I don't know if this problem is limited purely to the issue model, or if it exists elsewhere in other models as well...

The addition of the safe_attributes method, well, really the entire paradigm of setting attributes this way, severely hamstrings other developers’ abilities to write plugins that extend the issue model. accepts_nested_attributes_for goes out the window, because now all the submitted for elements for those get ignored and thrown into the bit bucket.

To some extent, I can understand why this was done, but I don’t understand doing it without providing a way to easily EXTEND the safe_attributes array, without having to set some constant to replace it completely. Hopefully this can be remedied soon, as I have plugins dependent upon extending the Issue model :

Same here, also had problems with issue model because of safe_attributes, and I'd like to have a clean way to extend it.

FYI, you may have a workaround using controller hooks: in my case, I noticed that safe_attributes= just prevents affecting the attributes, so I used controller_issues_new_before_save and controller_issues_edit_before_save to assign attributes manually (you can see it here, even if I'm not really proud of it ;))

For now i'm just tweaking the codebase manually, at least until I hear a response on how they want to handle this project side (if they want to handle it at all or if this was all by design...). What you posted isn't that hackish, it would work pretty well. Its still just messier than it needs to be :(
Fixed in r4491. You can now extend safe attributes for a given model using:

```ruby
Issue.safe_attributes 'foo', 'bar'
```

or makes safe attributes conditional:

```ruby
Issue.safe_attributes 'foo', 'bar', :if => lambda { |issue, user| issue.author == user}
```

You can have a look at redmine/safe_attributes.rb.