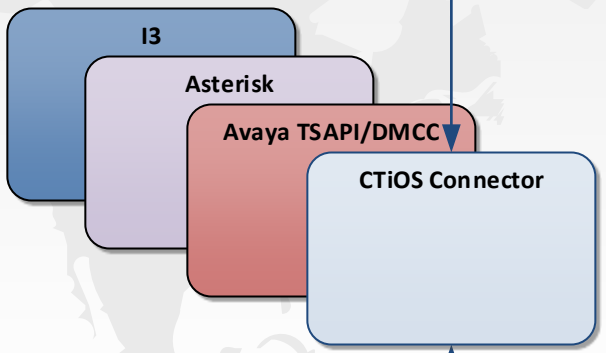
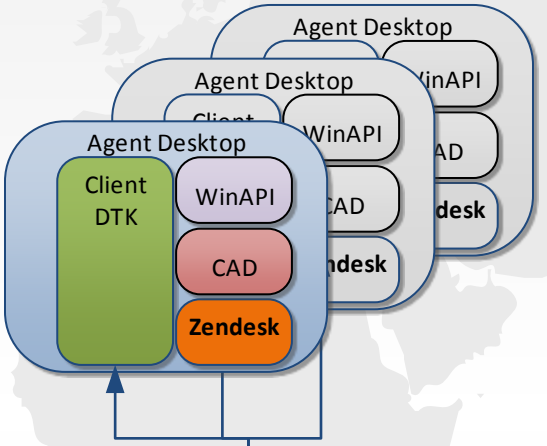
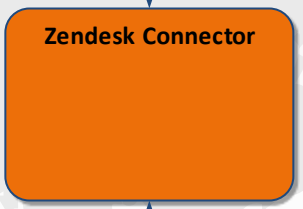
DTK and Connector Architecture

DTK Server provides a highly extensible platform by utilizing configurable “Connectors”. Each “Connector” provides specific functionality enabling rapid and flexible integration. Client DTK is an optional component complements DTK server and is implemented to support client side application integrations.







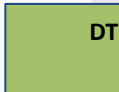
Published Telephony Events

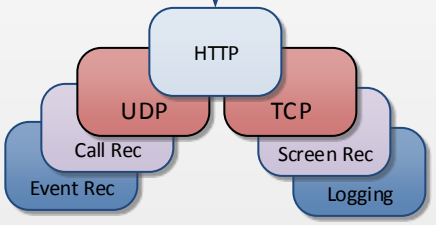
- DTK.Telephony.Call.Started
- DTK.Telephony.Call.Ringing
- DTK.Telephony.Call.Answered
- DTK.Telephony.Call.Ended
- DTK.Telephony.Call.Alerting

- DTK.Telephony.Agent.Login
- DTK.Telephony.Agent.Logout
- DTK.Telephony.Agent.Ready
- DTK.Telephony.Agent.NotReady

Subscribed Events

- DTK.Data.Update





Other DTK Connectors

Published Events

- DTK.Data.Update

Subscribed Events

- DTK.Telephony.Call.Started
- DTK.Telephony.Call.Ringing
- DTK.Telephony.Call.Answered
- DTK.Telephony.Call.Ended
- DTK.Telephony.Call.Alerting

- DTK.Telephony.Agent.Login
- DTK.Telephony.Agent.Logout
- DTK.Telephony.Agent.Ready
- DTK.Telephony.Agent.NotReady

DTK Architecture

Technological changes brought about by product updates, vendor replacement, as well as mergers and acquisitions are costly and unavoidable. DTK has been designed from the ground up to be a powerful, flexible, and cost effective integration solution that minimizes time, effort, and expense of adapting to new or different technologies.

At the heart of DTK is a core that:

- **Optimizes performance by employing an Event-centric architecture**
- **Delivers tremendous flexibility and adaptability through configurable "Connectors"**
- **Utilizes a Service Oriented Architecture**
- **Employs and facilitates agile integration strategies and practices**
- **Provides centralized logging and solution metrics**
- **Minimizes integration time and expenses**
- **Leverages decoupled architecture enabling "Connector" reusability**

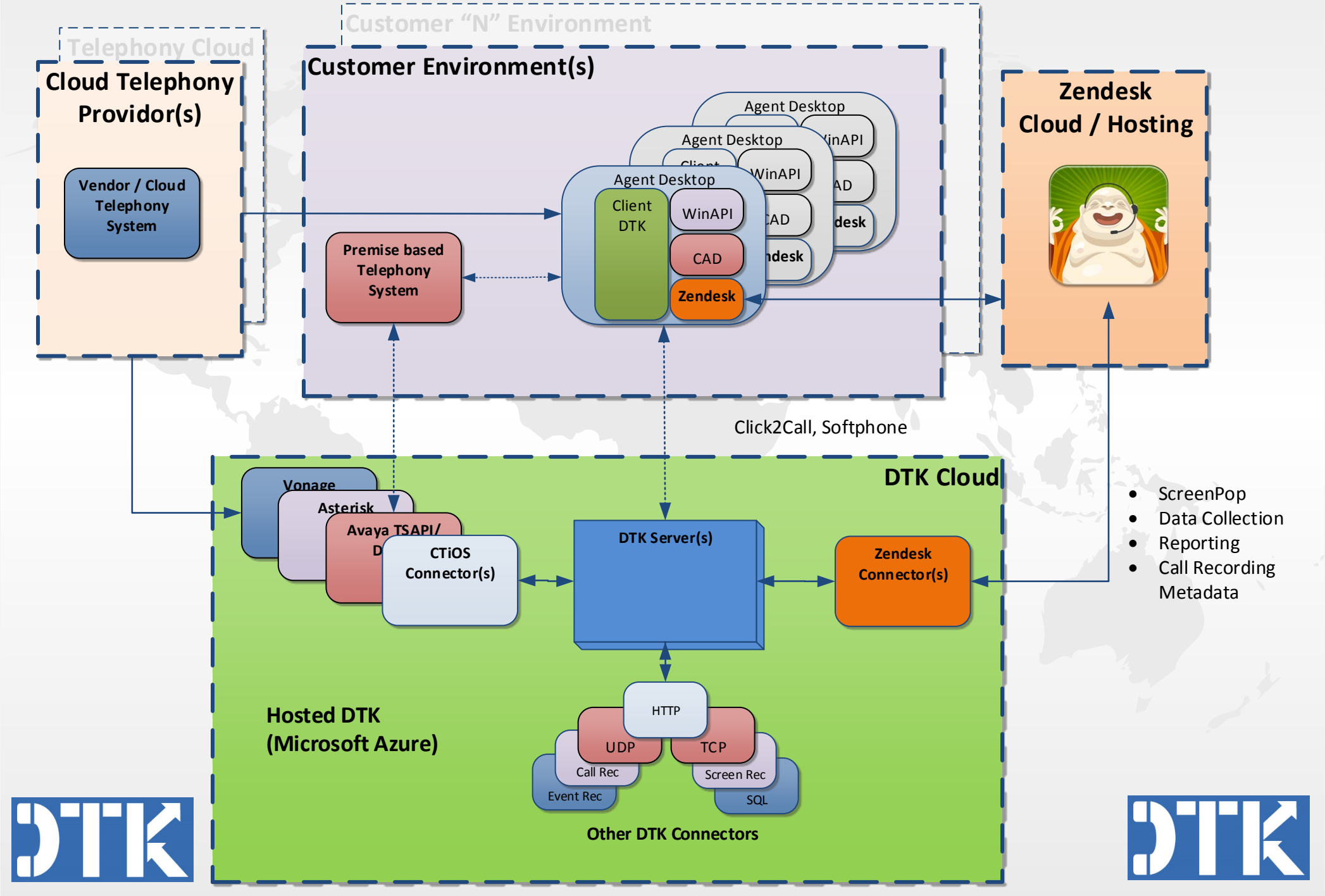
DTK Connectors are the "eyes and ears" of the DTK and are responsible for:

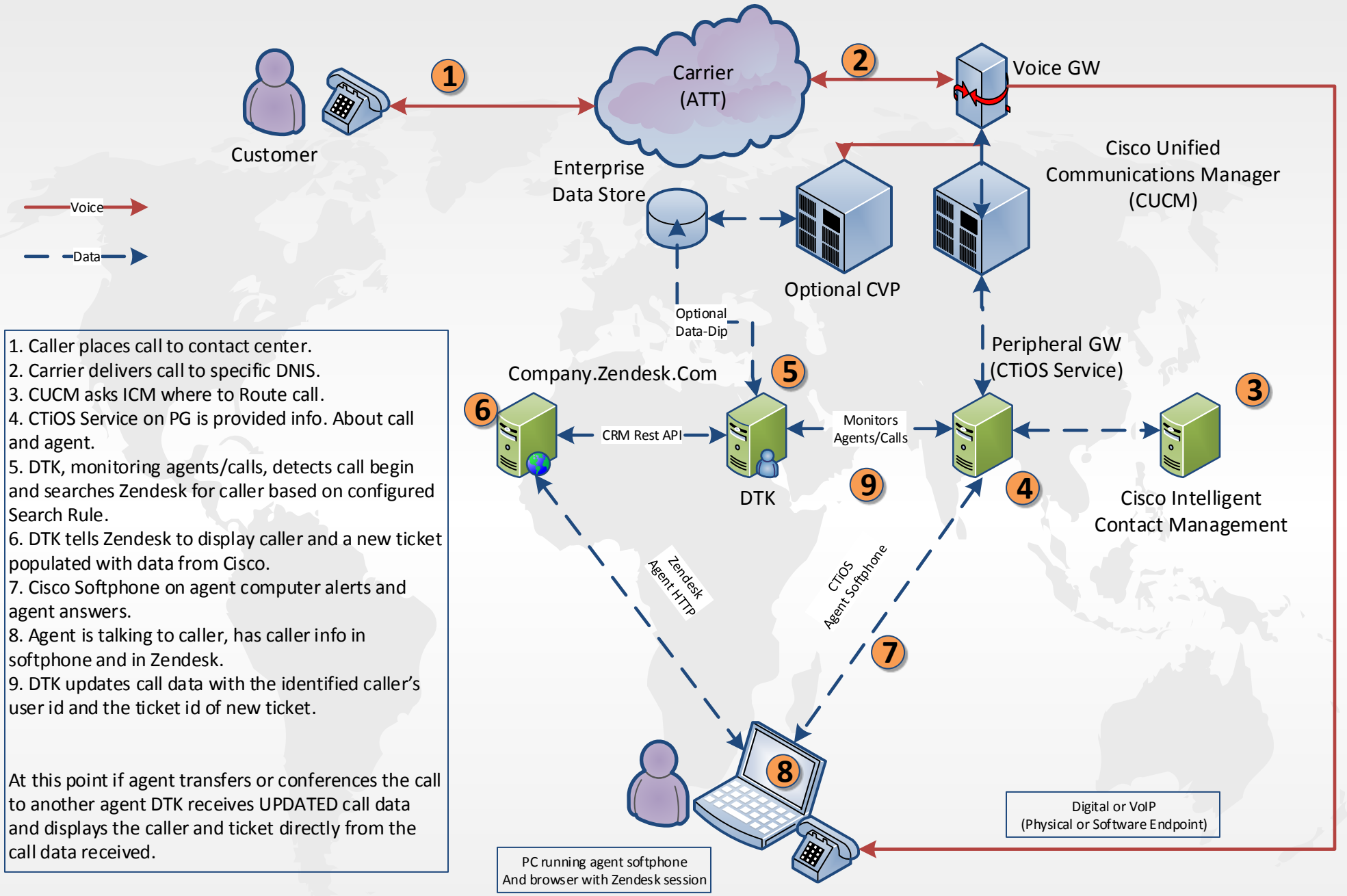
- **Producing and consuming events for specific products and applications**
- **Filtering source events to minimize system activities**
- **Providing and requesting services specific to applications**
- **Translating data between sources**
- **Encapsulation of unique system functionality**
- **Consistent event structure across various vendor telephony platforms (Cisco, Avaya, Asterisk etc.).**

"Adapt or perish, now as ever, is nature's inexorable imperative." - H. G. Wells

"It is not the strongest of the species that survives, nor the most intelligent that survives. It is the one that is the most adaptable to change." - Charles Darwin

DTK Hosted Environment

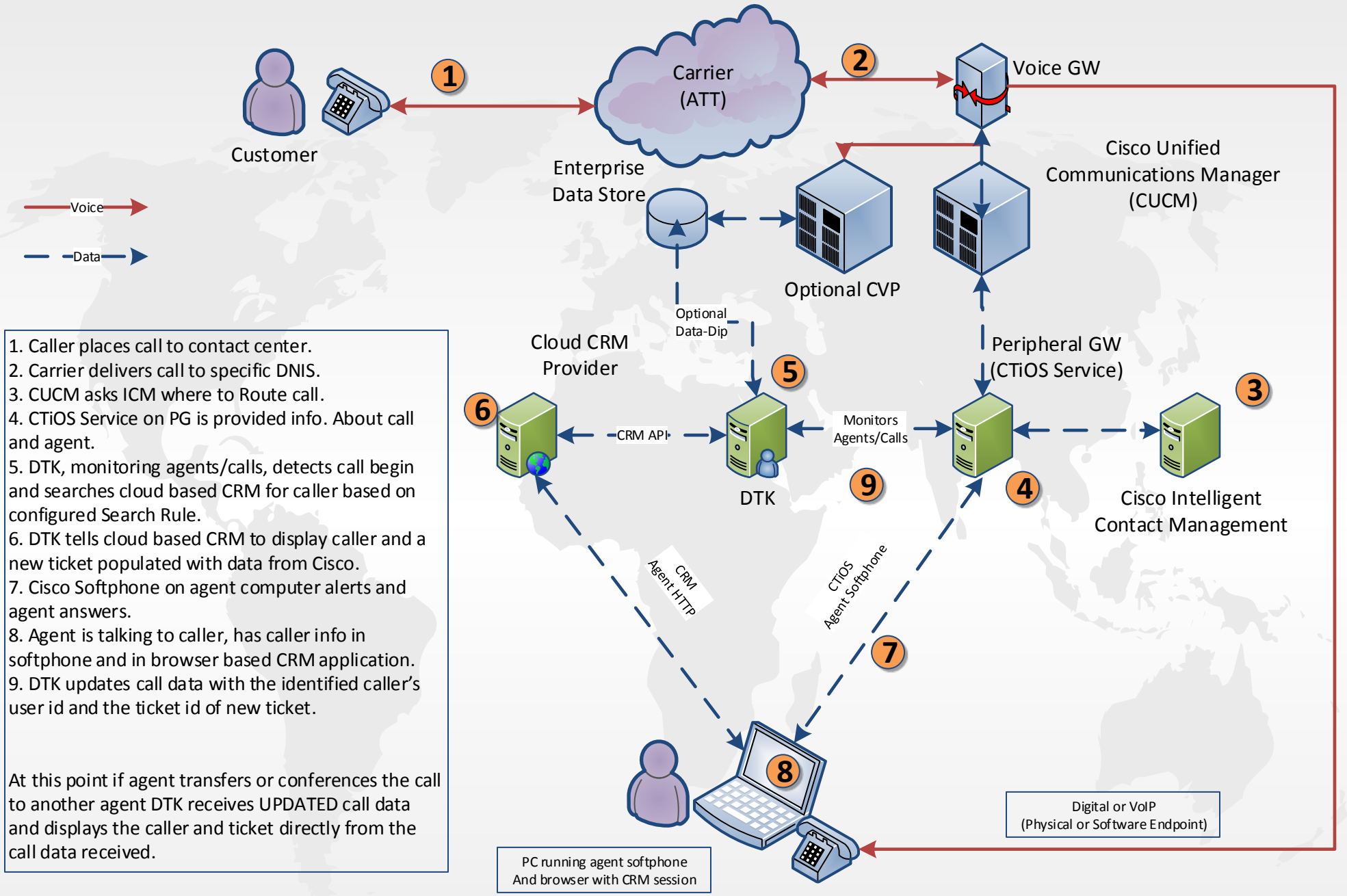




1. Caller places call to contact center.
2. Carrier delivers call to specific DNIS.
3. CUCM asks ICM where to Route call.
4. CTiOS Service on PG is provided info. About call and agent.
5. DTK, monitoring agents/calls, detects call begin and searches Zendesk for caller based on configured Search Rule.
6. DTK tells Zendesk to display caller and a new ticket populated with data from Cisco.
7. Cisco Softphone on agent computer alerts and agent answers.
8. Agent is talking to caller, has caller info in softphone and in Zendesk.
9. DTK updates call data with the identified caller's user id and the ticket id of new ticket.

At this point if agent transfers or conferences the call to another agent DTK receives UPDATED call data and displays the caller and ticket directly from the call data received.

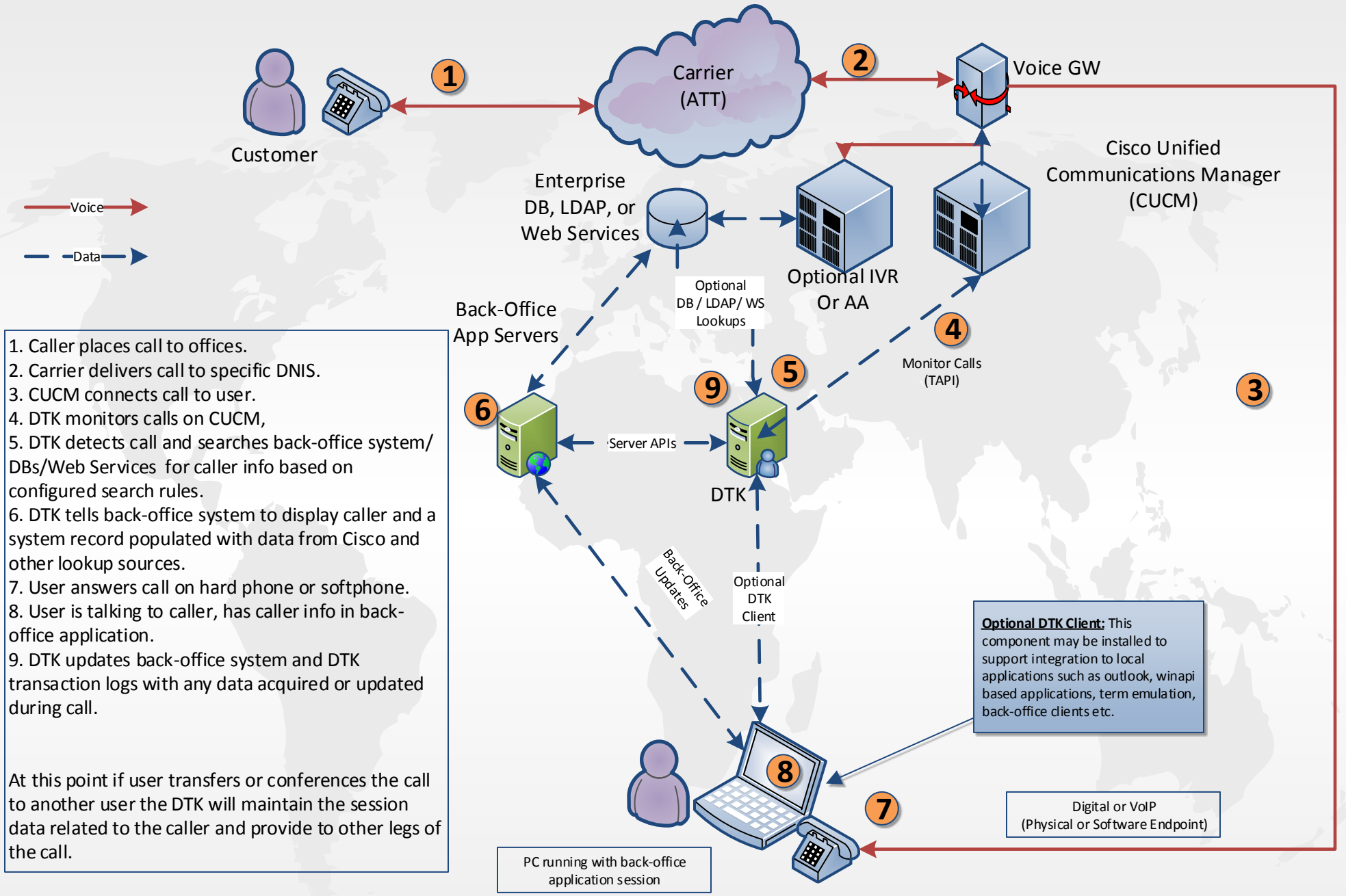
Physical Cisco UCCE / CTiOS



1. Caller places call to contact center.
2. Carrier delivers call to specific DNIS.
3. CUCM asks ICM where to Route call.
4. CTiOS Service on PG is provided info. About call and agent.
5. DTK, monitoring agents/calls, detects call begin and searches cloud based CRM for caller based on configured Search Rule.
6. DTK tells cloud based CRM to display caller and a new ticket populated with data from Cisco.
7. Cisco Softphone on agent computer alerts and agent answers.
8. Agent is talking to caller, has caller info in softphone and in browser based CRM application.
9. DTK updates call data with the identified caller's user id and the ticket id of new ticket.

At this point if agent transfers or conferences the call to another agent DTK receives UPDATED call data and displays the caller and ticket directly from the call data received.

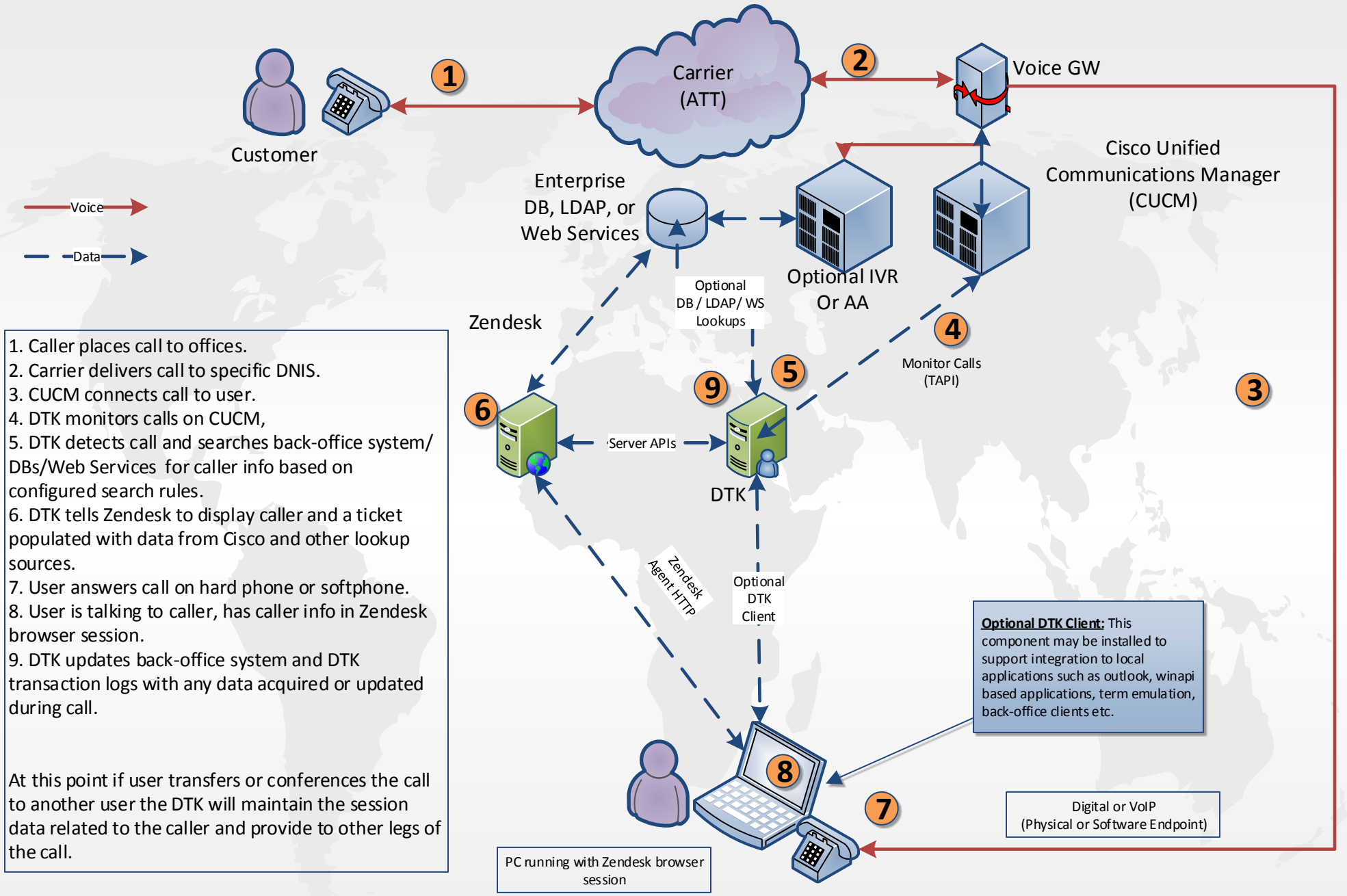
Physical Cisco UCCE / CTiOS



1. Caller places call to offices.
2. Carrier delivers call to specific DNIS.
3. CUCM connects call to user.
4. DTK monitors calls on CUCM,
5. DTK detects call and searches back-office system/ DBs/Web Services for caller info based on configured search rules.
6. DTK tells back-office system to display caller and a system record populated with data from Cisco and other lookup sources.
7. User answers call on hard phone or softphone.
8. User is talking to caller, has caller info in back-office application.
9. DTK updates back-office system and DTK transaction logs with any data acquired or updated during call.

At this point if user transfers or conferences the call to another user the DTK will maintain the session data related to the caller and provide to other legs of the call.

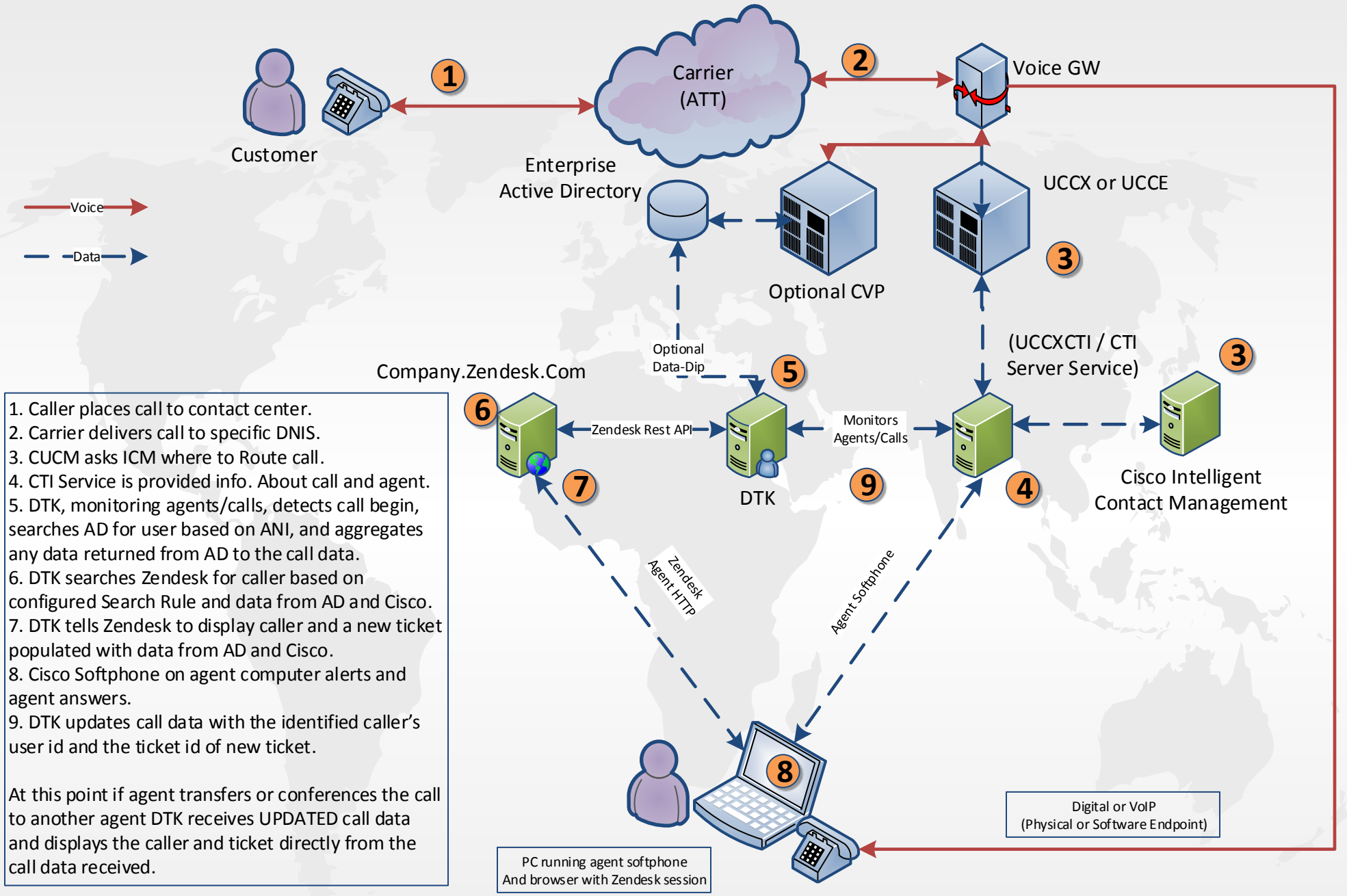
Physical CUCM



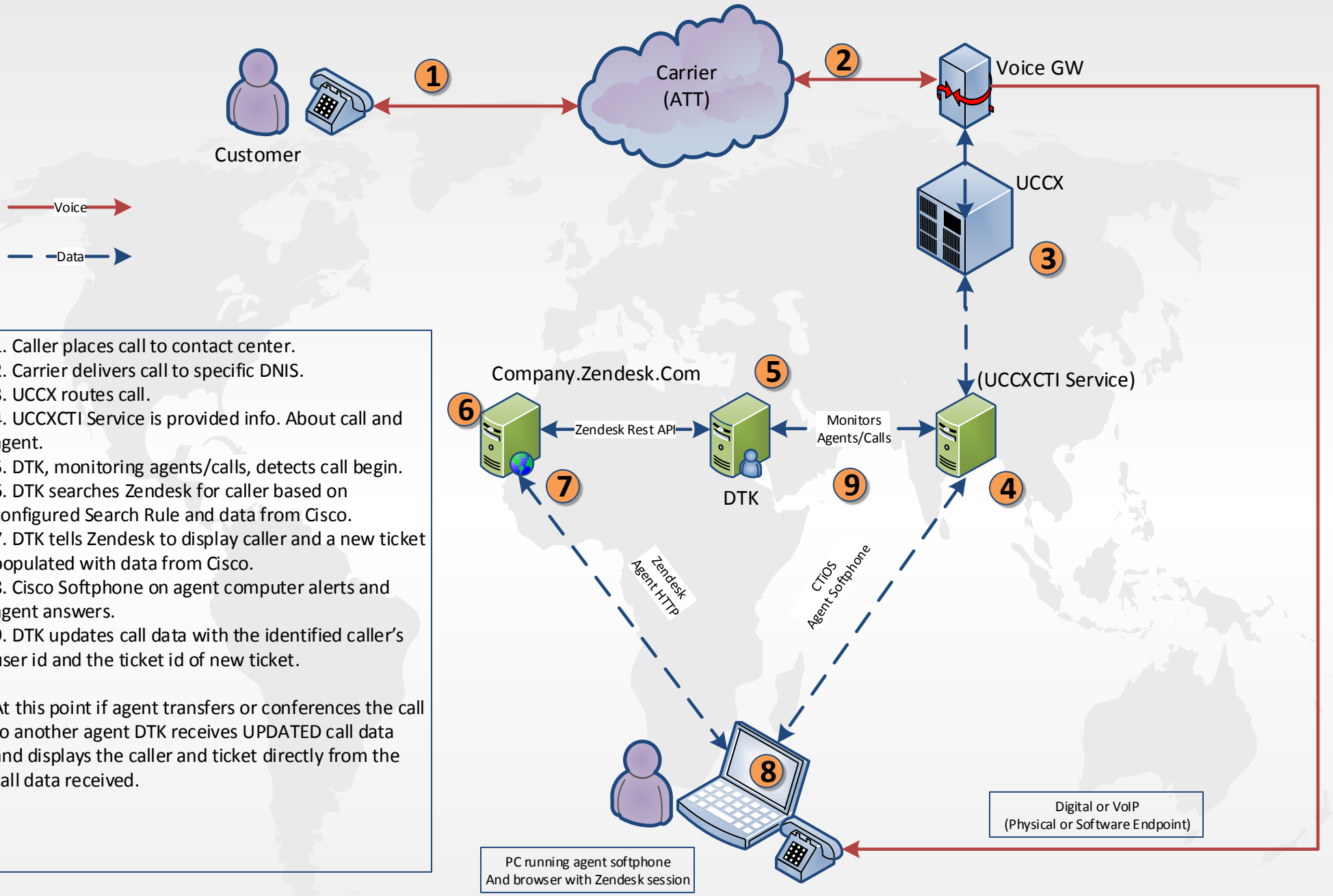
1. Caller places call to offices.
2. Carrier delivers call to specific DNIS.
3. CUCM connects call to user.
4. DTK monitors calls on CUCM,
5. DTK detects call and searches back-office system/ DBs/Web Services for caller info based on configured search rules.
6. DTK tells Zendesk to display caller and a ticket populated with data from Cisco and other lookup sources.
7. User answers call on hard phone or softphone.
8. User is talking to caller, has caller info in Zendesk browser session.
9. DTK updates back-office system and DTK transaction logs with any data acquired or updated during call.

At this point if user transfers or conferences the call to another user the DTK will maintain the session data related to the caller and provide to other legs of the call.

Physical CUCM



Physical Cisco GED 188

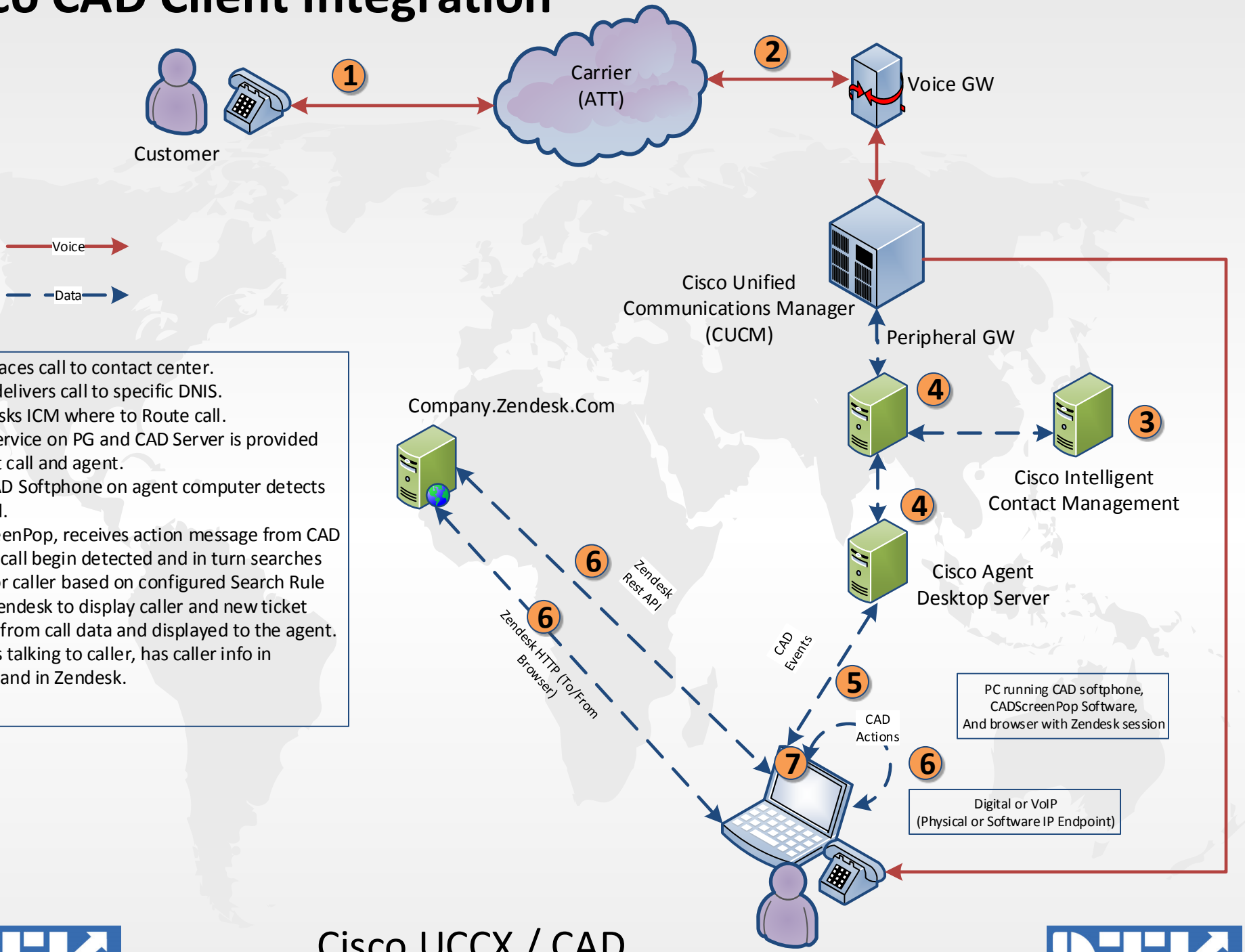


1. Caller places call to contact center.
2. Carrier delivers call to specific DNIS.
3. UCCX routes call.
4. UCCXCTI Service is provided info. About call and agent.
5. DTK, monitoring agents/calls, detects call begin.
6. DTK searches Zendesk for caller based on configured Search Rule and data from Cisco.
7. DTK tells Zendesk to display caller and a new ticket populated with data from Cisco.
8. Cisco Softphone on agent computer alerts and agent answers.
9. DTK updates call data with the identified caller's user id and the ticket id of new ticket.

At this point if agent transfers or conferences the call to another agent DTK receives UPDATED call data and displays the caller and ticket directly from the call data received.

Physical Cisco UCCXCTI

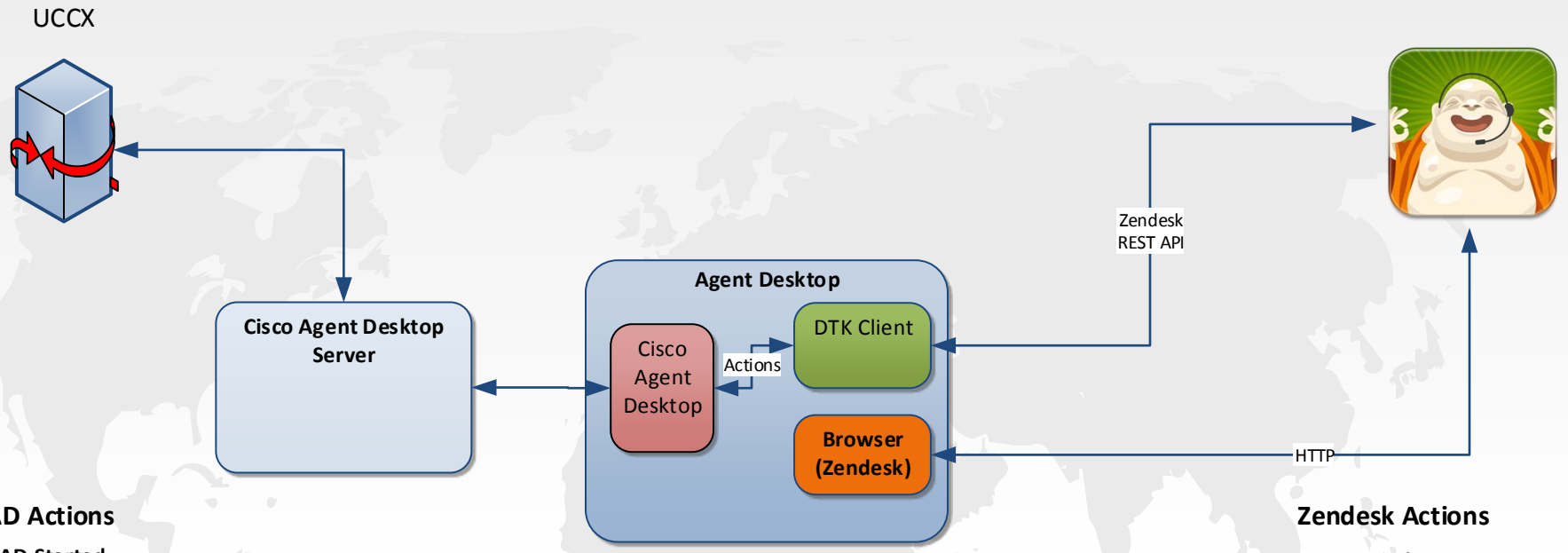
Cisco CAD Client Integration



1. Caller places call to contact center.
2. Carrier delivers call to specific DNIS.
3. CUCM asks ICM where to Route call.
4. CTiOS Service on PG and CAD Server is provided info. about call and agent.
5. Cisco CAD Softphone on agent computer detects call started.
6. CADScreenPop, receives action message from CAD client that call begin detected and in turn searches Zendesk for caller based on configured Search Rule and tells Zendesk to display caller and new ticket populated from call data and displayed to the agent.
7. Agent is talking to caller, has caller info in softphone and in Zendesk.

Cisco UCCX / CAD Client Deployment Model

Cisco CAD Client Integration



CAD Actions

- CAD Started
- CAD Agent Login
- CAD Agent State Change
- CAD Call Answered
- CAD Call Dropped
- CAD Agent Logout
- CAD Exit

Zendesk Actions

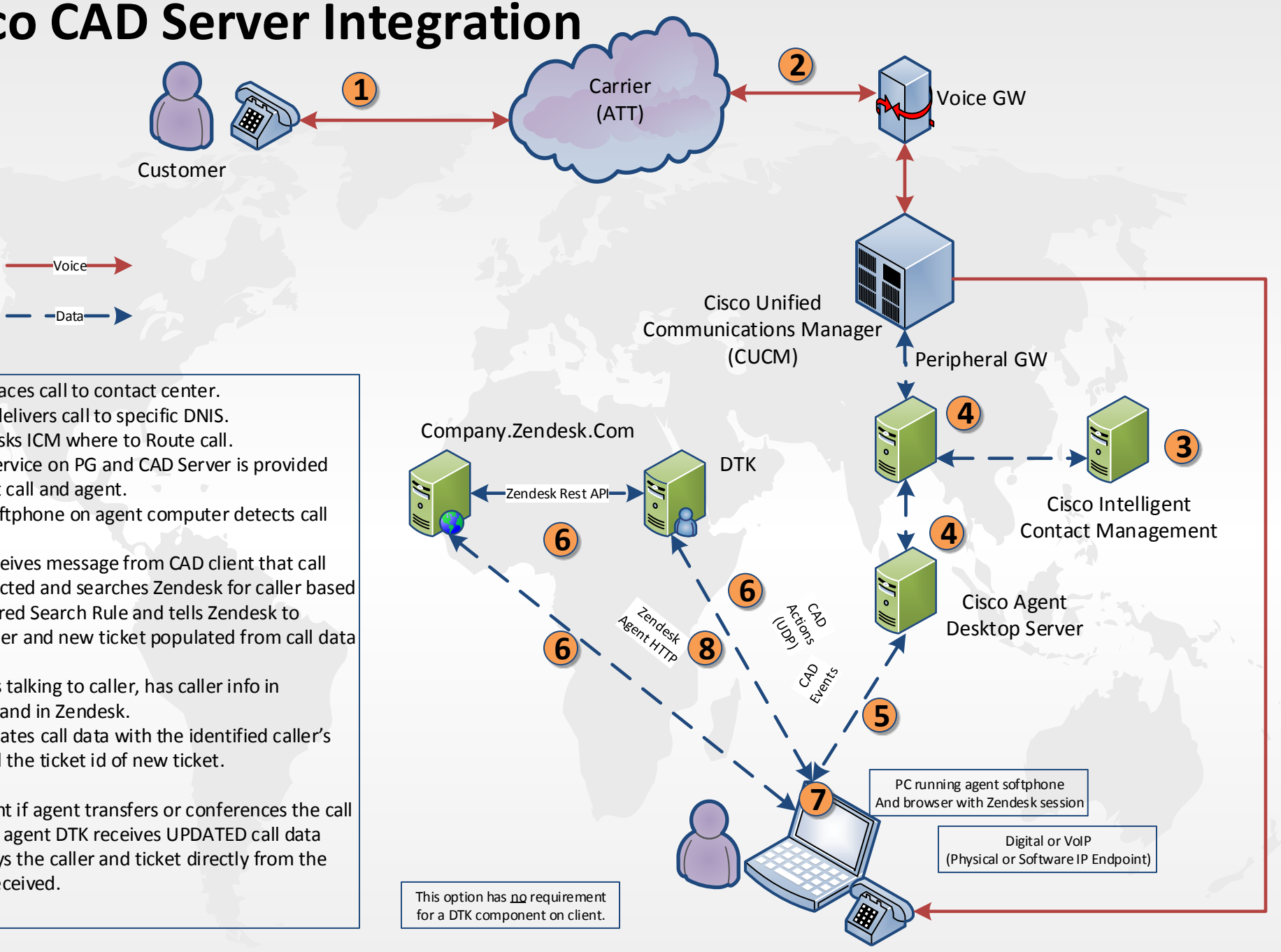
- ZEN Find User
- ZEN Create Ticket
- ZEN Create User
- ZEN Display User
- ZEN Display Ticket

CAD Action => Application Function

CAD Started => Start CADScreenPop
CAD Agent Logon => ZEN Find User
CAD Agent Ready => ZEN Find User (if needed)
CAD Agent Not Ready => ZEN Find User (if needed)
CAD Call Answered => ZEN Find User, ZEN Create Ticket, ZEN Display User, Zen Display Ticket
CAD Call Dropped => Clear CADSCreenPop call state
CAD Agent Logout => Clear CADSCreenPop agent state
CAD Exit => Shutdown CADScreenPop

Cisco UCCX / CAD Client Deployment Model

Cisco CAD Server Integration

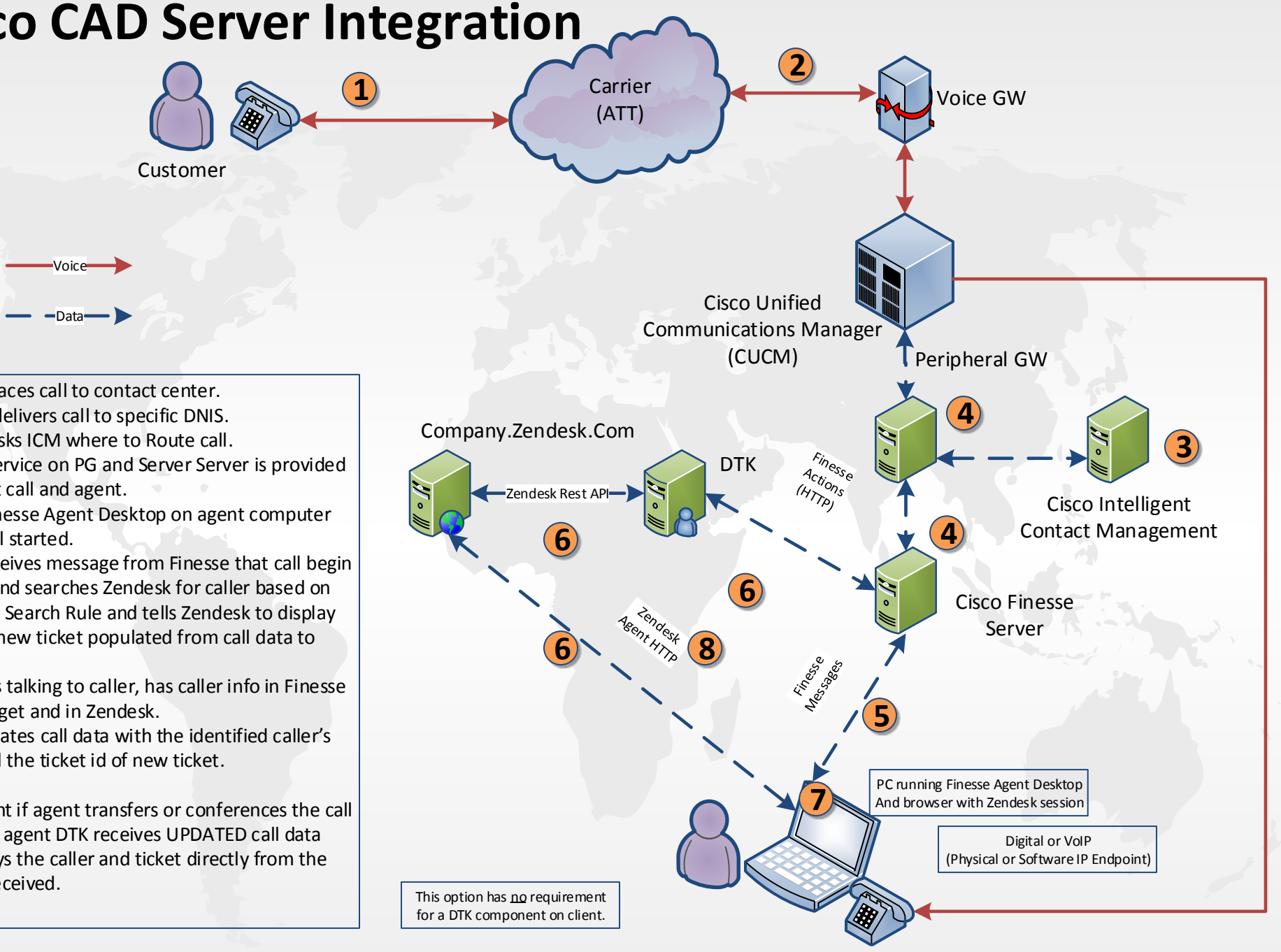


1. Caller places call to contact center.
2. Carrier delivers call to specific DNIS.
3. CUCM asks ICM where to Route call.
4. CTiOS Service on PG and CAD Server is provided info. about call and agent.
5. Cisco Softphone on agent computer detects call started.
6. DTK, receives message from CAD client that call begin detected and searches Zendesk for caller based on configured Search Rule and tells Zendesk to display caller and new ticket populated from call data to agent.
7. Agent is talking to caller, has caller info in softphone and in Zendesk.
8. DTK updates call data with the identified caller's user id and the ticket id of new ticket.

At this point if agent transfers or conferences the call to another agent DTK receives UPDATED call data and displays the caller and ticket directly from the call data received.

Cisco UCCX / CAD Server Deployment Model

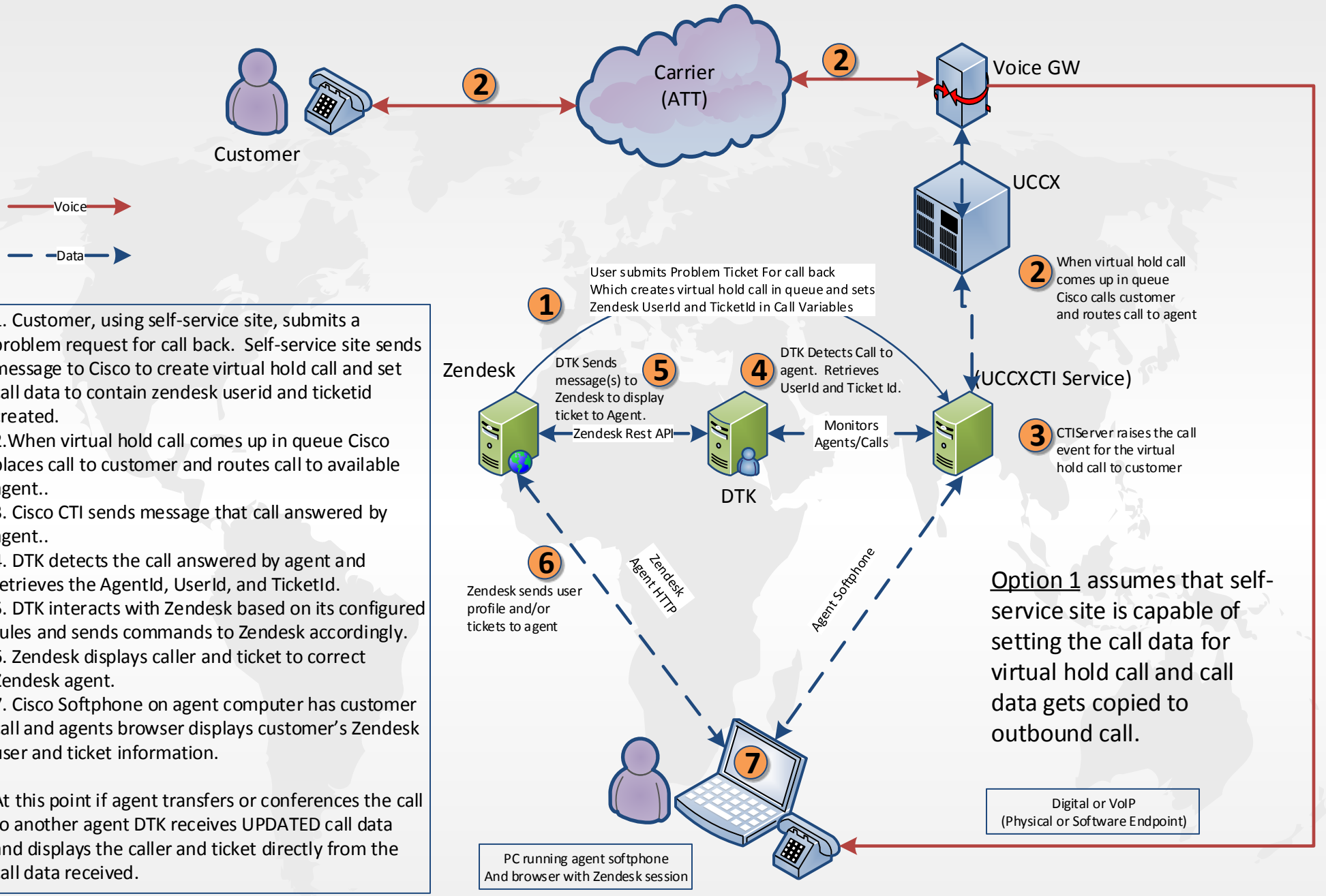
Cisco CAD Server Integration



1. Caller places call to contact center.
2. Carrier delivers call to specific DNIS.
3. CUCM asks ICM where to Route call.
4. CTiOS Service on PG and Server Server is provided info. about call and agent.
5. Cisco Finesse Agent Desktop on agent computer detects call started.
6. DTK, receives message from Finesse that call begin detected and searches Zendesk for caller based on configured Search Rule and tells Zendesk to display caller and new ticket populated from call data to agent.
7. Agent is talking to caller, has caller info in Finesse phone gadget and in Zendesk.
8. DTK updates call data with the identified caller's user id and the ticket id of new ticket.

At this point if agent transfers or conferences the call to another agent DTK receives UPDATED call data and displays the caller and ticket directly from the call data received.

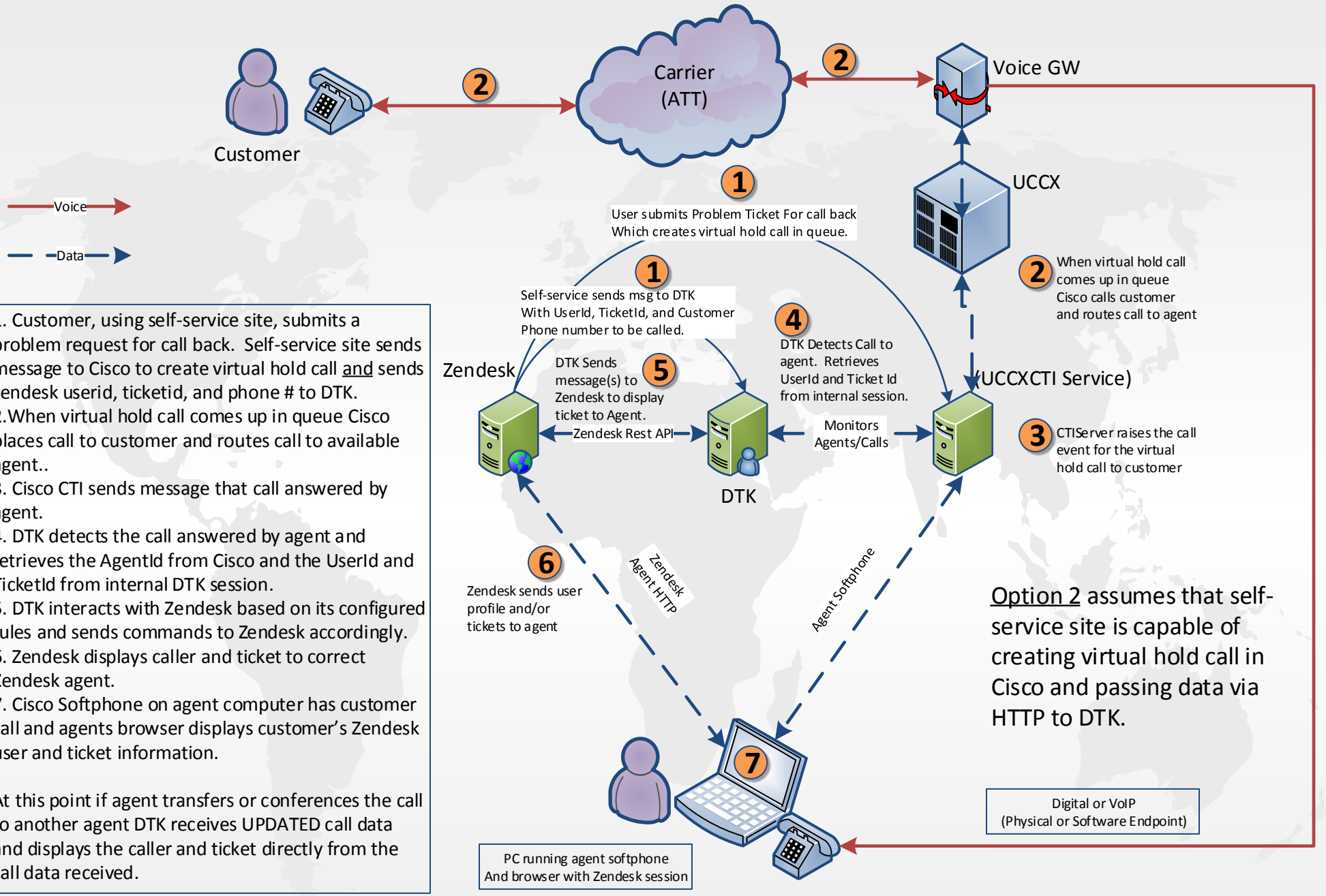
Cisco Finesse Server Deployment Model



1. Customer, using self-service site, submits a problem request for call back. Self-service site sends message to Cisco to create virtual hold call and set call data to contain zendesk userid and ticketid created.
2.When virtual hold call comes up in queue Cisco places call to customer and routes call to available agent..
3. Cisco CTI sends message that call answered by agent..
4. DTK detects the call answered by agent and retrieves the AgentId, UserId, and TicketId.
5. DTK interacts with Zendesk based on its configured rules and sends commands to Zendesk accordingly.
6. Zendesk displays caller and ticket to correct Zendesk agent.
7. Cisco Softphone on agent computer has customer call and agents browser displays customer's Zendesk user and ticket information.

At this point if agent transfers or conferences the call to another agent DTK receives UPDATED call data and displays the caller and ticket directly from the call data received.

Zendesk / Cisco Integration Opt 1

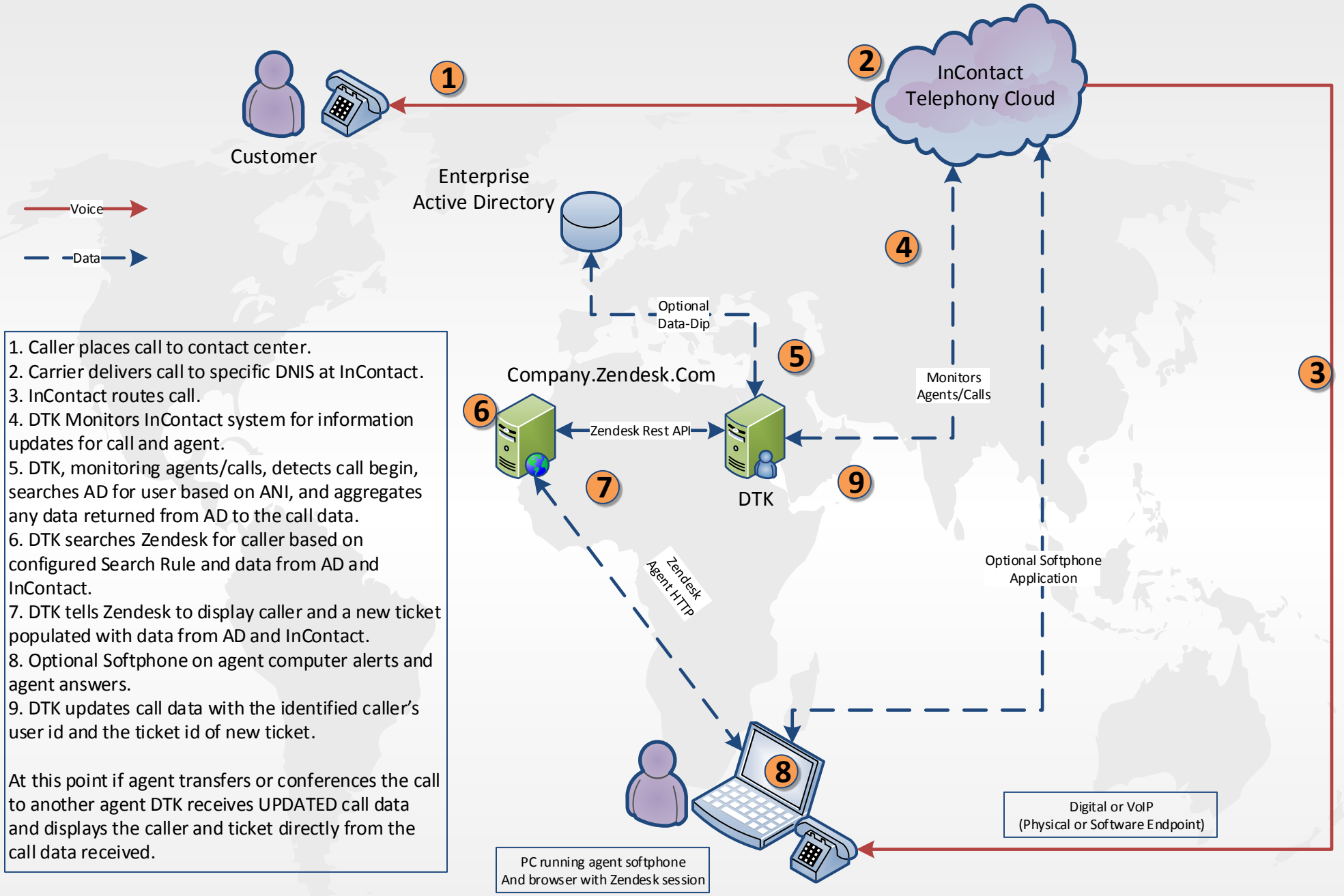


1. Customer, using self-service site, submits a problem request for call back. Self-service site sends message to Cisco to create virtual hold call and sends zendesk userid, ticketid, and phone # to DTK.
2.When virtual hold call comes up in queue Cisco places call to customer and routes call to available agent..
3. Cisco CTI sends message that call answered by agent.
4. DTK detects the call answered by agent and retrieves the AgentId from Cisco and the UserId and TicketId from internal DTK session.
5. DTK interacts with Zendesk based on its configured rules and sends commands to Zendesk accordingly.
6. Zendesk displays caller and ticket to correct Zendesk agent.
7. Cisco Softphone on agent computer has customer call and agents browser displays customer's Zendesk user and ticket information.

At this point if agent transfers or conferences the call to another agent DTK receives UPDATED call data and displays the caller and ticket directly from the call data received.

Option 2 assumes that self-service site is capable of creating virtual hold call in Cisco and passing data via HTTP to DTK.

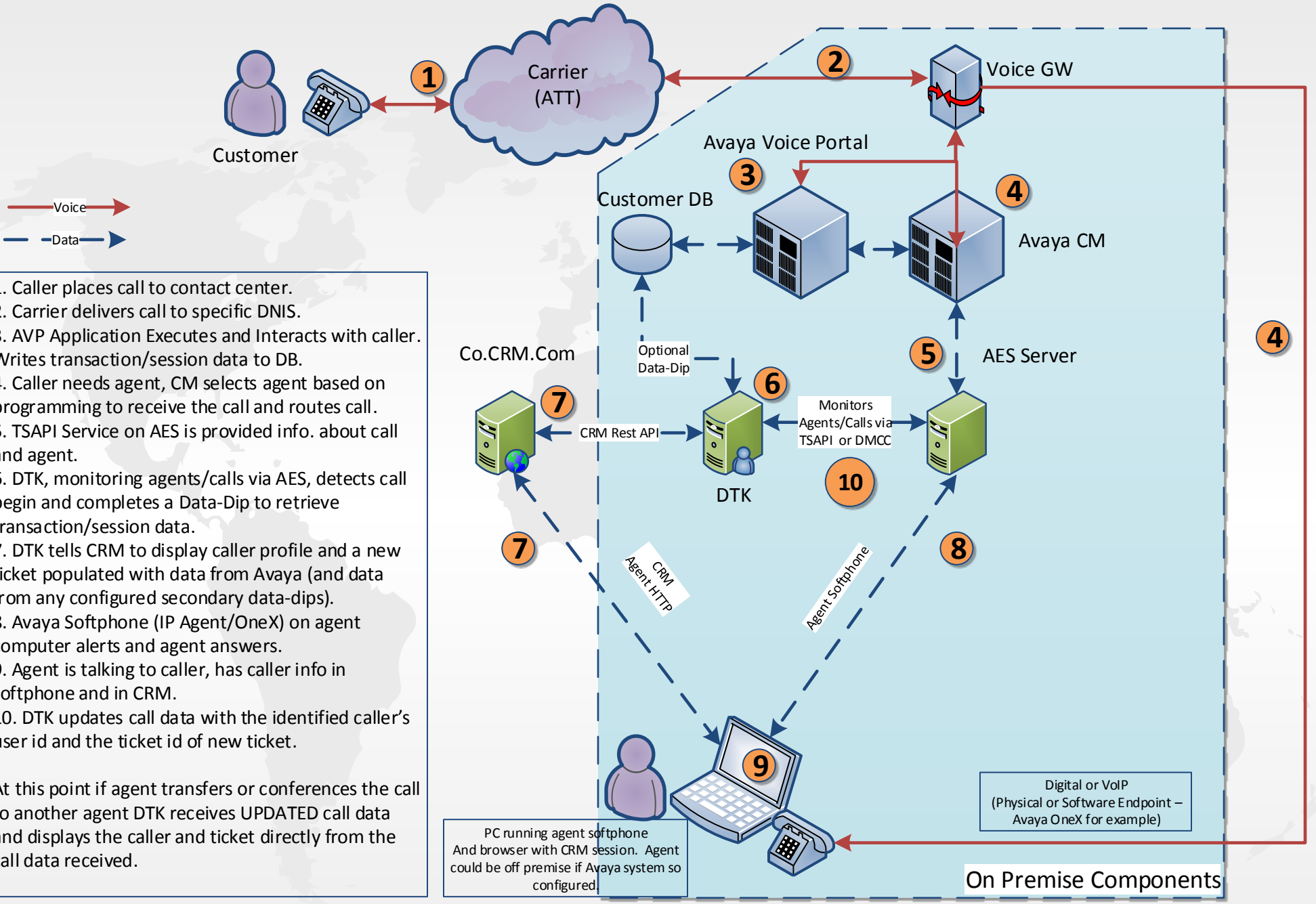
Zendesk / Cisco Integration Opt 2



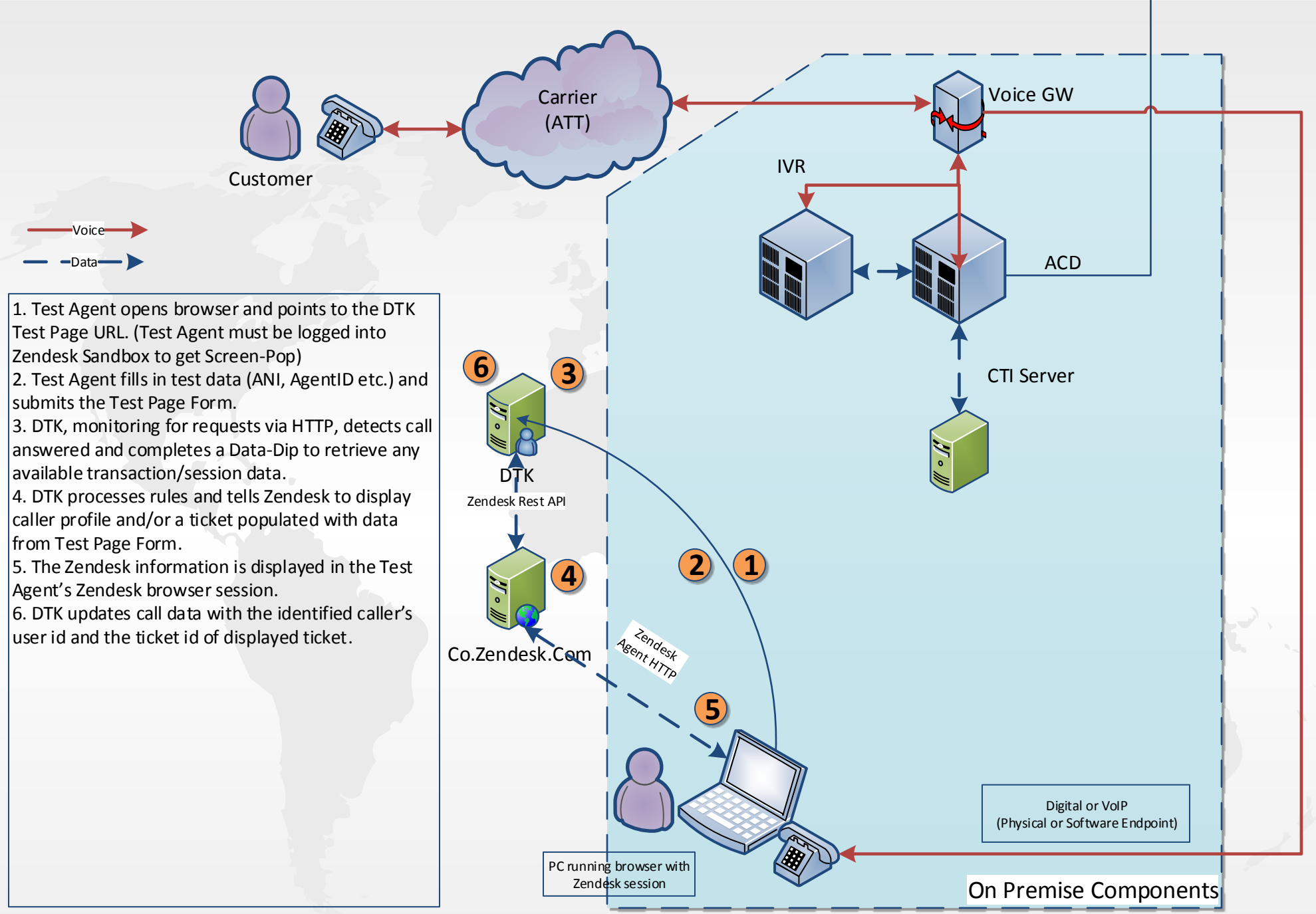
1. Caller places call to contact center.
2. Carrier delivers call to specific DNIS at InContact.
3. InContact routes call.
4. DTK Monitors InContact system for information updates for call and agent.
5. DTK, monitoring agents/calls, detects call begin, searches AD for user based on ANI, and aggregates any data returned from AD to the call data.
6. DTK searches Zendesk for caller based on configured Search Rule and data from AD and InContact.
7. DTK tells Zendesk to display caller and a new ticket populated with data from AD and InContact.
8. Optional Softphone on agent computer alerts and agent answers.
9. DTK updates call data with the identified caller's user id and the ticket id of new ticket.

At this point if agent transfers or conferences the call to another agent DTK receives UPDATED call data and displays the caller and ticket directly from the call data received.

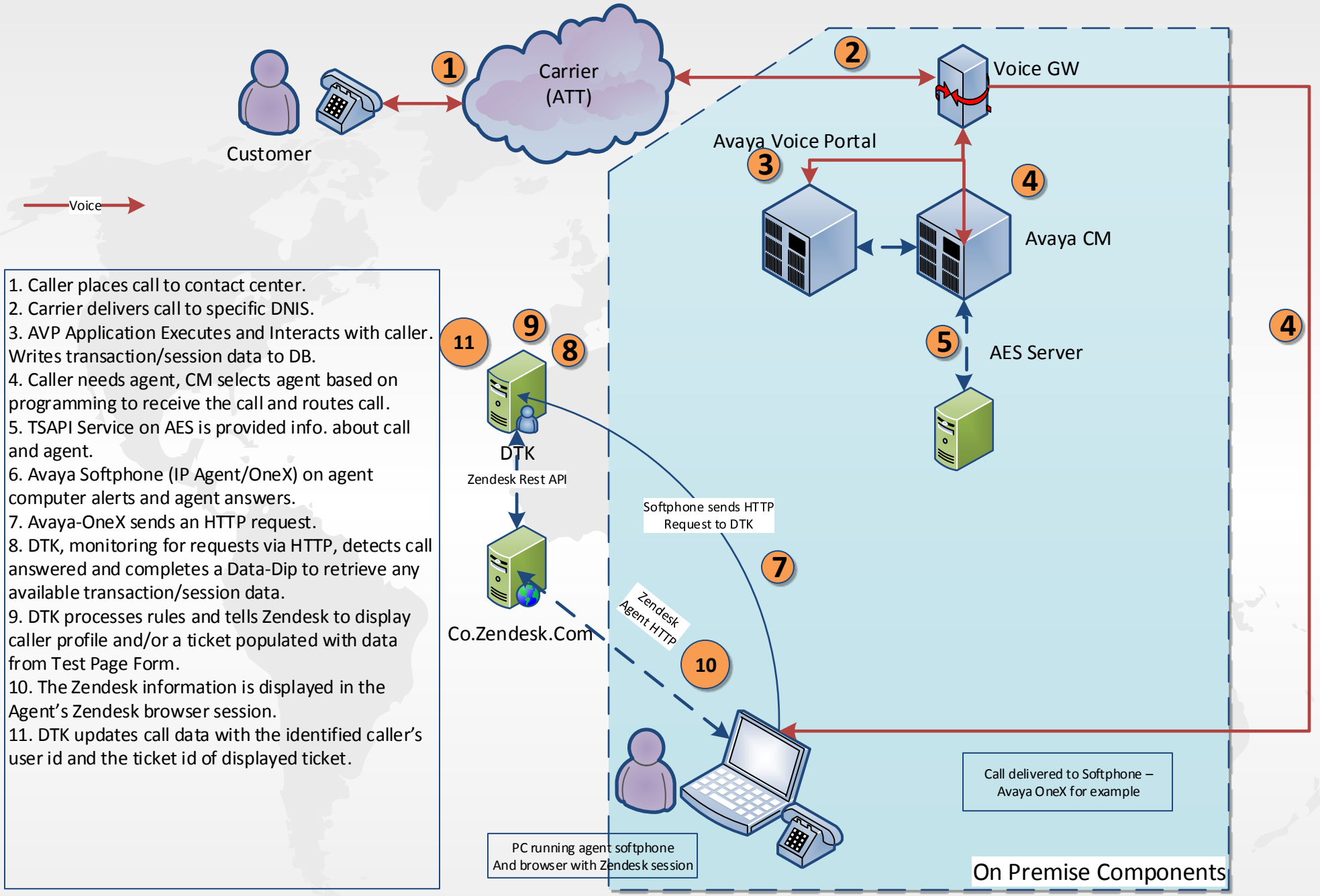
Physical InContact



Physical Avaya / AES

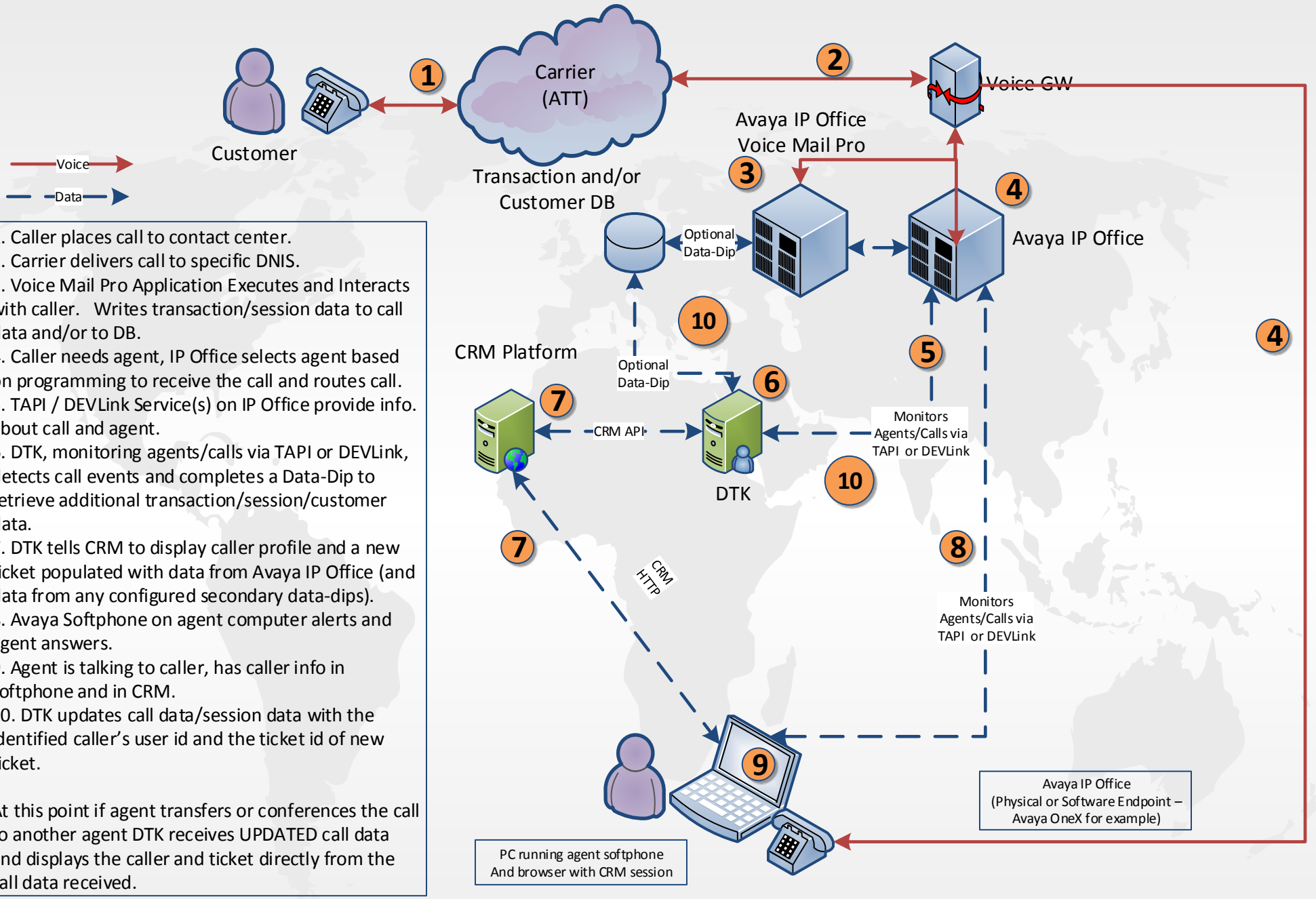


Hosted Demo / Simulation



1. Caller places call to contact center.
2. Carrier delivers call to specific DNIS.
3. AVP Application Executes and Interacts with caller. Writes transaction/session data to DB.
4. Caller needs agent, CM selects agent based on programming to receive the call and routes call.
5. TSAPI Service on AES is provided info. about call and agent.
6. Avaya Softphone (IP Agent/OneX) on agent computer alerts and agent answers.
7. Avaya-OneX sends an HTTP request.
8. DTK, monitoring for requests via HTTP, detects call answered and completes a Data-Dip to retrieve any available transaction/session data.
9. DTK processes rules and tells Zendesk to display caller profile and/or a ticket populated with data from Test Page Form.
10. The Zendesk information is displayed in the Agent's Zendesk browser session.
11. DTK updates call data with the identified caller's user id and the ticket id of displayed ticket.

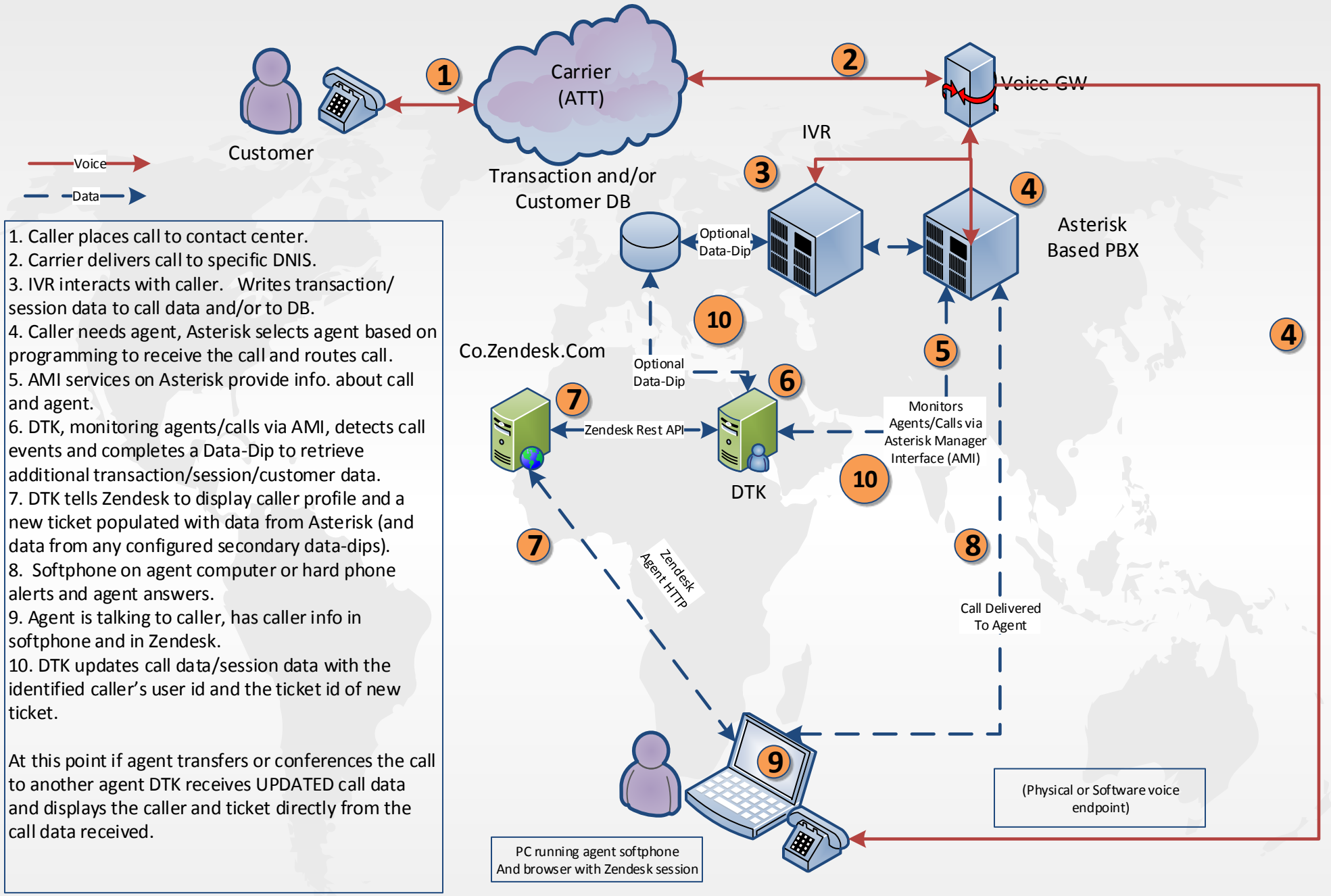
Hosted Environment via HTTP



1. Caller places call to contact center.
2. Carrier delivers call to specific DNIS.
3. Voice Mail Pro Application Executes and Interacts with caller. Writes transaction/session data to call data and/or to DB.
4. Caller needs agent, IP Office selects agent based on programming to receive the call and routes call.
5. TAPI / DEVLink Service(s) on IP Office provide info. about call and agent.
6. DTK, monitoring agents/calls via TAPI or DEVLink, detects call events and completes a Data-Dip to retrieve additional transaction/session/customer data.
7. DTK tells CRM to display caller profile and a new ticket populated with data from Avaya IP Office (and data from any configured secondary data-dips).
8. Avaya Softphone on agent computer alerts and agent answers.
9. Agent is talking to caller, has caller info in softphone and in CRM.
10. DTK updates call data/session data with the identified caller's user id and the ticket id of new ticket.

At this point if agent transfers or conferences the call to another agent DTK receives UPDATED call data and displays the caller and ticket directly from the call data received.

Physical Avaya IP Office



1. Caller places call to contact center.
2. Carrier delivers call to specific DNIS.
3. IVR interacts with caller. Writes transaction/ session data to call data and/or to DB.
4. Caller needs agent, Asterisk selects agent based on programming to receive the call and routes call.
5. AMI services on Asterisk provide info. about call and agent.
6. DTK, monitoring agents/calls via AMI, detects call events and completes a Data-Dip to retrieve additional transaction/session/customer data.
7. DTK tells Zendesk to display caller profile and a new ticket populated with data from Asterisk (and data from any configured secondary data-dips).
8. Softphone on agent computer or hard phone alerts and agent answers.
9. Agent is talking to caller, has caller info in softphone and in Zendesk.
10. DTK updates call data/session data with the identified caller's user id and the ticket id of new ticket.

At this point if agent transfers or conferences the call to another agent DTK receives UPDATED call data and displays the caller and ticket directly from the call data received.

Physical Asterisk