



Université Bordeaux 1
351 cours de la Libération
33405 TALENCE
France



Noldus Information Technology
Nieuwe Kanaal 5
6709 PA Wageningen
Nederlands

Development of tools for a project of maritime safety

Student	: François Tessier
Placement supervisor	: Wil van Dommelen
Teacher in charge	: Emilie Lalague
Period	: from Monday 4 May 2008 to Friday 24 July 2008

Contents page

INTRODUCTION

I. CONTEXT	4
A. OVERVIEW	4
B. PRODUCTS	4
1. <i>The Observer XT</i>	4
2. <i>EthoVision XT</i>	5
3. <i>Other products</i>	6
C. THE RESEARCH AND DEVELOPMENT DEPARTMENT	7
D. POSEIDON PROJECT	7
E. DEVELOPMENT ENVIRONMENT	8
II. PROBLEM ANALYSIS	8
A. VISUALIZE AIS DATA IN ETHOVISION	8
III. REALIZATION	10
A. DOCUMENTATION	10
B. AIS TO GPX	11
C. MAP DOWNLOADER	11
1. <i>Extract from GPX files</i>	13
2. <i>Output files</i>	13
3. <i>Map from GPX</i>	14
D. TXTTOTRK CONVERTER.....	14
E. ETHOVISION XT	15
F. LIMITS.....	16

CONCLUSION

Introduction

Noldus Information Technology, Thales, ESI and some universities grouped themselves to launch Poseidon Project. This project try to find solutions to complex notions of “systems of systems” and particularly to maritime safety domain. The main goal is to improve security of ships and to anticipate hypothetical collision or maritime problems. This subject is topical because of fresh outbreak of act of piracy (around Somalian coast for example).

My internship at Noldus IT, Nederland, concerned Poseidon Project. I had to find, study and implement solution to track ships in order to analyze their moving and extract conclusion of it. To do that, I had to use a Noldus software: EthoVision XT.

First, I'll explain in details the context of this assignment. Then, I'll present and analyze problems and how to resolve them. Finally, I'll show solutions I implemented.

I. Context

A. Overview

Noldus Information Technology is a software publisher specialized in behavioral research. This private company was founded in November 1989 by Dr. Lucas Noldus after completion of his PhD at the Department of Entomology (study of insects), Wageningen University. He still serves the company as the Chief Executive Officer.

Noldus is located at Wageningen (Nederland) and has some offices in United States, France, Hungary and recently in China. The company counts around 100 employees most of them working at the head office. The goals of this company are summarized in one commercial sentence: *"To become the world's leading developer of tools and services for behavioral research"* [Lucas Noldus]. So Noldus Information Technology develops, markets and supports software, instruments, integrated systems and services for behavioral research. These allow scientists and practitioners to enhance the quality of their data, to increase productivity, and to make optimal use of human or animal resources.

B. Products

The company produces lot of software which can detect face expressions or record and analyze interaction between a user and a system for example. The two oldest and also main ones are The Observer XT and EthoVision XT. My internship in Noldus concerned the last one.

1. *The Observer XT*



The Observer XT allows analyzing, calculating and representing lots of human or animal behavior data. Historically, The Observer XT was the first program written by Lucas Noldus himself and running on TRS-80 Model 100 (processor: 2.4 MHz, RAM: 32 K, ROM: 32 K). Today, it's a powerful multithreaded program which needs a recent machine to work efficiently.

It can be applied to study observable behavior such as activities, postures, gestures, facial expressions, movements, and social or human-system interactions. The collection of observational data can be provided by a lot of Data Acquisition Systems (EEG, EMG, other Noldus Software: uLog, FaceReader...) connected to the Observer. This software allows quantifying behaviors thanks to a code scheme system. Then it is possible to compute multiple analyses: behavior analysis, numerical analysis, reliability analysis, etc.

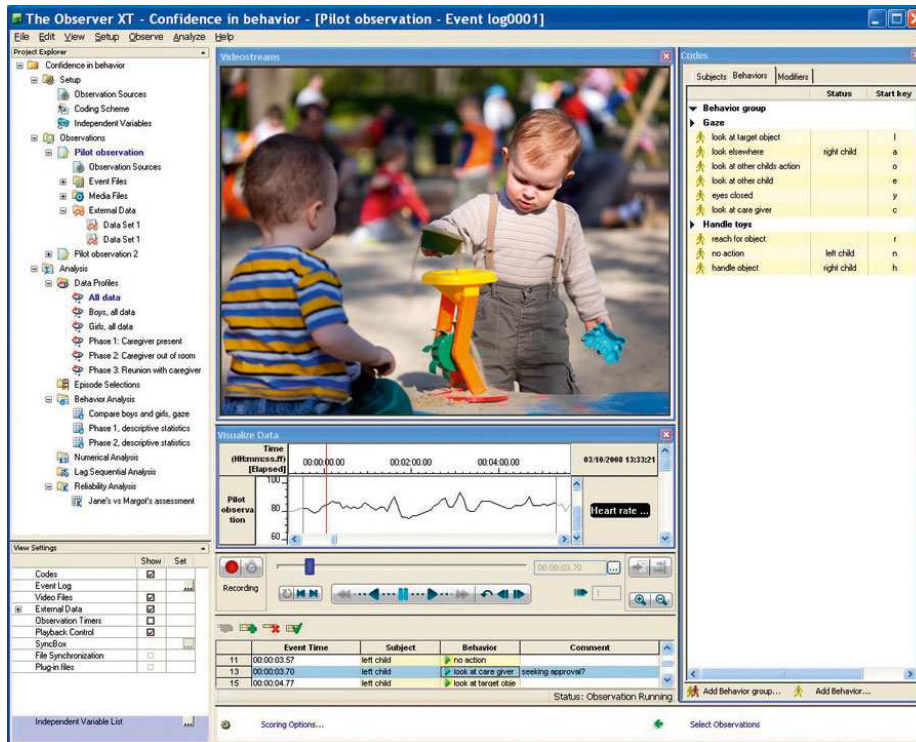


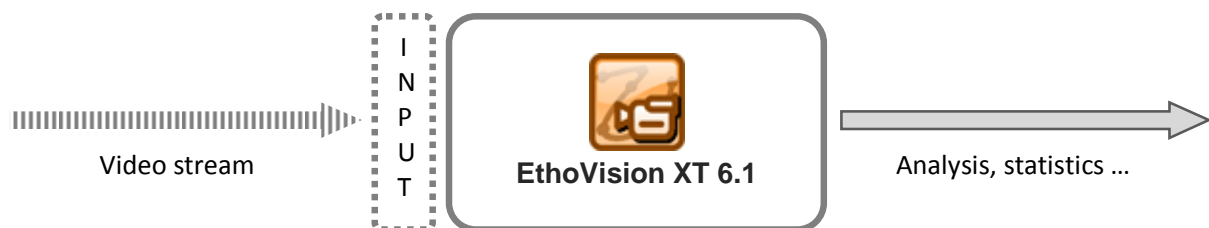
Figure 1 - Screenshot of the Observer: video + data + coding scheme

2. EthoVision XT



First named Video Tracking & Motion Analysis System, EthoVision XT is a VTS program (Video Tracking System) specialized in animal behavior field. It's the second tool developed by Noldus IT since 1993.

This tool takes as input a video stream from file or from live camera and tracks subjects on it. Next, it analyses all these tracks and give to the user lots of meaningful statistics about moving, behavior, social interactions, etc...



A new version of this software is published approximately every two years. It's the time needed for Noldus to create functionalities and answer to needs of research domain.

➤ **Working**

To use EthoVision, the user has to create an “experiment”. An experiment (as the meaning of EthoVision), is composed by settings, trials, arenas and lots of optional parameters.

In settings, the user has to specify the source of video stream (MPEG file, plugged camera), the number of subjects present in the video and the number of arena he wants to create.

A trial defines subjects and arena to analyze.

An arena is a zone which the user wants to analyze.

Once the acquisition is done, we can see the tracks on a background image grabbed from video as input.

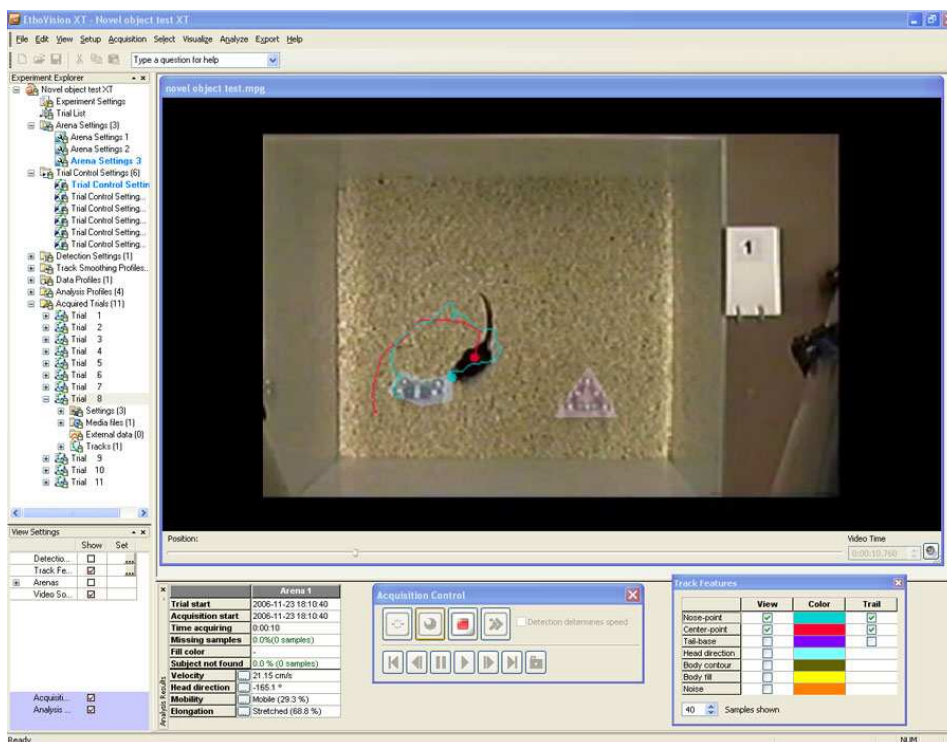


Figure 2 - Screenshot of EthoVision XT

3. Other products

➤ **Catwalk XT**

CatWalk XT is an instrument for quantitative assessment of footfalls and gait in rodent models.

➤ **Pocket Observer**

This is the version of the Observer designed to run on portable device such PDA or Smartphone.

➤ **UltraVox**

This is a system for the automatic detection of ultrasonic vocalizations, such as mouse pup calls in neurotoxicological studies or rat vocalizations in drug testing.

➤ **MatMan**

This little software performs matrices calculations and statistics.

➤ **Theme**

Theme detects statistically significant time patterns in sequences of behaviors. These so called T-patterns are based on the timing of events, relative to each other. This can give striking new insights into the structure of behavior that cannot be found with the naked eye or with any other method.

➤ **FaceReader**

This is a tool capable of automatically analyzing facial expressions. It detects faces in a video stream and identifies the current expressions.

C. The Research and Development department

Software from Noldus are mainly used in research domain. That's why the company needs to progress constantly. To do that, Noldus is present at the most part of research conferences and the majority of its employees work in the "Research and Development" cell in Wageningen and Budapest (Hungary). I integrated this cell during my assignment.

The R&D department counts around 30 people whose few doing a PhD divided into two teams: VOS and VTS. The first team takes care of the Observer and the second of EthoVision. I belonged to the VTS team during this three-month internship. In parallel with the development and the improvement of the existing products, this team is involved in projects with exterior partners. One of these projects is the Poseidon project.

D. Poseidon Project



During this assignment, I worked on a project named Poseidon.

Poseidon is a big research project grouping Noldus, Thales, ESI and some University together. Its aim, beyond complex notions of "systems of systems", is to allow flexibility, adaptability and evolutivity in the domain of maritime safety systems.

Noldus intervenes by suggesting a tool allowing tracking ships and analyzing their moving. EthoVision XT can track and analyze animals' moving and behaviors. The idea is to modify EthoVision XT to use its skills to also track ships by using their AIS module (Automatic Identification System).

E. Development environment

EthoVision XT is software written in C++ for Windows environment. This project is composed by more than 100 subprojects, each implementing a category of functionalities (video treatment, motion analysis, etc...).

Microsoft Visual Studio 2008 with lots of useful add-ons is used to develop. Microsoft Visual Source Safe is used as version control system.

The development is guided by lots of rules partly extracted from reference work in C++. That's why the code is very clear, whole documented and the compilation doesn't accept any warnings.

II. Problem analysis

A. Visualize AIS data in EthoVision

Great majority of ships are equipped by an AIS module. This AIS module sends data about identification, position, speed, etc ... to Vessel Traffic Services. These data are used to steer ships and to ensure their security.

<code>^M^M</code>	<code>rawtostampedMsg^L</code>	<code>48^UT</code>
<code>1180656780.231^EOH^!AIVDM,1,1,,1,19NS6Q002H0?sdDNR7o7g6=10000,0*40</code>	<code>rawtostampedMsg^L</code>	<code>48^UT</code>
<code>^M^M</code>	<code>rawtostampedMsg^L</code>	<code>48^UT</code>
<code>1180656780.234^EOH^!AIVDM,1,1,,1,13:4o8003g0@4S`N7rMHE6ch0000,0*69</code>	<code>rawtostampedMsg^L</code>	<code>48^UT</code>
<code>^M^M</code>	<code>rawtostampedMsg^L</code>	<code>48^UT</code>
<code>1180656780.234^EOH^!AIVDM,1,1,,1,13QAC2001p0@2<<MsWKR71e60000,0*22</code>	<code>rawtostampedMsg^L</code>	<code>48^UT</code>
<code>^M^M</code>	<code>rawtostampedMsg^L</code>	<code>48^UT</code>
<code>1180656780.235^EOH^!AIVDM,1,1,,1,13P71n@P1U0@3GVMenwS6BOT0000,0*2B</code>		

[AIS file](#)

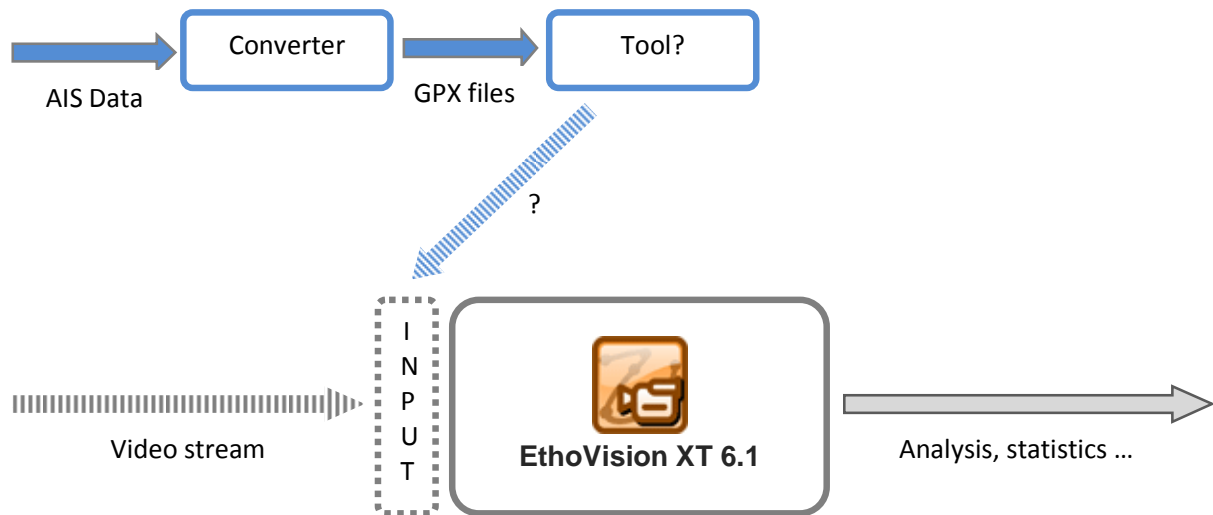
Noldus got from the Maritime Research Institute of Netherlands a DVD containing AIS data from ships next to Dutch coast during a week. The main goal is to use EthoVision XT to visualize these data. EthoVision XT now accepts video stream (from a MPEG file or from a live camera) and extracts of this a background image. Then, it tracks each subjects and create a formatted file called "Trackfile". A Trackfile is a binary file created by Noldus which contains information about a track for an experiment: (x, y) coordinates, area, elongation rate, etc...

Object Index: 0 Number Of Samples: 359 Sample Rate: 25 Duration: 14.36					
Hr.	Object index: 0				
	X	Y	Area	ChangedAr	Elongation
1	-1	-1	-1	-1	-1
2	279	171	-1	-1	-1
3	278	170	-1	-1	-1
4	276	169	-1	-1	-1
5	275	168	-1	-1	-1
6	273	166	-1	-1	-1
7	280	171	-1	-1	-1
8	289	177	-1	-1	-1
9	287	176	-1	-1	-1
10	286	175	-1	-1	-1
11	284	174	-1	-1	-1
12	283	173	-1	-1	-1
13	289	178	-1	-1	-1
14	297	183	-1	-1	-1
15	296	182	-1	-1	-1
16	294	181	-1	-1	-1
17	293	180	-1	-1	-1
18	292	179	-1	-1	-1
19	298	182	-1	-1	-1
20	305	186	-1	-1	-1
21	303	186	-1	-1	-1
22	302	185	-1	-1	-1
23	301	184	-1	-1	-1
24	299	184	-1	-1	-1

Figure 3: Example of Trackfile. A tool used at Noldus allow to display and modify these Trackfiles.

So, the first idea is to create a fake experiment without video stream but with a background image and a Trackfile fitting with the map for each ship. For a second step, it could be better to allow natively EthoVision XT to take AIS files as input.

A student who did an internship last year in Noldus began to work on this project. She mainly did some documentation and thought about some part of solution. A first step was extracted: convert AIS data to a “universal” format. GPX format was chosen because it’s used by almost all GPS hardware. A GPX file is an XML file containing GPS data (latitude, longitude, elevation ...). By this way, we can stretch tools created for Poseidon project to all domains using GPS data.



My assignment consisted in developing a converter from AIS data to GPX files and finding, studying and developing tools to solve this problem: How to give GPS data as input in EthoVision XT to create analysis?

III. Realization

A. Documentation

The first days in Noldus were reserved to documentation about the company and especially about VTS project. Training was organized for me to learn how EthoVision works. Thanks to the strong programming rules applied by developers, I read a lot of documentation. Internally forum, wiki and Doxygen¹ were especially powerful for developers. I began to familiarize me with the code. Lots of reference book was available in the library.

¹ Doxygen is a documentation generator for lots of programming languages.

B. AIS to GPX

The first tool I created was to convert AIS data to GPX files. Because it's often not necessary to reinvent the wheel, I used a project by Brian C. Lane called "AIS Parser SDK". This project is more a library than a SDK which allow to extract all data in AIS file. Next I had to integrate this library in a C++ console application in order to create valid GPX files.

Some documentation about the specification of GPX format was necessary. Learn the C++ language and the good practice associated with also took some time.

To do this "conversion", the algorithm is simple : for each line read in AIS file, this program convert this line according to GPX format and write it in a GPX file which name is the id of treated ship.

C. Map Downloader

As explained earlier, EthoVision need a background image on which it can draw tracks of subjects. In the actual version of EthoVision, the map is extracted from the video stream. As we have no videos to extract background image, we need to get this image (a map in this case) and save it as the background image in EthoVision.

To do that, I created a tool named Map Downloader. This tool was written in C++.Net for the graphic user interface (GUI) and in "standard" C++ for the treatment. Map Downloader displays an instance of Google Map. The user can select a zone of earth (by moving on the map and zooming) and download this map as a PNG and Bitmap file.

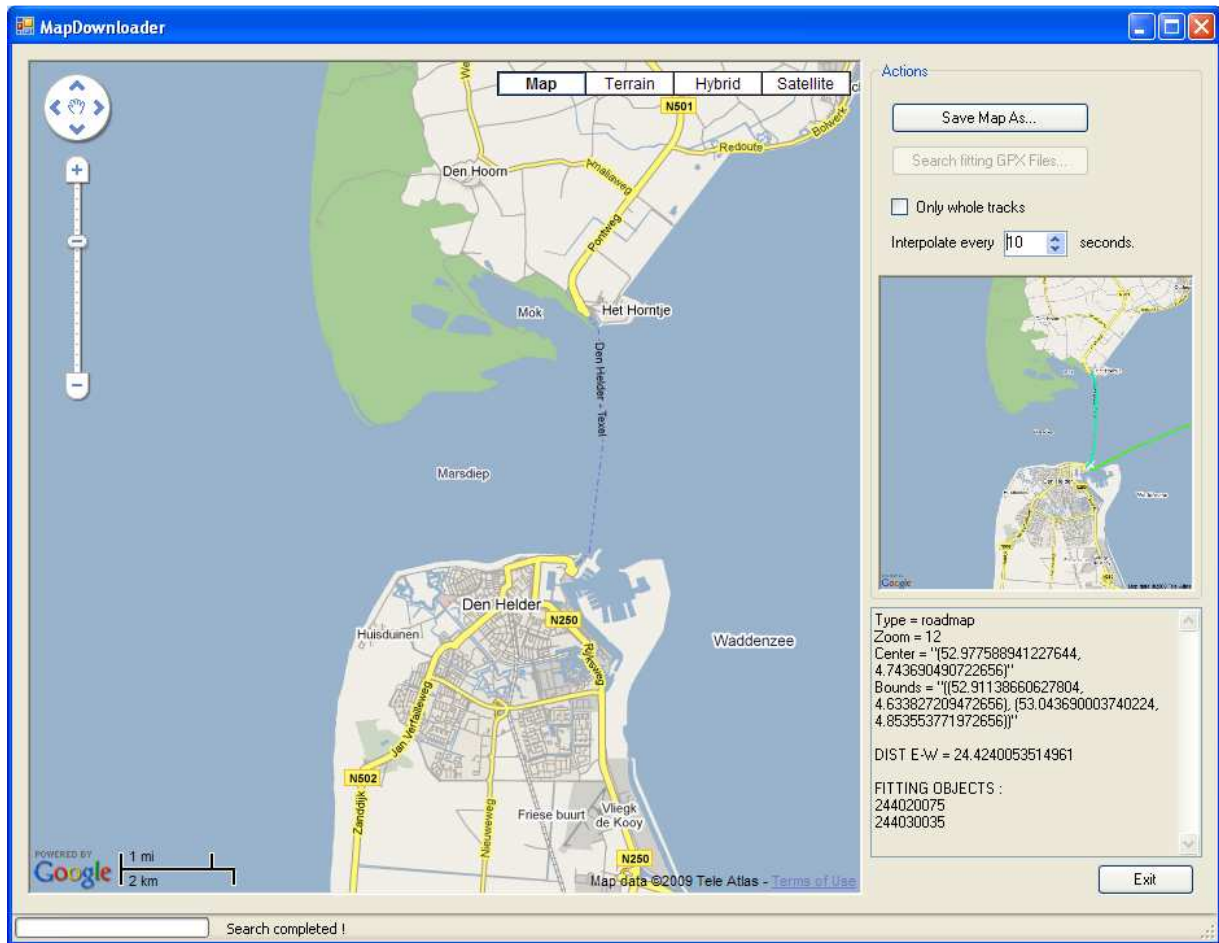


Figure 4 : Screenshot of MapDownloader with tracks preview

To allow this functionality, I created an HTML file with some javascript using the Google Map API. Map Downloader display this HTML file by means of the Dotnet control WebBrowser. When the user move or zoom on the map, an event is triggered by javascript and some variables like coordinates of the center point of the map, coordinates of the corner of the map or zoom level used are updated. A function called on a button click in Map Downloader can get back these data in order to get the same map as an image file by using Google Static Map API. This functionality of Google Map allows getting a map image from a URL. This URL have to take some parameters like coordinates of the center points, size of output image and span or zoom level. To get this file, I chose to use a famous open source program, wget, by a system call.

When you got a map, you can select a folder containing GPX files. MapDownloader will extract all data from these files and check, for each file, if it contains at least one coordinate fitting with the map (or the whole track if specified). If it's, the file is rewritten time-equidistant by creating needed points by linear interpolation. A text file containing the same information but with coordinates converted in (x, y) system according to the map is also created.

1. Extract from GPX files

The first thing to extract from GPX file is evident: Be sure that file as input is a GPX file. To do that, and after a test on extension, I wrote a function using MSXML and a XML schema (also called XSD) to validate each GPX file according to the GPX Schema 1.1.

A GPX file is a XML file. It implies a tree structure and so, a better algorithm than line by line reading. To extract data from GPX files, I chose to use the recursive Depth First Search algorithm. This famous algorithm matches exactly with the XML structure.

For each node covered, a “switch” allow to extract data or not according to the node. For example, the “rtept” tag contains coordinates of a track point. So if the node covered is one of these, we extract these coordinates and save them in a vector. But if we are on a “ele” tag corresponding to elevation, we’ll pass over it because this data doesn’t present any interest for us.

2. Output files

When data are extracted from GPX files and reduce to fit with the map, data are made equidistant by time. It means that if the user specified a value of 10 seconds, we have to produce a GPX file as output containing coordinates every 10 seconds. But a GPX files can contain sometimes more than one point for a given time. For example, you can have this:

```
<?xml version="1.0" encoding="UTF-8"?>
<gpx version="1.1"
  creator="AIStoGPX by Noldus"
  xmlns="http://www.topografix.com/GPX/1/1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.topografix.com/GPX/1/1
http://www.topografix.com/GPX/1/1/gpx.xsd">
  <rte>
    <name>445</name>
    <rtept lat="51.973" lon="5.63456">
      <time>2008-07-22T09:30:24Z</time>
      <src>445</src>
    </rtept>
    <rtept lat="51.973" lon="5.63453">
      <time>2008-07-22T09:30:34Z</time>
      <src>445</src>
    </rtept>
    <rtept lat="51.973" lon="5.63453">
      <time>2008-07-22T09:30:34Z</time>
      <src>445</src>
    </rtept>
```

Example of GPX file

I chose to dismiss these points.

A function allows to convert (lat,lon) coordinates to (x,y) coordinates.

3. Map from GPX

Another useful function allows to give as input GPX files. It extracts data from them and gets a map containing all tracks. There is no GUI for this function now.

This function needed a lot of documentation and particularly about Mercator projection. This projection system is used to represent the earth which is a sphere on a plan. It one of the more used projection and it the one used by GoogleMap. I encountered the main difficulties with MapDownloader on this. Lots of formula were necessary to calculate bounds of needed map, pixel per degree in longitude or pixel per degree in latitude...

D. TxtToTrk Converter

EthoVision create from video stream a background image and Trackfiles. One for each subject. With MapDownloader, we got back a map to be the background image and GPX and Text files time-equidistant. We need now to convert these text files to Trackfiles readable by EthoVision.

Text files from MapDownloader were made time-equidistant and converted to (x, y) coordinates. TxtToTrk Converter can take these files as input and create Trackfiles from them.

Another particularity of EthoVision is that this software create (and need) Trackfiles which have the same number of track points. This notion is called "trackset". If there is only one subject on the video stream, one Trackfile will be created containing only track points of the analyzed subject. But if there is more than one subject, a Trackfile is created for each one. As you have to specify before acquisition how much subjects you want to track, EthoVision XT creates Trackfiles and add track points on each at the first second of acquisition. If a subject is not present on the video, all these track points during this absence period will be created with negative (x, y) coordinates.

If Trackfiles of a trial have not the same number of track points, EthoVision will crash. That's why, an option in TxtToTrk Converter allow making a trackset.

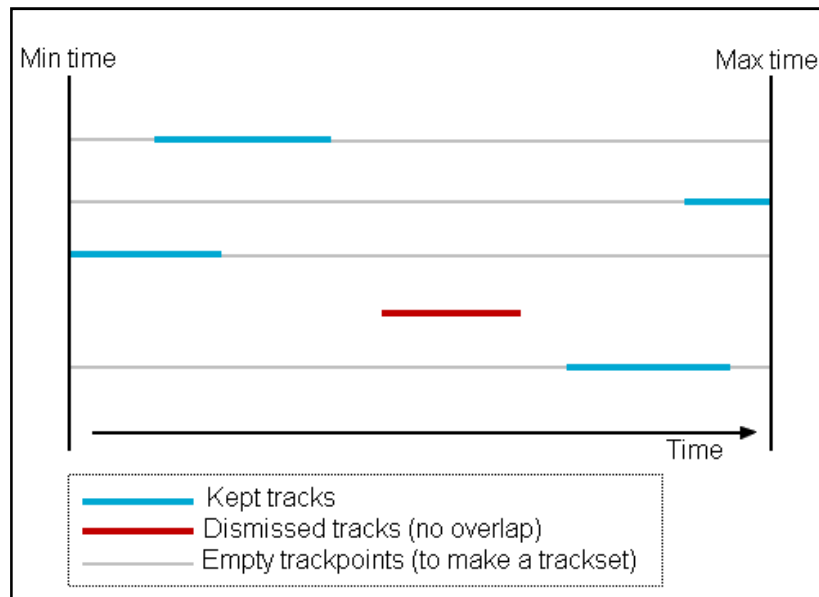


Figure 5: Explanation of system of trackset

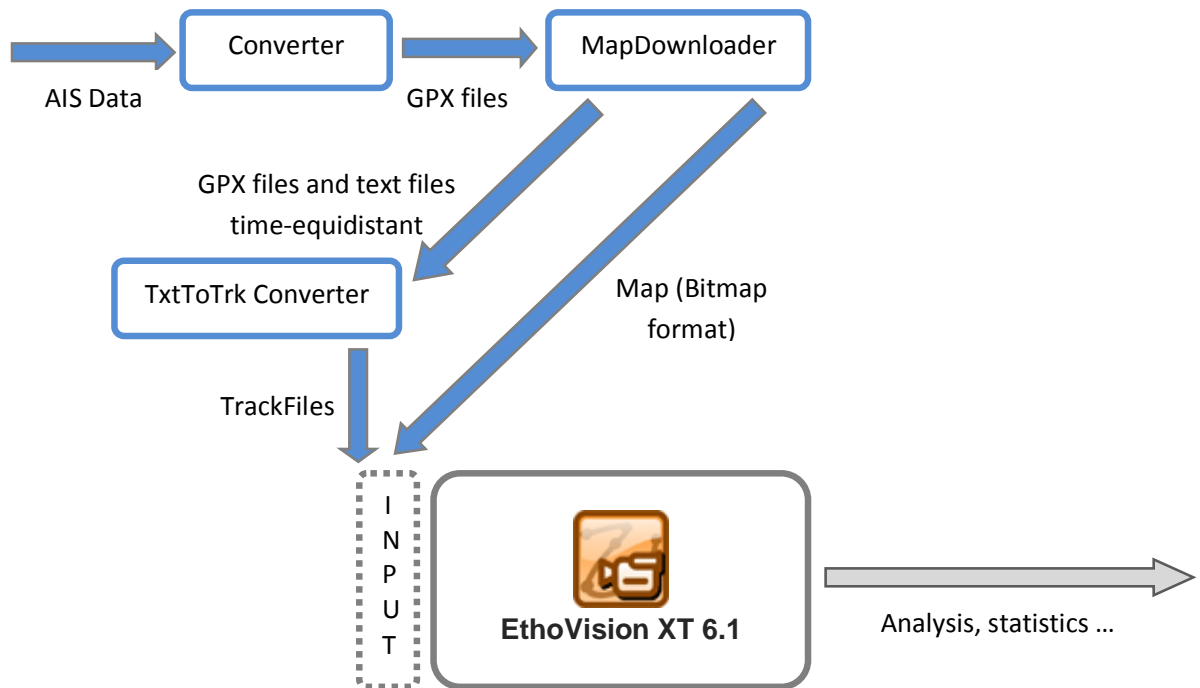
E. EthoVision XT

Modifications on EthoVision XT were one of the more difficult parts of this project. I had to modify a big existing project (more than 100 sub-projects and around 7000 code files). So, the first part of this consisted in understanding the architecture of this software. Then, I had to select all sub-projects on which I had to do some modification and analyze them to understand their working.

I mainly modified three projects: the one about graphic user interface (GUI), the one about arena settings to allow giving as input a background image instead of video file and the one about acquisition to allow giving to EthoVision XT trackfiles created by TxtToTrk Converter.

The code needed to do these actions is not large but the difficulty was to find how and where to write this code by following the strict coding rules imposed.

Once this part done, the process to create a fake experiment from AIS files is as follow:



The next steps for this project will concern this part, namely improving the import of background image and trackfiles in EthoVision XT.

F. Limits

These tools and modifications encounter some limitations. The main one is about time in EthoVision XT. If trackfiles contain trackpoints interpolated by seconds, when we give these trackfiles as input in EthoVision XT, we can see that acquisition is really fast. It's due to number of frames per seconds. EthoVision XT uses by default 25 frames per seconds when it creates a trackfile. When we give a trackfile created by TxtToTrk Converter, EthoVision XT will extract 25 trackpoints (25 seconds) to create one second of acquisition. A solution could be to duplicate each trackpoints 25 times. But it'll multiply size of trackfiles by 25. The best way to resolve this is to modify EthoVision XT to allow it to accept 1 frame per second.

Another limitation was about the fact that all created tools were prototype with lots of tests, lot of useless code and a bad error management. Particularly MapDownloader and TxtToTrk Converter were not optimized. To resolve it, Noldus IT hired me for three weeks after my internship.

Conclusion

This internship was really enriching. Beyond my work at Noldus, I discovered new country, new people and new way to work.

I did a lot of progress particularly about programming. I also discovered the difficulty to intervene in existing project. By using Microsoft DotNet technologies, I participate to impose new programming approach.

Although tools I created are now used for a testing phase, it probably needs new eyes to improve treatment and usability. I began it by working three weeks after this internship from home. It was an interesting experience too.