TSUNAMI, AN UNIFIED IN-FIELD MEASUREMENT AND ALIGNMENT SOFTWARE FOR EXPERIMENTS AND ACCELERATORS AT CERN LARGE SCALE METROLOGY SECTION

P. SAINVITU, P. DEWITTE, J.-C. GAYDE, D. MERGELKUHL, D. MISSIAEN
CERN, Geneva, Switzerland

ABSTRACT

TSUNAMI, The Survey Unified Notepad for Alignment and Measurement Interventions is a new software designed to replace two existing software, mainly dedicated to data acquisition and beam component alignment using geodetic instruments. The use of different programs has been justified historically by the different needs, equipment and types of user. The two main ones have been written in obsolete VB 6.0 and VBA. Maintained for 20 years by different persons the code is now a mix of procedural and Object Oriented programming.

The motivation is to create a single, modular and easy to maintain software written in a popular language for Windows applications (C#), that can be used in a "Standard mode" guided through well-defined steps or in a free "Advanced mode".

The approach is to build the application as a collection of wizards guiding the user through alignment and measurement modules, composed of more basic ones such as management and compute sub-modules.

This paper presents the functionalities and the development strategy of TSUNAMI.

INTRODUCTION

At the "European Organization for Nuclear Research (CERN)" the Large Scale Metrology section is responsible for dimensional measurement and alignment of particle accelerators, of their transfer lines and of the physics experiments.

For this purpose, among the main instrumentation in used is the following (see Figure 1 from left to right):

- Theodolites, total stations and trackers to measure angles and (or not) distances;
- Levels to measure offsets to horizontal planes;
- Ecartometers to measure offsets to vertical planes;
- Tilt Gauges to measure inclinations.

Together with many accessories such as (see Figure 2 from left to right):

- Prisms; • Levelling staffs; • Tilt gauge supports.

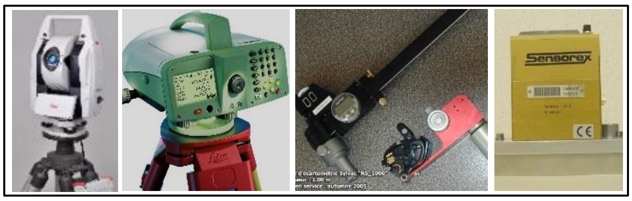

Figure 1: Main types of instrument

Nevertheless, the two programs share the following:

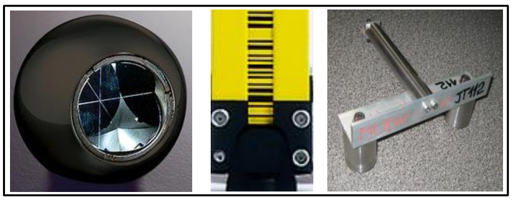

Figure 2: Main types of accessories

To acquire the geodetic measurements, two software applications are used in the CERN survey team: SFB, acronym for Survey Field Book, is used in the accelerators and Carnet4000 is used in the physics experiments.

Historical background

The design of the two programs was carried out for the construction and maintenance of the Large Hadron Collider (LHC) in the mid-90s. They run on a Windows® platform and they were developed in Macro Excel VBA and Visual Basic 6. Their codes were maintained over the years by different surveyors.

Their functionalities are historically different and adapted to the location and environment of use:

- In the accelerators: the alignment procedures are well-known and controlled sequences are needed to ensure the same quality in all operations;
- In the experiments: the diversity of the measurements and the constantly moving configurations require a maximum of flexibility at all levels.

The main divergences are shown in the Table 1.

	Accelerator	Experiment
Measurement:	Repetitive	Diversify
Precision:	Well-defined	Variable
Sequences:	Controlled	Non-existent
Main users:	Industrial Support	CERN STAFF
Coordinate Syst.:	Global	Local
Most processing:	Postpone	Immediate

Table 1: Sub-section divergences

- The business model (elements to measure or align, points, survey networks, geodetic measurements, …);
- The instruments used and their communication libraries;
- Calls for libraries of common calculations;

Main drawbacks

During the last couple of decades, most of the maintenance of the two programs consisted in adding new instruments, replacing computing dependencies and adding the new functionalities brought by those. The same work had to be done on both software.

These changes have been performed by several different surveyors. The resulting code is sometimes confused. It contains a lot of duplicated parts and mixes procedural and object-oriented programming.

The languages have been outdated and show some limitations.

Tablets can be useful in field but the existing GUI are not design to be efficient on touch screens.

The challenge

It was decided by the CERN survey team to merge the two programs into a single one in order to:

- Unify the accelerator and experiment survey data acquisition tools;
- Facilitate the maintenance and the integration of new features and devices;
- Open doors to more up to date interfaces (touch screen, 3D).

A complete rewriting of the software has been chosen to dispose of the obsolete languages and of the code which is hard to maintain.

The challenge is to create a single application that fulfil all the constraints (see next paragraph) of the two modes of use.

The software solution will be named TSUNAMI acronym for "The Survey Unified Notebook for Alignment and Measurement Interventions".

CONSTRAINTS

The project is starting from beginning but is tighten by a large number of constraints including those which originally explains the existence of two separate programs.

Multi-User

Two type of user must coexist. The *advanced user* must have access to all the functionalities in a flexible way while the *guided user* must be assisted step by step through measurement procedures.

Different languages for the displayed information should be available, at least English and French which are the CERN official ones. Nevertheless, it could be interesting to easily add other language in case of future foreigner manpower support contractors.

Use cases

The application has to cover the use cases of the initial applications and fill the needs not covered by the present programs. The left part of the Figure 3 shows the main use cases. They represent all possibilities of alignment and measurement involving polar, distance, tilt, and offset measurements.

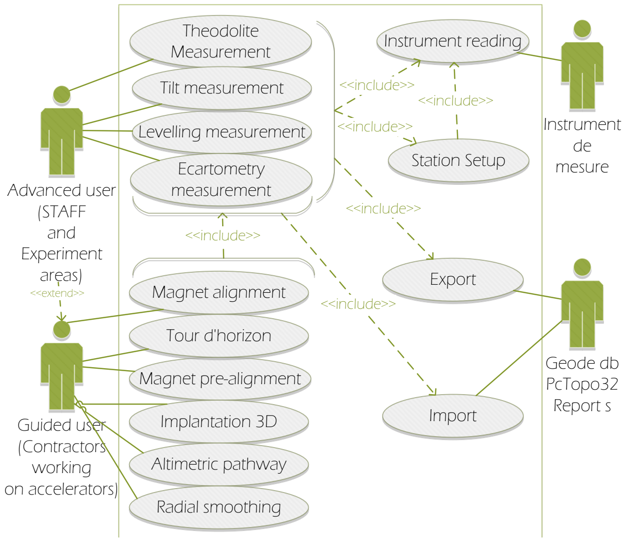

Figure 3: General Tsunami use cases.

Different views of the data.

The main types of data processed by the program are points, components, instruments and observation data. The components are survey networks, accelerator elements or parts of detectors to be aligned and geometrically described by bunches of points. These points can be reference points, station positions or fiducial marks. The measurement of those points by the instruments generate observation data.

Those data must be visualized as a detailed list but also as a tree structure and 3 dimensional view to ease the understanding of their structures.

The dependencies

TSUNAMI is depending on several external dynamic libraries and executable files. The Figure 4 shows the data flow through the different dependencies of calculations and communications. Most of the computation is processed by executable files developed in the CERN survey team. Those requires the writing of proprietary formatted input and output files.

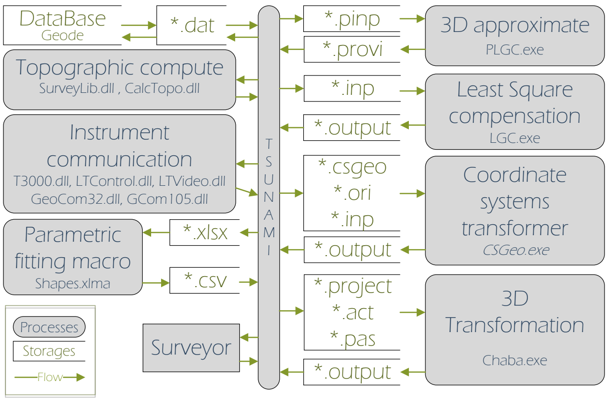

Figure 4 : Gane-Sarson Diagram (Data-flow)

Coordinate systems

The application should deal with different types of coordinate systems. In the experiments, two main coordinate systems are in use. They are Cartesian, the survey system is linked to the gravity and the physiscits one is linked to the nominal beam line. In the accelerators the main system used is the CERN Global Coordinates system (CCS).

Environment of use

As an acquisition data application, the software will be mainly used in the field, inside the tunnels and the caverns, therefore it will run on a laptop or a tablet. To fit with the tablet use, most of the interaction must be possible with tactile gestures.

As the software will be intensely used during the period of shutdown, a great concern should be applied to the ergonomics.

Note that due to its specificities, the application is implemented and will be maintained by surveyors from the Large Scale Metrology section.

METHOD

Development process

The development is following an "Unified Process" method. It is iterative and incremental. It is divided in phases and time boxed cross-discipline iterations. It is centred on the architecture, risk-focused and driven by the use cases. The Figure 5 shows the evolution of the work in different disciplines and the release of the iterations over the progress of the project.

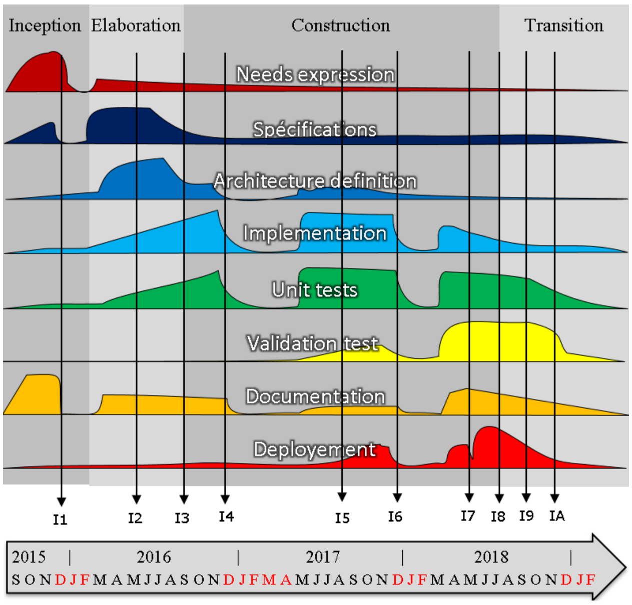

Figure 5: Relative emphasis of disciplines over time

One of the main tool used in the method to formalise the needs, the interactions, the structural and the behavioural decisions, is the *Unified Modeling Language* (UML). UML provides a standard way to visualize the design of the system through a serial of views and diagrams. Figure 3, Figure 4 and Figure 6 are 3 examples of those diagrams.

Platform & development language

As all the dependencies run under the Microsoft© Windows® Operating System and the instrument API* are delivered with .NET† wrappers. It was logical to stay on a Windows environment and to choose C# as programming language as it seems to be one of the most commonly used and taught for the .NET implementation. It is easy to learn and promoted by Microsoft.

Workflow

The main workflow of the application can be described in an activity diagram (Figure 6). The first step is to identify the user as *advanced* or *guided* and therefore allow or not the *advanced* and *management* modules.

* Application Programming Interface (API) is a set of subroutine definitions, protocols, and tools for building software and applications.

† .NET Framework is a software framework developed by Microsoft.

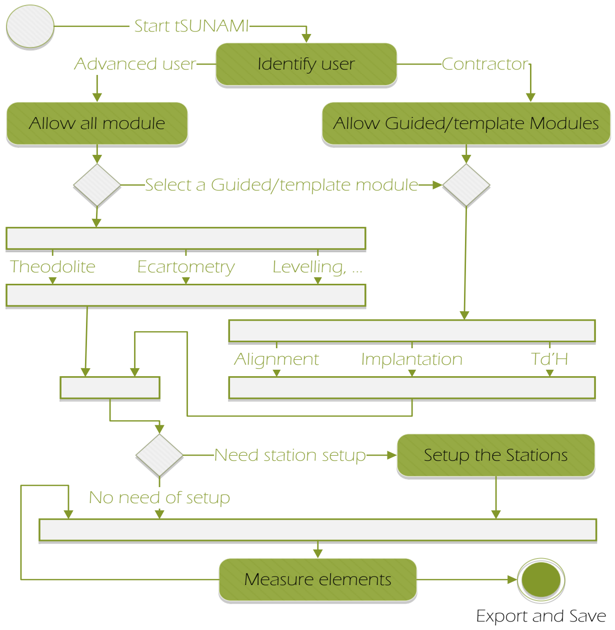

Figure 6: Main workflow

After the selection of a desire module, an instrument setup is run if necessary. Then the measurement can be acquired or the alignment performed. Finally, the acquisition must be exported and/or saved.

Architecture

To ensure the quality of the resulting software application, architectural choices are done to improve the non-exhaustive list of indicators shown in the Figure 7.

Figure 7: Tsunami quality criteria

Reusability, extensibility and maintainability are provided by the intense use of "design patterns" which are general reusable solutions to commonly occurring problems within a given context. They are decreasing the coupling of the software components making them easier to reuse and the application easier to extent. Their formalisation gives a common terminology which added to the existence of a naming convention help the maintenance.

In order to increase the reusability and the extensibility, the application is built as a set of modules operable separately, in parallel or included one in each other. The Figure 8 shows an example of module imbrication. The main module is TSUNAMI itself, it is composed of a *log*, a *module manager* and a serial of *advanced* and *guided* modules.

Figure 8: Modules imbrication

Those *advanced* and *guided* modules are made of *measurement* modules and *element manager*. While the *advanced* modules have access to the complete functionalities of the *measurement* modules, the *guided* ones can only access them sequentially according to procedure steps. The *measurement* modules contain *instrument* and *measure manager* and some *adapter* modules ensuring the communication with the computing dependencies.

The modules which need a graphical user interface (GUI) play the role of presenter (middle-man) in a module-view-presenter pattern. This architectural design pattern facilitates the testing and allows the replacement of a view without any other modifications.

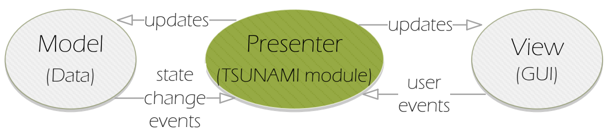

Figure 9: Model View Presenter

The interoperability is ensured by the use of XML‡ as a normalized language for the exchange of data. The vocabulary and grammar are defined by the structure of the objects which is exchanged (use of .NET XmlSerialization which create XML file from the state of an object in the memory).

The transparency is provided by the existence of a *log* file and a *module manager*. The *module manager* allows

‡ Extensible Mark-up Language is a mark-up language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

the user to browse the structure of the modules hierarchy (see Figure 10).

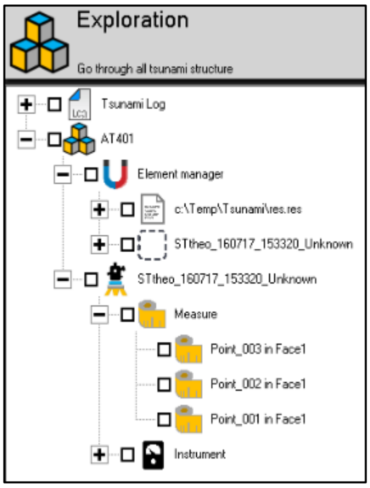

Figure 10: Module manager

The *log* is accessible at any time and keeps track of all the events (including exceptions catching) associated with their respective senders (see Figure 11).

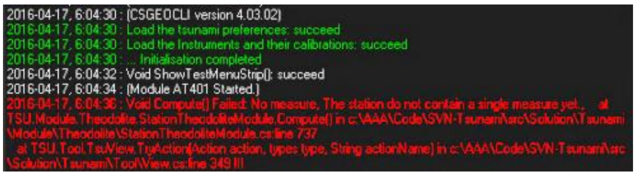

Figure 11: Example of reported logs

The conviviality is brought, amongst others, by:

- The size and the self-explaining character of the controls (see example in Figure 12);

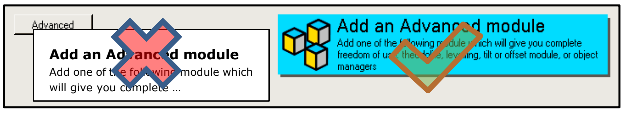

Figure 12: Action button

- The customized design of the instrument graphical interface such as shown in the Figure 13;

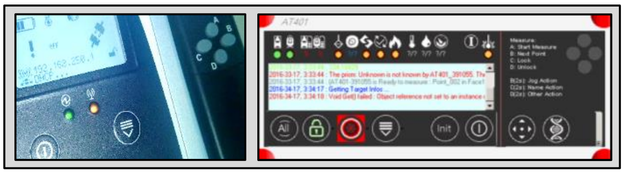

Figure 13: GUI for the Leica© At40x®

- The possibility to visualize data as a list, as a tree hierarchy or as three dimensional views.

---

§ UT is a software testing method by which individual smallest testable part of an application source code is tested independently from the others.

STATUS OF THE DEVELOPMENT

First iterations

The first iterations of the method defined the needs, most of the specifications and a large part of the global architecture, all formalized in a serial of artefacts in UML.

Implementation and unit testing§ has been realized for the *management* modules as well as some of the *advanced measurement* modules.

The TSUNAMI environment is shown in the Figure 14, on the left side the main menu gives access to the global settings and the allowed modules. The right space is reserved for the display of the measurement modules. On the bottom the log keeps track of all the events.

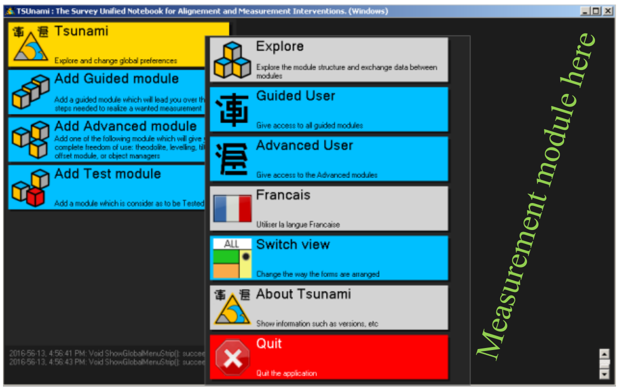

Figure 14: TSUNAMI main view

The Figure 15, the Figure 16 and the Figure 17 show 3 of the *measurement* modules involving *computing* and *management* modules. They give a maximum of flexibility in terms of procedure to follow, of information to introduce and tolerances to accept.

The *manager* modules views are implemented as lists and tree views (see *element manager* in Figure 15 and *measure manager* is Figure 17).

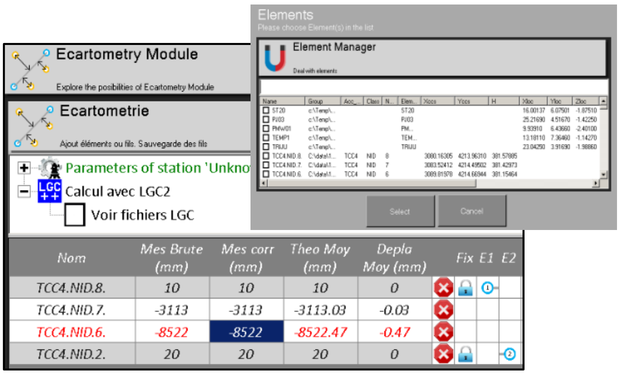

Figure 15: GUI of the *line offset measurement*

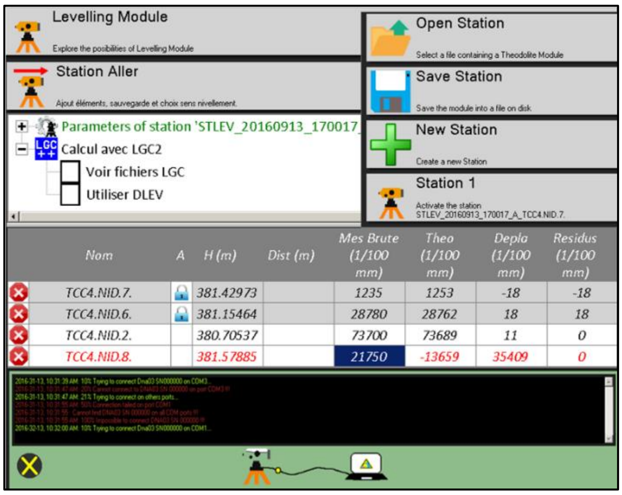

Figure 16: GUI of the *levelling measurement module*

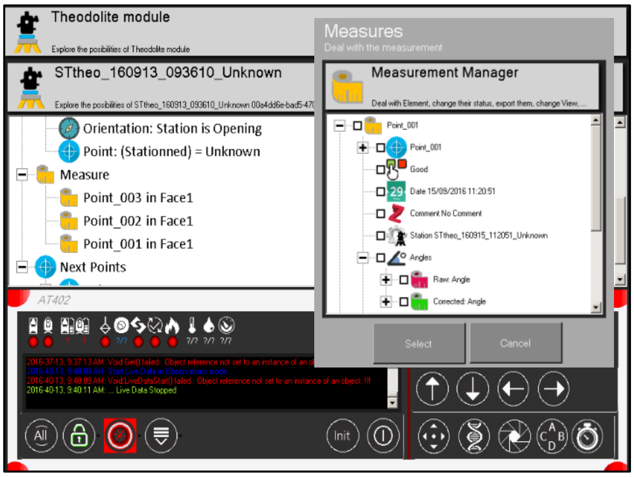

Figure 17: GUI of the *theodolite measurement module*

Next iterations

The next iterations to come will involve architectural choices and the implementation of:

- The 3D visualization of the data;
- The *tilt measurement* module;
- The *guided* modules.

The *guided* modules have to sequentially link the user with a list of *advanced* modules functionalities. This could be done by a sequence of dialog boxes that lead the user through a series of well-defined steps. Those sequences could be done through hard-coding[**] or preferably through XML configuration files. Those files could be created manually or by recording macro commands in the GUI.

CONCLUSION AND OUTLOOK

The development of the TSUNAMI software solution is in progress. It follows an incremental method of development that delivers the application by iterations. The program will replace the two existing ones to unified the accelerator and experiment survey data acquisition tools.

The first iterations have delivered a large part of the global architecture. Solutions have been found to fulfil the requirements and to satisfy the identified constraints.

The application is designed to ensure conviviality, reliability and transparency. This is achieved with the introduction of custom user controls and of log modules.

The architecture of the program and the related documentation should ease the maintenance. The achieved modularity should stimulate the extension of the software. The modularity also allows the reusability of the code in future projects that could require measurement or geodetic data management.

The core functions such as *ecartometry*, *levelling* and *theodolite measurement* have been successfully implemented. The next iterations will deliver the tilt functionalities as well as the *guided modules.*

Full-scale tests will be performed by members of the CERN survey team during next year. Corrections will be applied and feedback will be taken into account. A validated version should be available for the next long shutdown of the CERN accelerators.

REFERENCES

[1] A.-V. NAEGELY, G. CHAMBLAS, P. ANDREO and M.-L. RUIZ, SFB Guide Utilisation 2014, Genève, 2014.

[2] P. DEWITTE, "Fusion SFB-Carnet 4000 - Spécifications SFB," Genève, 2014.

[3] P. SAINVITU, "Fusion SFB-Carnet 4000 - Spécifications Carnet4000," Genève, 2014.

[4] S. MAHMOUDI, "Expertises digitale et logicielle," Mons, 2015.

[5] E. GAMMA, R. JOHNSON , R. HELM and J. VLISSIDES, "Design patterns, software engineering, object-oriented programming," 1994.

[6] D. MERGELKUHL, "Comparison of SU Acquisition Software CARNET4000 and SFB to evaluate a possible Unification," Genève, 2014.

[**] Hard –coding refers to the software development practice of embedding directly into the source code what may be considered as an input or configuration data.