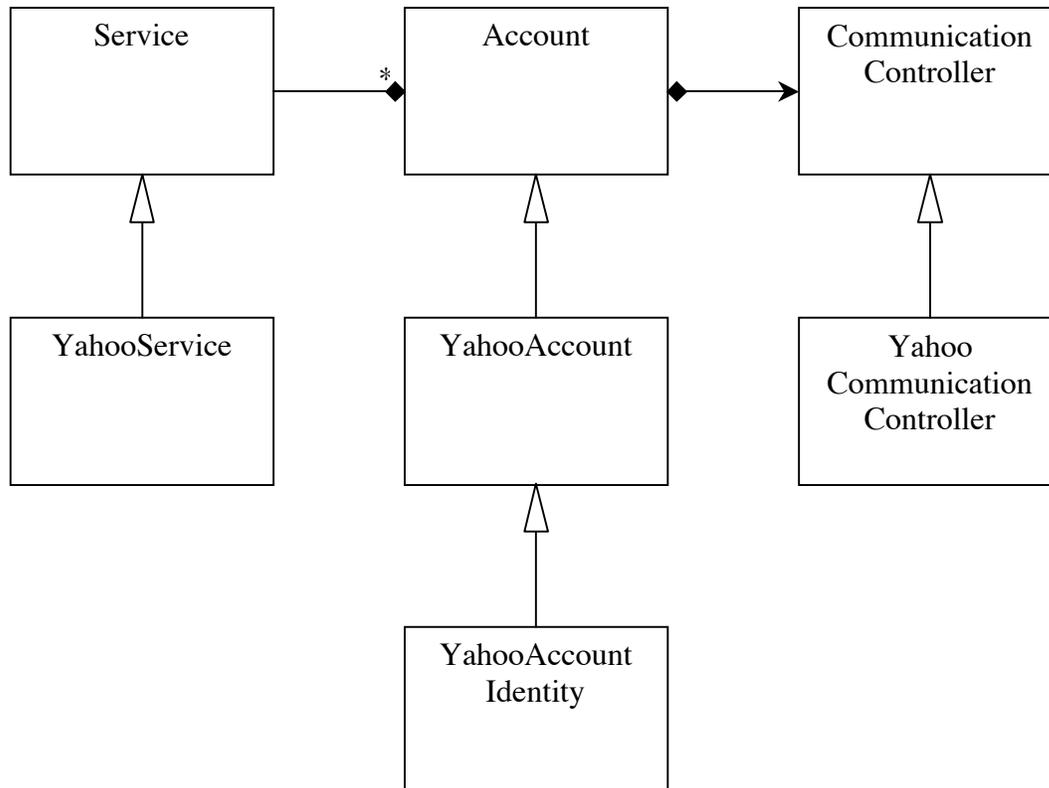


Moving to support for Multiple Accounts.

We have long had the goal of supporting multiple concurrent accounts for a given service. In an effort to help move towards this goal, here are some ideas on how this could be supported. (Note that this is a quick-rough draft and likely full of errors!!!!)

Attached is a “pseudo-UML” class diagram describing a proposed relationship between Services, Accounts, and CommunicationControllers. The basic premise here is that the Account object becomes the center of “navigation” to access other information about this



particular connection. In the above, we have multiple account objects pointing to a single service object, representing that there can be multiple accounts for a single service. Also inherent in this design is that the account points to the communicationController object responsible for communicating on behalf of that account. It is possible that multiple accounts could point to a single communication controller, but in practice there would probably be a one-to-one correlation here.

Notice that the account to service relationship is bi-directional. The service keeps a list of all accounts associated with that Service.

Also notice the concept of a new Subclass of an account, which is an “alias” for a true account. Yahoo calls these “Identities” And there can be multiple identities all pointing to the same master Account.

The basic changes between this design and the existing design are:

1. Service no longer knows about CommunicationControllers
2. There is no longer a [CommunicationController sharedInstance] since there can be more than one communicationController for a given Service Type.
3. Account now stores the CommunicationController object and is responsible for instantiating it.
4. Items in other parts of the code will now store an account object instead of a Service or CommunicationController.

We have not established a firm timeline for this migration, but in the work I did for MessageItem, some of this migration to an “account-centric” universe has already been accomplished. Obviously, not all libraries will currently support multiple connections, but for those that do, I would like to move towards supporting them as soon as possible.

Please let me know your feedback on this idea as soon as possible. This will take some work (especially in the Service Specific Files and Buddy Areas) but is doable.