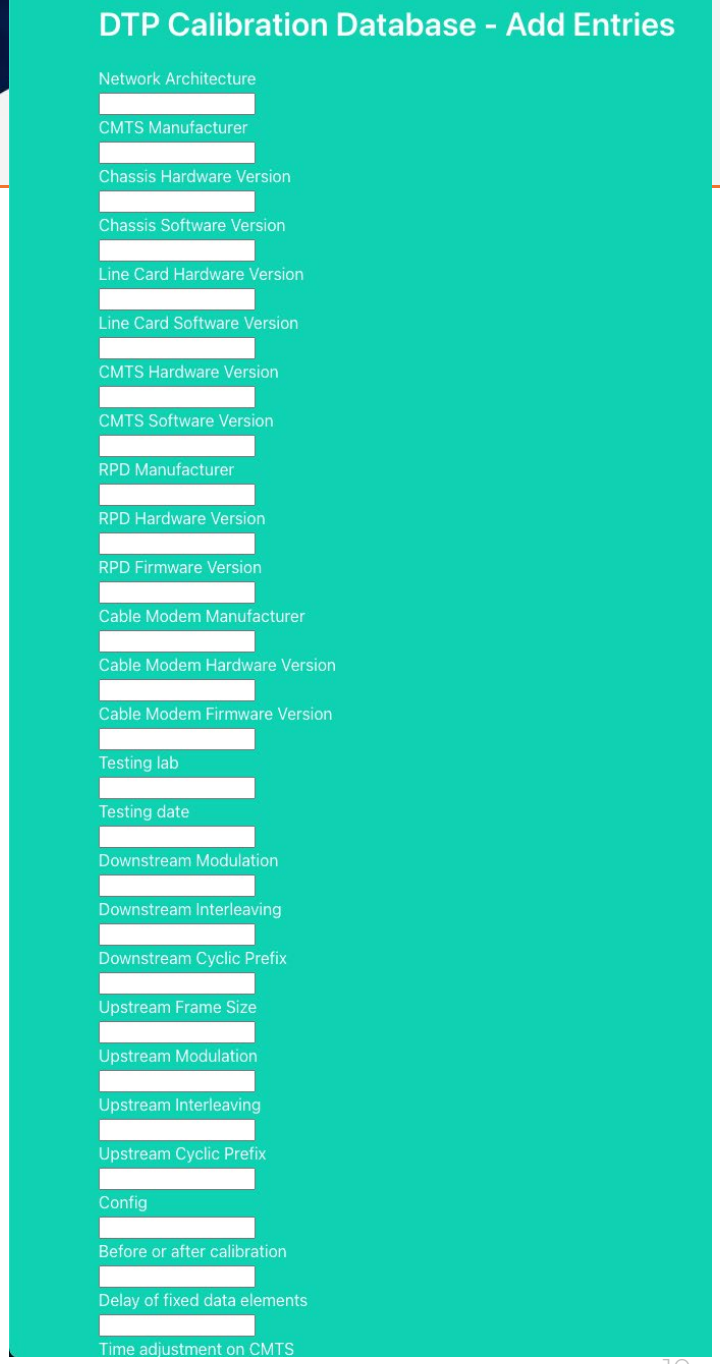
Web GUI

Implemented in Python using Flask

- Allows for easy addition of libraries or plugins for an extension
- Comes with a built-in development server and fast debugger
- Additional modules for handling forms and enabling login using username and passwords were implemented using Flask extensions

The web GUI provides the user with the capability to

- add new calibration entries to the database
- read existing entries from the database
- update existing entries in the database
- delete any existing entry from the database



Database

- Since the structure of the data to be stored in the database is not expected to change frequently, a relational database was selected as the database type
- Amongst the different relational database engines available, MySQL was chosen as it is open-source and provides sufficient flexibility to run on any operating system
- A Python script was developed to instantiate and create the database schema allowing for any possible changes in the future
- Database reliability could be enhanced by utilizing the RDS Multi-AZ (Availability Zone) functionality provided by AWS wherein a standby database instance is automatically created in another AZ and data is synchronously replicated between the two instances

Data format

JSON is used as the payload format

- lightweight
- suitable for both human reading and machine parsing

A Lambda function is developed

- run a database read query based on the received query/payload parameters (CMTS-CM Hardware-Firmware versions)
- return the result (timing parameters) along with a valid HTTP status code

Security

Application security is considered at two levels

1. API level: Access to the API can be restricted by either using HTTP request parameters-based authorization (such as username/password) or by using token-based authorization (such as JSON Web Token, JWT)
2. Network level: Access to the virtual network where the API framework is hosted can be restricted to known CMTS IP addresses only

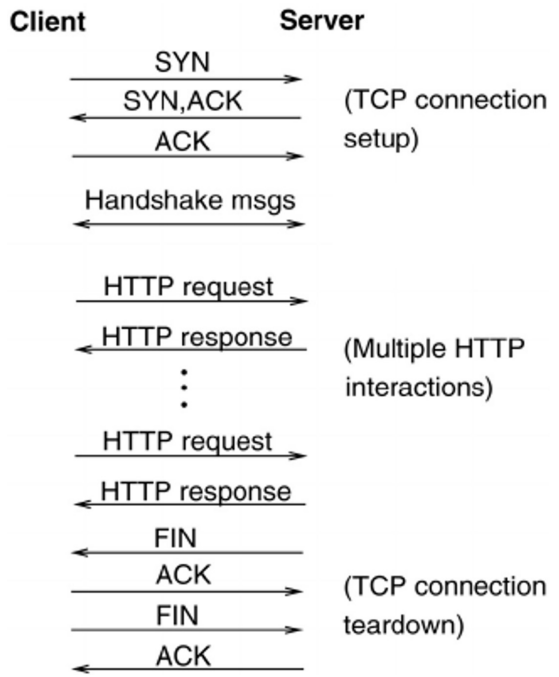
Cloud database message flow

The message flow between the AWS server and CMTS client is HTTP-based

- HTTP uses TCP as transport layer to provide reliable network transmission using acknowledgments

DTP API uses "*HTTP request*" and "*HTTP response*" to exchange information

- The CMTS (client) sends *Cal-data Request* to the server that is contained in the *HTTP Request*
  - The *Cal-data Request* message includes network architecture and hardware/firmware combinations
- The server sends *Cal-data Response* to the client that is contained in the *HTTP Response*
  - The data elements contained in the *Cal-data Response* message is listed in the next slide



Cloud database data structure

[1] Ruoyu Sun, Jennifer Andreoli-Fang, Elias Chavarria Reyes, John T. Chapman, et al., "DOCSIS Time Protocol Proof of Concept," in SCTE-Expo 2021, Atlanta, GA, October 11-14, 2021.

Cal-data Request only includes the first two data elements

- Network architecture: I-CMTS and distributed access architecture (DAA)
- Hardware/firmware combinations: make, hardware version and firmware version of I-CMTS chassis, I-CMTS line card, RPD (for DAA only) and CM

Cal-data Response includes all the data elements listed in the table

The initial considered "test configurations" include DS modulation scheme, DS interleaver, DS cyclic prefix, US modulation scheme, US cyclic prefix, and US frame size

These parameters will be adjusted in the near future depends on the DTP PoC phase 2 testing [1]

Network Architecture	Hardware & firmware combinations	Testing lab	Testing date	Test configurations	Additional time adjustment *y* (ns)	Constant Time Error (ns)
I-CMTS	Combo 1	CableLabs	11/1/2020	Config 1	200,000,000	-50
			11/1/2020		300,000,000	…
			11/2/2020	Config N	400,000,000	100
DAA	Combo 2	CableLabs	12/1/2020	Config 1	200,000,000	50
			12/1/2020		300,000,000	…
			12/1/2020	Config N	400,000,000	-100

Conclusion

DTP and DTP calibration are required

- DTP provides such sync signals in the backhaul over HFC networks
- DTP needs automated calibration in the field to guarantee time accuracy

The DTP cloud-based database fill the gap of DTP calibration

This paper presented the DTP calibration method and design of a cloud app

- The cloud app is prototyped on AWS
- The web GUI is implemented in Python using Flask allowing an engineer to add, read, and delete the DTP calibration data entries
- The API uses HTTP protocol. Calibration data message flow and data structure are designed
- The database uses JSON as the data format
- Security and reliability enhancement features will be added based on costumers' requirements

Future work of automated DTP calibration includes

- Collecting calibration data in test labs
- CMTS will need to add the corresponding feature to inquiry and apply the calibration data automatically
- Proof-of-concept test for the AWS cloud app and automated DTP calibration is planned in the near future

ATLANTA, GA
OCTOBER 11-14

SCTE
a subsidiary of CableLabs

Thank You!

Roy Sun, Ph.D., Lead Architect, *CableLabs*, r.sun@cablelabs.com

Rahil Gandotra, Mark Poletti, *CableLabs, Inc.*
Jennifer Andreoli-Fang, *Amazon Web Services (AWS)*
Elias Chavarria Reyes, *Hitron Technologies, Inc.*
John Chapman, *Cisco Systems, Inc.*