**Tasks of PATH Dynamic Transit Research**

**(Thru Feb. 2016)**

The dynamic transit research conducted by California PATH is the essential foundation of two relevant projects: Integrated Dynamic Transit Operation (IDTO) and Interactive Transit Station Information System (ITSIS).

**Task 1:** To parse the GTFS files into MySQL database tables

* To get to understand the format of GTFS files, start from the static data;
* To generate database tables according to the “GTFS Realtime Data Feed Sources.docx” and “Existing Transit DB tables.docx”, including
  + **Table Agency**
  + **Table Routes**
  + **Table Stops**
  + **Table Route\_stop\_seq**
  + **Table RunPattern**
  + **Table Schedules**
  + **Table Route\_point\_seq**
  + **Table Points**
  + **Table Fare (Not required for now)**
  + **Table Calendar\_dates**

- Before this task comes to a database, use CSV files with the same structure as the outputs instead.

**Task 2:** To generate a table of connection (transfer) stop pairs

* To generate a table containing connection stop pairs by setting constraints of their physical distance and reachability through walking. The definition of table “Transfer” is also given in “Existing Transit DB tables.docx”.
  + **Table Transfers**
* Follow the rules for generating the entities of this table described in the docx file, especially work on the approach of querying Google map to get the walking distance and time.
* Process the data of Tri-Delta agency first, other agencies such as Bart may be involved.

**Task 3:** To parse the GTFS realtime data into database tables

* To generate database tables according to the “GTFS Realtime Data Feed Sources.docx” and “Existing Transit DB tables.docx”, including
  + **Table gps\_fixes**
  + **Table TransitETA** (Under discussion)
* Before this task comes to a database, use CSV files with the same structure as the outputs instead.
* Study on the preliminary algorithm of getting Estimated Time of Arrival (ETA) based on the information types obtained from the transit real time feed. For example, through the APIs provided by 511.org, the ETA information is a direct output (see below), which has NOT included vehicle location data that are more basic and could be used for improved ETA prediction algorithm.

**Appendix:** Example of ETA information querying through 511.org API:

Usage: <http://services.my511.org/Transit2.0/GetNextDeparturesByStopCode.aspx?token=62652106-4cff-41bc-9878-d18ffca92029&stopcode=60335>

Output：

<RTT>

<AgencyList>

<Agency Name="VTA" HasDirection="True" Mode="Bus">

<RouteList>

<Route Name="22 Core (Regular)" Code="22">

<RouteDirectionList>

<RouteDirection Code="East" Name="Eastridge Transit Ctr via El Camino Real">

<StopList>

<Stop name="EL CAMINO REAL & S CALIFORNIA AVE" StopCode="60335">

<DepartureTimeList>

<DepartureTime>8</DepartureTime>

<DepartureTime>20</DepartureTime>

<DepartureTime>33</DepartureTime>

</DepartureTimeList>

</Stop>

</StopList>

</RouteDirection>

</RouteDirectionList>

</Route>

<Route Name="522 Core (Rapid)" Code="522">

<RouteDirectionList>

<RouteDirection Code="East" Name="Eastridge Transit Center">

<StopList>

<Stop name="EL CAMINO REAL & S CALIFORNIA AVE" StopCode="60335">

<DepartureTimeList>

<DepartureTime>24</DepartureTime>

<DepartureTime>30</DepartureTime>

<DepartureTime>46</DepartureTime>

</DepartureTimeList>

</Stop>

</StopList>

</RouteDirection>

</RouteDirectionList>

</Route>

</RouteList>

</Agency>

</AgencyList>

</RTT>