

Addendum to 18 November 2010 customer debrief minutes

The System Operator (SO) held a workshop on 18 November 2010 to brief participants in the SFT trials on the SFT project and performance, results from the parallel operations, and outcomes of the scenario testing. The minutes have been placed on the SO website.

From that meeting, the SO undertook to review the possible mechanisms for identifying and highlighting constraints that were very sensitive to demand changes. The SO has reviewed the options presented at a very high level and makes the following response. The SO can undertake further investigation and development on any or all of the options below; however, such development will be subject to the CAPEX process mandated by the System Operator Service Provider Agreement and industry indication of priorities.

Option	Comment
1. Modifying the NBT to allow customers to identify sensitive constraints in earlier schedules	<p>This would appear to be a relatively straight forward option to implement, as it would require a different NBT value assigned to each study mode (SDPQ, SDS etc...) so that the 'earlier' schedules could identify constraints that were just outside the normal NBT.</p> <p>This may not resolve MEL's issue because all it does is increase the number of constraints written by SFT. It doesn't offer any way of deciding whether any of the newly exposed constraints are particularly sensitive to demand and are therefore likely to become a problem if the demand forecast changes.</p> <p>Participants may develop experience over time as to which constraints identified in this way are more likely to crop up as 'real' constraints closer to real time, and may be happy with that. In fact, it is something that would be of potential use to SO as well (statistical likelihood of particular branch flow violations below the normal NBT in planning time becoming published constraints in the SDPQ timeframe).</p> <p>We suspect there may be a technical hitch if we drop the SFT NBT below 0.85. The extra constraints generated wouldn't be visible to participants since they would fall below the publication threshold.</p>
2. Generating two SFT runs with modified demands to allow the calculation and publishing of the sensitivity of the constraint to demand	<p>Technically this is possible, but would be quite unwieldy & would obviously increase the solver workload significantly. It would also introduce another level of complexity into the workflow process (error handling for the primary & secondary runs, managing the incremental demand between the two), and a secondary set of constraints which would need to be handled in a much different way.</p>
3. Using the difference in the demand schedule between runs to approximate and publish the sensitivity of the constraint to demand	<p>Again technically possible, and avoids the additional solver workload of explicitly solving SFT twice for each trading period with a different modelled demand.</p>

As a general comment, the SO was a little hard pressed to decide how we would identify 'constraint sensitivity', since a constraint consists of two coefficients & a RHS. Our best guess is that it would be the sensitivity of the evaluated LHS to demand changes (therefore giving a clue as to whether a demand change is likely to bring the constraint above the normal NBT or not).

It's not clear how the constraint would be calculated – the numerator is presumably the change in the evaluated LHS, but is the denominator the change in island load, the change in regional load (& how do you define region) or the change in a particular bus load (in which case each constraint would need to be associated with a particular bus load for sensitivity purposes)?

In addition, whilst there will always be errors in the load forecast, the greater uncertainty arises from the projected generation profile used to meet the load.

We suggest participants and the SO adopt a 'wait and see' approach to this issue and then determine if this is one they wish to take up. The problem can then be better defined, funding organised, and a solution implemented.