INIT Innovations in Transportation

Leading Provider of Intelligent
Transportation Systems for Public Transport

System Design Document

Chapter 3.2.3 Interface GIS Full Capability, Appendix B

Customer: Tri-County Metropolitan Transportation
Author: Andrea Gräser
Version: 3.2
Status: Jan-3-2011

Addresses

INIT GmbH
Kaeppelestrasse 4–6
76131 KARLSRUHE
Germany

postmaster@init-ka.de
www.init-ka.de

Tel. +49 721-6100-0
Fax +49 721-6100-399

INIT Inc.
1420 Kristina Way, Suite 101
CHESAPEAKE, VA, 23320
USA

postmaster@initusa.com
www.initusa.com

Tel. +1 757-413-9100
Toll Free +1 877-462-4648
Fax +1 757-413-5019

INIT Canada Est, Inc.
14 Place de Commerce, bur. 360
Île-des-Sœurs, MONTRÉAL
Québec, H3E 1T7
Canada

postmaster@initag.com
www.initag.com

Tél. +1 514-766-2836
Sans Frais +1 877-766-2836
Fax +1 514-766-1578

Revision record:

Version	Date	Author	Comments / Modifications
1.0	February 2010	Andrea Gräser	Initial Version (based on standard version "Interface GIS-Full_Capability_v3_0.doc")
3.0	August 2010	Patrick Neumann	Changed version number. No change of content.
3.1	December 2010	Robert Kestel	Update based on FDR comments
3.2	January 2011	Robert Kestel	Update based on FDR comments

Table of Contents

1 Introduction

This interface definition describes a GIS data exchange format which can be used to provide the INIT system with GIS background data.

The interface provides enough detail to support the INIT central map display, routing functionality on the central side, map matching functionality on the Copilot and also the navigation solution on the Copilot (not all features might be in use on the current project).

2   General

General characteristics of the file format are:

- All geo-coded files are delivered in ESRI shape format.

- One shape file can only contain features of the same geometry type (e.g. point and line features must not be contained in one and the same file).

- The defined attribute columns have to exist in the respective files. Though the order of the attributes within the file is not restricted. Also additional columns may reside in the files at any position, but won't be used in the INIT system.

- Optional fields are marked with '(o)'. Optional fields may be left empty, while the data column itself has to exist.

- Multi geometries (means separated geometries within the same shape object) are not supported.

- This interface description uses terms like 'start of a line segment', 'end of a line segment', 'right side of a line segment' and 'left side of a line segment'. These specifications refer to the mapping direction of the line segment as it is stored in the accordant *.shp file.

- The coordinate format is WGS84, Longitude/Latitude.

2.1   File Overview

This interface describes the following data files:

Streets	Geo-coded file / Contains street segments
CityDirectory	Geo-coded file / Contains location information
StreetDirectory	Contains further street and most notably address information
Areas	Geo-coded file / Contains region objects
Lines	Geo-coded file / Contains line objects (besides streets)
Points	Geo-coded file / Contains point objects
TurnRestrictions	Contains maneuver information (ban on turns)
LayerOrder	Contains the layer sequence used for drawing the object types.

3 File Specification

3.1 Streets

The filename is **Streets.shp**.

This file contains street segments as polylines.

Several options for providing a navigable street geometry are provided by the specification of this file. See more details below the record listing.

STREETS.SHP				
	FIELD	**TYPE**	**KEY**	**RANGE / REMARK**
1	Seg_ID	Integer	PK	ID of the street segment. Used for unique identification of the segment and for referencing from other files.
2	Prim_Name (o)	Char[256]		Primary street name. Field can be left empty if information is not available. If more than one primary name exists several names can be entered delimited by '/'. For the labeling of the street only the first name is used - on the street search all provided names are regarded.
3	Sec_Name (o)	Char[256]		Secondary street name. Field can be left empty if information is not available. If more than one secondary name exists several names can be entered delimited by '/'. For the labeling of the street only the first name is used - on the street search all provided names are regarded.

				Secondary street names usually contain the numeric signifier of the street, e.g. 'I5' or 'A8' if available.
4	Category	Integer		Categorizes relevance of the street segment in steps from 1 to 7. The relevance of the street decreases with a higher number. 1 - Main Roads 2 - First Class Roads 3 - 4 - Second Class Roads 5 - Third Class Roads 6 - 7 - Fourth Class Roads Usage in navigation system: The definition of the street category decides if a street segment must be taken into account for finding a specific path. E.g. if a path from San Francisco to New York is searched, only high class roads would have to be used in the routing algorithm for the main distance. Also the decision if a turn maneuver needs to be specifically announced may depend on a category switch of streets. (Otherwise information like 'follow the street' is sufficient.)
5	Type	Integer		Categorizes the average speed on this street segment in steps from 1 to 15. The speed of the street decreases with a higher number. The average speed is typically not equal to the allowed maximum speed on the street. Also the category of a street does not mandatorily indicate the average speed. A city street might be faster than an express highway on some

			occasions.
			The assignment of the type value to the concrete speed is done project and vehicle type specific on the Copilot. Therefore for the definition of the types it just needs to be ensured that the type classification is decreasing with higher values.
			The following listing gives a lead of how to provide the types. In brackets the speed values how they would be set on default in the Copilot for a slow vehicle are provided.
			1 - First class road fast (100 km/h)
			2 - First class road average (90 km/h)
			3 - First class road slow (80 km/h)
			4 - Second class road fast (55 km/h)
			5 - Second class road average (50 km/h)
			6 - Second class road slow (45 km/h)
			7 - Third class road fast (45 km/h)
			8 - Third class road average (40 km/h)
			9 - Third class road slow (35 km/h)
			10 - Fourth class road fast (30 km/h)
			11 - Fourth class road average (25 km/h)
			12 - Fourth class road slow (20 km/h)
			13 - Ferry (no speed information)
			14 - Miscellaneous (10 km/h)
			15 - Miscellaneous (10 km/h)

				Usage in navigation system: The categorizing of the average speed is used to determine the fastest connection between two points.
6	Style	Integer		Defines the graphical representation of the street on the map display. 1 - Freeway 2 - Expressway 3- Express Highway 4 - Highway 5 - City Street 6 - Miscellaneous Street 7 - Ferry Connection
7	One_Way	Integer		Defines the 'One Way' restrictions on this street segment. 0 - no 'One Way' restriction 1 - Access is only allowed in direction 'From -> To' 2 - Access is only allowed in direction 'To -> From' 3 - Access is denied in general Relevant are the restrictions which apply to the systems vehicle. E.g. if busses are allowed to access a street in both ways - even so this street has a 'One Way' restriction to the public - the set value for this segment should be '0'.
8	Length (o)	Integer		Specifies the 'true' length of a street segment in meters. The true length of a segment might vary from the polyline length since curves are represented by straight connections between points with the polylines.

			In case of ferry connections this field contains the travelling time in minutes.
9	Speed (o)	Integer	Allowed maximum speed. Currently not in use.
10	Ped_Zone (o)	Integer	Defines if a street segment is located within a pedestrian zone. 0 - not in a pedestrian zone 1- in a pedestrian zone
11	F_ZLev (o)	Integer	Defines the level of the street segment at the beginning of the segment. Access to an adjoining segment is then only possible if the relevant level settings on this node match. The value itself is hereby irrelevant. This field offers one of several possibilities to ensure geometrical unambiguousness, more information can be found below this record table.
12	T_ZLev (o)	Integer	Defines the level of the street segment at the end of the segment. Access to an adjoining segment is then only possible if the relevant level settings on this node match. The value itself is hereby irrelevant. This field offers one of several possibilities to ensure geometrical unambiguousness, more information can be found below this record table.
13	F_Node (o)	Integer	ID of the node at the start of the segment. Access to an adjoining segment is then only possible if they are sharing the same node at the conjunction. This field offers one of several possibilities to ensure geometrical unambiguousness, more information can be found below this record table. Attention: It is not mandatory to provide *F_Node*/*T_Node* information - but if provided it has to be provided for all segments.
14	T_Node (o)	Integer	ID of the node at the end of the segment.

				Access to an adjoining segment is then only possible if they are sharing the same node at the conjunction.
				This field offers one of several possibilities to ensure geometrical unambiguousness, more information can be found below this record table.
				Attention: It is not mandatory to provide *F_Node*/*T_Node* information - but if provided it has to be provided for all segments.
15	Roundabout (o)	Integer		Indicates if the the street segment is part of a roundabout.
				0 - no roundabout
				1 - roundabout

The geometry of the street network needs to be navigable. This means:

- Street segments which intersect in reality also have to be 'connected' in the GIS data. Connected means that the crossing streets share the same start-/endpoint at the intersection.

- Situations where street segments overlap in reality but do not enable access to each other (like on bridges and tunnel situations) have to be specified to ensure geometrical unambiguousness.

There are three possibilities to ensure geometrical unambiguousness:

1.) **Pure geometrical**

The geometrical unambiguousness can be assured through the geometries defined. In this case it needs to be strictly assured that street segments which cross each other but do not intersect (like on tunnels or bridges), must NOT share the same start or endpoint at the crossing.

Figure 1 - Not connected and connected line features

Figure 1 depicts this situation. On the left side it shows not connected street segments. These street segments represent a tunnel or bridge situation where the changing of the streets is not possible for a vehicle.

On the right side connected street segments are depicted. All four street features share a start or endpoint at the junction and therefore connect the streets on this position.

2.) **Using Level Information (*F_ZLev*, *T_ZLev*)**

The interface offers the possibility to provide levels of a street segment at the beginning and the end of the segment. The levels are defined with the values *F_ZLev* and *T_ZLev*.

For street segments which carry level information two segments are only considered as being accessible if their start/end points intersect AND the level at these points of the segments have the same value.

3.) **Using Node information (F_Node, T_Node)**

The interface offers the possibility to define id's for the nodes at the end and the beginning of the street segment. Street segments are only considered as accessible then if the start/end nodes of the intersecting segments have the same node id.

Attention: It is not mandatory to provide F_Node/T_Node information - but if provided it has to be provided for all segments.

1

2

Note

The using of 'Level Information' to ensure geometrical unambiguousness and 'Node Information' can NOT be mixed up. Either the 'Level Information' is used for all segments, or the 'Node Information' is used for all segments. The respective other fields remain empty.

3

4 **3.2 City Directory**

5 The file name is **CityDirectory.shp**. The file contains point geometries.

6 This file contains information about administrative locations. The provided point geometry defines the position where the
7 location name (name of the city/town) is displayed in the vehicle navigation system. Also the relevance of the location can be
8 defined which influences the size of the labeling and the zoom factor when it will be displayed.

9 Without the **CityDirectory.shp** file no location names can be displayed on the vehicle navigation map. No search
10 functionality for zip codes or locations can be provided. Also the **CityDirectory.shp** is a precondition for providing a
11 **StreetDirectory.shp**.

12

CITYDIRECTORY.SHP			
FIELD	**TYPE**	**KEY**	**RANGE / REMARK**
1	Loc_ID	Char[9]	PK.1	Unique id of the location within the country (defined by *Country_ID*).
2	Country_ID	Char[3]	PK.2	Defines the country the location is located in. Corresponds to the internationally specified country indicators. E.g.: CDN - Canada D - Germany USA - United States of America ...
3	Name1	Char[50]		Name of the main locality (e.g. New York).
4	Name2 (o)	Char[50]		Name of the sub locality (e.g. Manhattan). The provision of *Name1* remains mandatory even if a *Name2* is provided.
5	City_ID	Char[9]		Unique id of the administration unit which comprises several locations, like a commune e.g. or a city with several sub locations.
6	PostalCode (o)	Char[10]		Zip code of the location. The provision of this information does not put the provision of the *PostalCode* in the **StreetDirectory.shp** aside.
7	Level	Integer		Defines the relevance of the location. Values from 1 to 15 are possible, while the relevance decreases with increasing numbers 1 - Biggest and most relevant locations ... 15 - Smallest and less relevant location

[1]

Each zip code location is represented with one entry in the **CityDirectory.shp**. Therefore several entries for locations with the same location and sub location name (only with different zip code) will be provided. To decide which entry will be taken for the location labeling is decided by the defined *Level* of the entry. The entry with the highest level (= smallest number) is used.

It is also possible to provide the CityDirectory.shp file without the PostalCode (zip code) information. The functional impact in this case is relevant for the vehicle navigation system. An address search here starts with selecting the City OR the PostalCode where the street in question is placed in. If the PostalCode information is not provided the search always has to start with giving the City name, to start with giving the PostalCode is not possible in this case. Especially if users are not familiar with the PostalCodes of the area and typically use City names for an address search the provision of PostalCodes therefore can be disregarded.

Regarding this it is recommended that the most relevant entry of a city is provided with the correct (applicable to the city size) *Level*, while the additional entries for providing the different zip code areas of the city should be defined with a minor level (15).

3.3   Street Directory

The file name is **StreetDirectory.shp**. The file does not contain geometries.

The file defines which street segments defined in the **Street.shp** file belong to the logical same street. E.g. the street 'Forest Ave.' might be split up in several segments due to intersections with other streets. With the **StreetDirectory.shp** it is defined that these segments belong together.

Also zip code, house number and house numbering rules (like even or odd numbering) are defined per street segment.

If the **StreetDirectory.shp** file is not provided any functionality related to house numbers or zip codes can not be provided on the central side. Also the vehicle navigation system will not be able to provide any street search functionality.

	FIELD	TYPE	KEY	RANGE / REMARK
	STREETDIRECTORY.SHP			
1	Seg_ID		FK	Refers to the street segment in **Street.shp**.
2	Country_ID	Char[3]	FK	Defines the country the street segment is located in. Corresponds to the internationally specified country indicators. E.g.: CDN - Canada D - Germany USA - United States of America ...
3	Loc_ID		FK	Refers (together with the *Country_ID*) to the locality defined in **CityDirectory.shp**.
4	Street_ID	Char[11]		Street segments which logically belong together and carry the same name have to have the same *Street_ID*.
5	Range_ID (o)	Integer [1 - 999]		Subdivides logically adjoined streets according to zip code area changes. Explanation: Street segments which logically belong together carry the same *Street_ID*. If this logically created street passes more than one zip code area the *Range_ID* must be set to group all segments which reside in a particular zip code area.
6	Postal_L (o)	Char[10]		Zip code on the left side of the street.
7	Postal_R (o)	Char[10]		Zip code on the right side of the street.
8	L_F_Addr (o)	Char[11]		House number on the left side of the segment, at the beginning of the segment. If applicable with house number add-on (like 11a)
9	L_T_Addr (o)	Char[11]		House number on the left side of the segment, at the end of the segment. If

				applicable with house number add-on (like 11d)
10	R_F_Addr (o)	Char[11]		House number on the right side of the segment, at the beginning of the segment. If applicable with house number add-on (like 10a)
11	R_T_Addr (o)	Char[11]		House number on the right side of the segment, at the end of the segment. If applicable with house number add-on (like 10d)
12	Addr_Pol_L (o)	Integer		Defines the house numbering rule on the left side of the street segment. 1 - odd 2 - even 3 - consecutive empty - unknown
13	Addr_Pol_R (o)	Integer		Defines the house numbering rule on the right side of the street segment. 1 - odd 2 - even 3 - consecutive empty - unknown

3.4 Areas

The file name is **Areas.shp**.

This file contains polygons which are provided for display purposes.

	FIELD	TYPE	KEY	RANGE / REMARK
1	Name (o)	Char[126]		Representative name of the regional object.
2	Type	Integer		Defines the type of the regional object. **Administration** 22 = Municipal area 23 = Administrative area **Urban Area** 16 = Building area **Miscellaneous Area** 5 = Pedestrian refuge island 11 = Pedestrian zone 8 = Shopping centre **Airport** 9 = Airport 10 = Runway **River** 3 = River big 18 = Harbor 20 = River small

				Building		
				110	=	Cultural facility (museum, theatre, ...)
				111	=	College / University
				112	=	Fire department
				113	=	Exhibition hall
				114	=	Government and administration building
				115	=	Historical building
				116	=	Recreational building
				117	=	Medical building
				118	=	Shopping malls / centers
				119	=	Landmark
				120	=	Railway building
				122	=	Commercial center
				123	=	Sport facility (building)
				Green		
				21	=	Cemetery
				24	=	Golf course
				28	=	Park
				Industry		
				26	=	Large industrial facility
				Island		

			27 = Island area
			Ocean
			14 = Ocean
			National Park
			15 = National Park
			6 = Nature Park
			Public Facility
			17 = School
			7 = Sport facility
			25 = Hospital
			Parking
			13 = Parking garage
			12 = Parking lot
			Lake
			19 = Lake
			Forest
			4 = Forest
			Customer Specific Areas
			10001 = Public Transportation Benefit Area (PTBA)
			10002 = ADA Service Areas

1

2 3.5 Lines

3 The file name is **Lines.shp**.

4 This file contains polylines which are provided for display purposes.

5

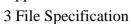| LINES.SHP			
	FIELD	**TYPE**	**KEY**	**RANGE / REMARK**
1	Name (o)	Char[126]		Representative name of the linear object.
2	Type	Integer		Defines the type of the linear object. **River** 33 = River big 36 = River small **Railroad** 34 = Railroad 3400 = Metro 3401 = Commuter Train 3402 = Electric Trolley **Tunnels / Bridges** 137 = Bridge 138 = Tunnel

1

3.6 Points

The file name is **Points.shp**.

This file contains point objects which are provided for display purposes.

5

POINTS.SHP				
	FIELD	**TYPE**	**KEY**	**RANGE / REMARK**
1	Name (o)	Char[126]		Representative name of the point object.
2	Type	Integer		Defines the type of the point object. **Rail** 58 = Railway station **Airport** 78 = Airport **Cultural Facility** 42 = Museum 48 = Theater 69 = Cinema **Store** 97 = Grocery/Drug store 52 = Shopping Center **Ferry**

			74	=	Ferry Terminal
			Public Buildings		
			71	=	City hall
			72	=	Town hall
			73	=	College / University
			77	=	Hospital
			91	=	School
			30005	=	Elementary School
			30006	=	Middle School (Junior High)
			30007	=	High school
			93	=	Police
			95	=	Library
			100	=	Court
			30001	=	City Government Site
			30002	=	County Government Site
			30003	=	Federal Government Site
			30004	=	Employment Center
			30013	=	Fire Station
			Park		
			45	=	Park
			Parking		
			44	=	Park & Ride

				46 = Parking garage
				47 = Parking lot
				Post
				135 = Post office
				Sport
				55 = Sport facility
				Tourism
				56 = Tourist information
				57 = Landmark
				127 = Historic landmark
				Public Transportation Facilities
				30008 = Operation Base
				30009 = Transit Center
				Radio Facilities
				30010 = Radio Tower
				Street Network Features
				30011 = Road Undercrossing
				30012 = Bridge

3.7 Turn Restrictions

The file name is **TurnRestrictions.csv.**

The file does not contain geo-references and is delivered in the 'comma separated values' format.

The column order is not variable.

This file contains bans on turns which apply for turning maneuvers from one street segment to another. This information is necessary on situations where streets intersect and are accessible, but e.g. the traffic rules prohibit a left turn at this intersection.

Only street segments which intersect and are accessible according to the definition in **Streets.shp** can be listed in this file to add a ban on a specific turn.

TURNRESTRICTIONS.CSV				
	FIELD	**TYPE**	**KEY**	**RANGE / REMARK**
1	From_Seg_ID		FK	Defines the street segment from which coming a ban on turn shall be added. Refers to the *Seg_ID* defined in **Streets.shp**.
2	To_Seg_ID		FK	Defines the street segment towards which a ban on turn shall be added. Refers to the *Seg_ID* defined in **Streets.shp**.
3	Node		FK	Defines the node over which the turn shall be prohibited. Attention: The node information can and has only to be given if the *F_Node* and *T_Node* fields in **Streets.shp** are used to ensure geometrical unambiguousness of the street network! Use the attribute *From_Segment_Pos* alternatively if this is not the case.
4	From_Segment_Pos	Integer		This attribute only has to be provided if the *From_Segment* and the *To_Segment* intersect each other at the start point as well as the end point. In this case it needs to be defined if the ban on turn applies to the start of the *From_Segment* or to the end of this segment.

				0 - Start of segment
				1 - End of segment
				Alternatively the *Node* attribute can be used if node id's were provided in the **Streets.shp**.

1

3.8   Layer Order

The file name is **LayerOrder.csv.**

The file does not contain geo-references and is delivered in the 'comma separated values' format.

The column order is not variable.

This file defines the layer sequence for displaying the layer. The sequence is defined top down - first mentioned layers are on top, lastly mentioned layers are at the bottom.

Currently this table is only used for vehicle navigation GIS data sets. For the central side data provision we have fixed layer order which we could adapt (if necessary/wanted) at system setup. LAYERORDER.CSV				
	FIELD	**TYPE**	**KEY**	**RANGE / REMARK**
1	Group	Char		Defines the 'group' of the object.
				Possible values are:
				- Streets
				- Areas
				- Lines
				- Points
2	Type	Integer		Defines the type of the object.
				In case of the Group 'Streets' this attribute field remains empty. Otherwise the

			entries accord to the defined types in the files **Points.shp**, **Lines.shp** and **Areas.shp**.

1