ETS Plugins

The following pages will describe a proposed ETS Plugin Framework (**EPF**). The goal of **EPF** is to integrate Document Viewers with scientific and multi-media applications, so that students can benefit from sophisticated data visualization and **3D** graphics tools. Interactive multimedia presentations can help students intuitively understand scientific concepts, while preparing for exams such as the chemistry, biology, physics, and mathematics **GRE**s.

A second goal of **EPF** is to provide machine-readable structural representations of publication manuscripts, which Document Viewers may use to introduce additional pedagogical content: review questions, student instructions, glossaries, reading assignments, and so forth. For this technology, each publication may provide a "Semantic Document Infoset" (**API**), which divides manuscripts into textual units (sections, paragraphs, sentences, etc.) and identifies document elements such as glossary terms and figure illustration. ETS plugins can then examine a publication's **API** so as to determine how to augment the underlying document with additional instructional and/or multimedia features.

*How **EPF** enables multi-application networking*

EPF refers not to a single plugin, but a toolkit for implementing ETS plugins to be embedded in many different applications. These plugins should be sufficiently similar that students or instructors familiar with an ETS plugin in one context (chemistry, for example) would quickly understand how to use plugins present in a different context. An important **EPF** feature is that distinct ETS plugins would be able to communicate with each other. In particular, Document Viewer plugins would send data to plugins for scientific or multimedia applications so that students could access multimedia content linked to test-preparation materials.

For a concrete example of advanced functionality that can be achieved by connecting two distinct **EPF** plugins, consider a student reading the **GRE** Chemistry Practice Book published by ETS. This book has sample questions such as (number 4, page 11) **The molecular geometry of thionyl chloride, $SOCl_2$, is best described as** *(A) trigonal planar, (B) T-shaped, (C) tetrahedral, (D) trigonal pyramidal, or (E) linear.*

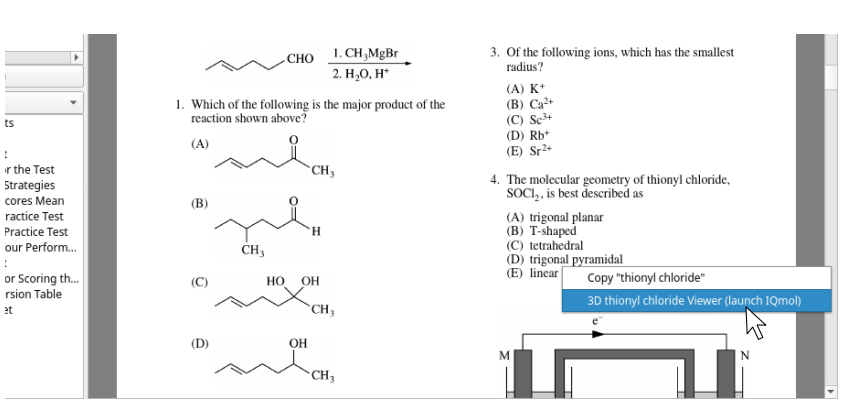
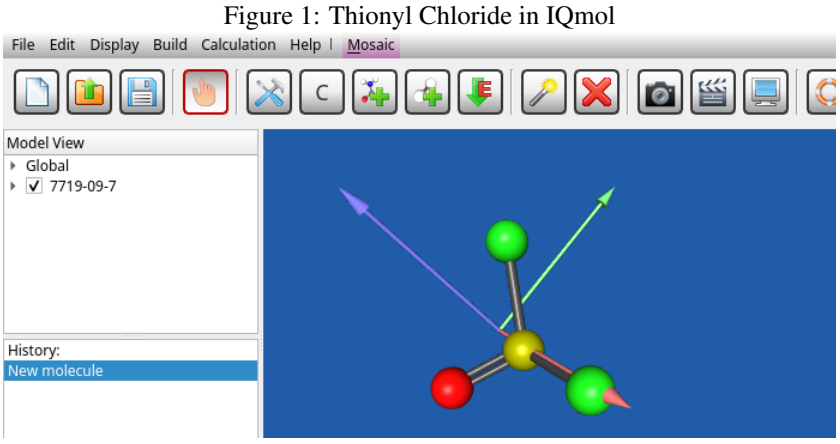


Figure 1: Thionyl Chloride in IQmol



To understand this question/answer, it may help students to view a **3D** model of thionyl chloride, which can be done through molecular visualization software such as IQmol. Accordingly, this specific question in the book may be associated with Molecular Data file for $SOCl_2$ (this file is available from the Chemical Abstracts Service database). The relation between the specific textual location (where the practice Question 4 is presented) and

the supplemental Molecular Data file would be asserted in the Document Infoset, and read by a document viewer (e.g., **XPDF**). The **XPDF** plugin would then launch IQmol and send the molecular file to the IQmol ETS Plugin, with instructions to load this file into an IQmol session (see Figure 1). The end result would be that the student, with a single click (such as selecting a visualization action from a context menu on the practice question) have access to an interactive **3D** graphic representing thionyl chloride. (Of course, analogous functionality would be available for any chemical compound with multimedia files in formats like Molecular Data, Protein Data Bank, or Chemical Markup Language).

The data sent between **EPF** applications may be more complex than a request to open a single multimedia file. Suppose a student reading the GRE Chemistry practice exam launches IQmol a second time — perhaps in conjunction with a later question (95) about the molecular structure of glucose. In this case, the plugin can send information not only about the present request but about the student's prior usage; in particular the fact that he or she had previously viewed the $SOCl_2$ file. The **EPF** plugin on the IQmol side can then load the prior file along with the new one, so the student can browse back to prior screens if desired (see the Model View panel on Figure 2).

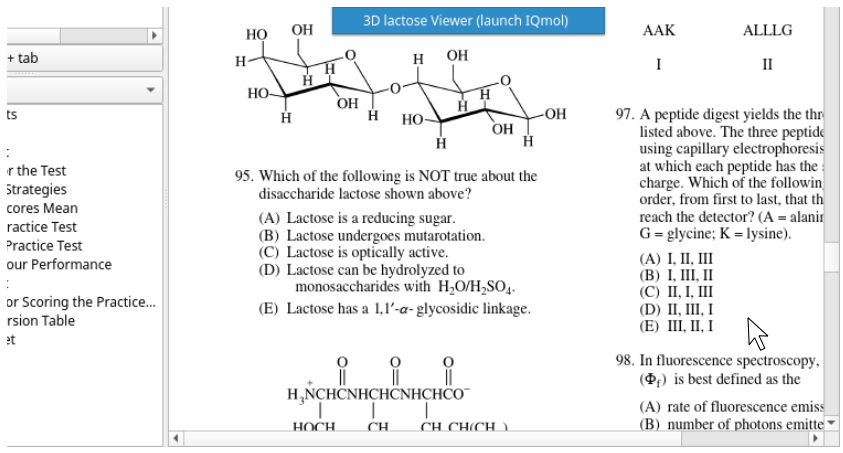
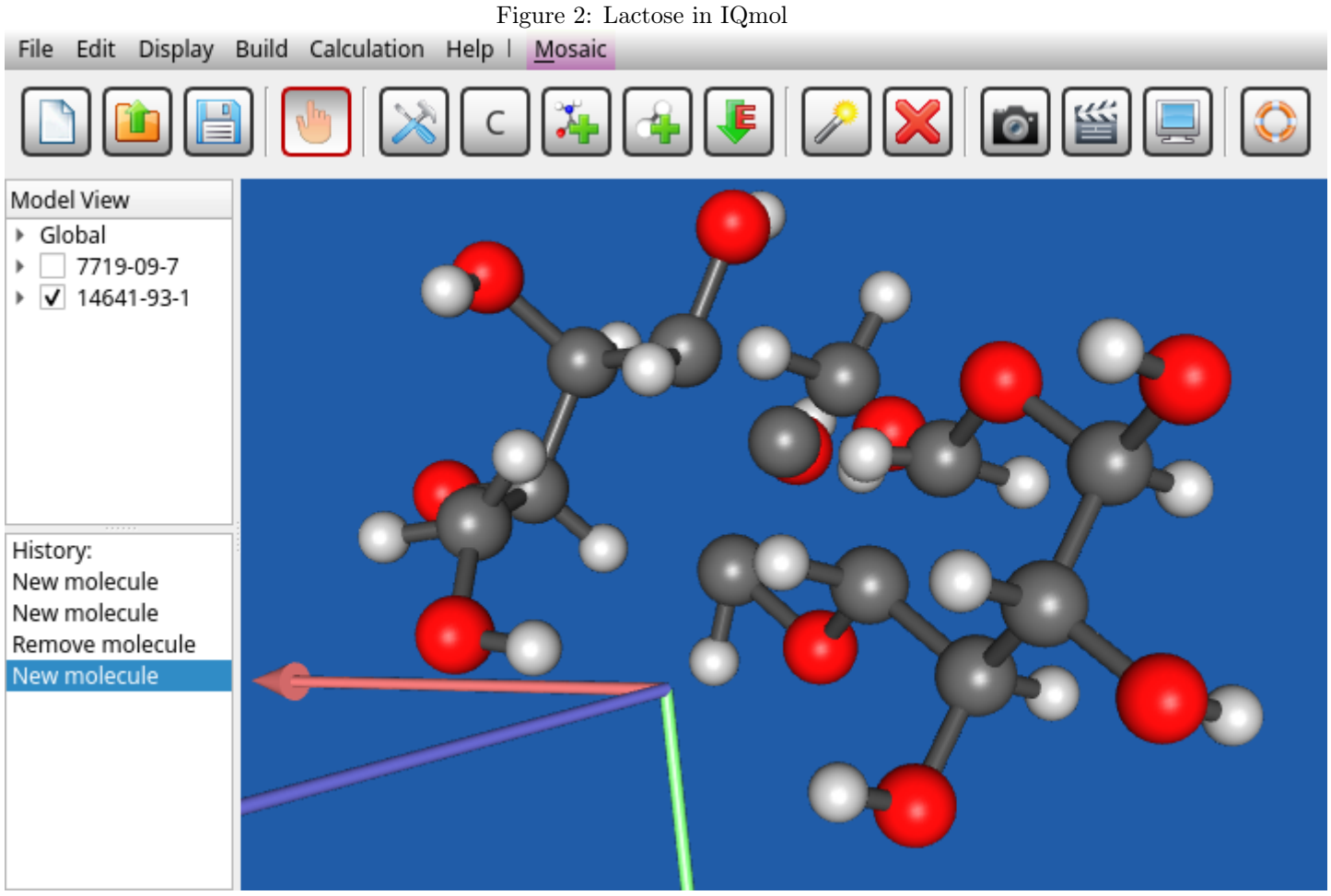


Figure 2: Lactose in IQmol



In addition to data visualization, scientific applications can help students understand concepts which are covered by a test. For example, a later **GRE** Chemistry practice question concerns Orbital Angular Momentum. To understand this topic, students may benefit from hands-on experience cal-

culating and visualizing Molecular Orbitals in IQmol. In this scenario, once again, the practice book may be linked to IQmol through the Orbital Angular Momentum question. However, in this case, instead of showing a single molecule, IQmol could load an interactive tutorial — provided by the ETS Plugin — explaining the Canonical Orbital Surfaces features in IQmol and enabling students to explore these with a variety of different molecules.

In general, the functionality provided by each ETS plugin will depend in part on the host application where the plugin is embedded. An IQmol plugin would load cheminformatic files and may activate IQmol's analytic capabilities in the domain of chemistry, whereas a plugin in Data Visualization applications (such as ParaView) could open quantitative data sets with 2D or 3D views (via surfaces, scatterplots, bar charts, etc.) and activate statistical calculations. Certain functionality, however, would be shared among all ETS plugin, including a dialog window to show basic plugin information (see figure at right) and also a more detailed review of data transmitted between applications via plugins. The "request info" tab allows students, instructors, and plugin developers to see information about the request which caused the current application to be launched or to open a specific file.

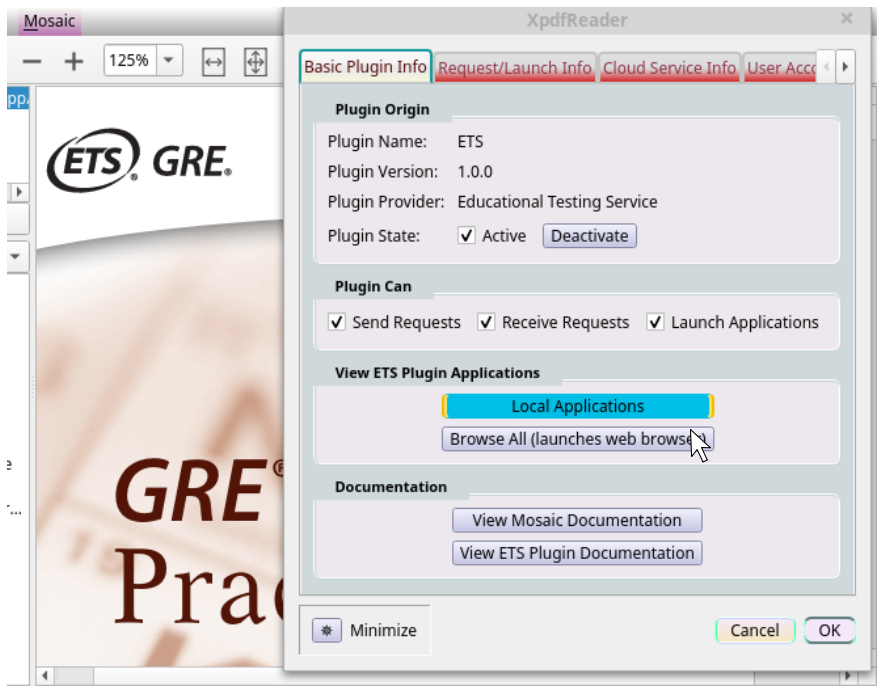


Figure 3: Request Information in IQmol