改动日志：

14-09-14

1. 拓扑
   1. 子路段(Link)

几何：直线段，不是多线段，连接连个Node得到的直线段，

Link表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 说明 |
| 1 | LinkID | int | LinkID |
| 2 | FNodeID | int | 起始结点 |
| 3 | TNodeID | int | 终止结点 |
| 4 | RoadType | int | 道路等级：0高速路；1快速路；2主干道；3次干道；4支路；5  辅道；6匝道；7渠化道 |
| 5 | RoadName | string | 道路名称 |
| 6 | FlowDir | string | 用来表示Link的是单向还是双向，**2：**Link是双向；  1：交通流方向与Link数字化方向相同，即Arc.Dir==1的那段Arc存在交通流，相应的也要生成**boundary、kerb、surface**；而Arc.Dir==-1的那段Arc不存在交通流，相应的不能生成**boundary、kerb、surface**  -1：交通流方向与Link数字化方向相反，即Arc.Dir == -1的那段Arc存在交通流，相应的也要生成**boundary、kerb、surface**；而Arc.Dir == 1的那段Arc不存在交通流，相应的不能生成**boundary、kerb、surface** |
| 6 | GeoType | int | 表示几何类型  0表示直线段  1 表示弧段 |
| 7 | Other | int | 备用字段 |

* 1. 子路段结点(Node)

几何：点，位于Link的起点或终点处

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 说明 |
| 1 | NodeID | int | 结点ID |
| 2 | NodeType | int | 1交叉口结点；0非交叉口结点 |
| 3 | AdjLinkIDs | char | 从正北向开始，以顺时针方向记录相邻弧段，用‘/’分隔； |
| 4 | NorthAngles | char | 结点-弧段夹角，具体数字取整，仿AdjLink顺序，用‘/’分隔； |
| 5 | ConnState | char | 转向连通状态，设有n条AdjLink，则有n\*n个符号表示，0表示禁行，1表示通行。先后顺序AdjLink相仿 |
| 6 | CompositeType | int | 结点类型：1平交2非平交 |
| 7 | Other | int | 备用字段 |

1. NodeType生成规则：

if Node.AdjLinkIDs.Length > 2

NodeType = 1;

else

NodeType = 0;

1. CompositeType

if NodeType == 1

if exist (adjlink.z !=0 )

CompositeType = 2

else

CompositeType = 1

else

CompositeType = null;

* 1. 有向子路段（Arc）

几何：以Link为几何参考，截头截尾，平移得到

注意：判断是否生成与Arc对应的**Kerb、boundary、Surface**时是需要判断，Link.FlowDir字段，若Arc.FlowDir==Link. FlowDir或Link. FlowDir ==2，需要生成**Kerb、boundary、Surface，否则，**Arc. FlowDir!= Link. FlowDir且 Link. FlowDir!=2，不能生成**Kerb、boundary、Surface,但是还要生成Lane、LaneConnector**

表1.3 Arc表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 说明 |
| 1 | ArcID | int | ArcID |
| 2 | LinkID | int | 所属LinkID |
| 3 | FlowDir | int | 1，与Link数字化方向相同；-1，与Link数字化方向相反 |
| 4 | LaneNum | int | 车道数 |
| 5 | Other | Int | 备用字段 |

* 1. 车道（Lane）

定义Lane具有几何，且几何为线段（两点间连线），有Link截头截尾偏移而来，第i个车道的偏移距离为（Lane1.Width+ Lane1.Width+… +Lanei-1.Width+ Lanei.Width/2），数字化方向同Arc

表1.4 Lane表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 说明 |
| 1 | LaneID | int | 车道ID |
| 2 | ArcID | int | 所属ArcID |
| 3 | Position | int | 在所属Arc中，从左到右的序号，从1开始 |
| 4 | Change | string | 允许的变道。  "None"：不允许变道  "Left"：仅可向左变道  "Right"：仅可向右变道  "Both"：允许向左向右变道 |
| 5 | LeftBoundaryID | int | 左边界BoundaryID， |
| 6 | RightBoundaryID | int | 右边界BoundaryID， |
| 7 | VehClasses | String | 车道通行车辆属性,不同车辆用‘\’分隔，0：All，  1：Car，2：HGV（载重汽车），3：Bus，4：Tram（有轨电车） |
| 8 | LaneClosed | int | 车道封闭属性，0：车道开放、1：车道封闭 |
| 9 | Width | double | 车道宽度，默认值为3.5 |
| 10 | Other | int | 备用字段 |

* 1. 车道连接器（LaneConnectors）

重新定义车道连接器，添加几何，**几何定义为多线段**，便于交叉口内部的交通运行轨迹的限定。生成流程见《LaneConnectors生成流程》



表1.5 LaneConnectors表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 说明 |
| 1 | ConnectorID | int |  |
| 2 | fromLaneID | int | 起始车道ID |
| 3 | toLaneID | int | 到达车道ID |
| 4 | TurningDir | string | 描述的转向，如直、左、右等  Straight, Left, Right, UTurn |
| 5 | fromArcID | int | 起始Arc |
| 6 | toArcID | int | 终止Arc |
| 7 | fromLinkID | int | 起始车道所属Link |
| 8 | fromDir | int | 起始车道对Link的相对方向 |
| 9 | toLinkID | int | 到达车道所属Link |
| 10 | toDir | int | 到达车道对Link的相对方向 |
| 11 | Other | int | 备用字段 |

1. 几何
   1. 停止线(StopLine)

几何：直线段，通过连接相邻连个车道边界线的起点/终点得到，

表2.1 停止线（StopLine）

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 说明 |
| 1 | StopLineID | int | 唯一标识停止线 |
| 2 | NodeID | int | 停止线所属的结点 |
| 3 | LaneID | int | 停止线所属的车道，定义车道与停止线为一一对应关系 |
| 4 | StyleID | int | 停止线的样式  包括线性和颜色 |
| 5 | Other | int | 备用字段 |

* 1. 车道界限(Boundary)

几何：直线段，以Link为几何参考，通过截头截尾，平移得到；

分隔对象车流的双黄线截头截尾的长度各端截头截尾的最小值；

一段Boundary不属于多个Link。

表2.2 界限（Boundary）

（包含中心线、护栏、绿化带、车道分界线、车道边缘线）

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 字段名 | 类型 | 说明 |
| 1 | BoundaryID | int | 唯一标识一个界线 |
| 2 | StyleID | int | 附件1 |
| 3 | Dir | int | 定义与link数字化方向一致,默认值为1,与Link相反定义为-1 |
| 8 | Other | int | 备用字段 |

* 1. 导向箭头(TurnArrow)

几何：点，第一个导向箭头距离停车线4-5米，点由线性参考得到

导向箭头

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 类型 | 含义 |
| 1 | ArrowID | int | 若点状符号标志为导向箭头，则ArrowID为唯一表示一个导向箭头的ID |
| 2 | StyleID | int | 见附件2 |
| 3 | ArrowType | int | 当导向箭头具体表示的是指示方向时为1，表示前方指示方向为0 |
| 4 | Serial | int | 停车线前的导向箭头的编号是0，逆着车流方向递增 |
| 5 | ANGLE | double | 图标切线方向与正北方向顺时针方向夹角 |
| 6 | ArcID | int | 标线所属的Arc的ID |
| 7 | LaneID | int | 导向箭头所属的车道ID |
| 8 | PrecedeArrowIDs | string | 当前导向箭头逆着车流方向的上一组导向箭头ID，从左向右用’\’分隔 |
| 11 | Other | int | 备用字段 |

* 1. 控制点(Kerb)

一个Arc对应4个控制点，定义一个Arc对应的控制点的编号：Arc 的右上角的控制点编号为0，右下角编号为1，左下角编号为2，左上角的编号为3。

Kerb point生成规则：

1. 以Link作为线性参考的几何，注意Link是分隔对象车流的线段，one way时为左侧道路边缘线。偏移方向和数字化方向与Arc相同，偏移距离为路面宽度，如果端点处为交叉口，则需要截取与相邻入口（出口）路段车道数对应的头或尾；
2. 外侧点，线性参考，截头截尾，偏移路面宽度，得到的线段终点为0控制点，偏移得到的线段起点为1控制点；
3. 内侧点，线性参考，截头截尾，不偏移，得到的线段终点为3控制点，偏移得到的线段起点为2控制点；

 

外侧 内侧

生成流程



控制点Kerb

点，以Link为几何参考，通过截头截尾，偏移得到，生成是注意偏移距离为该有向子路段的所有车道的宽度和

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| KerbID | int | 唯一编码，控制点编号 |
| ArcID | int | 控制点所属的Arc |
| Serial | int | 右上角0，右下角1，左下角2，左上角3 |
| PointX | Double | 控制点的X坐标 |
| PointY | Double | 控制点Y坐标 |
| Other | int | 备用字段 |

* 1. 路面(Surface)

路面用来现实道路范围，包括路面的物理范围和车行范围。路面与有向子路段相关联，其初始值有车道数及车道宽决定；为了方便，定义路网中，只能有线段，不能出现多线段。若修改控制点（kerb），则路面也需要做相应的修改。



控制点集生成规则

1. 相邻Link在node(in)处有出口，Arc(Loc)有逆时针方向第一个Link(Next)，且Link(Next)有出口Arc(NextOut)，规则为kerb(0)- kerb(1) –node(out) - kerb(2) - kerb(3) – node(in) – kerb(NextOut 0)，如下图中的规则1，
2. 相邻Link为单向，没有出口Arc，规则为kerb(0)- kerb(1) –node(out) - kerb(2) - kerb(3) – node(in)，如图规则2
3. 当前Link为断头路，入口Node的相邻路段数为1，不用截头结尾，规则kerb(0)- kerb(1) –node(out) - kerb(2) - kerb(3) – node(in)，如下图中的规则3

 

规则1 （相邻有出口） 规则 2（相邻为单向）



规则 3 （断头路）

生成流程



路面Surface

几何：多边形，生成规则，连接ControlID对应的点

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| SurfaceID | int | 唯一编码，路面编号 |
| ArcID | int | 与有向子路段对应的路面 |
| ControlIDs | string | 控制点集，控制点集由kerb和node组成，以该有向子路段方向上的0编号的控制点开始记录，第三、六个位置上记录的是NodeID，如下图中分别是101\102\26\103\104\27\106  105\106\27\107\108\28  控制点集的生成规则有3个 |
| Other | int | 备用字段 |