

## UA.proto

### IG NO.9

```
syntax = "proto2";
```

```
package UPS;
```

```
// UPS send world id to Amazon after create a new world
```

```
message USendWorldId{  
    required int64 world_id = 1;  
}
```

```
// Amazon message to tell UPS pickups()
```

```
message APacPickup{  
    required int32 whid ; //warehouse address  
    required int64 shipment_id = 2; //shipid returned by APack  
    optional string ups_username = 3; //the optional ups username to combined the order with the  
specified ups user  
    required int32 x = 4; //dest addr  
    required int32 y = 5;  
}
```

```
// Ups Response to APacPickup
```

```
message UPacPickupRes{  
    required int64 tracking_id = 1; //track id for the shipment  
    optional bool is_bound = 2; //judge whether the attached username has error  
    required int64 shipment_id = 3; //shipid in the APacPickup  
    required int32 truck_id = 4; //truck_id to pickup this package  
}
```

```
// Ups message to tell Amazon truck arrived
```

```
message UsendArrive{  
    required int32 truck_id = 1;  
    required int32 x = 2; //WH addr  
    required int32 y = 3;  
}
```

```
message AProduct{  
    required int64 product_id = 1;  
    required string description = 2;  
    required int32 count = 3;  
}
```

```
// Amazon package meta info
```

```

message APackage{ //destination address
    required int32 x = 1; // coordinate(x,y)
    required int32 y = 2;
    required int64 shipment_id = 3; // = package id
    repeated AProduct items = 4;
}

// Amazon message to tell UPS load completed and start to deliver
message ASendAllLoaded{
    repeated Apackage packages = 1; //loaded package(s)
    required int32 truck_id = 2; //the truck that carries the loaded packages
}

// UPS message to tell Amazon package delivered
message UPacDelivered{
    required int64 shipment_id = 1;
}

message UAmessage{
    optional USendWorldId world_id;
    optional UPacPickupRes pickup_res;
    optional UsendArrive send_arrive;
    optional UPacDelivered pac_delivered;
    optional UBindRes bind_res;
    optional UResendPackage resend_package;
}

message AUmessage{
    optional APacPickup pickup;
    optional ASendAllLoaded all_loaded;
    optional ABindUpsUser bind_upsuser;
}

//-----Extra-----

// Two UPSs Extra with Amazon:
// 1. Amazon user can rebind shipment to ups account after packing up
message ABindUpsUser{
    required int64 shipment_id = 1;
    required string ups_username = 2;
}

// UPS response to ABindUpsUser
message UBindRes{

```

```
    required int64 shipment_id = 1;  
    required bool is_bound = 2;  
}
```

```
// 2. express deliver: UPS tell Amazon whether it can offer express service  
message UExpressAvailable{  
    required bool express_available = 1; //UPS calculate whether it has enough express truck  
}
```

```
// 3. UPS request: resend unreceived package  
message UResendPackage{  
    required int64 shipment_id = 1;  
}
```