P-21. Expanding Bytran Capabilities with Hardware Sensors and Device Interconnect

D. Pliutau [a]

[a] *bytran.org, 1083 Independence Blvd., P.O. Box 134, Virginia Beach, VA 23455, USA*

We describe new functionality implemented in the bytran software[1] since its original release including the upcoming update to version 1.3. The new features include the support of external and built in hardware sensors for semi-empirical modeling and bytran communication capability for remote control and data retrieval.

The hardware sensor support was introduced to provide an option to acquire empirical data from ambient meteorological sensors (temperature, humidity and total pressure) for calculations as an alternative to manual entry. The sensors supported by bytran 1.2 include the external SensorTag module over Bluetooth[2], and the currently limited use of the built in cell phone sensors[3]. The Raspberry Pi's Sense Hat[4] support is planned for version 1.3.

To enable simplified integration of bytran into existing hardware systems without the need in source code modifications the capability to control bytran remotely was implemented in version 1.3. Such remote operation may be carried out over the Internet (with Bluetooth and the Serial Port support planned) using the newly introduced bytran communication protocol[5] and a variety of computer languages and programming environments. In particular, bytran execution over the Internet is based on the WebSockets protocol. WebSockets communication may be carried out using a wide array of programming languages including HTML5 and JavaScript from within a web browser. As such simple web-based applications may be developed to remotely control and retrieve calculation results from bytran[6] running on dedicated hardware such as a cell phone.

The above new features were meant to improve usability and make it relatively easy to integrate bytran into existing hardware systems. The near-term future plans for further bytran improvements include code optimizations to increase speed and the implementation of the Hartmann-Tran lineshape profile.

Recently Qt bindings with the Python language have gained attention due to the increased popularity of the Python language. Libraries implementing such capability include PySide and the PyQt. There has been recent work to extend their support to the Android and iOS operating systems. Originally developed as a separate branch

[1] D. Pliutau, K. Roslyakov, *Earth Sci. Inform.*, 10, 395 (2017).
[2] http://www.ti.com/sensortag
[3] http://doc.qt.io/qt-5/compatmap.html
[4] https://www.raspberrypi.org/products/sense-hat/
[5] http://www.bytran.org/protocol.htm
[6] http://www.bytran.org/websockets.htm

PySide is to become included[7] into the main Qt development environment at the end of 2018. This opens a future possibility of creating a hybrid application by combining the bytran's cross platform Qt interface with the HAPI's Python computational code to function under mobile and desktop environments.

[7] http://blog.qt.io/blog/2018/02/22/qt-roadmap-2018/