568 Final Project Protocol Writeup

IG 3

wc148, ny38, yw314, yx143, fz48, yt134,,hw210,cx44

04/09/2019

Preface:

This writeup describes the workflow and how an Amazon server and a UPS server interact. This writeup consists of four parts: UPS workflow description, Amazon workflow description, Workflow Diagram, XML Protocol.

The first two parts are a detailed presentation of the diagram in the third part. The last part defines the content of all the requests and ACKs.

This is a premature protocol that is subject to change.

UPS workflow description

The UPS server MUST connect with the world first.

<ACreate> When the UPS server receives an ACreated request, it MUST send ACK to the Amazon, choose an available truck (in “idle”, “arrive warehouse” or “delivering” status) and send it to the specific warehouse. The truck MAY change the destination if the UPS receives other ACreate signals.

<UArrived> The UPS MUST send an UArrived signal to the Amazon when the truck arrived at a warehouse. The truck will stop and wait for loading the package. If the UPS doesn’t receive an ACK after sending the signal to Amazon, it MUST resend the message.

<ALoaded>When receiving an Aloaded, UPS server would know that the truck has finished loading a package and would start to deliver the package. UPS MUST send an ACK back to Amazon to inform it that the package has started to be delivered.

<UDelivered> After a piece of product is delivered to the customer, UPS SHOULD send a signal to inform Amazon that specific package is delivered. After receiving an ACK, the status of the truck might be “idle” or “delivering”.

Amazon workflow description

The Amazon server MUST connect with the world first.

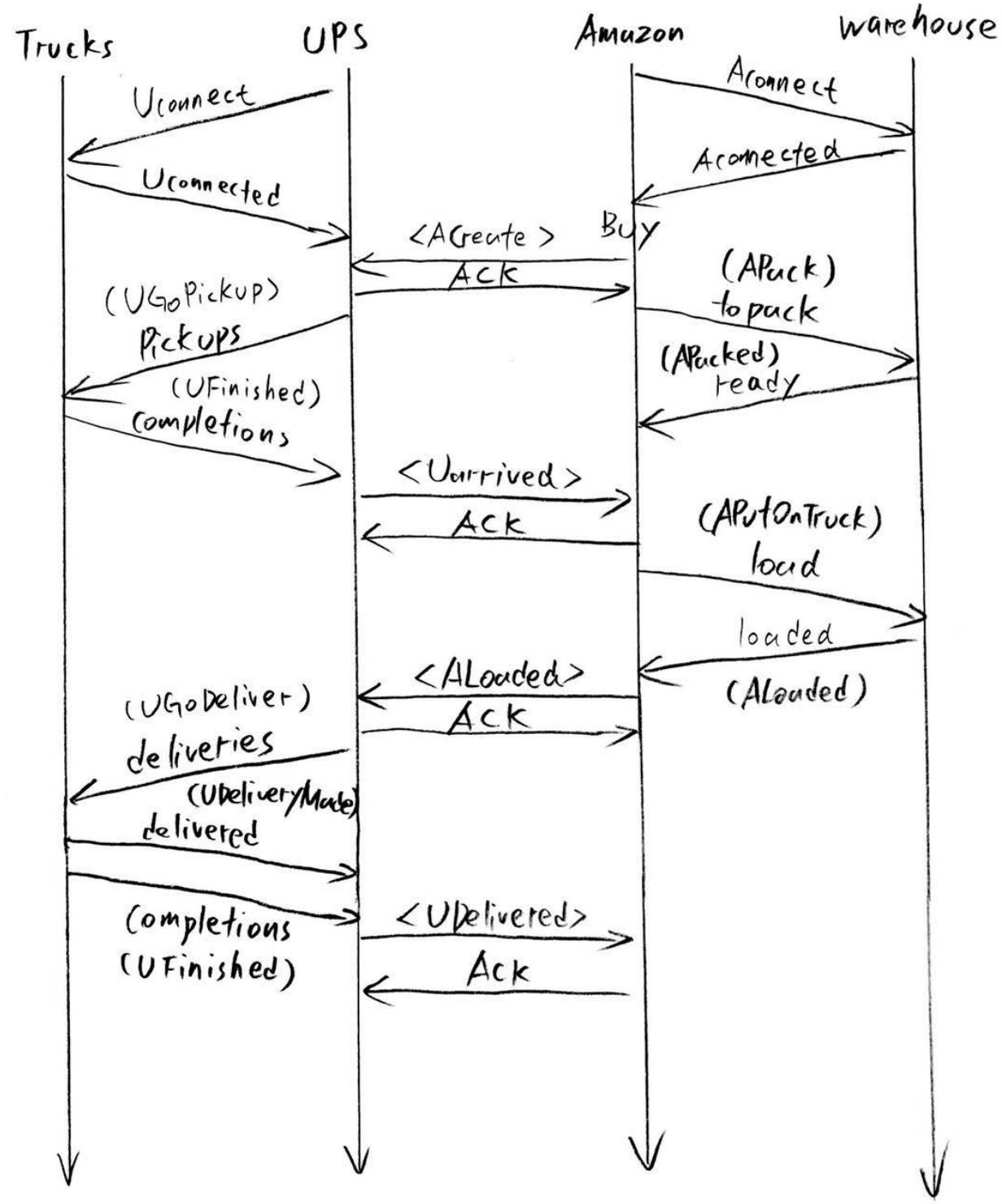
<ACreate> As soon as an order is placed, if there is no truck at the warehouse, the Amazon server MUST send an ACreated request to the UPS server to ask for a truck. If there are already some trucks waiting to be load, the Amazon server MAY not send a new request. If the Amazon server doesn't receive an ACK for this request in a certain time period, it MUST resend the same request.

<UArrived>The Amazon server MUST receive an UArrived signal from the UPS and the Packed response from the World before starting loading package. The truck id SHOULD be included in the Uarrival signal for future truck selection on the Amazon server. The Amazon server SHOULD reply with an ACK to the UPS.

<ALoaded>When a truck is loaded with a certain number of packages or it has been waiting for a certain period of time, this truck becomes ready to deliver. The Amazon server SHOULD send a signal to inform the UPS that this truck is ready to deliver packages. If the Amazon server doesn't receive an ACK for this signal in a certain time period, it MUST resend the same signal.

<UDelivered> When the Amazon server receives a UDelivered signal, it SHOULD update the status of the package on the Amazon website. The Amazon server SHOULD reply with an ACK to the UPS.

Workflow Diagram:



XML protocol

```
<ACreate seq=”SEQ” packageid=”PACKAGE_ID”>
        <location x=”X” y=”Y”/>
        <destination x=”X” y=”Y”/>
</ACreate>

<UArrived seq=”SEQ” truckid=”TRUCK_ID” packageid=”PACKAGE_ID/>

<ALoaded seq=”SEQ” truckid=”TRUCK_ID”/>

<UDelivered seq=”SEQ” truckid=”TRUCK_ID” packageid=”PACKAGE_ID”/>

<ACK ack=”ACK”/>       # ACK should be SEQ+1
```

Protobuf protocol

```
message ACreate {
        required int64 seqnum = 1;
        required int32 packageid = 2;
        required int32 whid=3;
        required int32 desX = 4;
        required int32 desY = 5;
}

message UArrived {
        required int64 seqnum = 1;
        required int32 truckid = 2;
        required int32 packageid = 3;
}

message ALoad {
        required int64 seqnum = 1;
        required int32 truckid = 2;

}

message UDelivered {
        required int64 seqnum = 1;
```

```
        required int32 truckid = 2;
        required int32 packageid = 3;
        required int32 desX = 3;
        required int32 desY = 4;
}

message ACreate{

  required int32 packageid = 1;
  required int32 whid = 2;
  required int32 desX = 3;
  required int32 desY = 4;
  required int32 wh_x =5;
  required int32 wh_y=6;
  optional int32 upsid = 7;
  optional string password = 8;
  required string itemname=9
  required int32 itemnum=10
}

message UArrived{

  required int32 truckid = 1;
  required int32 packageid = 2;
}

message ALoad{
  required int32 truckid = 1;
  required int32 packageid = 2;
}

message UDelivered{

  required int32 truckid = 1;
  required int32 packageid = 2;
  required int32 desX = 3;
  required int32 desY = 4;
}



message UAMessage {
        repeated ACreate creates = 1;
```

```
        repeated UArrived arrives = 2;
        repeated ALoad loads = 3;
        repeated UDelivered delivered = 4;
}
```

Ups’s port: 34567
Amazon’s port: 45678