



Enforced Work Flow

1. Overview

The JMU AcademiCS Application Framework is a set of classes/interfaces that allow for the creation of GUI-based desktop applications. For example, one could use the JMU AcademiCS Application Framework to construct a business suite that consisted of a word processor, spreadsheet, and drawing program. The JMU AcademiCS Application Framework both makes it easier for developers to implement applications and ensures that the user experience is the same across different applications.

2. Notation and Typographic Conventions

{A | B} indicates A inclusive or B.

{A ^ B} indicates A exclusive or B.

{A & B} indicates A and B.

User actions begin with an uppercase letter.

Document states are italicized.

3. Document States

A document may be *empty* (e.g., when an application is started), *unmodified* (e.g., when a document is first opened), *modified* (e.g., after its opened and edited), *unnamed* (e.g., after its created but before it is saved), and *named* (e.g., after it is created and saved). However, only five of the eight possible compound states are possible.

4. User Actions

The application framework supports the following user actions: Close, New, Open, Print, Quit, Save, and SaveAs.



Enforced Work Flow

5. Enforced Work Flow

The work flow that is enforced by the application framework can be represented in two different ways. The first involves the actions that are enabled in each compound state and the second involves the way in which an action cause state transitions. This section contains both.

5.1 Compound States and Corresponding Enabled Actions

Different user actions are enabled in each of the compound states. Specifically:

1. If *{empty}* the user may *{New ^ Open ^ Print ^ Quit}*.
2. If *{unmodified & unamed}* the user may *{Close ^ New ^ Open ^ Print ^ Quit ^ SaveAs}*.
3. If *{unmodified & named}* the user may *{Close ^ New ^ Open ^ Print ^ Quit ^ SaveAs}*.
4. If *{modified & unnamed}* the user may *{Close ^ SaveAs ^ Print}*.
5. If *{modified & named}* the user may *{Close ^ Save ^ SaveAs ^ Print}*.

Note that this is not the work flow used by most applications. Specifically, note that when the system is in a *modified* state the user may not Quit (whereas most applications will allow the user to Quit but will confirm before doing so). Instead, the use most Close before using Quit.

5.2 States and Transitions

The way in which a document can transition between states is described in the following UML State Diagram. The actions that are enabled in each compound state correspond to the possible transitions.

Note that this diagram includes an Edit transition (in blue) that is not part of the application framework itself but is, instead, application-specific.

Enforced Work Flow

