

## 11.4 Import of Data from Land Survey Sources

To reach a useful domain coverage, it is necessary to import geo data from existing resources. Commercial systems in the area of location-based services often import geo data from hundreds of geo spatial databases [145]. To investigate import issues, two geo data resources are exemplarily used:

- *TIGER/Line* file [171],
- *ATKIS* geo database [5].

The TIGER/Line files are created from the Census Bureaus *TIGER (Topologically Integrated Geographic Encoding and Referencing)* database of geographic information. TIGER was developed at the Census Bureau to support the mapping and related geographic activities required by the decennial census and sample survey programs.

The files contain geographic objects such as roads, railways, rivers, lakes, and political boundaries covering the entire United States. As the files contain polygonal data about these objects, the type and name, they are ideal sources for Nimbus domains. From this data set, a subset of about 1300 locations in the San Francisco Bay Area was used to create configuration files for the location servers [59, 60].

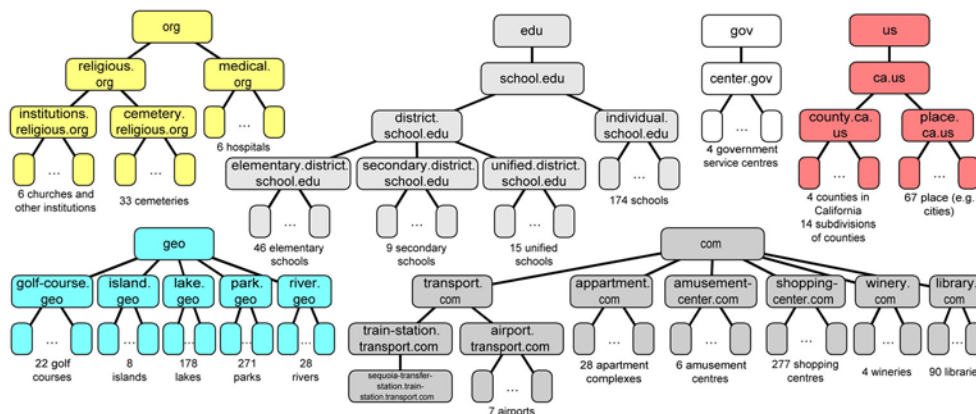


Fig. 11-8. Hierarchies imported from the TIGER/line files

The resulting hierarchies are presented in fig. 11-8. As the TIGER/line files were already close to the intended representation, the import was easy. Mainly, the import component has to parse the files, collect the polygons and map them to the respective hierarchies.

### 11.4.1 Import of Data from ATKIS Data Files

In contrast to the TIGER/Line files, the import process from the ATKIS geo spatial source was more complex. *ATKIS (Amtliches Topographisches Kartographisches Informationssystem*, in English: *official topographic cartographic information system*) provides geospatial data for governmental purposes in Germany. The ATKIS system unites the distributed data of different land survey offices of the German states. Usually, offices using these data have direct access to ATKIS databases, but private persons or companies can purchase parts of the data exported from the database and exported as *EDBS* data [4, 108, 109, 92]. For the Nimbus project, data covering an area of 84 km<sup>2</sup> in the area of Hagen was purchased. It contains approx. 80000 EDBS records which were converted to approx. 8000 Nimbus domains.

The entire import can be divided into eight steps:

#### 1. Parsing and Interpreting EDBS Records

EDBS files are text files. Each text line defines a nested multirecord which contains record-specific data and a number of subrecords. Fig. 11-9 outlines a part of the syntax of EDBS files.

```

EDBS      = EDBSHeader {EDBSLine}
EDBSHeader= "EDBS" mapID ...
EDBSLine  = Lineheader (ULOBNN | ULTANN)
ULOBNN    = "ULOBNN " ULOB0000 {ULOB100F} {ULOB200F}
ULOB0000  = GaußKrügerCoord (* defining the start value *)
ULOB100F  = ULOB1000 {ULOB110F} {ULOB1200}
ULOB1000  = GaußKrügerCoord linetype (*defining a line segment *)
ULOB110F  = layer ATKISclass objectIDright objectIDleft ...
ULOB200F  = ...

```

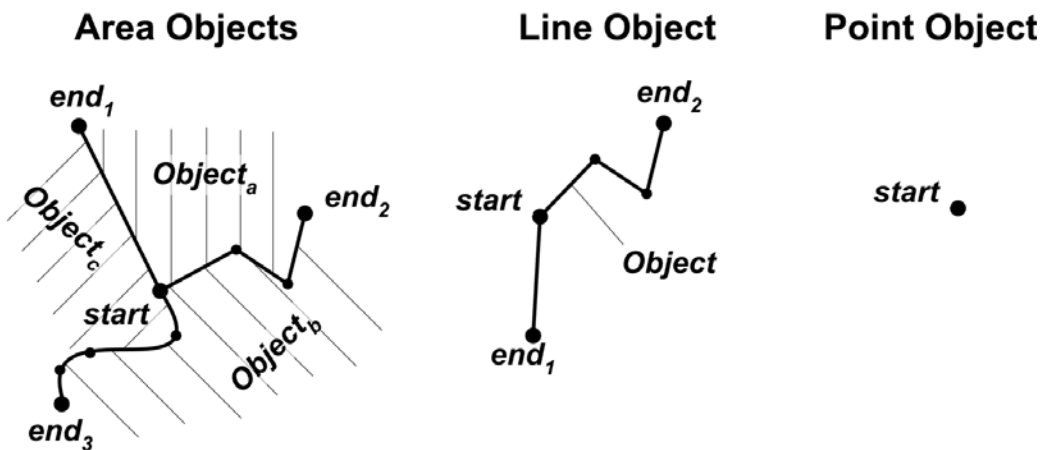
**Fig. 11-9. Part of the EDBS syntax**

The main task of the first step is to parse the record elements such as Gauß-Krüger coordinates and to build the deep structure of an EDBS record. As the internal format for further computation describes degrees of latitude and longitude, coordinates have to be converted.

#### 2. Merging Records to Raw Spatial Objects

The next step merges certain EDBS records to a single spatial object. This is because a single EDBS record often does not describe a spatial object, but only a part of it. Fig. 11-10 shows the possible contents of an EDBS records.

Each record contains a *start* coordinate and no, one or multiple *end* coordinates. For each end coordinate a sequence of segments exists which can be straight lines or curves. A record may define *area* objects, a single *line* or a *point*.



**Fig. 11-10. Objects represented by EDBS records**

Line and point objects directly represent spatial objects. Line objects represent, e.g. streets, where point objects represent river sources or archeological sites. Area objects, in contrast, have to be merged by several EDBS records. Each record defines a certain segment of a border of a number of area objects. In this step, different segments of a single object have to be collected and combined to a number of closed polygons. Unfortunately, EDBS defines which objects reside on the right and left sides of a segment, but not, how these segments are connected. Thus, the collection process has to check, if the end points of different segments are identical in order to construct closed polygons. As related EDBS records do not necessarily appear nearby in the EDBS file, this is a time and memory consuming process. In addition, this step converts curves into a number of straight lines as the Nimbus area representation does not support curves.

In our example, the 80000 EDBS records were converted to approx. 13000 spatial objects.

### 3. Merging Raw Spatial Objects to Geo-Objects

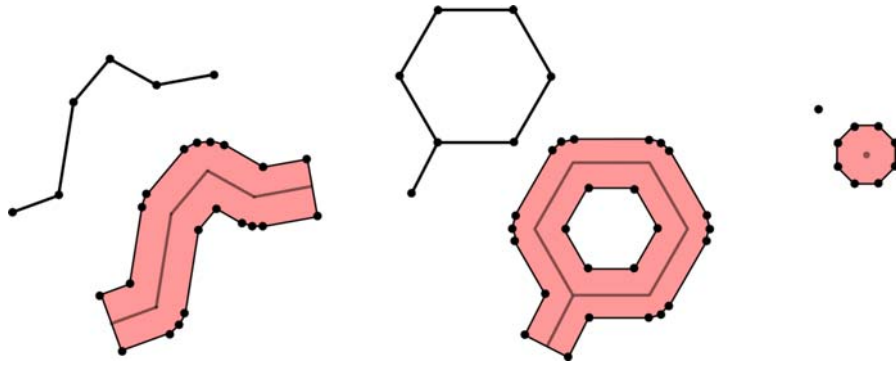
Geographic objects exported from the AKTIS database are cut according to the Gauß-Krüger grid (i.e. 1 km<sup>2</sup> cells). Larger objects, e.g. rivers, are thus represented as several closed areas, each of them fits perfectly into a Gauß-Krüger cell. This simplifies the rendering process for certain map applications, but is inadequate for location servers. So the import process has to collect related areas and connect them to the original area.

Another type of merging is necessary for street objects. ATKIS represents a street by segments, each defining the part from one crossroad to another. As domains should store all parts of an area, all street segments are merged to a single object.

In total, the import merged the 13000 objects from the last step to approx. 8000 objects.

### 4. Converting Line- and Point-like Geo-Objects to Areas

Some objects in the original representation are line- or point-like. Since Nimbus requires domains to have a certain extension, the import process has to create polygons from lines and points. For this, the *buffer* operation offered by the JTS/OpenGIS toolkit (fig. 11-11) is used.



**Fig. 11-11. Generate polygons from line and point objects**

A buffer encloses all points within a certain distance to the line or point. If the line forms a ring (a street representing a roundabout), the result is a polygon with a hole. Single points are presented by octagons.

Unfortunately, the required buffer size is not part of the original data. The import process chooses an appropriate size using the object class. E.g., streets get fixed widths for every street type such as highways, country roads, or streets in towns.

## 5. Finding Names of Geo-Objects

An important attribute of a domain apart from its geometry is its *name*. As a next step, an appropriate name has to be looked up. For named objects such as rivers, historic sites, suburbs, streets, etc. ATKIS already contains an appropriate name. For other objects (forests, farmlands, housings etc.) often no name is available. In these cases, Nimbus uses the object's class together with the unique EDBS object ID (e.g. TowerA01ZR65). To manually assign names, the Nimbus import tool allows the definition of own names for each object.

## 6. Finding the Hierarchy and the Position Inside the Hierarchy

The set of ATKIS objects inside the database is flat, i.e. the objects are not related to each other. To build the required Nimbus structures, objects need to have an appropriate position inside the hierarchies. The import process uses two criteria: the object's class and its physical location. It maps objects according to a number of rules such as:

- object class is `housing` and location is inside a suburb:  
`<name>.<suburb>.<city>.<country>`
- object class is `river`:  
`<name>.rivers.geo`
- object class is `street` and location is inside a city:  
`<name>.<city>.roads.traffic`

It is possible to override these rules and manually assign the hierarchy and position inside the hierarchy.

## 7. Integrating Meta Data

Up to now, the domains do not contain any meta data. ATKIS objects contain a classification reusable inside the Nimbus framework. Some ATKIS classes are, however, not very expressive: e.g. universities, schools and kindergartens all have the same classification "area with a special function". Nimbus extends the ATKIS class list to distin-

guish these objects and allows the manual assignment of classes by the Nimbus import tool.

## 8. Generating Location Server Files

As a last step the XML files are generated. In addition to concrete domains derived from the ATKIS geo-objects, abstract domains (such as `rivers.geo`) have to be generated. Fig. 11-12 shows a generated XML file containing the domain name, the domain type (abstract or concrete), the definition of the multipolygon and the class definitions (original ATKIS class and own class).

After the generation process is completed, domains can be reloaded for manual editing with the *DomainEdit* tool (fig. 11-13).

```
<?xml version="1.0" encoding="UTF-8"?>
<lsi>
  <config
    domain="Fernuni.fley.hagen.de"
    servertype="concrete"
    polyarea="P(5122.7004,N,00729.8548,E,000.0m*
              5122.7021,N,00729.8578,E,000.0m*
              5122.7514,N,00729.8518,E,000.0m* ...
              5122.7004,N,00729.8548,E,000.0m) "
    atkisclass="2114"
    lsiclass="22110"/>
</lsi>
```

Fig. 11-12. A generated XML file

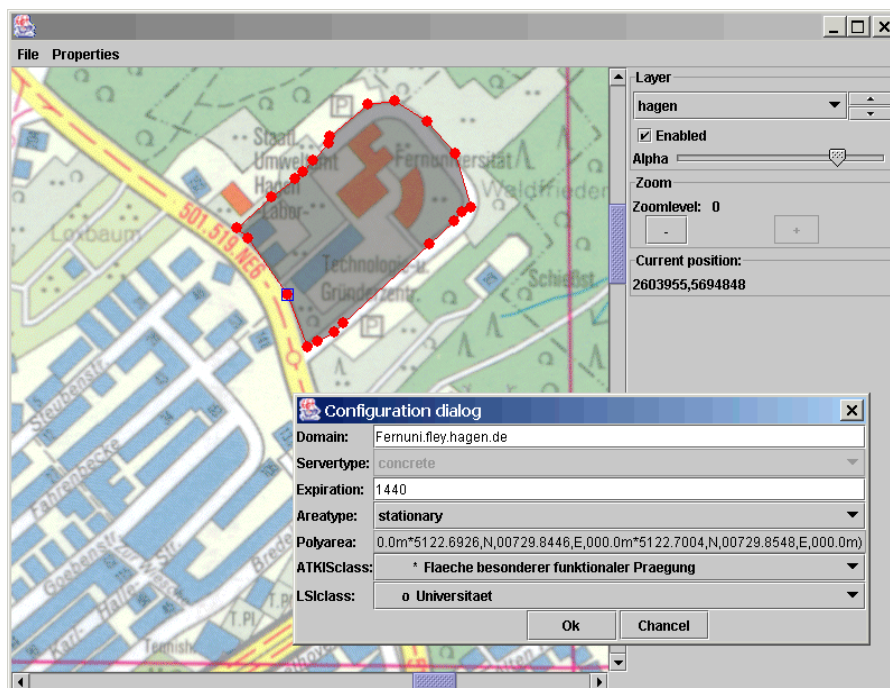
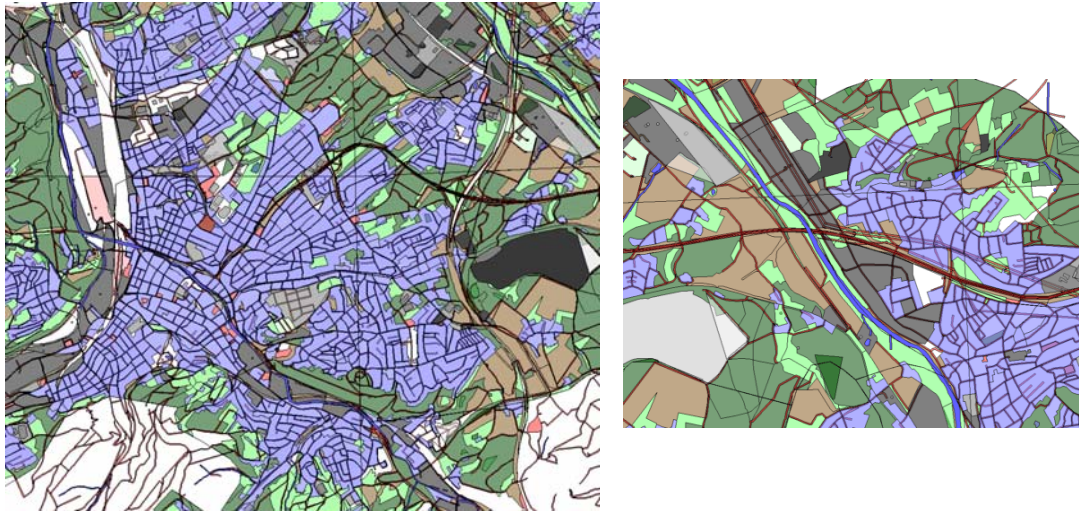


Fig. 11-13. The DomainEdit tool

### 11.4.2 ATKIS Import Result

The imported approx. 8000 domains in the area of Hagen form the basis of evaluations presented in chapter 12. Fig. 11-14 gives an impression of the imported data.



**Fig. 11-14. Maps of imported ATKIS data**

The data set contains

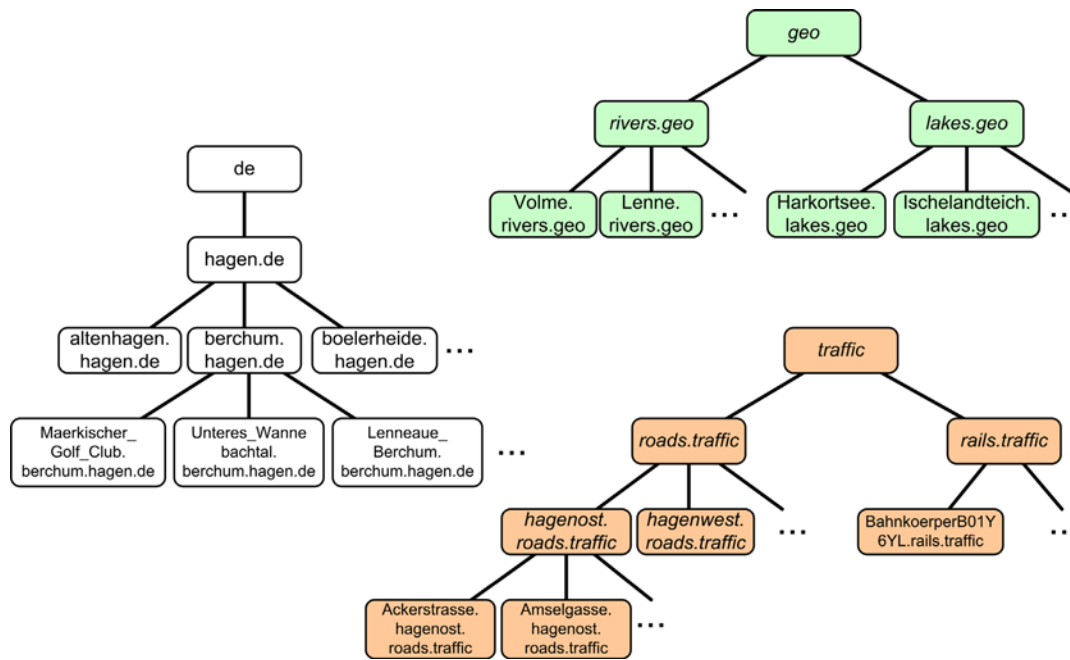
- geographic objects: rivers, lakes, sources;
- nature and agriculture: forests, farmlands, meadows;
- industrial areas and plants: quarries, industrial areas, water treatment plants, power stations, power lines;
- inhabited areas: suburbs, housing areas;
- special places: sports grounds, graveyards, market places, archeological sites;
- infrastructure and traffic: streets, parking places, rails, rail stations.

Housing areas are represented by 'blocks', i.e. the areas are enclosed by their adjoining streets. Individual buildings or houses are not distinguishable.

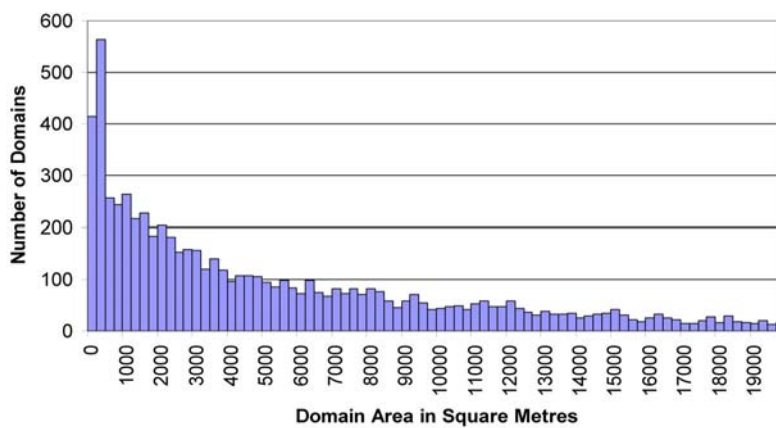
Fig. 11-15 shows the resulting hierarchy structure of imported domains with three hierarchies: A `de` hierarchy containing the city of Hagen, parts of the city and semantic areas with a relation to these parts. A `geo` hierarchy contains those domains with a geographical meaning, currently rivers and lakes. Finally, a `traffic` hierarchy contains roads and railways.

The average area covered by a domain is about 40000 m<sup>2</sup> and half of the domains are smaller than 5000 m<sup>2</sup>. The distribution of area sizes is presented in fig. 11-16.

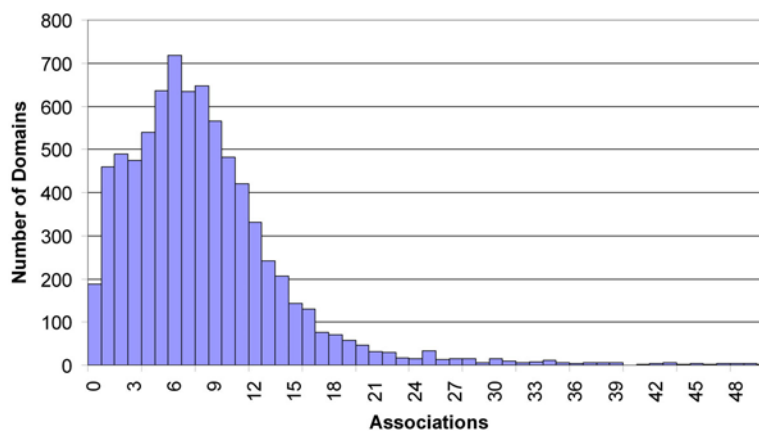
Table 11-3 gives a statistical overview of the generated domains. The table presents the number of subdomains and associations. As not all 8000 domains can be listed, names starting with '\*' indicate all subdomains of a specific domain. For these rows, average values for the number of subdomains and associations are presented. Fig. 11-17 presents the distribution of associations among the domains.



**Fig. 11-15. Hierarchy structure of imported domains**



**Fig. 11-16. Area size distribution (areas up to 20000 m<sup>2</sup>, class width 250 m<sup>2</sup>)**



**Fig. 11-17. Distribution of associations**

**Table 11-3. Imported domains in the area of Hagen**

<b>Domain</b>	<b># Subs</b>	<b>Example Subdomain</b>	<b># Assoc.</b>
de	1	hagen.de	0
hagen.de	25	altenham.hagen.de	97.4
altenham.hagen.de	69	SportplatzB01Y90D-43.altenham.hagen.de	45
*.altenham.hagen.de	0	-	11.5
berchum.hagen.de	192	Unteres_Wannebachtal.berchum.hagen.de	24
*.berchum.hagen.de	0	-	10.8
boelerheide.hagen.de	122	Kampfbahn_Boelerheide.boelerheide.hagen.de	41
*.boelerheide.hagen.de	0	-	11.0
delstern.hagen.de	124	AbfallbehandlungsanlageA01ZQS2.delstern.hagen.de	63
*.delstern.hagen.de	0	-	12.0
eckesey.hagen.de	102	Wielandplatz.eckesey.hagen.de	37
*.eckesey.hagen.de	0	-	9.4
eilpe.hagen.de	350	Kampfbahn_Struckenberg.eilpe.hagen.de	126
*.eilpe.hagen.de	0	-	12.8
elsey.hagen.de	208	Elseyer_Dorfplatz.elsey.hagen.de	69
*.elsey.hagen.de	0	-	9.9
emst.hagen.de	291	Marktplatz.emst.hagen.de	173
*.emst.hagen.de	0	-	14.0
eppenhausen.hagen.de	189	Goldene_Pforte.eppenhausen.hagen.de	124
*.eppenhausen.hagen.de	0	-	10.5
fley.hagen.de	278	Fernuni.fley.hagen.de	83
*.fley.hagen.de	0	-	9.7
halden.hagen.de	189	TC_Halden_2000.halden.hagen.de	59
*.halden.hagen.de	0	-	11.9
haspe.hagen.de	232	Freibad_Hestert.haspe.hagen.de	148
*.haspe.hagen.de	0	-	11.3
hassley.hagen.de	73	TurmA01ZR65.hassley.hagen.de	37
*.hassley.hagen.de	0	-	11.3
helfe.hagen.de	75	Waldfriedhof.helfe.hagen.de	35
*.helfe.hagen.de	0	-	10.0
herbeck.hagen.de	217	Archaeol_FundstaetteA01ZR4B.herbeck.hagen.de	59
*.herbeck.hagen.de	0	-	9.4
hohenlimburg.hagen.de	296	Platz_der_7_Kurfuersten.hohenlimburg.hagen.de	212
*.hohenlimburg.hagen.de	0	-	8.7
holthausen.hagen.de	143	Holthausen_Bachtal.holthausen.hagen.de	52
*.holthausen.hagen.de	0	-	11.8
innenstadtnord.hagen.de	386	Funckepark.innenstadtnord.hagen.de	146
*.innenstadtnord.hagen.de	0	-	12.3
innenstadtsued.hagen.de	160	Bodelschwinghplatz.innenstadtsued.hagen.de	103



**Table 11-3. Imported domains in the area of Hagen (continued)**

<b>Domain</b>	<b># Subs</b>	<b>Example Subdomain</b>	<b># Assoc.</b>
*.innenstadtsued.hagen.de	0	-	14.9
innenstadtzentrum.hagen.de	120	Hagen_Hbf.innenstadtzentrum.hagen.de	54
*.innenstadtzentrum.hagen.de	0	-	11.6
quambusch.hagen.de	131	Wald__ForstB01GG7S.quambusch.hagen.de	37
*.quambusch.hagen.de	0	-	11.1
reh.hagen.de	330	Stadion_Kirchenberg.reh.hagen.de	158
*.reh.hagen.de	0	-	10.1
vorhalle.hagen.de	279	Vossacker.vorhalle.hagen.de	48
*.vorhalle.hagen.de	0	-	10.2
wehringhausen.hagen.de	193	WohnbauflaecheB01Y6U2.wehringhausen.hagen.de	346
*.wehringhausen.hagen.de	0	-	15.9
west.hagen.de	690	BrunnenB01GGHP.west.hagen.de	155
*.west.hagen.de	0	-	8.8
geo	2	lakes.geo	0
lakes.geo	41	Harkortsee.lakes.geo	0
*.lakes.geo	0	-	2.4
rivers.geo	18	Volme.rivers.geo	0
*.rivers.geo	0	-	50.8
traffic	2	roads.traffic	0
rails.traffic	5	BahnkoerperB01Y6YL.rails.traffic	0
*.rails.traffic	0	-	22.2
roads.traffic	3	hagenost.roads.traffic	0
hagenost.roads.traffic	698	Amselgasse.hagenost.roads.traffic	0
*.hagenost.roads.traffic	0	-	13.2
hagenwest.roads.traffic	809	Alexanderstrasse.hagenwest.roads.traffic	0
*.hagenwest.roads.traffic	0	-	11.5
intercity.roads.traffic	885	Herdecker_Strasse.intercity.roads.traffic	0
*.intercity.roads.traffic	0	-	6.6