01 Transportation Business
02 Extensible Entity Data Model
Transportation Business
Early Days

Rider

Vehicle

Driver
Transportation Today

- Personal Transport
- Food Delivery
- Bikes and Scooters
- Business Rides
- Non emergency Health Rides
- Freight Delivery
- Fleets
- Many more ....
Key Themes

Privacy and Security
Protect customer data

Users & Businesses as Customers
Support users as well as business customers

One Customer, Many Roles
Customers with multiple product centric profiles, beyond riders/drivers

Relationships between Customers
Employees, restaurants chains, multinational enterprise hierarchy

Agility & Extensibility
Easy to add new line of business
Privacy and Security

- Fraud
- Encryption
- Field Level Authorization
One Entity, Many Roles

Customer Types: Rider, Driver, Courier, Restaurants, Employee, ... 

Rider: 
- Eater
- Biker

A Restaurant

Common Requirements: 
- Fraud, Safety, Privacy, Encryption, Financial Compliance, ...

Specific Requirements: 

<table>
<thead>
<tr>
<th>Rider</th>
<th>Good Rating, Valid Payment method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driver</td>
<td>Car, Good Rating, Valid driver licence</td>
</tr>
<tr>
<td>Courier</td>
<td>Bike or Car</td>
</tr>
<tr>
<td>Restaurants</td>
<td>Physical location</td>
</tr>
<tr>
<td>Employee</td>
<td>Employment status</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Extensible Entity Data Model
Data Model for Entity

- **UUID**: Unique identifier
- **Base model for all customers**: Common schema shared by all customers.
- **Schema Extensions**: Specific extensions per type of customer and line of business.

```protobuf
message Customer {
  UUID id = 1;
  BaseCustomer base = 2;
  map<string, common.Extension> extensions = 3;
}

message Extension {
  google.protobuf.StringValue name = 1;
  // A serialized schema conforming message,
  // schema controlled by products
  google.protobuf.Any proto = 2;
}
```
Ride Sharing Data Model

Polymorphic User Data Modeling
Enable the same Uber account to have multiple product centric profiles

Rapid Data Model Extensibility
- Quickly introduce new roles for users, reduced time to market
- Extend existing users with schema validation as business evolves
- Enable self-serve extensions
Enable first-class entities for new business verticals

Rapid Data Model Extensibility
- Quickly introduce new types of business customers, reduced time to market
- Extend existing business with schema validation as they use multiple Uber products
- Enable self-serve extensions
Data Model for Relationships

Relationships between entities in a NoSQL distributed infra

- **User** → **Employee** → **Business**
- **Business** → **Branch** → **Business**

<table>
<thead>
<tr>
<th>SourceUUID</th>
<th>DestUUID</th>
<th>RelationType</th>
<th>Extensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>SourceType</td>
<td>DestType</td>
<td>type</td>
<td>additional data on the relationship edge</td>
</tr>
</tbody>
</table>

- The type and uuid of the source customer (either user or business)
- The type and uuid of the destination customer.
- Unique type for relationship between these two entities
- Additional data on the relationship edge
Employee Relationships

Enable many to many relationships between users and businesses

Flexible Model for Relationships in a NoSQL Backend
- Set up user to business relationships, allowing coherent product experience for all employees of a business
- Set up enterprise chains to get global view of accounts
Platform Integrations

- Seamless, common integrations with major platforms
  - Communications
  - Payments
  - Onboarding & Compliance
- Shield platforms that should not care about specific customer types via views
- New types and their semantics are correctly recognized throughout the Uber ecosystem
Thank you!