

Presentation of Tplan 29.1 (or The End of the affair)

March 23, 2011

Hello all

The subtitle is borrowed from a novel by Graham Greene, otherwise no links to that. My use of that phrase is to stress that I have decided that version 29.1 is my last major effort to complete a process which started in 65/66 and which have been my sole professional occupation since that. (Systems like Tplan can always be improved, but my own engagement will be reduced in the future)

I will not tell “the story of my life” but concentrate on the major parts of the development of Tplan the last 10 years. A natural starting point is the most frustrating period of my life: The years 95-99. At that point in time Tplan was a DOS- application. I still regard that as a fairly good system. I had a sufficient happy group of customers and reasonable economic income; in short on the surface things appeared quite satisfactory.

At this point in time we got what I like to describe as the Windows- hysteria. It didn't matter if a system was pure rubbish with no qualities; the only requirement was that it could run under Windows. I have always been a warm supporter of a standard for user interfaces. My opinion is that the introduction of Windows is one of the most important steps forward in the computing world, but Windows on its worst was sheer hell in the initial phases. At that point in time it put the development of sensible software backwards many years, and many good systems were suddenly obsolete. In hindsight I realize that this perhaps was a process we had to go through.

I had to adapt to modern times, and silly enough I made a new user interface in DOS where I simulated the Windows functionality. (One of my biggest programming efforts ever) This was finished in -97, and now I was really in for a big disappointment. With pride I presented my new product to potential customers. Most were fairly amateurish timetablers; however they had enough PC- knowledge to utter the following intelligent remark: “This is not pure Windows” There we got stranded in the paddock and never got off to the races i.e. to demonstrate the real qualities of Tplan. Furthermore, in 1999/2000 we realized that my DOS/Windows would not cope with future Windows systems (NT in particular.)

My customer base started to decline (not because the qualities of Tplan). It does not take much imagination to understand that I was on a losing track. As luck would have it, I at point in time came in contact with Lasse Storr-Hansen. He offered generously to convert my DOS-system to a Windows platform which I gratefully accepted. I guess Lasse was aware of the data entry problems in Tplan, but believe he had not fetched all problems with the more difficult adjustment phase. (In hindsight I hope that Lasse does not regret too much his original somewhat naïve offer.) Regardless:

1. This was to first major step in salvaging a good DOS system for the future Windows world. It was more or less a direct copy of the earlier DOS- system and was released in 2002 as version 26. We added a number a facilities to simplify life for the timetabler.

The version was well received, but we still had a small customer base expanding slowly. I made 2 important decisions:

- As a result of finally being rid of all headaches with screen interfaces and Windows standards, I experienced a real brainstorm. I realized that the problem area I had been working with for almost 40 years possibly could be handled in a far more general way i.e. I could possibly generalize my algorithms considerably. That vision was far too tempting to be left untouched. I regarded the whole enterprise as a Win- or Win – situation: If my new ideas didn't represent a considerable improvement, I could comfort myself with that the ideas I been using all the time was in fact fairly good. If on the other hand I got a considerable improvement, well then I had found what I was looking for! Finally in 2005 I was ready to present Tplan version 28. *The result surpassed all my expectations. The user interface in the older version 26 and version 28 was very similar. The latter version represents however a quantum leaps with regard to algorithmic improvements. At the same time we made the screen interface considerably improved*
- I was not that keen on expanding the user base for Tplan too fast. I was more interested in building a solid user group which could collaborate and create a firm base for the continued use of Tplan. The result of that is the large group of Danish gymnasiums currently using Tplan

2. The second major step in creating Tplan as the system I wanted it to be, was the algorithmic improvements following from replacing version 26 with version 28.

In the period 2004-2008 we did a number of important additional things. I wrote an English User Manual (A considerable task ripe for a revision now. My new working title will be: “Timetabling without tears”.) In addition I worked a lot with more complex timetabling problems like Component Setting and Teacher Pools. Lasse on his side was working with important interfacing work with other school administrative system like for instance Lectio and Ludus.

Still during 2008 I felt not too happy about the situation as a whole and the reason being: We had a very good data entry system while the final adjustment phase was far from efficient enough. I noticed that old hands in timetabling preferred techniques like the previous DOS system due to the new Windows processes being both complex and slow. Novices had to fight with this to the best of their abilities and often this was a mixed blessing. The reason for this can be blamed on both Lasse and me. We had both made operating programs for adjusting a timetable, but both had underestimated the time doing so; in short we had delivered lousy solutions for that part of Tplan. Current Windows uses drag/drop- techniques for very many operations, and we lacked that. A Drag/drop is of course very simple to perform in Windows, but our problem was much more complicated than that: Having done a Drag/drop this info had to be sent to Hmtp for evaluation of the timetabling consequences and the result of complex operations sent back to be presented for the user. Luckily Lasse detected the possibility of using a “floating” Kickout- list. (Which means that this list may be onscreen simultaneously with the timetable.) This is vital for making drag/drop at all in Tplan.

That was the situation autumn -08, and I decided to do my last major programming effort. The following is a summary of what has been done for almost 3 years now:

- The most important thing was a complete redefinition of the internal data structure in both Wintp and Hmtp. I speak only for the latter and state that this had considerable impact on how Hmtp had to be organized. The main changes were done in data structures common to both Wintp and Hmtp. The normal user will not notice directly the changes in internal data structure, but he will surely notice the immediate consequences. Just above I was talking about slow and lousy programming. Well, that has disappeared now. What follows in the next

paragraph will of course be the most convincing answer to why a new internal data structure is so important.

- We have implemented a complete drag/drop technique which I am perfectly happy with for the moment. (Experiences from users might modify this.). We will make separate documentation of these facilities, and I will not say too much here. This is of course vital for all users of Tplan. I am very experienced in timetabling and believe I know all tricks of the trade. Even so, when finally having access to a sensible drag/drop, I reduce the time needed for doing manual adjustments enormously. I have never before tested a specific facility in Tplan to such an extent. That process has had many ups and down. Sometimes I have been utterly frustrated by all the mess produced onscreen and at other times being happy when things fitted nicely together as I intended to.
- The two points above explains most of the time spent for producing version 29. But during that process Lasse explained to me why I absolutely needed to replace my old FORTRAN compiler (16-bit) with a newer version (32/64- bit). I invested in that and got considerably additional work. Today I am perfectly happy with the new compiler. I automatic got a Hmtp running twice as fast as before. (You used 28.6 last year which were made using the new compiler and saw for yourself the difference in speed.) But there more to it than that. During this conversion process I detected an “algorithmic snag” allowing me to increase speed with a factor 2 compared with 28.6 i.e. Tplan runs app. 4 times faster as older versions of version 28. We are now backing a version running faster than the old outdated version 26.
- We have simplified the Run Dialog.
- We have reorganized how the user gets access to run options and eliminated options which a standard user never needs and some run options also have additional parameters.
- We have reorganized the print files from Hmtp with the hope that the user more easily finds the various important info depending on how far the timetabling has progressed.
- The checking of formal errors (Inspect Button and initial stage of Transfer program) is now performed by Wintp and not Hmtp. Lasse’s way of doing that is far better than my way, and I am impressed by the current status of this facility.
- We have removed the possibility of creating DOS-files for the final tables. No one has asked for such a facility for the past 8 years. Wintp has far better ways of listing the final result than Hmtp. More usual than that is to convert the final result to Excel –files or perhaps even more usual is to send the result to school administrative systems like Lectio or Ludus. (In the long run I expect everybody to use such methods for the final result.)
- There is also some algorithmic improvements in version 29 compared with version 28 but these are far from the same drastic change we saw from version 26 to 28. I have in particular worked with the more problematic aspects of timetabling like for instance component setting. This will be described in a separate paper.

3. The third major step in creating Tplan as I intended to be, is the vastly improved interface in version 29 compared with older versions of Tplan. In some respects I also like to speak of a quantum leap here too. I feel that the average user will appreciate even more the improvements in improved user interface in version 29 compared with version 28 than any improvements I have done with the algorithms in later years. (And which I of course regard as being more important.)

Having been through this considerable development work for almost 3 years, I have toyed with a number of additional ideas which could be a basis for a future version 30. I will not explain all this here. I will only state that my ambition levels goes much farther than mainly Danish gymnasiums. I

feel I have a perfect tool for coping with the most complex timetabling problems. (Like for instance UK schools.)

However, I have a more major concern now and go back to the start of this paper: Version 29 is my last major development effort. Due to aging years I refuse to be in the “first front line” with respect to further extended use of Tplan. At best I may play a consultative role in the continued use of Tplan. This decision is definitive regardless of consequences. I shall take some initiative to secure the future of Tplan, but don't know what will happen.

I shall give a short comment to my own work. My evaluation of that has had a lot of ups and downs. Sometimes I felt that I had no method at all, and that the whole was only a collection of reasonable guesswork. At other times I imagined I had detected some new basic mathematical truths. These days I have an in-between attitude: I am aware that I have not found the ultimate and final truth about the general school timetabling problem, and I am also of the opinion that the search for such a thing is without purpose. On the other hand I am proud of the behaviour of Tplan from a pragmatic point of view. I have not seen any other timetabling system being able to solve realistic timetabling problems with the same completeness and qualities as Tplan. I am humble for having created a “baby” being infinitely cleverer than myself or any other human timetabler. From a professional point of view I can live with such a conclusion. If someone else has produced a similar good alternative to Tplan, I pay my due respects since I am well aware of the work behind such an effort.

If I should give any rationale for devoting my career to school timetabling, I can only say: **I did it just for the fun of it, and I believe I have beaten(or at least gotten drawn with) my “invisible opponent” ; meaning the infernal complexities of the combinatorial world .**

Yours

Harald Michalsen

Dr. Techn.