will not be used and an alternative mode of operation may be effected.

It is appreciated with respect to Figures 1 and 4 that the FEATURES_OS array is an implementation of the FEATURES list 23 and the hardcoded bit masks are stored at 24. The SUPPORTEDFEATURES_OS array is an implementation of the SUPPORTEDFEATURES list 30 with the bit masks stored at 31. The FEATURES_SPP array is an implementation of the FEATURES list 40 with the hardcoded bit masks stored at 41. The SUPPORTEDFEATURES_SPP array is an implementation of the SUPPORTEDFEATURES list 50 with the bit masks stored at 51. The SPPFEATURES_OS is returned as report 22.

Bit masks are utilized to facilitate the comparison operation. The use of bit masks is not, however, a requirement. Other known feature indication storage arrangements, such as any bit map arrangement, could be utilized to the same effect.

Another example of two software entities that may utilize the invention are two independent processes running within the same computer system controlled by the same OS. Any two software processes which are capable of using an InterProcess Communication (IPC) mechanism to implement interface 14-15 may use this invention.

The above described embodiment was explained in terms of mutually supported features. However, features which are not mutually supported could be included for reporting purposes. That is, the OS could obtain and report on SPP features which the OS does not need to support. These features would be considered optional.

Each of, or at least one of, the software entities maintains/constructs a list of features supported by both. The bits representing optional features may be dynamically referenced to determine whether to use the feature or to effect the alternate mode.

While the invention has been described in its