The Theory of Computation
Bernard M.E. Moret
Addison-Wesley-Longman, 1998

Solution Manual
Chapter 6, Problem 6.12

Why is it that both P and PSPACE have complete problems, while POLYL does not, particularly when the argument for the latter is one based on viewing the class as an infinite union of increasingly larger classes? We can view P as the infinite union of TIME(n^k) and PSPACE as the infinite union of SPACE(n^k) and try to apply the same argument. (There is a technical problem with viewing P in that way, since the component classes, TIME(n^k), are not model-independent; this problem does not arise with the component classes for PSPACE and can safely be ignored.) The key difference is in the effect of the chosen reduction on the component classes. The component classes of POLYL are closed under logarithmic-space reductions, as shown in Exercise 6.11. But the component classes of P are *not* closed under logarithmic-space reductions, since a logarithmic-space reduction can take any polynomial amount of time. Thus having a P-complete problem in some class TIME(n^k) does not imply that any problem that reduces to it will also be in that class; it could well be only in TIME(n^j) for $j > k$. The same flaw appears for PSPACE: its component classes, SPACE(n^k), are not closed under polynomial-time reductions, which can use any polynomial amount of space. (Interestingly, however, these component classes *are closed* under logarithmic-space reductions, so that the argument used for POLYL applies to PSPACE under these circumstances and shows that PSPACE cannot have complete problems under logarithmic-space reductions. But P and NP do have such problems: so should we conclude that PSPACE is strictly larger than P or NP?)