An occam Model of XCHANs

Fringe Presentation

Peter H. WELCH

School of Computing, University of Kent, Canterbury, UK

`P.H.Welch@kent.ac.uk`

Abstract. Øyvind Teig, in [1], proposed a higher level channel construct (XCHAN) that attempts to reconcile those wedded to asynchronous message passing with the synchronous form in CSP. Sending a message does not block the sender, but the message may not get sent: the sender receives a success/fail result on each send. The XCHAN provides a conventional feedback channel on which it signals when it is ready to take a message. Being ready means that it has space (if it is buffered) or a reading process has committed to take the message (if it is not buffered). Sending to a ready XCHAN always succeeds; sending to an XCHAN that is not ready always fails. The sender can always wait for the signal from the XCHAN (whilst ALTing on, and processing, other events) before sending. We can model an XCHAN by a process in occam-π. Buffered XCHANs are easy. Zero-buffered XCHANs are a little harder, because we need to maintain end-to-end synchronisation. However, occam-π's extended input (??) and output (!!) primitives enable the process implementing the XCHAN to be hidden from its users. Unfortunately, extended outputs are not yet in the language, but their semantics can be simulated by making the receiving process read twice and ignore the first (which is just a signal whose taking must commit the reader to its second read). An important message is that sane higher level synchronisation mechanisms are usually not hard to implement efficiently via the low level CSP primitives offered by occam-π. Although not yet measured for XCHANs, it is likely that such simulation in occam-π will have competitive performance with direct implementation elsewhere. occam-π code and documentation for this model is available from [2], together with the slides from this presentation.

Keywords. XCHAN, occam-pi, concurrency, synchronisation, simplicity

References

[1] Oyvind Teig. XCHANs: Notes on a New Channel Type. In *Communicating Process Architectures 2012*, pages 155–170. Open Channel Publishing, August 2012.
[2] Peter H. Welch. An occam Model of XCHANs, 2013. `https://www.cs.kent.ac.uk/research/groups/plas/wiki/An_occam_Model_of_XCHANs`.