Herschel "Determining User Wishes" Lessons Learned

Overview:

During this session the audience was polled how to best determine user wishes. No clear way forward was found, the views of the audience were split. Generally it was felt that that it is not easy to engage persons not directly involved in the science ground segment. On the other hand such external persons can provide essential inputs to avoid that the facility is tuned implementing the wishes of the expert in-crowd.

Key Points:

1. **Involve all kinds of users (from naïve outside users to expert insiders) in the development as soon as possible.** This needs to cover all areas: Software, archive products, documentation. From: H34, H51, H102, H113, H140 and H179.

2. **DP workshops and Webinars helped to gather the feedback of real users of the system.** Such feedback needs to be formalised in tickets, with a mechanism in place to ensure that CCBs act on the input of the out-crowd instead of focussing to implement the wishes of their in-crowd colleagues. From: H83, H212 and H213

3. **Additional ways to gather requirements.** Involvement of users is essential: Agile practices encourage the communication with users, adding users or including a representative of them inside the development team. Also scrum sessions (http://en.wikipedia.org/wiki/Scrum_%28software_development%29) are considered to be a good way to extract user requirements. Regardless the methodology used, uses cases are an invaluable tool for analysis, extracting test cases, documentation, and also for planning. Additional discussions on the most efficient way to gather and formalise requirements, and to which level of detail they should be provided, were considered to be necessary. From: H7 and H179

4. **All stakeholders, also developers, should be engaged in requirements definition.** Such a dialogue is essential to manage user expectations, and derive cost-effective solutions where the key requirements are implemented with a reasonable effort. In software development new problems and solutions arise at any phase of the development process, as all the stakeholders get a better understanding of the problem. So all of them should be able to collaborate in order to get the best solution. From: H92

5. **Don't promise forward compatibility** Software systems evolve to cope with new requirements and general advances in the technology. While backwards compatibility – reading data generated with a previous software version with new software versions – can be achieved, forward compatibility – reading data generated with a recent software version by a previous software version – cannot be ensured. From: H92

6. **Pair software developers with astronomers and end other users of the system.** An efficient way to steer software development is to pair developer with end users. From: H194

Discussion Points:

The audience was polled whether

1. **Helpdesk questions should be public.** Under the assumption there is a "private" button, 60% of the participants preferred that helpdesk questions should be public viewable and therefore accessible by search engines

2. **The content of the software ticketing system should be public.** Under the assumption there is a "private" button and the software ticketing system is "read only" for persons outside of the science ground segment , 70% of the participants preferred that the ticketing system remains private

Additional Points that arose during the session:

1. **ESA should showcase Herschel (and other missions) in big conferences to engage the out-crowd Herschel community.** From H214

2. **The opportunity for all stakeholders of Herschel, users and developers, to meet should be continued also at this stage of the mission.** From H87, H104, H117 and H125.